AMBIENTE DE ANÁLISE DE SENTIMENTOS BASEADO EM DOMÍNIO

Leonardo Falcão Koblitz

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Civil, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Civil.

Orientador: Nelson Francisco Favilla Ebecken

Rio de Janeiro

Dezembro de 2010

AMBIENTE DE ANÁLISE DE SENTIMENTOS BASEADO EM DOMÍNIO

Leonardo Falcão Koblitz

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA CIVIL.

Examinada por:

_____

Prof. Nelson Francisco Favilla Ebecken, D. Sc.

_____

Prof. Alexandre Gonçalves Evsukoff, Dr.

_____

Prof. Antônio Cesar Ferreira Guimarães, D. Sc.

_____

Prof[a]. Beatriz de Souza Leite Pies Lima, D. Sc.

_____

Prof[a]. Marta Lima de Queirós Mattoso, D. Sc.

RIO DE JANEIRO, RJ - BRASIL
DEZEMBRO DE 2010

Koblitz, Leonardo Falcão

Ambiente de análise de sentimentos baseado em domínio / Leonardo Falcão Koblitz. – Rio de Janeiro: UFRJ/COPPE, 2010.

XIII, 101p.:il.;29,7 cm.

Orientador: Nelson Francisco Favilla Ebecken

Tese (Doutorado) – UFRJ/COPPE/ Programa de Engenharia Civil, 2010

Referências Bibliográficas: p. 79-84.

1. Análise de Sentimentos. 2. Mineração de textos. 3. Adaptação de domínio. I. Ebecken, Nelson Francisco Favilla, II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Civil. III. Título

Este trabalho é dedicado a minha mãe Maria Laura (in memoriam)

AGRADECIMENTOS

Ao meu orientador, professor Nelson Ebecken, por sua competência e disponibilidade.

Ao amigo Marcel Waintraub pelo seu apoio e amizade.

Ao amigo Antonio Cesar por apontar a direção.

Ao amigo Marcos Pinheiro pelos seus conselhos.

Aos meus amigos da SEINF Adino, Luiz Adelino, Mauro e Rangel.

Aos colegas do Instituto cujo convívio é parte importante em minha vida.

Ao Dr. Claudio Rangel e Dra. Márcia Kikoler pelos seus conselhos.

A equipe de desenvolvimento do software GATE pelo apoio que recebi ao longo deste trabalho.

Ao Instituto de Engenharia Nuclear na figura dos meus chefes Julio Cezar Suita e Edison de Oliveira Martins Filho por incentivarem o aperfeiçoamento profissional na Instituição.

À minha filha Ana Paula.

À Vera Lúcia pelo carinho e por estar ao meu lado.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

AMBIENTE DE ANÁLISE DE SENTIMENTOS BASEADO EM DOMÍNIO

Leonardo Falcão Koblitz

Dezembro/2010

Orientador: Nelson Francisco Favilla Ebecken

Programa: Engenharia Civil

Cada vez mais as pessoas colocam suas opiniões e sentimentos em diversos tipos de serviços disponíveis na Web. Sites de *microblogging* como o *twitter*, redes sociais ou fóruns têm se tornado o meio comum para elas se expressarem.

Elas colocam de forma espontânea, gratuita e em tempo real, opiniões sobre os mais diferentes assuntos. A análise destes dados constitui uma fonte importante e rica para se entender e se antecipar às expectativas e frustrações das pessoas a respeito de um produto, um serviço ou mesmo sobre pessoas ou fatos.

Entretanto, cada domínio ou serviço de Internet tem suas peculiaridades. Jargões específicos de um domínio, gírias ou mesmo características próprias dos serviços para as pessoas colocarem as suas opiniões diferem de maneira significativa, o que compromete a utilização de sistemas de aprendizado de máquina desenvolvidos anteriormente para outros domínios.

Com isto em mente, foi proposta uma estratégia para permitir a análise de sentimentos baseada em domínio, a qual estabelece os passos para se montar rapidamente um ambiente de análise de sentimentos e conteúdo de acordo com o domínio sendo examinado.

Esta estratégia contempla desde o processo de anotação do corpus, os passos necessários para a criação de anotações de acordo com o domínio, criação de léxicos semânticos e o desenvolvimento e validação dos classificadores.

Para testar esta estratégia foi desenvolvido o sistema JULGAR, cujo núcleo está baseado no ambiente computacional GATE utilizado para o processamento de linguagem natural.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

SENTIMENT ANALYSIS DOMAIN BASED ENVIROMENT

Leonardo Falcão Koblitz

December/2010

Advisor: Nelson Francisco Favilla Ebecken

Department: Civil Engineering

More and more people show their opinion and feelings at several available Web services. Microblogging sites, such as the twitter, social networks or forums have become the ordinary media for these people to express themselves.

In real time, they say spontaneously and at no cost what they think about different matters. These data analysis is an important resource to understand and to know in advance people's expectations and frustrations about a product, a service and even people or facts.

However, each Internet site or service has its own characteristics. Sites' specific jargons, slangs or even specific characteristics of services where persons express their opinions don't have a pattern, making difficult the use of learning systems previously developed for other sites.

For this purpose a strategy was proposed a strategy that allows the analysis of feelings based on site and that establishes steps to quickly create an environment for the analysis of feelings according to the site being examined.

This strategy comprises making notes on the corpus, the necessary steps for creating annotations according to the site, lexical semantic creation and the development and validation of the classifiers.

In order to test this strategy, it was developed the JULGAR system, whose core is based on the computational environment GATE, which is employed for the processing of natural language.

INDICE

INDICE DE FIGURAS

INDICE DE TABELAS

INDICE DE EQUAÇÕES

1 INTRODUÇÃO

1.1 Descrição do problema

Cada vez mais as pessoas estão emitindo opiniões por meio de comentários textuais na Internet. Usando *blogs*, fóruns ou *chats*, ou mais recentemente, utilizando serviços de *microblogging* como o *Twitter*[1] ou o *Tumblr*,[2] elas opinam em tempo real sobre os mais diferentes assuntos.

Avaliações sobre a qualidade de um produto ou serviço ou sobre um político são postadas em uma magnitude tão grande e rápida que a análise delas utilizando os métodos atuais não tem condição de acompanhar e extrair uma riqueza de informações colocadas de maneira gratuita e direta pelos próprios usuários, eleitores ou consumidores.

Contudo, cada área ou domínio tem um vocabulário próprio e cada serviço de Internet possibilita uma maneira distinta para seus usuários se expressarem. Por exemplo, um site de venda de produtos, como, o *Amazon.com* disponibiliza para seus clientes um espaço grande para eles expressarem suas opiniões. Outros, entretanto, como o *Twitter*, limitam os usuários a apenas cento e quarenta caracteres para eles exporem as suas opiniões.

Entretanto, a maneira de o usuário manifestar sua opinião sobre um determinado domínio não depende só do espaço dado a ele pelo serviço de Internet, mas também pela forma como estes comentários são escritos. Estes podem ser escritos de maneira formal, ou por meio de gírias, símbolos ou abreviações específicas do domínio ou do serviço de Internet utilizado.

Estas características próprias de cada serviço de Internet por si só configuram um domínio a ser analisado.

Sendo assim, a análise dos comentários postados pelos usuários pode produzir resultados errados quando realizadas sem se considerar as peculiaridades de cada domínio sendo examinado e as características específicas do serviço de Internet utilizado.

A análise de um comentário feita sem se considerar as peculiaridades do domínio sendo examinado, pode resultar em uma interpretação errada ou até mesmo induzir a uma conclusão contrária do pretendido pelo usuário ou cliente de um produto

[1] http://twitter.com

[2] http://tumbblr.com

ou serviço. Como exemplo, a frase “compre o livro” pode ser positiva quando retirada de um comentário postado em um site de venda de livros e negativa quando analisada em um contexto de avaliação de filmes.

Ou seja, para avaliarmos a polaridade de uma opinião ser positiva, negativa ou neutra, devemos considerar não só as peculiaridades do domínio sendo analisado, mas também levar em conta as características específicas do meio ou serviço de Internet de onde elas foram extraídas.

1.2 Objetivo

Diversos softwares e ferramentas estão disponíveis para a realização da análise de sentimento. Cada um deles com características próprias.

BARTLETT e ALBRIGHT (2008) propõem várias estratégias para realizar análise de sentimentos utilizando o software SAS *Text Miner*[3]. Entretanto, eles concluíram que as estratégias testadas não produziram o resultado esperado, sendo o modelo produzido por um algoritmo de aprendizado de máquina o que apresentou os melhores resultados.

PAK e PAROUBEK (2010) desenvolveram um software para analisar os sentimentos contidos em mensagens do *Twitter*.

A empresa SentiMetrix[4] oferece um software que possui um módulo para análise de sentimentos ou reputações.

Contudo, os exemplos acima analisados, apresentam limitações quanto ao domínio em que eles podem ser utilizados ou por não terem integrado a eles uma metodologia para incorporar características específicas de um domínio particular o que limita a acurácia dos resultados.

Com estas considerações em mente, este trabalho tem como objetivo definir, implementar e validar uma estratégia descrita no capítulo 5 para a realização de análise de sentimentos baseada em domínio.

Para servir como ponto inicial para este trabalho, foi escolhido para estudo e análise de um domínio particular a área nuclear.

Visto ser uma área com um alto grau de controvérsia e debates acalorados, a sua análise torna-se uma boa escolha para subsidiar o estudo para o conhecimento e

[3] http://www.sas.com/text-analytics/text-miner/index.html

[4] http://www.sentimetrix.com/newsite/sentimetrix/en

entendimento necessário para a construção de um ambiente configurável de avaliação de sentimentos baseado a domínio.

Como apoio para validar esta estratégia foi desenvolvido um ambiente computacional, denominado JULGAR. Tal ambiente é configurável em função da aplicação ou domínio sendo avaliado. É possível entre outras coisas, a incorporação de classificadores de subjetividade e polaridade desenvolvidos e testados externamente, a definição de vários tipos de listas, como por exemplo, listas de *stop words*, listas com palavras ou expressões com polaridade positivas ou negativas referentes ao domínio sendo analisado.

O ambiente JULGAR permite também vários tipos de análise de conteúdo dos textos sendo analisados.

1.3 A área nuclear

A área nuclear, embora tão presente em nossas vidas, como na geração de energia elétrica, na medicina, e em inúmeros outros campos de aplicação, sofre um estigma em todo o mundo, que a associa diretamente ao seu uso em armas atômicas ou aos acidentes radioativos ocorridos no passado.

Vários estudos foram feitos (JUNIOR, 2007, KUGO, 2005a, 2005b, 2008) com o intuito de se entender melhor como as pessoas pensam, o que as motiva e que ações poderiam ser feitas para mitigar estas percepções tão negativas sobre a área nuclear.

RIBEIRO (2007) apresenta um estudo detalhado sobre as percepções das pessoas quanto à área nuclear. Conclui que o risco percebido pelas pessoas entre outras coisas é advindo de seus valores, experiências e de seus grupos sociais, mas que pode ser melhorado entre outras ações, através da segmentação do público de acordo com seus modelos mentais dominantes. Os jovens foram identificados como o grupo mais promissor para se dirigir os esforços para melhorar a percepção quanto a área nuclear.

RIBEIRO (2007) também ressalta a importância da escolha correta das palavras a serem usadas nas palestras. Por exemplo, a repetição excessiva de palavras ou expressões como uso pacífico ou segurança tem o efeito contrário ao desejado nas mentes das pessoas.

Outra pesquisa muito esclarecedora foi feita na Inglaterra (COSTA-FONT, 2008). Baseada em um relatório sobre a percepção das pessoas sobre resíduos

radioativos (*Eurobarometer*[5] 227 *report*), constatou-se que ao contrário do que se supunha, o aumento de conhecimento sobre energia nuclear faz diminuir a aceitação das pessoas a ela.

Ou seja, a importância do conhecimento em relação às atitudes sobre energia nuclear é encoberta por crenças ideológicas.

Revelaram ainda, que a percepção das pessoas é influenciada não só pela filiação política, mas também pelo o sexo. Uma das conclusões encontradas foi que as mulheres são mais receosas do que os homens ao emprego da energia nuclear.

Com o intuito de identificar os interesses básicos do público japonês relativo ao gerenciamento de resíduos radioativos no Japão, foi desenvolvido um sistema de mineração de textos para avaliar os sentimentos expressos em comentários deixados em um site na *Web* (KUGO, 2005a, 2008).

Este sistema permitiu a interação entre peritos treinados especificamente para este trabalho e o público em geral. As preocupações e desconfianças sobre a política de tratamento de resíduos foram então identificadas com sucesso, as quais forneceram pontos significativos quando da concepção de novas diretrizes para as políticas de comunicação e de gerenciamento de resíduos a serem implementadas no Japão.

A compreensão de como os sentimentos são expressos, a identificação dos fatores externos que influenciam estes sentimentos é de suma importância, não só para a área nuclear, mas também para entender, prever e propor ações de mitigação dos preconceitos às novas tecnologias, as quais por não serem familiares aos indivíduos sofrem um preconceito negativo.

1.4 Organização da tese

O Capítulo I apresenta a descrição do problema a ser resolvido e o objeto da tese. Descreve também a importância da escolha do domínio nuclear para análise de sentimentos por este ser um domínio onde os sentimentos são expressos de uma maneira apaixonante.

O Capítulo 2 define a análise de sentimentos e os seus principais desafios.

---

[5] *Eurobarometer*, site utilizado pela comunidade européia para conduzir pesquisas de opinião, http://ec.europa.eu/public_opinion/index_en.htm.

O Capítulo 3 faz uma revisão bibliográfica focando nos tópicos filtro de subjetividade, as dificuldades ao se deparar com um novo domínio e como encarar esta tarefa e por fim como a polaridade e a intensidade dos sentimentos são tratadas.

O capitulo 4 analisa em profundidade o esquema de anotação utilizado neste trabalho.

O capítulo 5 mostra uma visão geral dos experimentos, descrevendo o ambiente de desenvolvimento e a estratégia empregada na construção dos corpora, dos classificadores de subjetividade e polaridade e da ferramenta desenvolvida neste trabalho.

O capítulo 6 analisa os resultados encontrados.

Por fim o capítulo 7 apresenta as conclusões e indica os trabalhos futuros.

O apêndice A descreve as principais funcionalidades do sistema JULGAR e o apêndice B apresenta algumas regras escritas na linguagem JAPE para extrair anotações dos domínios que foram analisados.

O apêndice C apresenta um glossário com os termos mais utilizados.

2 ANÁLISE DE SENTIMENTOS

A análise de sentimentos[1] também chamada de análise de opiniões ou computação afetiva, tenta classificar textos atribuindo a eles uma orientação, a qual pode ser positiva, negativa ou neutra.

A palavra sentimento define o que uma pessoa sente a respeito de algo, pode ser também uma atitude mental de aprovação ou não a respeito a um determinado assunto, ou mesmo pode ser uma opinião ou uma reflexão.

Um sentimento, contudo não é expresso de uma maneira clara e precisa, sendo freqüentemente expresso de uma maneira sutil e complexa. Além disso, o autor pode misturar informações objetivas com subjetivas, escrever sobre outros tópicos diferentes do que ele está realmente comentando.

Para nos referirmos aos vários estados emocionais ou mentais usamos um termo geral conhecido como, estados particulares (*private state*). Onde um *private state* é definido como um estado que não está aberto a uma observação ou verificação objetiva.

Assim, uma pessoa pode ser observada por afirmar que Deus existe, mas não pela sua crença que Deus existe. Ou seja, a crença desta pessoa neste senso é particular a ela (WILSON, 2007).

Além de ser um problema intelectualmente desafiador, a aplicação da análise de sentimentos é totalmente útil em várias áreas.

A comunidade acadêmica tem tido muito interesse neste assunto, pelo fato de cada vez mais as pessoas utilizarem a Internet para externar opiniões sobre vários assuntos.

Através de *blogs*, *chats*, *newsgroup*s são realizadas avaliações de equipamentos, produtos, livros ou mesmo de pessoas ou fatos.

Pode-se citar como aplicação imediata da análise de sentimentos a pesquisa de mercado, a mineração na Web e o gerenciamento da relação com os clientes. Até em jogos on-line a interação com os jogadores pode ser aprimorada pela percepção da intensidade emocional dos comandos emitidos durante as partidas.

---

[1] A análise de sentimentos é um subproblema da mineração de opiniões

A mineração na Web pode entre outras coisas, poupar que as empresas gastem tempo e dinheiro fazendo pesquisas sobre pontos de seus interesses, extrair opiniões de pessoas que influenciam outras através de seus *blogs* e fazer uma análise em tempo real do que as pessoas pensam.

A importância da análise de sentimentos não deve ser subestimada, outrossim, a sua utilização em determinados contextos deve ser utilizada com muito cuidado e tato, visto que o governo dos Estados Unidos da América (EUA) está patrocinando estudos e o desenvolvimento de um software[2] para processar grandes volumes de dados não estruturados para extrair opiniões ou sentimentos sobre determinados assuntos.

Estas informações em conjunto com o *Military Commissions Act*, o qual permite declarar qualquer pessoa como um inimigo do estado, deve ser vista com muito cuidado. Pois ela possibilita que as autoridades americanas possam decidir arbitrariamente que a distribuição de um determinado material por um indivíduo possa ser classificada como material de suporte ao terrorismo e assim representar uma ameaça as liberdades individuais.

A detecção de falsos positivos em tais materiais sendo analisados pode injustamente classificar alguém como inimigo do estado, visto a paranóia devido ao terrorismo que nos cerca atualmente.

2.1 Definições

O objetivo da análise de sentimentos é entender como o leitor pode interpretar uma emoção em um texto e com isto desenvolver programas que executem esta tarefa.

Certos tipos de emoções quando expressas através de ironias ou metáforas estão fora do escopo desta análise por serem muito complexas.

A análise de sentimentos envolve a identificação de:

1. Expressões ou palavras que expressam sentimentos;
2. A polaridade (positivo/negativo/neutro) e intensidade das expressões e
3. O relacionamento destas com o assunto sendo examinado.

---

[2] http://www.truthout.org/cgi-bin/artman/exec/view.cgi/65/23200/printer

Alguns autores (WHITELAW *et al.*, 2005 e WILSON 2007) utilizam três dimensões para a detecção automática de emoções: Avaliação, Potência e Intensidade.

a) A dimensão Avaliação é a mais direta podendo ser positiva ou negativa. Geralmente é expressa através de adjetivos.

Ex: O filme foi bom.

b) A dimensão potência expressa o grau de identificação de se o escritor se identifica ou não com o significado da sentença. Esta dimensão é subdividida em três subdivisões.

b.1) Proximidade (perto/longe): Eu gostaria de encontrar o Paulo. Eu gostaria de encontrar o gerente. (proximidade social)

b.2) Especificidade (claro/vago): Expressa se o objeto é referenciado de uma maneira clara e direta. Ex: Eu esqueci o livro. Eu esqueci meu livro. (geral/particular)

b.3) Certeza (confiança/dúvida): Expressa a certeza do escritor quanto ao conteúdo do texto, ou seja, se o escritor está inteiramente convencido sobre o que ele está escrevendo. Ex: Supostamente é um grande filme. Definitivamente é um grande filme.

c) A dimensão Intensidade (mais/menos) é utilizada para reforçar um sentimento. Por exemplo, as frases “Este é simplesmente o melhor filme.” ou “Em que porcaria nós estamos?” (praguejando)

Os pesquisadores têm adotado basicamente dois modelos para enfrentar os desafios da análise de sentimentos.

O primeiro grupo (PANG *et al.*, 2002) tenta incorporar o conhecimento lingüístico em seus sistemas, apresentando um alto grau de complexidade neste tipo de abordagem. A outra linha de pesquisa, a ser explorada por este trabalho, utiliza sistemas de aprendizado de máquina (AUE, 2005, McDONALD, 2007) que possuem um custo menor, são mais portáteis para outros domínios e linguagens e mais robustos quanto aos erros gramaticais.

2.1.1 Corpus

Corpus[3] é um grande conjunto estruturado de textos usado para análise estatística, verificação de ocorrências e validação de regras lingüísticas considerando um universo específico. Em um corpus pode haver anotações para pesquisa lingüística. Por exemplo, o uso de P0S (*Part-of-Speech*), onde a informação sobre cada palavra (verbo, substantivo, advérbio etc.) é adicionada ao Corpus. Outro exemplo seria a utilização de lemas[4], que indicam a base de cada palavra ou anotações (*gloss*[5]).

Os documentos para serem analisados pelos algoritmos são transformados em uma representação vetorial, onde cada palavra é uma dimensão e indica uma determinada característica do texto. Através da utilização de modelos estatísticos podemos aprender as estruturas complexas da linguagem presente em um corpus de texto (OGURI, 2006).

2.1.2 Tipos de modelos

A maioria dos trabalhos sobre análise de sentimentos considera duas linhas distintas.

A primeira linha utiliza o modelo Saco-de-Palavras (*Bag-of-Words* ou *BoW*) o qual representa os documentos pelas palavras ou *tokens*[6] que o compõem.

Tipicamente um *token* é uma única palavra, mais também pode ser também uma seqüencia de caracteres, uma frase, um endereço de *e-mail*, um *link URL* ou um acrônimo (KONCHADY, 2008). Contudo, neste texto a palavra *token* irá se referir a uma única palavra e as palavras *token* e "palavra" serão utilizadas como sinônimos.

O modelo Saco-de-Palavras tem como característica não considerar a ordem dos *tokens* no documento e tenta produzir a partir delas um classificador baseado nas freqüências das palavras contidas em vários documentos.

---

[3] O plural de corpus é corpora

[4] O lema da palavra *best* é *good*

[5] É uma explanação ou definição de uma palavra obscura dentro de um texto.

[6] *Token* é uma instância de um tipo.

Tal linha simplesmente analisa as co-ocorrências[7] de expressões dentro de uma curta distância ou de padrões que são utilizados na extração de informação para analisar relações entre expressões.

Outra linha considera o modelo de orientação semântica (*semantic orientation*) a qual tenta fazer uma avaliação das características das palavras.

Orientação semântica é um método utilizado para a avaliação das características das palavras. Classifica as palavras como bom ou ruim e então contabiliza o escore total de palavras boas e ruins para o texto sendo avaliado.

Neste tipo de modelo a palavra pode apresentar uma direção de orientação positiva ou negativa (elogio / crítica) e uma intensidade (leve ou forte). O texto é classificado segundo a média entre as orientações semânticas boas e más relativas as frases que contêm advérbios e adjetivos.

2.1.2.1 Bag of Words

Nesta técnica onde cada palavra que compõem um documento é uma variável por si só com um dado peso atribuído a ela.

Este peso pode ser apenas um ou zero representando a presença ou ausência da palavra, o número de vezes que a palavra aparece no documento ou uma outra medida de peso utilizada.

Neste tipo de representação a posição da palavra não é levada em conta, o que faz que este tipo de modelo perca a capacidade de capturar a semântica do texto.

Esta perda, entretanto, é compensada pela facilidade de representação do modelo, pois ele pode ser aplicável a textos provenientes de diversas fontes.

Assim, os documentos são representados por vetores de atributos numéricos em que cada valor do atributo é dado pela função de calculo do peso das palavras sendo utilizada.

A este conjunto de vetores de documentos chama-se espaço vetorial.

---

[7] Co-ocorrência é a propriedade temporal de duas coisas acontecerem ao mesmo tempo. Por exemplo, quando ouvimos o termo “aloha” nós pensamos imediatamente no Hawaii.

2.1.2.2 Técnicas baseadas na *WordNet*

A *WordNet*[8] é uma grande base de dados léxica do idioma inglês. Nela, substantivos, verbos, advérbios e adjetivos são agrupados em conjuntos de sinônimos (*synsets*), que são grupos de palavras que podem ser sinônimas dependendo do contexto. A *WordNet* é constituída de nós (palavras) conectados por arestas (relações com sinônimos). No momento há um grupo de trabalho na construção de uma *WordNet*-br para o idioma português do Brasil (SILVA, 2006).

A Figura 1 mostra os resultados obtidos pela pesquisa dos sinônimos referentes a palavra *good* sendo realizada através da interface *on-line* da *WordNet,.*

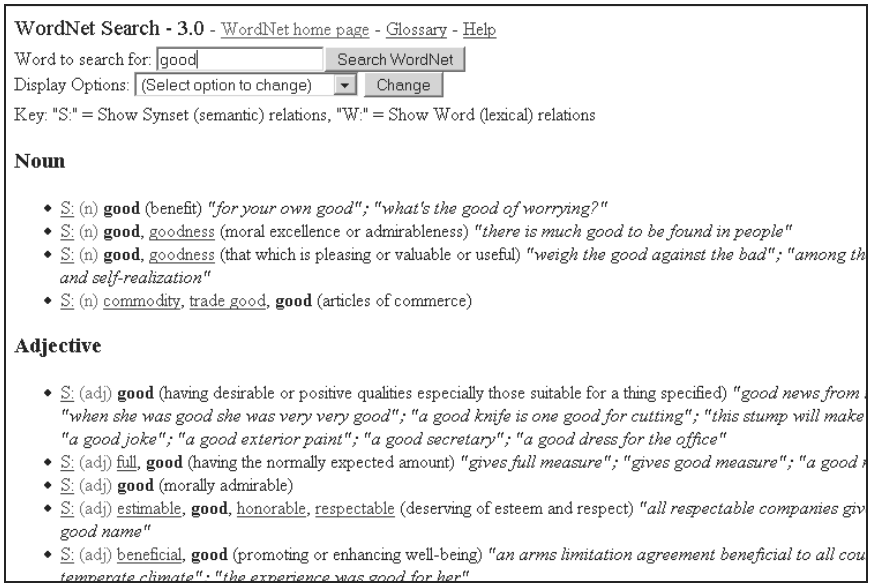

Figura 1: WordNet Search, sinônimos referentes a palavra good.

2.1.3 Aprendizado de máquina

Um algoritmo de aprendizado de máquina aprende sobre um fenômeno pela observação de um conjunto de ocorrências deste fenômeno.

O aprendizado de máquina pode ser supervisionado ou não supervisionado. No aprendizado supervisionado é fornecido ao algoritmo de aprendizado um conjunto de exemplos de treinamento já rotulados com a classe a ser aprendida.

Já para o algoritmo de aprendizado não supervisionado são fornecidos

[8] http://wordnet.princeton.edu

exemplos não rotulados e o algoritmo tenta determinar uma maneira para agrupá-los, ou seja, formar *clusters*.

A maneira de como os *clusters* são agrupados depende do tipo de método utilizado. Na primeira maneira (*single-link*) a similaridade de dois grupos é dada entre os membros mais próximos entre os dois grupos e na outra maneira (*complete-link*) é dada pela distância maior entre dois membros de cada grupo.

No caso de aprendizado de máquina supervisionado sendo estudado, temos sentenças ou documentos, que são usados como exemplos para o aprendizado de máquina. Baseado nestas ocorrências, um modelo é construído, o qual pode ser usado para predizer características de futuros exemplo deste fenômeno (CUNNINGHAM, 2010).

Ou seja, um modelo de predição baseado em atributos[9] (*features*) prediz um rótulo *y* a partir de um espaço Y, dado uma entrada x baseada num vetor de *features* $f(x,y) \in R^k$, que mapeia qualquer par (x,y) para um vetor de K reais.

O vetor de *features* é definido pelo usuário e fornece uma abstração conveniente para capturar uma variedade de tipos de dicas conhecidas que ajudam na extração da informação.

O modelo então associa um vetor de pesos *w* correspondente ao vetor de features *f* e o valor do rótulo predito será o de *y* com o maior valor de $w.f(x,y)$.

Fundamental para o desenvolvimento do classificador é a escolha das *features*, pois algumas delas aumentam o *recall*[10] na identificação da polaridade.

A acuracidade, entretanto, só é melhorada através da combinação de várias *features*. Na escolha das *features* devemos considerar:

- As que capturam negação (são as mais importantes);
- As que capturam as relações entre as instâncias das dicas de subjetividade;

---

[9] No caso de extração de sentimentos, considerando as polaridades dos documentos, o espaço *Y* consistiria de positivo, negativo ou neutro.

[10] *Recall* e *Precision* são duas métricas utilizadas para avaliar o desempenho dos classificadores.

- As que representam interdependências complexas;
- As que identificam uma "influencia pragmática" sobre a polaridade.[11]

Por fim, para analisarmos os resultados da aplicação de um modelo construído (algoritmo) sobre um conjunto de textos, é necessário haver um recurso padrão.

Este recurso o qual consideramos conter as informações corretas que queremos obter após a aplicação do algoritmo sobre os textos, o qual pode ser um dicionário, um tesauro ou um corpus previamente anotado é chamado de *Golden Standard*.

O *Golden Standard* é utilizado para a verificação da precisão do modelo sendo construído.

Máquinas de Vetor de Suporte[12] (SVM) são técnicas populares de aprendizado de máquina supervisionado utilizadas entre outras coisas para a classificação de textos. Estas técnicas utilizam uma função chamada *kernel* para mapear um espaço de pontos de dados, os quais não são linearmente separáveis em um novo espaço (MULLEN, 2004).

Desta maneira, uma SVM determina o hiperplano que separa os pontos de forma a colocar o maior número de pontos da mesma classe do mesmo lado, enquanto maximiza a distância de cada classe a este hiperplano. A distância de uma classe a um hiperplano é a menor distância entre ele e os pontos dessa classe e é chamada de margem de separação (OGURI, 2006).

Para se utilizar o SVM é necessário que cada instância de dados que é representada por um vetor contenha os valores em números reais. Caso os dados sejam categóricos é necessário primeiro convertê-los em dados numéricos

Para um melhor entendimento de SVM e de suas aplicações ver HSU (2009).

SEGARAN (2008) lista as principais aplicações das Máquinas de Vetor de Suporte. Na Tabela 1 são apresentadas as aplicações mais comuns utilizando SVM.

O outro algoritmo utilizado e analisado neste trabalho é o algoritmo *Perceptron Algorithm with Uneven Margins* (PAUM) que foi proposto por YAOYONG (2002).

---

[11] "Israel failed to defeat Hezbollar", do ponto de vista de Israel é considerado negativo

[12] Support Vector Machine

O algoritmo PAUM é um algoritmo de aprendizado supervisionado para classificadores lineares que tem apresentado como características ser simples, rápido e efetivo quando comparados com o algoritmo SVM.

Para entendê-lo melhor é definido a seguir o algoritmo *Perceptron* em que ele se baseia.

Tabela 1: Exemplos de aplicações de Máquinas de Vetor de Suporte.

SVM - Aplicações
Classificar expressões faciais
Detectar intrusos utilizando conjuntos de dados militares
Prever a estrutura de proteínas a partir de suas seqüências
Reconhecimento de letra escrita manualmente
Determinar o estrago potencial durante terremotos

O algoritmo *Perceptron* é um algoritmo baseado em redes neurais. É o algoritmo de rede mais simples que existe, pois ele é baseado em um único neurônio. Diferencia-se dos outros algoritmos por ser constituído em torno de um único neurônio não linear, isto é, do modelo de neurônio proposto por McCulloch-Pitts.

O objetivo do algoritmo *Perceptron* é classificar corretamente um conjunto de dados aplicados a duas classes (C_1 e C_2). Atribui ao ponto representado por x_1, x_2,.., x_n a classe C_1 se o valor do *Perceptron* *y* for +1 e a classe C_2 se o valor do *Perceptron* for -1 (HAYKIN, 2001).

O algoritmo PAUM é então uma extensão do algoritmo *Perceptron* especialmente projetado para tratar de problemas de classificação com duas classes onde o conjunto de exemplos positivos é pequeno comparado com os negativos.

Estes tipos de problemas são comuns nas áreas de recuperação de informação e reconhecimento de face, por exemplo.

2.1.4 Esquemas de anotação

Anotação é o ato de etiquetar, comentar ou marcar uma mídia com alguma informação (metadado) para identificar o tipo ou o conteúdo da mídia.

A meta de um esquema de anotação para o caso deste estudo, o qual tem como objetivo compreender como os sentimentos ou opiniões são expressos nos textos é identificar e categorizar estas expressões em uma sentença.

Assim para a construção de um corpus específico para a área nuclear, foi necessária a seleção de textos que expressassem diferentes opiniões sobre energia nuclear e da definição de um esquema de anotação manual para registrar as características subjetivas presentes nestes textos.

Estas anotações serviram de base para um melhor entendimento do domínio e conseqüentemente na construção do classificador.

Foram avaliados dois esquemas de anotação. O primeiro baseado na Linguagem Avaliativa (*Appraisal Theory*) e o segundo utilizado neste trabalho é baseado no esquema de anotação estendido por WILSON (2007).

WHITELAW *et al.* (2005), utilizam no seu trabalho a Teoria da Linguagem Avaliativa para fazer uma análise semântica mais detalhada dos textos, visto que os dois métodos descritos anteriormente (*BoW* e Orientação semântica) perdem aspectos importantes quando na realização da análise dos textos.

A teoria da linguagem avaliativa é a linguagem usada para se expressar opiniões, um aspecto importante da linguagem. Através dela é formada uma taxonomia de tipos de atitudes ou outras propriedades semânticas.

Na Linguagem Avaliativa, as unidades atômicas são expressões, tais como, “extremamente chato” ou “muito bom”, e não palavras individuais. Podemos definir elas como Grupos de Avaliação (*appraisal groups*), como um grupo coerente de palavras que expressam juntas uma atitude particular.

A Figura 2 apresenta uma classificação simplificada baseada na linguagem avaliativa.

Embora similares estes dois esquemas de anotação têm enfoques diferentes. O esquema proposto por WILSON (2007) distingue como as opiniões estão sendo expressas. Por exemplo, se de uma maneira direta (Paulo é um chato) ou indireta (Maria poderia ser um pouco menos detalhista) e também não inclui uma

representação para os níveis aninhados de atribuição da origem do texto.

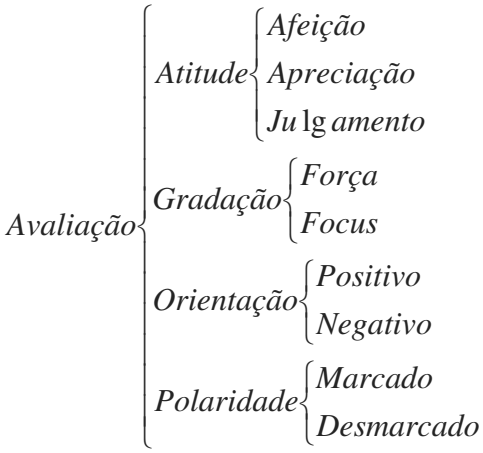

Figura 2: Classificação baseada na linguagem avaliativa

No caso da frase “Paulo é um chato, disse Maria” temos dois níveis. O primeiro nível é que de acordo com o escritor Maria acha Paulo um chato, e descendo um nível a mais, vemos que Maria está expressando uma opinião negativa sobre Paulo.

Além do mais, embora tenham sido utilizados em várias tarefas de análise de discurso, os conceitos chaves propostos pela Linguagem Avaliativa não foram avaliados empiricamente (WILSON, 2007).

3 ABORDAGENS PARA ANÁLISE DE SENTIMENTOS

A verificação do sentimento de um documento pode ser realizada em vários níveis com diferentes granularidades, a saber: nível de documento, de parágrafo, de sentença, de frase, de palavra ou mesmo no nível do orador.

Contudo, o nível a ser analisado depende do tipo de aplicação. Por exemplo, para processar um sistema de perguntas e respostas, uma análise no nível de parágrafo seria mais adequada, enquanto para um sistema de sumarização a análise apropriada seria realizada no nível de sentença ou frase.

PANG *et al.* (2004) verificaram que a precisão de uma análise feita no nível de documento pode ser aprimorada caso esta seja feita em conjunto com a análise no nível de sentença. Neste caso, cada frase que compõe o documento é classificada como objetiva ou subjetiva, através de filtro de subjetividade. São então escolhidas as *n* frases subjetivas principais que servirão de dados de entrada para um classificador de polaridade em nível de documento.

MACDONALD *et al.* (2007) propõe um modelo mais ambicioso, neste, as classificações que são realizadas em um nível influenciam a do outro nível. Ou seja, identifica o sentimento de um documento e de todos os seus subcomponentes (parágrafo, sentença, frase ou palavra). Ao contrário do trabalho de (PANG, 2004) as informações úteis para a classificação são passadas em ambas as direções.

A seguir é feita uma revisão mostrando as várias metodologias propostas para aprimorar a precisão destas classificações e outras áreas que se beneficiariam da utilização da análise de sentimentos.

Orientação Semântica (TURNEY, 2002) usa dois termos base (*excellent* e *poor*) como indicadores para os rótulos de classe positivo e negativo. Considera então, que os termos que tendem a co-ocorrer com a palavra *excellent* tendem a ser positivos e negativos com a palavra poor.

TURNEY (2002) utiliza para calcular o PMI-IR (*Pointwise Mutual Information and Information Retrieval*) referente a cada frase o mecanismo de busca *Altavista* utilizando o operador *NEAR*.

PMI-IR é uma medida de associação utilizada em IR[1]. Ela é definida como o *log* do desvio entre a freqüência observada de um bigrama[2] e a probabilidade deste bigrama se eles fossem independentes.

O operador *NEAR* retorna o número de documentos que contêm a palavra sendo pesquisada juntamente com um dos termos base (*excellent* ou *poor*). Para tal, basta que as duas palavras estejam dentro de uma faixa de palavras de tamanho *n*, não necessitando estarem juntas uma da outra.

Através da Equação 1 a orientação semântica da revisão é calculada pela média das orientações semânticas extraídas das revisões, ou seja:

OS(frase) = PMI(frase, *excellent*) - PMI(frase, *poor*) (Equação 1)

Onde:

OS: Orientação semântica;

PMI(a,b): medida de associação entre a e b utilizando PMI-IR (ver apêndice C);

poor e excellent: adjetivos bipolares

frase: palavra ou frase que se quer avaliar a polaridade em relação aos adjetivos bipolares;

A Figura 3 apresenta o exemplo (TURNEY, 2002) para o cálculo da orientação semântica de uma tupla.

Vários trabalhos realizados indicam que adjetivos e advérbios são bons indicadores de subjetividade das sentenças sendo avaliadas (WIEBE, 2002). Contudo, a utilização de adjetivos isoladamente é insuficiente para a determinação da orientação semântica.

Por exemplo, a palavra *imprevisível* pode ter uma conotação negativa quando da avaliação da direção de um automóvel e apresentar uma orientação positiva quando usada para descrever um filme.

Por isto, TURNEY (2002) recomenda a utilização de bigramas, por exemplo,

[1] Information retrieval

[2] Bigramas são grupos de dois *tokens*, é um caso especial de n-grama.

um adjetivo mais um substantivo ou um advérbio e um verbo, para se obter melhores resultados.

Tuplas: direção maravilhosa, assentos macios, etc.

Para cada tupla

Emitir um comando de busca no Altavista utilizando o operador NEAR com a palavra POSITIVA (*excellent*) e contar o número de documentos recuperados.

Emitir um comando de busca no Altavista utilizando o operador NEAR com a palavra NEGATIVA (*poor*) e contar o número de documentos recuperados.

Se o número de documentos recuperados relativos à palavra POSITIVA

for maior

então a orientação da tupla é positiva,

senão a orientação da tupla é negativa.

Fim para

Calcular a média final relativa às tuplas para calcular a orientação semântica do documento inteiro.

Figura 3: Algoritmo para calcular a Orientação Semântica de uma tupla.

Na Tabela 2 são apresentados os padrões de *tags* para se escolher os bigramas dos textos.

KAMPS *et al. (*2004) sugerem um conjunto de medidas de distância ou similaridade baseada nas relações de taxonomia da *WordNet* e na Teoria de diferenciação semântica de Charles Osgood[3], que usa vários pares de adjetivos bipolares para medir os aspectos afetivos das palavras.

[3] Psicólogo americano que desenvolveu uma teoria que usa um tipo de escala projetada para medir o significado de conotação de objetos, eventos e conceitos.

Primeira palavra	Segunda palavra	Terceira palavra (não é extraída)
JJ	NN ou NNS	Qualquer coisa
RB, RBR ou RBS	JJ	Nem NN nem NNS
JJ	JJ	Nem NN nem NNS
NN ou NNS	JJ	Nem NN nem NNS
RB, RBR ou RBS	VB, VBD, VBN ou VBG	Qualquer coisa

Tabela 2: Padrões de *tags* para se escolher bigramas dos textos.

Onde:

JJ = adjetivo;

NN = substantivo;

NNS = substantivo plural;

RB = advérbio;

RBR = advérbio comparativo;

RBS = advérbio superlativo;

VB = Verbo forma básica;

VBD = Verbo passado;

VBN = verbo particípio passado;

VBG = verbo gerúndio ou particípio presente.

Destes vários pares, o julgamento de uma palavra pode ser realizado considerando apenas três deles. A saber, os fatores de avaliação EVA (por exemplo, positivo e negativo), o fator de potência POT (forte e fraco) e o fator de atividade ACT (ativo e passivo). Destes três, o fator de avaliação é o que tem mais peso.

A distância mínima geodésica entre duas palavras pode dizer alguma coisa sobre a similaridade dos seus significados. Ou seja, a distância entre uma palavra e a palavra *good* pode ser utilizada para expressar uma relação fraca ou não de opinião positiva ao invés de uma escala precisa do grau de bondade de uma palavra.

Entretanto, esta relação mostra-se muito fraca, por exemplo, ao se calcular a distância entre as palavras *good* e *bad* obtém a distância igual a 4 (número de seqüências entre duas palavras menos um <good, sound, heavy, big, bad>), que mostra que as duas palavras são intimamente relacionadas mesmo sendo uma oposta da outra[4].

O autor sugere utilizar esta suposta fraqueza em nosso beneficio, pela utilização não apenas a distância mínima da palavra ao seu sinônimo, mas também a distância mínima em relação ao seu antônimo.

Ou seja, através da Equação 2 é calculada a orientação da palavra *honest* considerando a distância mínima entre ela e seu sinônimo e antônimo considerando o fator de avaliação (EVA).

$$EVA(palavra) = \frac{d(frase, bad) - d(frase, good)}{d(good, bad)}$$

Equação 2: Cálculo da orientação de uma palavra utilizando o fator de avaliação EVA.

Onde:

d(frase, bad): distância entre uma palavra e a palavra *bad*

d(frase, good): distância entre uma palavra e a palavra *good*

d(good, bad): distância entre as palavras *good* e *bad*

$$EVA(honest) = \frac{d(honest, bad) - d(honest, good)}{d(good, bad)} = \frac{6-2}{4} = 1$$

[4] Isto devido ao fato de existirem 14 sentidos para a palavra "*bad*" e 25 sentidos para a palavra "*good*" na *WordNet*

Estendendo a *WordNet* temos a *SentiWordNet*[5] que é uma fonte léxica para mineração de opiniões. A *SentiWordNet* atribui para cada *synset* (conjunto de sinônimos) da *WordNet* três classificações de sentimentos: positiva, negativa ou objetiva (neutra). Podemos observar que dependendo do sentido da palavra ela terá um grau maior ou menor quanto a sua classificação.

A Figura 4 mostra o resultado da análise da palavra *good* utilizando a *SentiWordNet*.

Para a palavra *good* são listados três sentidos. Para cada sentido há um triângulo invertido associado, no qual o vértice esquerdo representa a polaridade positiva, o vértice direito a polaridade negativa e o vértice inferior a polaridade nula da palavra. Dentro deste triângulo há uma esfera que é posicionada segundo o valor da polaridade da palavra. Por fim, para cada sentido da palavra é listado um ou mais exemplos de sentenças com a palavra sendo usada.

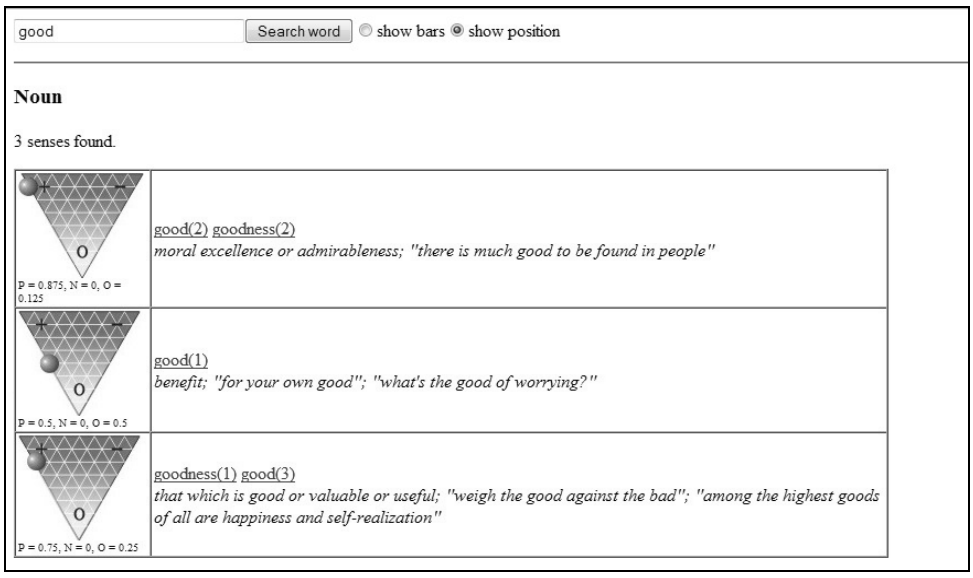

Figura 4: *SentiWordNet*, uma fonte léxica para mineração de opinião

Como a análise de sentimentos é um tipo de análise de subjetividade, a qual foca em identificar opiniões positivas e negativas, é necessário como primeiro passo

[5] http://sentiwordnet.isti.cnr.it/

da identificação da polaridade, a classificação das sentenças em objetivas e subjetivas. Uma sentença será considerada subjetiva se ela contiver uma ou mais expressões subjetivas em seu corpo.

WIEBE (2005) propõe um modelo para classificar sentenças em objetivas e subjetivas. Ela utiliza dois classificadores (um objetivo e outro subjetivo), um conjunto de dicas (regras) de subjetividade e um conjunto de textos não anotados.

As dicas podem ser classificadas em fortes ou fracas. Uma dica é considerada forte se ela tem uma alta probabilidade de ter um significado subjetivo e fraca caso ela tenha uma probabilidade baixa de subjetividade.

Assim, uma sentença é classificada como subjetiva se ela contiver duas ou mais dicas fortes.

O classificador objetivo classifica uma sentença como objetiva, pela falta de dicas, pela ausência de uma dica forte, na ocorrência de uma dica forte antes e depois da sentença sendo classificada e pela presença de pelo menos duas dicas fracas nas sentenças correntes, prévia e posterior combinadas.

De início são extraídos os padrões subjetivos e objetivos do conjunto de treino. Com estes padrões e as dicas de subjetividade um classificador *Naive Bayes* é treinado. Por fim, um conjunto de sentenças não anotadas é rotulado por este classificador e as sentenças que obtiveram melhor classificação são incluídas no conjunto inicial de testes, as quais serão utilizadas para se re-treinar os dois classificadores.

A Tabela 3 apresenta alguns exemplos traduzidos dos padrões extraídos por WIEBE (2005).

Outro classificador de subjetividade proposto por RILOFF (2003) parte do princípio que palavras que têm a mesma semântica aparecem em padrões similares. Por exemplo, “moro em” e “viajou para” irão co-ocorrer com muitas frases substantivas que representem localizações.

Padrões subjetivos	Padrões objetivos
<subj> acredita	<subj> aumentou a produção
<subj> está convencido	<subj> teve efeito
Agressão contra <np>	Delegação de <np>
Expressar <dobj>	Ocorreu sobre <np>
Suporte para <np>	Planos para produzir <dobj>

Tabela 3: Exemplos de padrões extraídos.

Onde:

subj = substantivo;

Dobj = objeto direto;

np = frase substantiva e

vp = frase verbal.

Através de um algoritmo de *bootstrapping*[6] e um conjunto inicial de substantivos subjetivos[7] (*seed words*) são procuradas nos textos palavras que aparecem no mesmo padrão de extração[8], tais quais as *seed words* iniciais e então, por hipótese considera que estas palavras encontradas pertencem as mesmas categorias semânticas delas.

Exemplo de categoria semântica: caminhão é um VEÍCULO e praia é um LOCAL.

As palavras VEÍCULO e LOCAL são uma categoria semântica.

[6] É um modelo de propósito geral para inferência estatística.

[7] Substantivo subjetivo é um substantivo que é derivado de um verbo, ex: dorminhoco derivado de dormir.

[8] O termo padrão de extração descreve estruturas que desempenham extração de informação

O modelo proposto por RILOFF (2003) tem os seguintes passos:

1. Começa com um pequeno conjunto de *seed words* e um corpus não anotado;
2. Cria um conjunto de padrões de extração para o corpus (milhares) Exemplo de um padrão: <sujeito> foi contratado;
3. Computa o escore para cada padrão baseado no número de *seed words* entre suas extrações;
4. Os melhores padrões são salvos e todas as frases substantivas são rotuladas como sendo da categoria semântica procurada;
5. Faz um re-escore dos padrões extraídos usando as *seed words* originais e as palavras recém rotuladas;
6. As palavras que foram postas no dicionário semântico são reavaliadas;
7. Para cada substantivo subjetivo é designado um escore baseado em quantos padrões diferentes extraíram o substantivo subjetivo sendo avaliado;
8. Somente os cinco melhores substantivos subjetivos ficam no dicionário semântico e
9. O processo reinicia agora usando o dicionário semântico revisado.

Um dos objetivos deste trabalho é examinar as características sobre um domínio específico, neste caso, a área nuclear.

Como a classificação de sentimentos é um problema específico de domínio, um classificador que desempenhe bem em um domínio, com certeza falhará, ou terá um desempenho muito abaixo do esperado quando aplicado a outro domínio. Ou seja, como um sentimento pode ser expresso de diferentes maneiras de acordo com um domínio, conclui-se que a construção de um classificador de sentimentos não é um problema simples (AUE, 2005).

Para construirmos um classificador temos três estratégias. A primeira é baseada em um modelo estatístico, o qual requer um material de treino. A segunda utiliza um modelo baseado em regras, que não se adapta automaticamente as novas características de um novo domínio. E a última estratégia é baseada na existência de um léxico específico para o domínio.

Ao migrarmos para um novo domínio nos deparamos com uma escassez de dados rotulados. A obtenção destes novos dados é cara e demorada, e os algoritmos tradicionais de aprendizado de máquina que requerem que tanto os dados rotulados e os não rotulados tenham sido extraídos da mesma distribuição.

Adaptação de domínio refere-se ao processo de adaptar um modelo de extração treinado para um domínio para outro domínio relacionado com somente dados não rotulados.

Ou seja, a adaptação de domínio é diferente do aprendizado semi-supervisionado, o qual assume que os dados rotulados e não rotulados sejam extraídos de um mesmo domínio.

Contudo, é assumido que embora os domínios sejam diferentes, a distribuição condicional das classes nos dois domínios permanece inalterada, ou seja, $P(y|x)=P(y_i|x_i)$ (GUPTA, 2009).

Vários métodos de adaptação de domínio (AUE, 2005, GUPTA, 2008, McINTOSH, 2006) foram propostos para se superar esta carência de dados anotados.

A idéia básica dos métodos de adaptação de domínio é escolher uma representação dos exemplos do domínio conhecido que faça que ele fique mais próximo da distribuição do novo domínio.

Entre as técnicas de adaptação de domínio temos:

- Selecionar os exemplos relevantes;
- Remover as *features* irrelevantes;
- Adicionar *features* relacionadas;
- Utilizar regularidades encontradas (propriedades) nos registros do domínio a ser adaptado para ajustar as classificações[9]

Para tal, foi montado um corpus anotado para a investigação de *features* específicas, dependentes ou independentes de domínio. Contudo em muitos casos de aprendizado de novos domínios a quantidade de dados anotados pode ser pequena ou simplesmente não existir.

Os sistemas de extração de informação usualmente requerem dois dicionários:

[9] Propriedade neste caso é qualquer regularidade encontrada no texto, como o nome do autor ser precedido pela palavra título.

um léxico semântico[10] e um dicionário de padrões de extração ou de conceitos (*concepts*)[11] para o domínio sendo analisado.

Adquiri-los automaticamente a partir de textos brutos é essencial então para superar este gargalo para se efetuar as tarefas de processamento de linguagem natural.

Existem dois modelos comuns para extrair léxicos semânticos: baseado em padrões *boostrapping* e pela similaridade distribucional[12]. A seguir são descritos algumas das soluções propostas para a confecção destes dicionários.

THELEN *et al.* (2002) mostram que de cada cinco termos gerados por um programa de aprendizado, três não estão presentes na *WordNet*, o que sugere que tais programas devem ser usados para produzir léxicos semânticos para domínios específicos.

Ou seja, as fontes disponíveis geralmente não contêm o vocabulário e o jargão especializado de um domínio.

Concluindo, embora para muitos a extração de termos de maneira automatizada produza no máximo os mesmos termos que seriam encontrados através de um procedimento manual minucioso, podemos observar que o uso de métodos automatizados consegue identificar termos raros que passariam despercebidos em uma análise feita de forma manual ou utilizando listas pré-compiladas de nomes.

Por exemplo, a utilização do software *Mutual Exclusion Bootstrapping* (MEB), descrito a seguir, encontrou nomes como *Uday* e *Igor*, os quais, não aparecem nas listagens de censo dos Estados Unidos da América.

Estas listas são extremamente completas e só não contêm os nomes de menos 0.001% da população (CURRAN, 2006).

THELEN *et al.* (2002) propuseram o algoritmo de *bootstrapping* BASILISK (*Bootstrapping Approach to SemantIc Lexicon Induction using Semantic Knowledge*) para produzir um léxico semântico independente de domínio. A Figura 5 mostra o algoritmo BASILISK.

---

[10] Dicionário de palavras rotuladas com a classe semântica, por exemplo, “pássaro” é um ANIMAL.

[11] Pode ser visto como uma expressão regular

[12] Tem por hipótese que termos similares aparecem em contextos similares

O BASILISK é um algoritmo de *bootstrapping* que foi projetado para aprender palavras pertencentes a uma categoria semântica. Tem como dado de entrada um corpus não anotado e um conjunto de palavras anotadas manualmente (*seed words*) que irão guiar o algoritmo na geração dos padrões de extração.

Ao final do processo são produzidos um léxico semântico e um dicionário de conceitos.

A grande vantagem de um algoritmo de *bootstrapping* é que ele pode identificar novos padrões de extração, os quais serão utilizados para encontrar novos termos, o que aumenta significantemente o *recall*[13].

As *seed words* podem ser geradas pelo conhecimento prévio de palavras do domínio ou serem geradas pela classificação de palavras de um corpus que apresentem a maior freqüência, as quais identificam os dez substantivos mais freqüentes por categoria.

Este algoritmo utiliza um conjunto de padrões de extração gerados automaticamente para identificar as palavras que pertencem a uma categoria semântica. Padrões de extração representam expressões léxico-sintática.

Por exemplo, o padrão "<Substantivo> foi contratado" poderia ser aplicado em sentenças que contenham o verbo contratar na voz passiva (RILOFF, 2003).

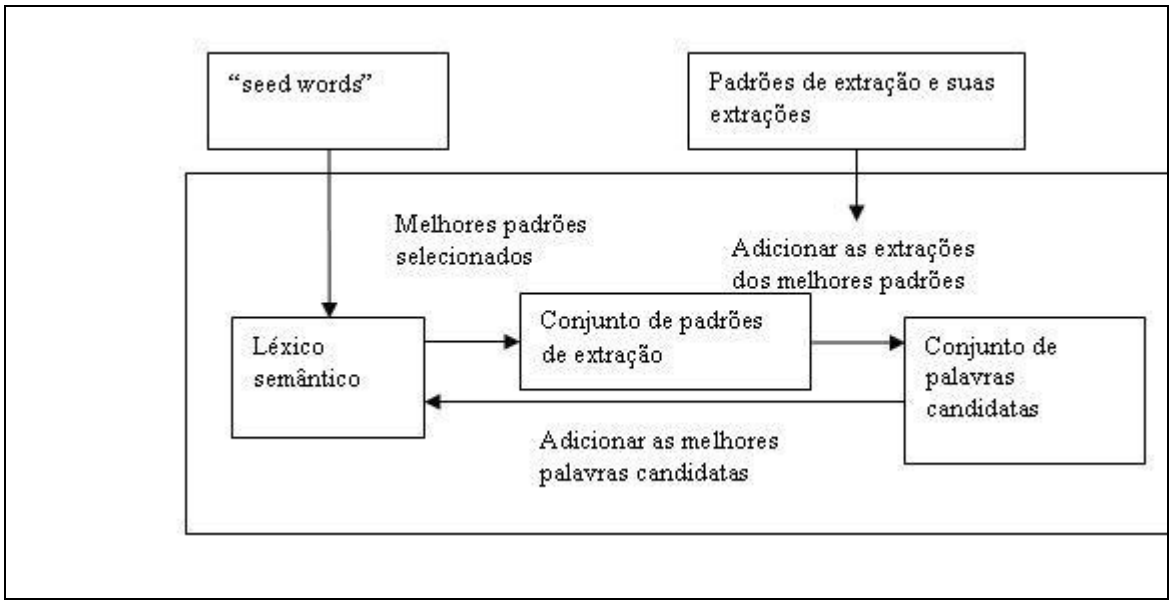

Figura 5: O algoritmo BASISLIK

[13] O *recall* mede quantos dos itens que deviam ter sido identificados foram realmente identificados.

Os melhores padrões são classificados segundo a métrica RlogF proposta por RILOFF (2004), os quais, serão utilizados na próxima iteração do algoritmo.

Na Equação 3 pode ser visto o calculo para classificação dos melhores padrões utilizando esta métrica.

$$R\log F(padrão_i) = \frac{F_i}{N_i} * \log_2(F_i)$$

Equação 3: Cálculo para seleção dos melhores padrões de extração.

Onde F_i é o número de palavras presentes no léxico e N_i é o número de palavras extraídas.

As palavras que serão adicionadas ao léxico são classificadas considerando-se todos os padrões que extraíram a palavra, não só os padrões que fazem parte do Conjunto de padrões.

Entretanto a Equação 3 apresenta um problema, pois a média pode ser altamente influenciada por um padrão que extraia um número muito grande de membros de uma categoria.

Para mitigar isto, é proposta a Equação 4 a qual estabelece quais serão as melhores palavras que serão incluídas no léxico. Ou seja, a equação é modificada para computar a média logarítmica do número de membros de uma categoria extraídos por cada padrão (THELEN, 2002).

$$AvgLog(palavra_i) = \frac{\sum_{j=1}^{Pi} \log_2(F_j + 1)}{P_i}$$

Equação 4: Cálculo para selecionar as melhores palavras que irão para o léxico.

Onde, P_i é o número de padrões que extraem a palavra *i* e F_j é o número de membros de uma categoria distinta extraídas pelo padrão *j*.

Para uma melhor compreensão, se considerarmos que a palavra “Brasil” estivesse no conjunto de palavras candidatas para ser ou não incluída no léxico semântico (Tabela 4).

Assumindo-se que os três padrões que extraíram a palavra “Brasil”, também extraíram outras palavras, sendo que algumas delas, sublinhadas, pertencem a mesma categoria semântica da palavra “Brasil”, teríamos (2 + 3 +2) / 3, o que daria o escore igual a 2,3 para a palavra “Brasil”. Este resultado significa que os padrões que

extraem a palavra “Brasil” também extraem, na média 2,3 palavras conhecidas relacionadas a categoria semântica locação.

CURRAN *et al.* (2006) declaram que modelos baseados em padrões de extração, como o BASILISK, embora possam ser implementados eficientemente, requererem um pré-processamento lingüístico mínimo e podem ser utilizados em grandes conjuntos de dados e são altamente dependentes de domínio.

Entretanto, em contra partida a estas vantagens, a adição ao léxico de somente de um termo que tenha um senso diferente predominante ou de um contexto que fracamente restringe os termos, pode rapidamente introduzir erros.

Ou seja, na avaliação de um algoritmo de *bootstrapping* um tema comum é o chamado *semantic drift*[14] onde termos ou contextos errados infectam a classe semântica (categoria).

Padrão	Extrações
foi morto no <np>	Brasil, tiroteio, confronto, Peru, El Salvador
<np> foi dividido	Brasil, o país, Peru, cartel de Medellin, o exercito, Colômbia
embaixador para <np>	Brasil, Peru, a UN, Panamá

Tabela 4: Exemplo do cálculo para se selecionar a palavra que será incluída no léxico

Para superar este erro, o algoritmo BASILISK só adiciona os cinco termos mais confiáveis em cada iteração, ao invés de todos os termos extraídos.

Contudo CURRAN *et al.* (2006) consideram esta estratégia insuficiente para tratar deste viés semântico.

Propõem que para tratar deste viés deve-se utilizar várias instancias independentes de *bootstrapping* para extrair em paralelo múltiplas classes semânticas.

Este algoritmo, chamado de *Mutual Exclusion Bootstrapping* (MEB) se diferencia do método proposto por RILOFF e JONES (1999) que usam padrões de

[14] Um viés semântico devido à inclusão de termos ou padrões errados aos conjuntos que irão servir na geração de novos termos e padrões na próxima iteração do algoritmo.

extração gerados pelo programa AutoSlog-TS.

CURRAN *et al.* (2006) utilizam um conjunto de unigramas (composto de 5 termos) retirados do corpus "Web 1T" disponibilizado pelo o buscador Google para ter um modelo o máximo possível independente de domínio. Para tal:

- Considera que o *token* do meio seja o termo e os outros dois *tokens* de cada lado do unigrama formam o padrão de extração (contexto). Emprega várias heurísticas para selecionar os termos e conceitos melhores.
- Assume também, que os termos somente têm uma única classe semântica e que os contextos somente extraem termos com uma única classe semântica, ou seja, as classes semânticas são mutuamente exclusivas quanto a termos e contextos.

McINTOSH e CURRAN (2006) revêem o algoritmo MEB estendendo a suposição de exclusão mútua entre classes ao incorporar uma nova "*pool* de padrões" acumulativos e novas funções ponderadas para termos e padrões de extrações.

DAI (2007) propõe um algoritmo de classificação baseado em *co-clustering*[15] para propagar a estrutura de classe e conhecimento de um domínio conhecido (rotulado) para um novo domínio.

Define D_i como o domínio com os dados rotulados e D_0 como o novo domínio a ser classificado.

Pressupõe que embora os domínios sejam diferentes existe uma relação entre eles. Por exemplo, existe uma relação entre um domínio sobre CARROS e outro sobre CAMINHÕES.

Desta maneira, as palavras similares presentes nos dois domínios descrevem categorias similares mesmo estando em distribuições diferentes.

O classificador então utiliza o conhecimento vindo do domínio D_i, que é usado como uma restrição e um grupo de palavras aparentemente não relacionadas extraídas da parte de *co-clustering*.

Nesta mesma linha CHEN *et al.* (2009) propõem um método para adaptação de domínio. Considera também que a distribuição entre os dois domínios têm que ser

[15] Dada uma matriz multidimensional, "*co-clustering*" se refere a uma *clusterização* simultânea por várias dimensões

relacionada. Seu método baseia-se no fato que existe um espaço de conceito no qual a distribuição de cada domínio é bem aproximada.

Por exemplo, se o domínio CONTABILIDADE contiver {qualificação, ano, experiência, CA,...} e o domínio PLANO DE SAUDE contiver {qualificação, psicologia, experiência, CCP,...}, pode-se extrair um conjunto de *features* que são independentes de domínio, no caso, {qualificação e experiência}.

O método minimiza a lacuna de distribuição entre os dois domínios e a perda dos dados rotulados presentes no domínio em virtude da extração do subespaço de conceito.

Como dito anteriormente a análise de sentimentos pode ser dividida em três passos. O primeiro que é identificar palavras ou expressões que exprimem sentimentos. Isto é feito quando separamos as sentenças subjetivas das objetivas.

O segundo passo é identificar como estes sentimentos são expressos Ou seja, identificar a polaridade deles (positiva, negativa ou neutra) e a sua intensidade.

Contudo, a identificação da polaridade de um documento não pode ser comparada ao problema de se classificar um documento como positivo ou negativo.

Inicialmente pode-se pensar que este problema poderia ser tratado como classificação baseada em tópicos (finanças, esportes etc.).

Entretanto na classificação de textos o foco é a identificação do tópico, enquanto na classificação de sentimentos o foco é na avaliação do sentimento do autor em relação ao tópico.

A resposta a esta questão é não, pois a classificação de sentimentos tem se mostrado mais difícil, enquanto os classificadores obtêm precisão de 90% quando se trata de tópicos a precisão dos classificadores para sentimentos é bem menor (PANG, 2002).

Tanto verbos, substantivos, adjetivos, advérbios ou colocações (*collocations*)[16] podem comunicar uma atitude (valência) positiva ou negativa.

[16] Seqüência de palavras ou de termos que co-ocorrem com mais freqüência do que seria esperado pelo acaso, por exemplo: "cirurgia cosmética".

As técnicas utilizadas para se saber as polaridades das palavras são:

- Consulta da polaridade prévia (*prior polarity*) de uma palavra contida em um léxico;
- Baseadas em um corpus, o qual usa restrições de co-ocorrência de palavras com polaridade similar ou oposta;
- Medidas estatísticas de associação entre palavras e
- Informações quanto às relações léxicas e *glosses* das palavras (*WordNet*).

A utilização de léxicos com polaridade das palavras tem sido uma fonte razoável para a análise de sentimentos e mineração de opinião. Entretanto o seu uso é limitado, porque tais fontes são de propósito geral e não refletem adequadamente as polaridades das palavras quando usadas em um domínio específico.

Nestes léxicos a polaridade atribuída a cada palavra é feita sem se considerar o contexto onde ela está sendo utilizada. Contudo sabemos que a polaridade de uma palavra pode ser influenciada entre outras coisas, pelo domínio (Ela é uma mulher quente), pelo papel sintático da palavra (objeto ou sujeito), pela modalidade (nenhuma razão absolutamente) e até pela perspectiva da pessoa que está expressando o sentimento.

Para melhorar a precisão na definição da polaridade de uma palavra, são descritos a seguir alguns métodos e modelos.

A Figura 6 apresenta um exemplo que mostra um método de cálculo da polaridade de um texto. Este cálculo é realizado apenas pela simples comparação do total de palavras com polaridade positiva e negativa (definidas em um léxico previamente).

Como se pode notar, este tipo de simplificação, a qual atribui a polaridade da sentença apenas pela comparação entre os totais de palavras com as polaridades positiva e negativa é falho e não deve ser aplicado.

Of course, that would not stop deregulation of the power industry altogether. The **blunderbuss** $^{-}$ of state initiatives will see to that. However, by prolonging **uncertainty** $^{-}$, it would **needlessly** $^{-}$ **delay** $^{-}$ the arrival of the **bonanza**$^{+}$ of **benefits**$^{+}$ that consumers **deserve**$^{+}$, and give them **legitimate** $^{+}$ grounds for their **cynicism** $^{-}$.

Figura 6: Exemplo de um texto com as polaridades positivas e negativas das palavras

Como podemos observar, embora este texto tenha mais palavras com polaridade negativa (cinco palavras marcadas no texto com o sinal "-") do que positiva (três palavras marcadas no texto com o sinal "+"), o leitor irá concluir que a opinião do autor é positiva ao invés de negativa.

Ou seja, não podemos considerar apenas as polaridades dos itens de um léxico para determinar a polaridade total do texto.

POLANYI (2004) propõe que a valência de um item pode ser alterada, fortificada ou enfraquecida pela presença de outros itens de um léxico, pelo tipo do gênero e a estrutura do texto e até por fatores culturais.

Enquanto alguns itens têm uma polaridade positiva ou negativa já herdada, as suas valências podem ser alteradas de acordo com o contexto através de modificadores de valência (*valence shifters*).

Temos como modificadores os de intensificação e os de negação. Os de negação são os mais óbvios, como em "João não é inteligente". Temos como exemplos as palavras nunca, nenhum ou ninguém.

Se combinarmos palavras com valência positiva com um modificador de negação, temos a modificação da valência de positiva para negativa.

Modificadores de intensidade como a palavra pouco, em "pouco eficiente" ou profundamente, em "profundamente suspeito" atuam para enfraquecer ou aumentar a força de um termo.

POLANYI (2004) mostra também que quando da presença de operadores modais da língua inglesa como *would*, *could*, *might* ou *possibly*, o calculo da atitude do texto considerando este contexto modal não deve ser tratado precisamente como em um contexto real.

Por exemplo:

- Mary is a terrible person. She is mean to her dogs.
- If Mary were a terrible person, she would be mean to her dogs.

Neste caso os operadores modais neutralizam a valência negativa de *mean* e *terrible*.

Ao analisarmos o texto também devemos considerar as restrições culturais. Dependendo do contexto uma pessoa pode ser classificada em lutador pela liberdade ou terrorista, ou seja, considerando o texto o poder da revolução que a América necessita, a palavra revolução é claramente positiva.

O cálculo da valência para a frase "*The very brilliant organizer failed to solve the problem*" é apresentado a seguir.

Enquanto a palavra "*very"* intensifica a palavra "*brilliant"*, a palavra "*failled"* inverte a polaridade.

A expressão solve "*the problem*" também é invertida por "*failled*", resultando em uma polaridade negativa com um grau de intensificação grande (-4).

A Figura apresenta o cálculo da valência para a sentença "*The very brilliant organizer failed to solve the problem"*.

brilliant	+2	(very)
Very brilliant	+3	-3 (failed)
failled	-1	-1
Solve the problem	+1	0 (neutralizado por failed)
pontuação		-4

Figura 7: Cálculo da valência para a sentença: "The very brilliant organizer failed to solve the problem".

NAIRN (2006) analisa as relações semânticas entre a sentença principal e as complementares para estimar a polaridade textual. Se considerarmos as frases a seguir, um sistema que computa as inferências textuais teria condições de deduzir que as frases (b) e (c) seguem a frase (a). Entretanto, há uma diferença clara entre os predicados "pretende" e "esqueceu", ou seja, não se pode afirmar precisamente que (c) é conseqüência direta de (b).

a. Mauro se esqueceu de fechar a porta.

b. Mauro pretende fechar a porta.

c. Mauro não fechou a porta.

Este modelo analisa construções factíveis simples e as interações entre verbos factíveis (fingir que) e implicativos (hesitar) para determiná-la se dado uma sentença *A*, pode-se concluir que a sentença *B* que a segue é verdadeira, falsa ou desconhecida. Alguns verbos podem diretamente implicar em uma ação (factível ou implicativa) positiva ou negativa, enquanto outros podem transmitir ambas as polaridades dependendo do contexto.

A Tabela 5 apresenta exemplos de tipos de verbos que expressam polaridades em função do contexto.

NAIRN (2006) considera que a polaridade de qualquer contexto depende da seqüência de contextos imediatamente acima. Ou seja, na frase “Adelino recusou a não proibir a caça”, a expressão “não proibir” é neutralizada totalmente pelo predicado “recusou”.

	Palavra	Polaridade relativa	
		Implicação	
Implicativos de dupla polaridade	Gerenciar para Esquecer de	(+) (-)	(-) (+)
Implicativos só positivos	Forçar para Negar que	(+) (-)	nenhum
Implicativos só negativos	Hesitar para Esforçar para	nenhum	(-) (+)
		Pressuposição	
Factível	Esquecer que	(+)	(+)
Contra factível	Inventar que	(-)	(-)
		Implicação	Pressuposição
Neutro	Querer que	nenhum	nenhum

Tabela 5: exemplos de verbos em uma sentença que podem implicar ou presumir em uma conclusão de polaridade de uma sentença subseqüente.

Cada contexto tem associado com ele dois conjuntos de contextos, com polaridades positiva e negativa. A polaridade do contexto então é calculada em termos dos seus conjuntos pais (positivos e negativos) em referência a um verbo *V*, o qual liga os dois contextos e ao seu valor contido em uma tabela de *lookup* (positivo ou negativo).

A Figura 8 apresenta as equações para o cálculo das polaridades positiva e negativa (NAIRN, 2006).

$$\otimes C =_{def} \{C\} \bigcup \begin{cases} \otimes_p(C) \; se \; sig^+ \; (V_p(C), C) = + \\ \Theta_p(C) se \; sig^- \; (V_p(C), C) = + \\ 0, \; outro \end{cases}$$

$$\Theta C =_{def} \{C\} \begin{cases} \otimes_p(C) \; se \; sig^+ \; (V_p(C), C) = - \\ \Theta_p(C) se \; sig^- \; (V_p(C), C) = - \\ 0, \; outro \end{cases}$$

Figura 8: Cálculo da polaridade, onde $\otimes C$ polaridade positiva e ΘC polaridade negativa

CHOI *et al.* (2008) afirmam que para o cálculo da polaridade de um texto é necessário mais do que a utilização do modelo *BoW*. Propõem para análise dos sentimentos no nível de sub-sentença um modelo de aprendizagem que incorpora regras de inferência inspiradas na semântica composicional.

A semântica composicional declara que o significado de uma expressão composta é função do significado de suas partes e das regras sintáticas pelas quais elas são combinadas.

Para então se determinar a polaridade de uma sentença, deve-se levar em conta de como as palavras que constituem a sentença interagem entre si.

Seu modelo de aprendizagem além de explorar a semântica composicional provê a flexibilidade necessária para manusear as complexidades da linguagem natural, tais como, encontro de palavras desconhecidas, exceções, falta de regras ou regras imprecisas.

Confirma em sua avaliação a importância de não só se considerar palavras negadoras, tais como, as palavras “não” ou “nunca”, mas também negadores (palavras

que alteram as fases) do tipo *function-words*[17] e *content-words*[18], tal qual eliminado na melhoria da precisão na determinação das polaridades das sentenças.

A Figura 9 lista um conjunto de regras de inferência inspiradas na semântica composicional (CHOI, 2008).

Regras		Exemplos
Polaridade (not[arg1])=	¬ Polaridade(arg1)	not [bad]
Polaridade ([VP]-to-[NP])=	Composicional(VP, P)	[refused] to [deceive] the man
……..		
Definição da função Composicional(arg1, arg2)		
Composicional(arg1,arg2) Se (arg1 é um negador) então ¬ Polaridade(arg2) Senão se (polaridade (arg1)== Polaridade(arg2)) então Polaridade(arg1) Senão a polaridade maior dos dados		

Figura 9: Regras de inferência composicional inspiradas na semântica composicional

[17] São palavras que servem para expressar relações com outras palavras na sentença ou especificam a atitude ou humor do orador (pronomes, conjunções, etc)

[18] Uma palavra tal qual um substantivo, verbo, adjetivo, que possui um significado léxico estável.

4 ESTUDO DO CORPUS

Quando queremos desenvolver um algoritmo de aprendizado de máquina é mais comum não termos a mão um conjunto de textos rotulados para a tarefa a ser executada

Quando isto ocorre devemos utilizar uma das técnicas descritas anteriormente para a adaptação de domínios.

No caso ideal, quando houver tempo e recursos, o melhor e talvez mais confiável, é anotar um conjunto de documentos para uma melhor compreensão de como os sentimentos são expressos no domínio sendo analisado.

Quando este for o caso, é necessária a escolha de um esquema de anotação para ser utilizado.

4.1 O Esquema de anotação

Para um esquema de anotação ser útil, este deve fornecer uma grande variedade de informações sobre as emoções e opiniões presentes nos textos, as quais poderão ser utilizadas no desenvolvimento e avaliação de sistemas de Processamento de Linguagem Natural (NLP).

O objetivo do esquema de anotação utilizado é então, representar os estados mentais e emocionais internos contidos nos documentos e distinguir a informação subjetiva da informação factual WIEBE (2003).

O esquema de anotação utilizado neste trabalho foi o esquema proposto por WILSON (2007).

Um processo de anotação segue os seguintes passos:

- Estabelecer o que se quer anotar;
- Selecionar o esquema de anotação;
- Selecionar os documentos;
- Pré-processar os documentos caso necessário;
- Treinar os anotadores;

- Aplicar o esquema de anotação aos documentos;
- Validar as anotações.

Dependendo do que se quer extrair, uma anotação pode ser realizada sobre cada palavra (*token*) de um documento, ou sobre faixas de texto que identifiquem uma entidade, por exemplo, os nomes de organizações.

Uma anotação possui um ou mais atributos. Por exemplo, a anotação de uma faixa de texto que exprima um sentimento dentro de um texto, pode ter como atributos a posição inicial e final desta anotação no texto, qual tipo de sentimento que ela expressa e a polaridade deste sentimento.

Na Figura 10 é mostrado um exemplo de anotação em que a frase "*Better busy than bored though*" foi anotada como sendo do tipo *Sentence* com os atributos de posição relativos ao texto onde ela está presente (*Start* e *End*) e com as *features* "classe" com o valor "pos" e "tamanho" com o valor do número de caracteres (inclusive os espaços em branco) que compõem a sentença.

	Atributos			
Type	Set	Start	End	Features
Sentence		9	38	{classe=pos, tamanho=29}

Figura 10: Exemplo de uma anotação de uma sentença.

Como dito anteriormente, a anotação de um documento é feita através da aplicação de um esquema de anotação manual.

Ou seja, a pessoa encarregada de anotar as características das expressões (o Anotador) seleciona uma faixa de texto que ela considera que expressa um sentimento e atribui a está faixa valores as características relativas a este tipo de expressão, tal como, se a expressão anotada tem uma conotação positiva ou negativa e qual a sua intensidade.

Entretanto ao se aplicar o processo de anotação, o anotador deve ter em mente que as regras definidas no esquema de anotação não são fixas. Pois elas devem ser anotadas e interpretadas segundo o contexto em que elas aparecem.

Quando as anotações forem realizadas por mais de um anotador é importante

se fazer uma avaliação para apurar o nível de concordância entre eles[1]. A medida mais utilizada neste caso é o coeficiente estatístico *Cohen´s Kappa* para se avaliar quando os anotadores tiverem um mesmo conjunto de questões que eles deveriam ou não concordar.

Este coeficiente é dado por *k* descrito na Equação 5.

$$k = \frac{P(a) - P(e)}{1 - P(e)}$$

Equação 5: Cohen´s Kappa

Onde, P(a) é a concordância observada entre os anotadores e P(e) é a probabilidade hipotética de concordância entre eles. O coeficiente de *Kappa* pode variar de –1 (discordância completa) a +1 (concordância completa).

A importância de tal avaliação é fundamental para se determinar o grau de dificuldade da tarefa. Muitas das vezes os anotadores terão discordância a respeito não só da faixa de texto que deverá ser anotada, mas também quanto ao valor da característica associada à anotação.

Além do mais o valor de concordância pode ser utilizado para se estabelecer um teto para o desempenho dos algoritmos de classificação.

As anotações dos documentos foram realizadas utilizando o software GATE (*General Architecture for Text Engineering*[2]) que é uma plataforma para processar linguagem natural e foi desenvolvida pelo grupo de NLP da universidade de Sheffield[3].

Na Figura 11 é mostrado um exemplo de anotação utilizando o ambiente computacional GATE.

As instruções de como realizar uma anotação utilizando o software GATE estão disponíveis no próprio site deste software[4].

Para mais informações sobre o esquema anotação utilizado, ver WILSON (2006), WILSON (2005), WILSON (2003), WIEBE (2002) e WIEBE (2003).

---

[1] Inter-annotation agreement

[2] http://gate.ac.uk/

[3] http://www.shef.ac.uk/dcs/

[4] http://www.cs.pitt.edu/mpqa/opinion-annotations/gate-instructions/

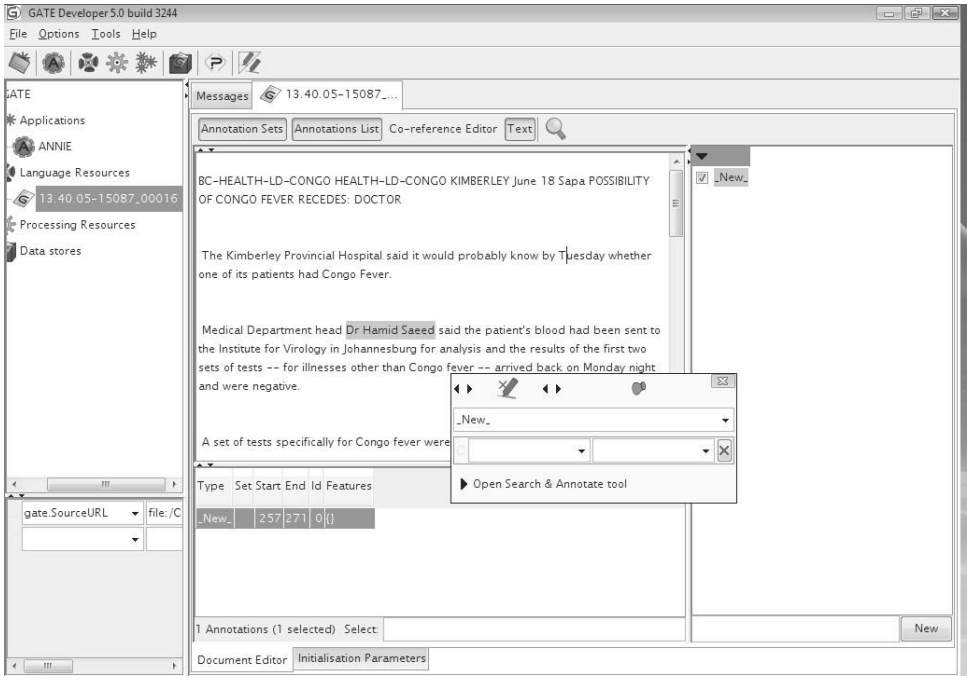

Figura 11: Tela de anotação de documentos utilizando o software GATE

4.2 Definições

Uma expressão subjetiva é qualquer palavra ou frase usada para expressar uma opinião, emoção, avaliação, postura etc.

Assim as sentenças subjetivas e objetivas são diferenciadas pela presença ou não de uma expressão subjetiva no corpo de uma sentença.

Um *Private State* (PS) é um termo geral usado para expressar opiniões, crenças, pensamentos, sentimentos, metas, avaliações, julgamentos etc. Eles são expressos usando uma linguagem subjetiva.

Um PS pode ser visto pelos seus componentes funcionais, ou seja, uma fonte (quem experimenta o PS) expressando uma atitude, possivelmente sobre um alvo. Contudo, nem toda atitude tem que ter um alvo.

O exemplo de uma menção explicita de um PS seria: “Paulo odeia Maria”. Neste exemplo de um PS, a fonte que expressa o PS é Paulo, que está mantendo uma atitude negativa (odeia) em relação a Maria (o alvo).

4.2.1 Representação de um PS

Podemos expressar um PS em um texto através de:

1. Menções explícitas de um PS;
2. Eventos de fala expressando um PS e
3. Elemento Subjetivo Expressivo (ESE).

São definidos cinco quadros para se representar os componentes funcionais de um PS. Cada quadro tem um conjunto de atributos próprios. Uma anotação é realizada então através do preenchimento de atributos dos quadros relacionados ao PS sendo anotado.

Este esquema de anotação descrito foi proposto por WILSON (2007), o qual incluiu os quadros de anotação de atitudes e de alvos ao modelo anterior de anotação utilizado em outros esquemas de anotação.

São eles:

1. Quadro de Private State (Private State Frame)
2. Quadro Evento de Fala Objetivo (*Object Speech Event Frame*)
3. Quadro Agente (*Agent Frame*)
4. Quadro Atitude (*Attitude Frame*)
5. Quadro Alvo (*Target Frame*)

1) Quadro de *Private State*

1.a) Este tipo de quadro é utilizado tanto para representar menções diretas de um PS quanto para se representar eventos de fala expressando um PS. Por eventos de fala consideram-se eventos tanto falados como escritos.

Ex: “Os países do leste europeu ficaram FRUSTADOS com as decisões tomadas pelo EUA”.

Neste caso há uma menção direta de um PS, “FRUSTADOS”.

1.b) Pode ser expresso também através de um evento de fala expressando um PS.

Os Eventos de fala são expressos por:

- Geralmente através de verbos;
- Por substantivos: preocupação ou vontade das pessoas ou
- Por adjetivos: assustador.

Contudo, nem sempre a presença de um evento de fala indica um PS.

Exemplo: O relatório está cheio de absurdos, “DISSE” Rangel.

Neste exemplo o Evento de fala é “DISSE”.

1.c) Elemento Subjetivo Expressivo (ESE)

Neste tipo os PS são expressos indiretamente através do uso de certo tipos de palavras e também pelo estilo de linguagem do escritor ou orador. Os ESE são utilizados para as pessoas expressarem a sua frustração, raiva, sentimento positivo etc.

Exemplo: “Eu previ uma fraude eleitoral, mas não um ROUBO TÃO DESLAVADO, disse Marcel”.

Exemplo: O relatório está CHEIO DE ABSURDOS, disse Adelino.

Neste exemplo CHEIO DE ABSURSOS é um Elemento Subjetivo Expressivo.

Ocasionalmente os PSs são expressos por ações físicas diretas. Neste caso chamam-se tais ações como ações de um PS, tais como, aplaudir, balançar os seus punhos agressivamente, franzir as sobrancelhas ou suspirar.

Ex: As pessoas começaram a APLAUDIR o jogador logo que ele entrou em campo.

2) Eventos de Fala Objetivos

Utilizados para distinguir um texto apresentado como factual de outro texto orientado a opinião.

Exemplo: “O doutor FALOU que o sangue do paciente foi enviado para análise”.

Esta sentença é objetiva embora contenha um evento de fala.

3) Quadro Agente

O esquema de anotação utiliza este quadro para anotar frases substantivas[5] que se referem às fontes ou origens de um PS, ou seja, são os que emitem o PS.

4) Quadro Atitude

A idéia por trás da anotação da atitude é a de se capturar a faixa de texto que expressa a atitude como um todo. Se a atitude de um PS for uma emoção negativa, então a anotação da atitude tem que englobar completamente o texto que expressa esta emoção negativa.

Exemplo: Márcio disse, "Isto é uma idéia ESTÚPIDA".

A anotação da palavra estúpida representa por inteiro a atitude negativa expressa. Mas, nem sempre a atitude é expressa assim tão fácil. No exemplo a seguir a faixa que representa a atitude expressa pelo PS é representada por várias palavras.

Exemplo: Maria está com MEDO QUE O MUNDO ESTEJA CHEGANDO AO FIM.

As anotações de atitude são ligadas as suas anotações de subjetivas diretas correspondentes.

As atitudes são categorizadas em categorias amplas, cujo objetivo é capturar distinções úteis para sistemas de perguntas e respostas ou outros sistemas de NLP.

Uma atitude pode ser classificada como sendo: sentimentos, concordância, argumentação, intenções, especulações ou outros tipos. Estas podendo ser positivas ou negativas. O outro atributo presente neste quadro é a intensidade que pode ser: baixa, média, alta ou extrema.

Aqui se encontra um ponto interessante para se melhorar a anotação dos valores atribuídos as intensidades pela utilização de uma medida *fuzzy*[6].

5) Quadro Alvo

Utilizado para anotar para onde a atitude está sendo expressa. É bom lembrar

---

[5] Uma frase substantiva pode ser apenas um substantivo, um pronome ou um grupo de palavras contendo um substantivo ou um pronome que funcionam juntos como o sujeito ou objeto de um verbo.

[6] Lógica *fuzzy*

que nem todas as atitudes têm um alvo. No caso da sentença “Rangel disse, Isto é uma idéia estúpida”, a atitude negativa se refere a palavra idéia.

As anotações de alvo são ligadas às suas anotações de atitude correspondentes.

Exemplo: Rangel disse, Isto é uma IDÉIA estúpida.

Na Figura 12 é apresentado um exemplo de anotação para a frase “Eu penso que as pessoas estão felizes porque Chavez caiu”.

Como se pode observar, o primeiro bloco desta anotação a fonte, o escritor, registra uma atitude do tipo argumentação positiva em relação ao alvo, neste caso “as pessoas”.

Já no segundo bloco a anotação é desmembrada em dois tipos de atitudes. A primeira retrata uma atitude positiva em relação à queda de Chávez e a segunda anotação de atitude mostra de maneira oposta uma atitude negativa, ou seja, um sentimento negativo em relação a Chávez.

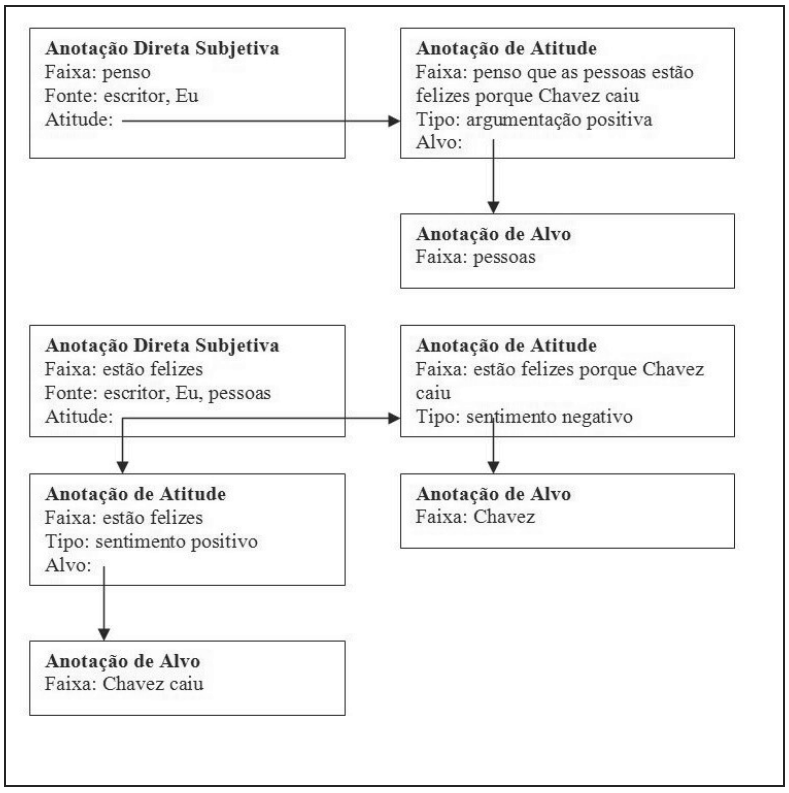

Figura 12: Esquema de anotação de uma frase utilizando o esquema de quadros.

5 PROPOSTA DE ESTRATÉGIA PARA ANÁLISE DE SENTIMENTOS

Pode-se resumir a estratégia proposta neste trabalho através dos seguintes passos:

- Criação do corpus anotado;
- Desenvolvimento de rotinas de extração e processamento de anotações específicas ao domínio;
- Criação de léxicos semânticos, expressões específicas ao domínio e outras listas com expressões de subjetividade e polaridade das palavras;
- Construção do classificador
    - Escolha das *features*;
    - Escolha do algoritmo a ser utilizado;
    - Parametrização do arquivo de configuração do classificador;
    - Avaliação dos resultados;
    - Salvar o classificador.
- Especialização para o domínio a ser analisado.

5.1 Visão Geral dos Experimentos

Todos os experimentos descritos a seguir tiverem como objetivo entender como os sentimentos são expressos em um domínio específico, estabelecer as *features* mais relevantes para a construção e avaliação de modelos de classificadores e também para o estudo das características mais importantes presentes nos textos relativos a um domínio sendo analisado.

Foram realizados dois experimentos para validar os passos propostos por este trabalho, para a avaliação dos classificadores produzidos e por fim para o melhor entendimento de como os documentos agrupados em um corpus se relacionam entre si.

Para a realização do primeiro experimento foram montados três corpus para o estudo do domínio Nuclear.

Diversos *sites*, *blogs* e fóruns da Internet foram pesquisados para se coletar textos com opiniões contra e a favor a área nuclear. Também foram coletados textos de outras fontes disponíveis na Internet. A partir dos textos retirados foi montado um corpus a favor da área nuclear, outro contra e por último, um conjunto específico de documentos com um viés tendencioso, mais político, que foram extraídos de *sites* ou publicações com um perfil mais ativista, como por exemplo, o site da organização ambientalista anti-nuclear *Greenpeace*[1].

Com a intenção de coletar um conjunto de documentos que não expressassem sentimentos sobre a área nuclear além dos que foram selecionados, também foi extraído um subconjunto de documentos referentes à área nuclear do corpus anotado MPQA[2]. O único critério de seleção destes documentos foi a presença da palavra nuclear no corpo do texto.

A anotação foi feita manualmente por dois meses e meio, onde foram anotados 281 documentos (contra, a favor, neutros e de ativistas). O processo de anotação realizado identificou e classificou inúmeras expressões e atribuiu valores as características relativas a estas expressões.

Para o segundo experimento foi utilizado o corpus produzido por PAK e PAROUBEK[3] (2010), o qual é composto 300.000 textos extraídos do *Twitter*, formando três conjuntos divididos uniformemente. Um com textos expressando sentimentos positivos, outro sentimentos negativos e o terceiro só com textos objetivos que não expressam opinião nenhuma.

5.1.1 Descrição do ambiente de desenvolvimento

A linguagem Java foi utilizada para a construção da ferramenta denominada JULGAR, desenvolvida neste trabalho. A sua escolha deve-se ao fato de ser uma linguagem amplamente usada, sua robustez e principalmente pela facilidade de

[1] http://www.greenpeace.org/international/

[2] Multi Perspective Question-Answer, http://www.cs.pitt.edu/mpqa/

[3] http://www.stanford.edu/~alecmgo/cs224n/twitterdata.2009.05.25.c.zip

integração com outros ambientes e APIs[4] (Interface de Programação de Aplicações). A descrição detalhada encontra-se no apêndice A.

O ambiente de desenvolvimento de software utilizado foi a plataforma Eclipse versão *Galileo* que é composta por um ambiente integrado de desenvolvimento (IDE), permite o uso de várias linguagens, tais como, Java, C e *Python* entre outras e pode ser extensível através da incorporação de vários tipos de ferramentas e plug-ins[5].

Para o processamento de linguagem natural foram utilizadas as APIs disponibilizadas pelo ambiente de processamento de textos GATE e o conjunto de bibliotecas Java para a análise lingüística de linguagem natural LingPipe[6].

Alguns módulos foram desenvolvidos primeiramente utilizando a linguagem *Python* devido a sua facilidade e posteriormente codificados em Java.

A linguagem JAPE (*Java Annotation Patterns Engine*) embutida no software GATE foi utilizada para parte do processamento dos textos analisados.

Dentre os principais *plugins* do Eclipse utilizados neste desenvolvimento destaca-se o *Visual Editor*[7] para a construção das telas e o UML2 *Tools*[8] para a construção dos modelos UML[9].

Para persistência dos dados foi utilizado o banco de dados PostgreSQL[10] versão 8.4, o software para recuperação de informação Lucene[11], o qual já vem integrado ao software GATE, ou simplesmente salvando os documentos no formato XML.

A ferramenta foi construída segundo o padrão de arquitetura de software MVC (Modelo-Visão-Controlador).

---

[4] *Application Programming Interface*

[5] Um *plugin* (também conhecido por *plug-in*, *add-in*, *add-on*) é um programa de computador usado para adicionar funções a outros programas maiores.

[6] http://alias-i.com/lingpipe/index.html

[7] http://www.eclipse.org/vep/

[8] http://www.eclipse.org/modeling/mdt/?project=uml2

[9] Unified Modeling Language

[10] http://www.postgresql.org/

[11] http://lucene.apache.org/

5.1.2 O software GATE

A seguir o software GATE é detalhado mais minuciosamente devido ao fato deste trabalho ser baseado principalmente nele quanto na anotação dos textos, no processamento e escolha das *features* e na construção e avaliação dos classificadores.

O software GATE é o maior projeto de código aberto para processamento de linguagem natural. O ambiente GATE disponibiliza ao desenvolvedor de software uma infra-estrutura completa para o desenvolvimento de componentes de software para os mais diversos tipos de aplicações.

Dentre as suas principais características pode-se citar:

- Suporte para: XML, HTML, PDF, MS Word, email, outros;
- Suporte para várias linguagens (de Urdu a Chinês);
- Modelagem e persistência de estruturas de dados específicas;
- Visualização e edição de anotações;
- Extração de instâncias para aprendizado de máquina;
- Algoritmos padrões (KNN, C4.5, *Naive Bayes*);
- Utiliza *plugins* para aprendizado de máquina: *Weka*, SVM *light*, *LingPipe*, ...;
- Linguagem específica para o processamento de anotações (JAPE).

O software GATE pode ser utilizado através da sua interface visual ou através do uso de suas APIs embutidas em um programa. Ele está dividido em três partes: Recursos de linguagem, Recursos de processamento e Aplicação.

Através do módulo de Recursos de Linguagem o usuário cria, define ou salva documentos, anotações, ontologias ou Corpus.

O módulo de Recursos de Processamento Lingüístico, que são na realidade algoritmos que atuarão sobre o texto, permite ao usuário escolher quais tipos de processamento irão ocorrer sobre os documentos.

Estão disponíveis vários tipos de processamento, tais como:

- Processamento de textos:
    - Segmentação do texto em *tokens* e sentenças;
    - Vários tipos de *Taggers* e *Parses*[12];
    - Analisadores morfológicos etc;
- Módulos de aprendizado de máquina;
- Construção de ontologias e
- Módulos para avaliação de concordância de anotações.

Uma observação importante sobre os recursos de processamento lingüísticos disponibilizados dentro do ambiente GATE é que vários deles não cumprem adequadamente a função que deveriam realizar. Por isto, é recomendável uma verificação criteriosa quanto na análise dos resultados obtidos, pois nem sempre dois recursos que deveriam funcionar da mesma maneira, funcionam como pretendidos.

Um exemplo é o recurso *ANNIE Sentence Splitter* que é usado para separar o texto em sentenças, mas que não funciona como esperado, devendo ser substituído pelo recurso de processamento de sentenças *RegEx Sentence Splitter.*

Estão listados na Tabela 6 alguns dos principais recursos de processamento lingüísticos do software GATE.

Nome	Função
Tokeniser	Separar o texto em unidades chamadas *token*.
RegEx Sentence Splitter	Separar o texto em sentenças.
POS tagger	Etiquetar os *tokens* em função da sua classe léxica.
Hash Gazetteer	Identificar e classificar através de listas de categorias pré-definidas
Morphological Analyser	Identificar a forma básica da palavra e seus afixos.

Tabela 6: Exemplo de alguns recursos de processamento lingüístico presentes no software GATE.

[12] Um *Tagger* cria um metadado sobre um *token* e um *Parser* determina a estrutura gramatical de um texto.

No recurso Aplicação é onde o usuário define os itens que farão parte do processamento e a ordem em que eles irão ocorrer sobre um documento ou um conjunto de documentos. Pode ser visto como fosse um tubo (*pipes*) no qual os documentos passam, são processados pelos vários tipos de recursos de processamento de acordo com a ordem que eles foram definidos.

A Figura 13 mostra a ordem de como quatro recursos de processamento definidos em uma aplicação serão aplicados sob um documento ou um Corpus.

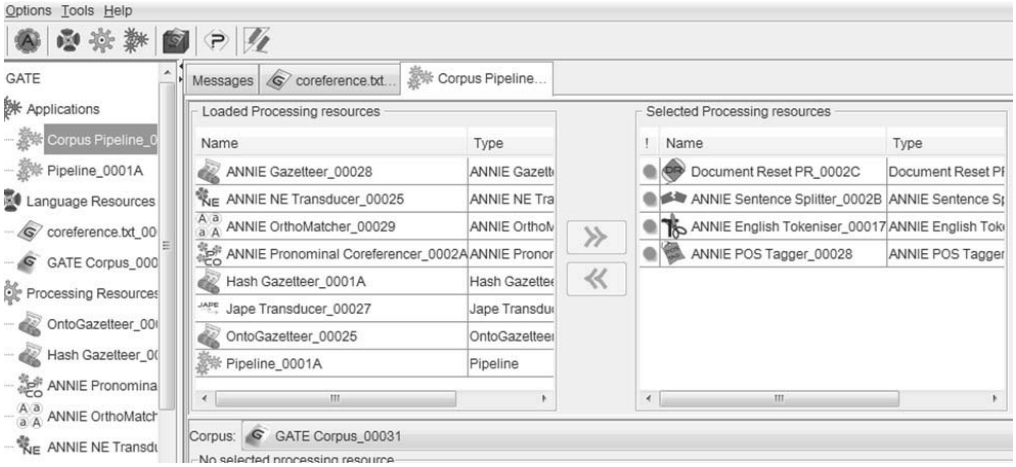

Figura13: Exemplo da aplicação de vários recursos de processamento sobre um texto

Para um melhor entendimento dos experimentos descritos nos próximos capítulos, é necessário se fazer a distinção entre alguns termos utilizados normalmente em outros ambientes e aplicações e como eles são definidos no ambiente GATE.

Embora o termo *feature* seja utilizado para atributo normalmente, não devemos confundir com as *features* presentes nas anotações descritas no GATE. Na Tabela 7 é apresentada uma descrição dos termos utilizados no ambiente GATE em relação a outros tipos de aplicação.

	Outros ambientes	GATE
Instância	Um exemplo do fenômeno sendo estudado.	Uma anotação.
Atributo	Uma característica das instâncias.	O valor de uma *feature* de uma anotação.
Classe	Um atributo, o qual, os valores são utilizados no processo de aprendizado de máquina.	Qualquer atributo referindo a instância corrente pode ser marcado como um atributo de classe.

Tabela 7: Descrição das definições de termos utilizados no ambiente GATE.

A linguagem JAPE presente no software GATE foi utilizada para extrair e processar informações específicas dos textos sendo analisados. Através dela expressões regulares são criadas para serem usadas sobre as anotações contidas nos documentos. Utiliza regras da forma LHS / RHS[13], onde LHS é o padrão a ser encontrado e RHS é a ação a ser realizada.

Na Figura 14 é apresentado um exemplo de uma regra programada na linguagem JAPE para criar uma anotação.

Por exemplo, considerando a regra JAPE a seguir:

```
({Token.string == Fortaleza}) :loc
→
:loc.Local = {tipo = cidade, regra = Localizacao}
```

Figura 14: Regra para anotação de localidades utilizando a linguagem JAPE

[13] Left Hand Side / Rigth Hand Side

Temos na primeira parte desta regra o padrão a ser encontrado. Neste caso quando uma anotação do tipo *token* contiver o valor da *feature string* igual a "Fortaleza" a segunda parte da regra será executada. Quando isto ocorrer será criada uma anotação do tipo "Local" com as *features* "tipo" e "regra" com os valores "cidade" e "Localização" respectivamente.

Um dos componentes mais importantes do GATE é o *Gazetteer* cujo papel é identificar nomes de entidades. Os *Gazetteers* são listas com uma entrada por linha, onde cada linha pode representar uma cidade ou uma organização. Todas as listas que são processadas pelo *Gazetteer* estão definidas no arquivo *lists.def*.

Para cada lista definida no arquivo *lists.def* são atribuídos dois parâmetros (*majorType* e *minorType*) cujo valores serão incorporados as anotações produzidas[14] pela a aplicação do componente *Gazetteer* sobre o corpus.

5.2 Criação do corpus anotado

Os algoritmos de aprendizado de máquina utilizados neste trabalho são algoritmos de aprendizado supervisionado e por isto há necessidade de fornecer para eles exemplos rotulados, anotados para a realização do aprendizado. Para tal é necessário se ter um corpus anotado.

Há dois tipos de anotações, a primeira é a anotação que é dada como exemplo para os classificadores. Neste caso, poderia ser qual o tópico no qual o documento está classificado (esporte ou finança).

O outro tipo de anotação é formado por um conjunto de anotações produzido pela aplicação de um dos recursos de processamento lingüístico presente no ambiente GATE.

Estas anotações podem ser criadas também pelo processamento delas por uma ou mais regras desenvolvidas através da linguagem JAPE, as quais podem extrair parte ou um todo de uma anotação e converter as informações lidas no tipo de anotação necessário para aplicação sendo desenvolvida.

Será através destas anotações que o algoritmo fará o aprendizado de máquina sobre o corpus fornecido.

Foram criadas várias listas para criação de anotações para a análise de

[14] Anotação do tipo Lookup.

sentimentos dos textos sendo analisados. Algumas delas são descritas na Tabela 8.

O corpus usado pelos classificadores foi composto de documentos anotados como sendo subjetivos ou objetivos. Para tal foi desenvolvido um algoritmo utilizando a linguagem JAPE para criar dois atributos para serem inseridos nas anotações do tipo *Sentence*[15].

O primeiro atributo chamado de "*sent_size*" tem como valor o número de caracteres da sentença. O outro atributo chamado de "*class*" com o valor "*subj*" para sentenças subjetivas e "*obj*" para sentenças objetivas. O atributo '"*class*" é o atributo que rotula os exemplos que serão utilizados pelos classificadores.

Nome	Descrição	Tipo mais genérico	Tipo mais específico
badwords	Lista com palavras chulas	word	negative
adjetivos	Lista com adjetivos	adjetivo	-------
dominiopositivo	Lista com expressões positivas do domínio nuclear.	dominio	positive
dominionegativo	Lista com expressões negativas do domínio nuclear.	dominio	negative
stopwords	Lista com palavras comuns	stop	-------
nounpositive	Lista com palavras com polaridade positiva	word	positive
nounnegative	Lista com palavras com polaridade negativa	word	negative

Tabela 8: Listas criadas para identificação automática de palavras ou expressões para a análise de sentimentos

[15] Tipo de anotação que corresponde a uma sentença presente no texto.

Na Figura 15 é mostrada uma regra escrita usando a linguagem JAPE para rotular as sentenças como sendo do tipo subjetivo.

Nesta regra, a primeira linha estabelece o nome da regra. As linhas de 2 a 4 são aonde é definido o tipo de anotação que se quer anotar, no caso anotações do tipo *Sentence* e o nome da variável de ligação (“frase”).

A linha sete é criada uma variável do tipo *AnnotationSet*, chamada também de “frase”, que recebe a variável de ligação. Na linha oito é criada uma variável do tipo *Annotation* que armazenará os valores das anotações.

Na linha nove é criada uma variável do tipo *FeatureMap*, *features*, a qual é usada para se adicionar uma nova informação em uma anotação. A linha dez coloca na variável *feature* uma nova anotação, *class* com o valor *subj*.

Por fim, na linha onze os novos valores são inseridos nas anotações do tipo *Sentence*.

Para se construir o corpus anotado para treinamento do classificador de polaridade foram desenvolvidos alguns algoritmos (ver apêndice B) para selecionar do corpus anterior apenas as sentenças rotuladas como subjetivas.

Estas sentenças selecionadas são então rotuladas como positivas ou negativas. Do mesmo modo como foi feito no passo anterior é criado para a anotação do tipo *Sentence* com os atributos “sent_size” e “*class*”, este último com os valores “pos” (positivo), “neg” (negativo) ou neutro.

```
1  Rule: Marcasentenca
2  (
3    {Sentence}
4  ):frase
5  -->
6  {
7    gate.AnnotationSet frase = (gate.AnnotationSet)bindings.get("frase");
8    gate.Annotation fraseAnn = (gate.Annotation)frase.iterator().next();
9    gate.FeatureMap features = Factory.newFeatureMap();
10   features.put("class", "subj");
11   outputAS.add(frase.firstNode(), frase.lastNode(), "Sentence",features);
12 }
```

Figura 15: Regra JAPE para anotar uma sentença como Subjetiva

Na Tabela 9 estão alguns sentimentos expressos no corpus nuclear que foram extraídos para melhor entendimento.

	Tipos de *Private State*	
	Expressive-subjectivity	Direct-subjective
Exemplos	absolutely urgent	can afford
	brought back	censored
	sustainable culture	collapse
	delays and regulatory battles persist	gearing up
	History sounds a cautionary note	opposed
	more clean	promoted
	no tomorrow	unacceptable

Tabela 9: Exemplo de expressões de private states retiradas do corpus nuclear anotado

5.3 Criação dos léxicos semânticos

Para se realizar a anotação automática das palavras com polaridade positiva e negativa, foi utilizado um subconjunto da lista de palavras que expressam subjetividade criada por WILSON *et al.* (2005).

Esta lista caracteriza cada palavra por sua polaridade (positiva, negativa, ambas ou neutra), pela sua indicação de subjetividade em relação a um contexto (*weaksubj* e *strongsub*[16]) entre outras características.

A partir dela foram criadas duas listas contendo palavras com polaridade positiva e negativa as quais foram utilizadas no processo de anotação.

Para este trabalho foram suprimidas as palavras que apresentassem polaridades neutras ou ambas e as palavras com indicação de subjetividade classificada como fraca para aumentar a precisão do classificador de subjetividade.

Em seguida foram configurados os ambientes para o domínio nuclear utilizando os softwares AutoSlog-TS e BASILISK.

Foi criada também, uma lista contendo palavras que tipicamente aparecem em textos contra a energia nuclear, tal como o nome da usina nuclear russa *Chernobyl* onde ocorreu um acidente nuclear em 26/4/1986.

5.4 Construção dos classificadores

Ao analisar um documento para saber como um sentimento está sendo expresso, necessitamos aplicar sobre ele um classificador que identifique se o sentimento em questão é positivo, negativo ou mesmo neutro a respeito de um determinado assunto.

Para desempenhar esta tarefa foi adotada a estratégia sugerida por PANG e LEE (2004), os quais propõem como primeiro passo separar as sentenças em sentenças subjetivas e objetivas e só depois classificar apenas as sentenças subjetivas para se obter a sua polaridade. Neste caso as sentenças objetivas serão consideradas com polaridade neutra.

Para tal foram desenvolvidos dois classificadores. O primeiro classificador tem como objetivo classificar as sentenças como sendo subjetivas ou objetivas. O segundo

[16] Palavras que foram classificadas como fortes ou fracos indicadores de subjetividade.

classificador é aplicado apenas sobre as sentenças subjetivas e tem como função classificá-las segundo a sua polaridade. Tal estratégia permite se conseguir melhores resultados do que aplicar diretamente sobre todos os documentos um único classificador de polaridade.

Os classificadores foram desenvolvidos utilizando o componente para aprendizado de máquina disponibilizado no ambiente GATE. O componente é o *Batch Learning PR* que permite o desenvolvimento e teste de um classificador de maneira fácil, bastando apenas definir alguns parâmetros.

Este componente quando aplicado a um conjunto de documentos produz além do modelo de linguagem baseado em *n-gramas*, o classificador em questão, vários arquivos de texto com informações sobre o treinamento realizado. Como exemplo pode-se citar os arquivos com vetores de *features*, *features* lingüísticas e matrizes documento-termo e um arquivo com os rótulos utilizados no treinamento (caso existam). As informações contidas nestes arquivos de texto podem ser utilizadas fora do ambiente GATE em outras aplicações.

O modelo de linguagem resultante pode ser, por exemplo, uma árvore de decisão ou um conjunto de regras, dependendo do tipo de algoritmo utilizado para o processamento lingüístico.

Os parâmetros para o *Batch Learning PR* são definidos por meio de um arquivo de configuração tipo XML descrito na Figura 16. Este arquivo apresenta duas seções principais. Na primeira é definido qual algoritmo será utilizado para processar os textos (SVM, *Naive Bayes*, PAUM, outros).

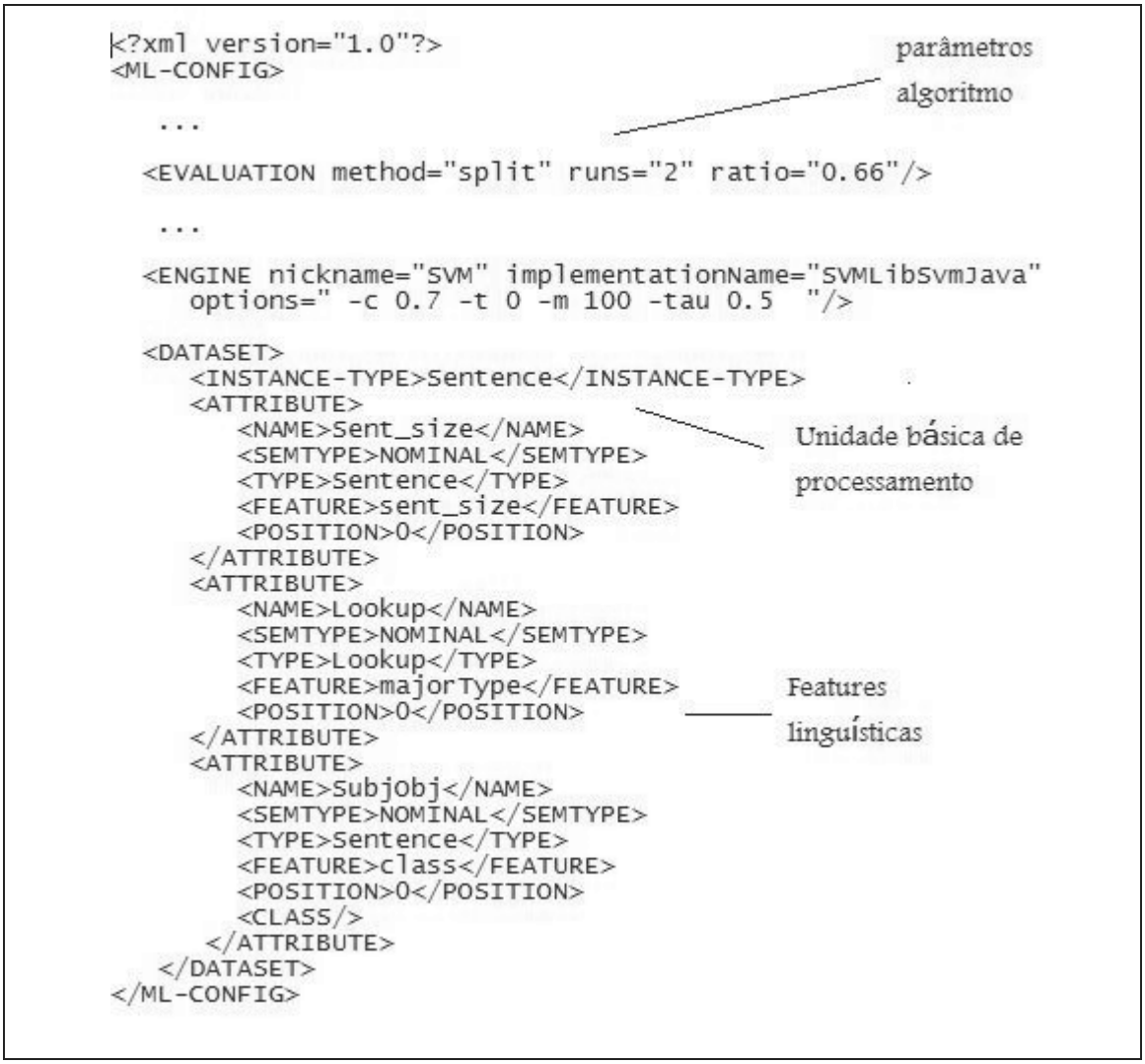

Figura 16: Arquivo de configuração do classificador de subjetividade.

Os dados lingüísticos são definidos na segunda seção. Nesta seção é estabelecido qual será a unidade para ser processada, por exemplo, anotações do tipo *Token*. Ou seja, estabelece que todas as anotações do tipo *Token* presentes no documento serão consideradas instâncias de treino.

Também é definido nesta seção um conjunto de atributos que irão caracterizar as instâncias de treinamento.

O componente possui vários modos de aprendizado. A Figura 17 apresenta os principais modos.

No modo de treino (*TRAINING*) o componente *Batch Learning PR* aprende a partir dos dados fornecidos e produz um classificador, o qual é salvo em um subdiretório chamado "*learnedModels.save"*.

Para avaliarmos o classificador produzido devemos escolher o modo de avaliação (*EVALUATION*) que realizará dependendo de como foi especificado no arquivo de configuração um teste *k-fold* ou *hold-out*[17].

Os resultados são apresentados por meio de várias métricas (*Precision*, *Recall* e *F-measure*) na janela de mensagens do GATE ao término da avaliação.

Onde:

- *Precision* mede quantos dos itens que o sistema identificou são realmente corretos sem considerar que o sistema falhou ao recuperar alguns itens corretos;
- *Recall* mede quantos dos itens que deveriam ter sido realmente identificados foram identificados sem considerar quantas identificações espúrias foram feitas.
- *F-measure* ou *F1 score* é a média harmônica de *precision* e *recall*.

Para um melhor entendimento destas medidas é apresentado na Tabela 10 um exemplo de uma matriz de confusão contendo informações sobre as classificações reais e as preditas feitas por um classificador. As equações 5, 6 e7 de acuracidade, *precision* e *recall* são mostradas adiante.

		Predita	
		negativa	positiva
Real	negativa	a	b
	positiva	c	d

Tabela 10: Matriz de confusão com as classificações reais e preditas feita por um classificador.

[17] Tanto *k-fold* e *hold-out* são métodos de validação cruzada, que possuem uma maneira específica para dividir os dados em um conjunto de validação e outro conjunto de treino.

Onde:

a: número de classificações negativas corretas

b: número de classificações positivas incorretas

c: número de classificações negativas incorretas

d: número de classificações positivas corretas.

Acuracidade: $\frac{a+d}{a+b+c+d}$ Equação 5: Acuracidade

precision: $\frac{d}{b+d}$ Equação 6: *Precision*

recall: $\frac{d}{c+d}$ Equação 7: *Recall*

Já o modo de aplicação (*APPLICATION*) lê o modelo salvo e aplica-o sobre os dados sendo analisados.

Outro modo disponível é o MITRAINING que pode ser usado para aprendizado on-line. Neste modo, os novos exemplos, documentos são adicionados ao arquivo de *features* já salvos dando continuidade ao treinamento.

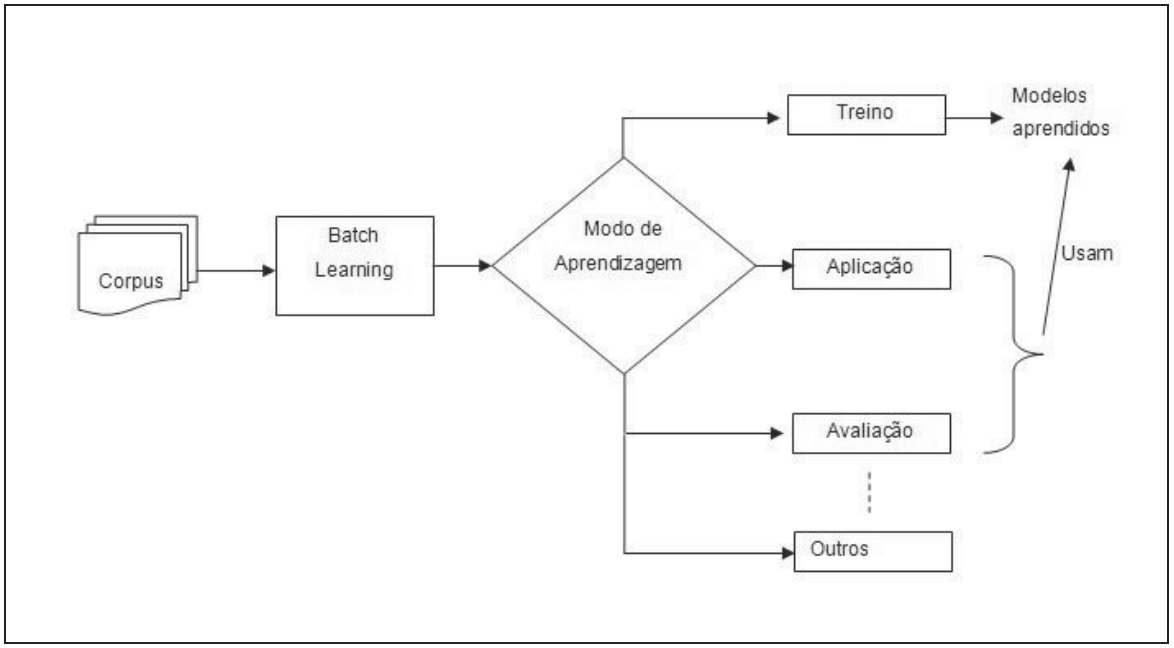

Figura 17: Os modos de aprendizagem do componente Batch Learning PR

Em resumo, para se utilizar o componente *Batch Learning PR* é necessário seguir os seguintes passos:

- Anotar alguns documentos com rótulos que se quer que o sistema de aprendizado anote os documentos novos;
- Pré-processar os documentos para se obter as *features* lingüísticas para o aprendizado. Estas features têm que ser da forma de anotações produzidas no ambiente GATE;
- Criar o arquivo de configuração;
- Configurar o componente e executá-lo;

Deve-se ressaltar que a escolha do tipo de *features* lingüísticas e do algoritmo utilizado dependerá do tipo de aprendizado de máquina que ser quer realizar. O ambiente computacional GATE permite três tipos de categorias.

Elas são:

- Classificação de textos;
- Reconhecimento de entidades (*Chunck recognition*) e
- Extração de relações.

Na Tabela 11 é apresentado para cada categoria o tipo de instância e as *features* lingüísticas que devem ser utilizadas.

	Tipo de instância	*Features* lingüísticas
Reconhecimento de entidades	Token	*Features* lingüísticas do *token* e outras anotações como *features*.
Classificação de textos	Unidade de texto para ser classificada	Documento, sentença ou *token*. No caso de sentença usar n-grama.
Extração de relações	Par de termos para serem relacionados	*Features* lingüísticas de cada um dos termos além das *features* lingüísticas relacionadas a ambos os termos.

Tabela 11: Tipo de instância e features lingüísticas utilizadas para cada tipo de categoria de aprendizado.

Por fim, para este trabalho de classificação de textos foram selecionados os seguintes parâmetros;

- Algoritmos: SVM e PAUM;
- Tipo de Instância: Sentença (Sentence);
- *Features* lingüísticas: *Token*, *Lookup* e *Spertus* (Descrita mais adiante);
- Tipo de n-gram: unigrama e bigrama:
- Tipo do valor dos n-gramas: TF*IDF[18]

A escolha dos dois algoritmos deve-se ao fato do SVM já ter se mostrado consistentemente como um algoritmo melhor do que os outros na classificação de textos (AUE, 2005, YE, 2005).

Para o uso dos algoritmos SVM, o software GATE disponibiliza duas implementações dos pacotes LibSVM[19] e o pacote SVM[light 20] .

Estes dois pacotes oferecem como opção para a função *kernel* o tipo linear e polinomial que são as mais utilizadas.

A escolha do algoritmo PAUM deve-se ao desejo de se fazer uma comparação entre ele e o algoritmo SVM, pois ele, como foi relatado anteriormente (YAOYONG, 2002), é um algoritmo simples e rápido que apresenta um desempenho similar ao algoritmo SVM, mas com tempo de treinamento muito menor.

Uma implementação do algoritmo PAUM também é disponibilizada no ambiente GATE.

A Figura 18 apresenta uma visão do fluxo do processo de criação de um classificador utilizando o ambiente GATE. Começa com os documentos sendo inseridos no recurso de linguagem Corpus

Depois são criados vários tipos de anotações pela aplicação dos recursos de processamento lingüístico definidos pelo usuário, as quais servirão como dados para o

---

[18] É uma medida estatística para avaliar a importância da palavra no documento

[19] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[20] http://svmlight.joachims.org/

aprendizado dos classificadores, cujos parâmetros (algoritmo e as *features* lingüísticas) a serem usados estão definidos no arquivo de configuração XML.

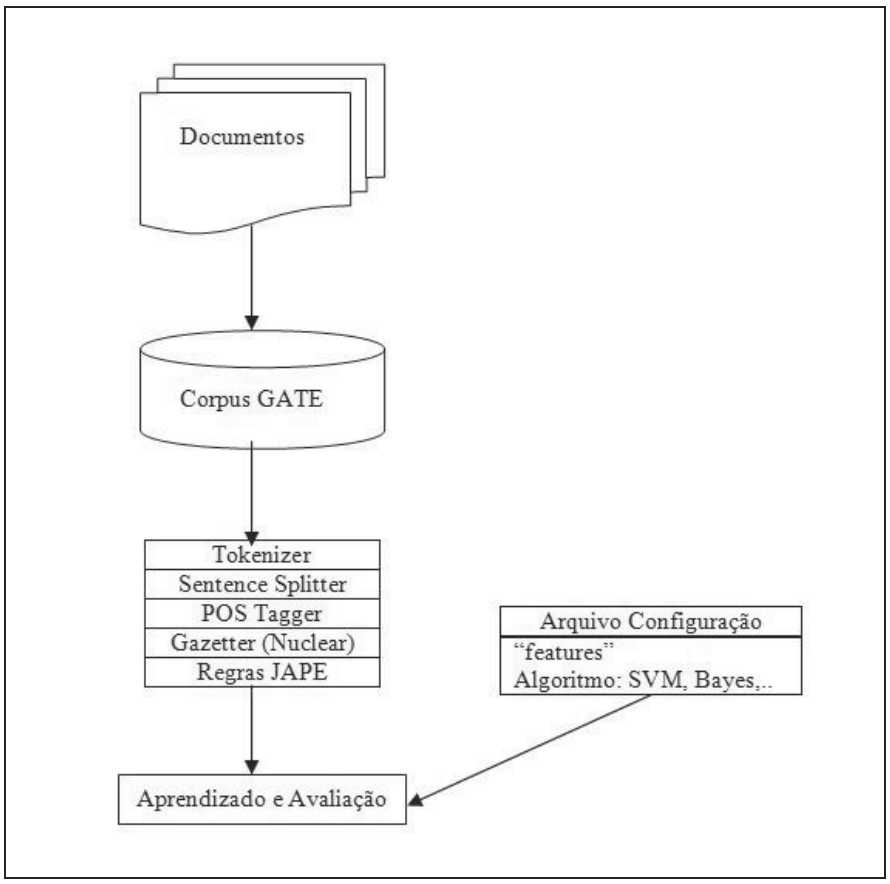

Figura 18: Fluxo de preparação dos documentos para criação do classificador

Várias *features* lingüísticas foram utilizadas para o processamento das sentenças sendo analisadas. Como visto na Tabela 11 para o processamento de sentenças deve-se se usar uma *feature* do tipo n-grama.

A importância de se utilizar um esquema do tipo *feature* n-grama é que os classificadores podem identificar a presença de palavras ou expressões que circundam o *token* sendo analisado, as quais podem alterar ou intensificar o sentido real que a sentença possui (*Contextual Valence Shifters*, POLANYI, 2004).

Por exemplo, as frases a seguir podem ter seu sentido alterado ou intensificado pela utilização destas palavras circundando as palavras que exprimem um *private state* dentro de uma sentença.

Ex: “It [the remote] is not as fast as a usual remote control”.

Ex: “John is less brilliant than Paul”;

Para estabelecer uma *feature* do tipo n-grama no ambiente GATE é necessário utilizar um tipo de *feature* definida no arquivo de configuração XML chamada NGRAM.

Este tipo de *feature* é usado para caracterizar uma instância de anotação em termos das seqüências de anotações de *features* que a compõem. Ou seja, caracteriza as anotações que são constituídas por mais de uma instância de anotação, como é o caso de uma sentença.

Como uma sentença é constituída de uma seqüência de *Tokens*, o NGRAM definido é composto de anotações do tipo *Token* com as *features* dos tipos *kind*, *orth* ou *category*.

A feature *kind* tem como valor se o *token* sendo analisado é, por exemplo, uma palavra, um número ou uma pontuação.

Já a *feature orth* indica se o *token* está em minúsculo ou se a primeira letra dele está em maiúsculo e finalmente o tipo *category* tem como valor o PoS do *token*.

Também foram criadas anotações baseadas nas observações sociolingüísticas listadas em SPERTUS (1997) para auxiliar os classificadores com mais uma informação para identificação de expressões de subjetividade.

A elas foram adicionadas algumas regras específicas para identificação de expressões típicas do domínio *Twitter*.

No *Twitter* os usuários enviam mensagens conhecidas como *tweets*. Estas mensagens têm características únicas. Elas, na sua grande maioria, são compostas de uma única sentença e utilizam símbolos e abreviações próprias. Por exemplo, é comum encontramos em um *tweet* as letras *RT*[21] que é uma abreviação de *ReTweet*, o qual indica que a mensagem deve ser retransmitida.

É comum também, encontrarmos nos *tweets* palavras escritas de maneira imprópria, na qual as letras do meio de uma palavra são digitadas inúmeras vezes para reforçar uma opinião. Por exemplo: "*I looooooooooooove you*".

A Figura 19 apresenta uma regra escrita na linguagem JAPE para identificar palavras escritas pelos usuários do *Twitter*, que usam repetição de letras no interior de uma palavra para enfatizar uma emoção.

[21] Para saber mais consultar http://www.brentozar.com/archive/2008/08/twitter-101/

```
Rule:twitter

Priority:50
(
  (
    {Token.string=~"[a-z]+a{3,8}[a-z]+"}|
    {Token.string=~"[a-z]+e{3,8}[a-z]+"}|
    {Token.string=~"[a-z]+i{3,8}[a-z]+"}|
    {Token.string=~"[a-z]+o{3,8}[a-z]+"}|
    {Token.string=~"[a-z]+u{3,8}[a-z]+"}
  ) |
  (
     ({Token.kind==punctuation, !Token.string=~"[\\.]" })[4,8]
  )
):rotulo
-->

:rotulo.spertus={rule="twiteer"}
```

Figura 19: Regra para identificar palavras com letras repetidas dentro dela.

Foi aplicado um processo de filtragem sob os textos dos *tweets,* onde foram removidos nomes de *links URL*, nomes de usuários *Twitter* , que são da forma *@Paulo* (o símbolo @ indica o nome do usuário) e *emoticons*[22].

Uma das regras escrita utilizando a linguagem JAPE foi baseada nas observações de PAK e PAROUBEK (2010) que identificaram através da análise de dois conjuntos de *tweets* classificados por polaridade (positivo e negativo) que os conjuntos de *tags POS* atribuídos aos textos não eram distribuídos uniformemente.

Assim cada conjunto de *tag* POS relativo aos conjuntos pode ser utilizado como indicador de polaridade da mensagem, *tweet*.

Por exemplo, a Tabela 12 mostra as conclusões de PAK e PAROUBEK (2010) relacionando as *tags POS* ao tipo de sentença.

[22] Um *emoticon* é uma expressão textual representando o estado de espírito do escritor da mensagem. Por exemplo, temos " :) " ou " : - (".

Sentença			
Objetivo		Subjetivo	
Regra observada	Tags	Regra observada	Tags
Contêm mais nomes comuns e próprios.	NPS, NP, NNS	Os autores escrevem sobre si mesmos ou referem-se a uma audiência (segunda pessoa).	VBD
Adjetivos comparativos são usados para declarar fatos.	JJR	Superlativos adjetivos são usados para expressar emoções e opiniões.	JJS
Verbos são usados na terceira pessoa.	VBN	Declarações são fortes indicadores de subjetividade.	UH
Verbos são mais usados no particípio passado.	VBZ	Contêm mais pronomes pessoais.	PP

Tabela 12: Lista de *tags* POS indicadoras de subjetividade e objetividade.

Onde:

NPS = Nome próprio plural;

NP = Substantivo próprio singular;

NNS = Substantivo plural;

JJR = Adjetivo Comparativo;

JJS = Adjetivo superlativo;

VBD = Verbo, passado simples;

VBN = Verbo, particípio passado;

VBZ = Verbo, terceira pessoa, singular, presente;

UH = Interjeição;

PP = Frase preposicional.

A Tabela 13 apresenta alguns tipos de anotações, chamadas neste trabalho de anotações SPERTUS, codificadas usando a linguagem JAPE, as quais anotam nos textos sendo analisadas expressões listadas por SPERTUS (1997) e os novos tipos propostos para identificar de como os usuários do *Twitter* registram suas emoções.

Features	Exemplo
Frases substantivas usadas como aposição[23]	you guys, you people
Sentenças iniciando com *you*, *your* ou *yourself*.	your so-called
Declarações imperativas	Have your fun
Vilões	Chernobyl
Palavras chulas ou obscenas	suck
Epítetos (frases curtas)	O verbo *get* seguido 10 caracteres

Tabela 13: Algumas das *features* sugeridas por SPERTUS (1997).

Outros tipos de anotações também foram utilizados pelos classificadores, como a anotação *Lookup* criada pela aplicação do componente *Gazetteer* sobre os textos que compõem o corpus juntamente com a anotação SPERTUS descrita anteriormente.

[23] Construção gramatical na qual dois elementos são colocados lado a lado, com um deles servido para definir ou modificar o outro

6 RESULTADOS

Os testes foram realizados utilizando os corpora criados para este trabalho e outras bases de dados criadas especialmente com intenção de mostrar algumas das funcionalidades que o sistema JULGAR possui. O apêndice A apresenta as principais funções do sistema JULGAR.

A avaliação dos classificadores de subjetividade e polaridade foi feita dentro do ambiente GATE utilizando o componente *Batch Learning PR* configurado com o parâmetro *EVALUATION*. Ao final do treinamento são fornecidas várias métricas sobre o treinamento realizado, como por exemplo, *recall* e *precision*.

Além das métricas, também são disponibilizados vários arquivos com informações relativas ao treinamento, como por exemplo, vetores de *features* e matriz documento-termo.

Os classificadores que foram incorporados ao sistema JULGAR foram testados com alguns conjunto de documentos previamente selecionados e também com textos recentes retirados de sites como o do Greenpeace.

A seguir são apresentados vários resultados acompanhados de comentários.

Dentre as funções disponibilizadas para o usuário destaca-se a capacidade de avaliação de similaridade entre os documentos do corpus.

A Figura 20 apresenta um exemplo de dendrograma[1] criado para um subconjunto do corpus Nuclear agrupando os documentos por similaridade de acordo com o tipo de *link* selecionado.

O dendrograma apresentado usa a notação de árvore binária endentada, no qual, é mostrado para cada agrupamento o valor da distância em que os documentos são agrupados.

Como se pode ver pela análise dos dendrogramas, os agrupamentos produzidos por *link* simples são mais profundos enquanto os produzidos por *link* completo são mais balanceados.

---

[1] Um dendrograma é um tipo de representação gráfica ou diagrama de dados em forma de árvore que organiza os dados em subcategorias que vão se dividindo em outras até chegar ao nível de detalhe desejado.

Por exemplo, considerando o dendrograma da esquerda da Figura 20, vemos que os documentos “nuclear002.txt” e “nuclear003.txt” são agrupados com o valor de distância 0.38. E o documento “nuclear001.txt” é agrupado a eles com uma distância de 0.47.

Uma das funções do sistema JULGAR é a análise de co-ocorrência dos *token,* palavras presentes nos documentos em relação a uma lista de itens que definem uma categoria.

Este tipo de análise é útil, pois mostra quais as palavras são mais utilizadas em função de um item da categoria. Estes itens podem ser nomes de políticos concorrendo a uma eleição ou marcas de automóveis.

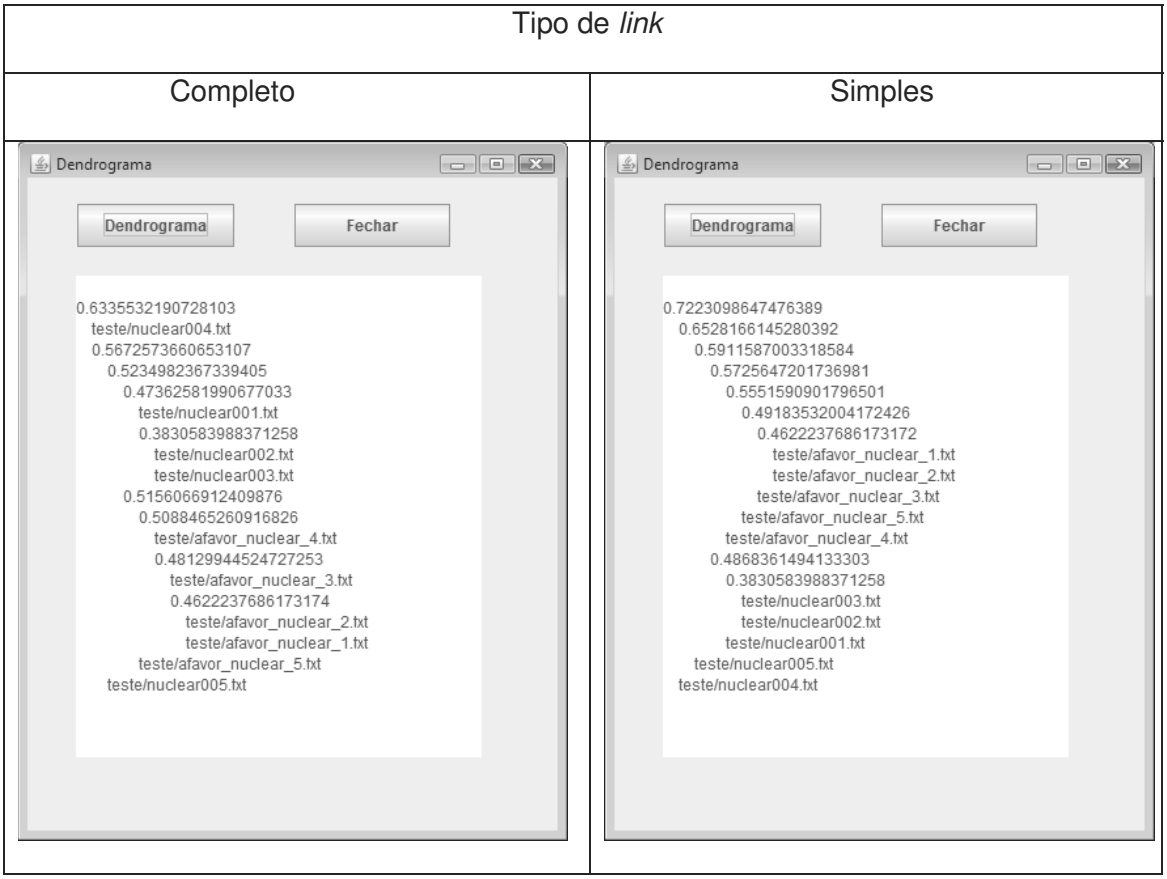

Figura 20: Exemplo de dois dendrogramas utilizando os tipos de link completo e simples.

Por exemplo, para se entender quais palavras presentes no corpus sendo Nuclear estão mais associadas a um determinado país, foi criada uma categoria contendo vários nomes de países.

Neste experimento foram criados dois corpora, um com documentos a favor e outro com documentos contra a energia nuclear.

Examinando a Figura 21 e a Figura 22 vemos que certas palavras são mais associadas a um determinado país dependendo da conotação positiva ou negativa do documento.

É o caso que constatamos ao examinar os resultados. Vemos que a palavra *Clamshell* (*Clamshell Alliance*, que é uma organização anti-nuclear) aparece em várias vezes em textos contra a energia nuclear quando associado à França (France) e não aparece nos textos a favor.

Esta análise fornece subsídios para identificar as palavras mais utilizadas em um contexto. Tal conhecimento permite traçar estratégias e priorizar quais pontos são mais importantes sobre um determinado assunto.

France	China	USA
nuclear=17	pounds=3	nuclear=16
French=5	kilograms=3	French=4
issues=4	person=2	energy=3
energy=4	emissions=2	technology=2
technologies=3	release=1	radiation=2
simply=3	people=1	plants=2
problem=3	nations=1	people=2
plants=3	involved=1	mongering=2

France	China	USA
nuclear=92	nuclear=65	national=7
energy=33	energy=25	victory=5
reactors=32	reactor=23	energy=5
reactor=32	reactors=20	atomic=4
French=31	French=15	reactor=3
Clamshell=23	plants=13	likely=3
radioactive=18	electricity=12	guarantee=3
industry=18	construction=	campaign=3

Figura 21: Tabela com palavras relacionadas aos nomes de países retirados de documentos favoráveis a energia nuclear.

Figura 22: Tabela com palavras relacionadas aos nomes de países retirados de documentos contrários a energia nuclear.

Outro tipo de análise de conteúdo presente no sistema é a análise de co-ocorrência de palavras.

Como mostrado na Figura 23, a palavra "*working*" co-ocorre com as palavras "*nuclear*" e "*liability*" em dois documentos e a palavra "*workers*" co-ocorre com elas em apenas um documento.

Nesta opção o usuário pode encadear várias palavras como dado de entrada para análise de co-ocorrência com as palavras contidas no corpus. Caso tenhamos as palavras "*nuclear*", "*liability*" e "*America*" teríamos como resultado que a palavra "American" co-ocorre com estas três palavras em três documentos

Alguns benchmarks foram realizados para avaliar os sentimentos sendo expressos.

Foram analisados vários conjuntos com textos com polaridade positiva e negativa e textos retirados de sites relativos ao setor nuclear.

A Figura 24 apresenta o resultado da aplicação da opção do menu "Análise de sentimentos" feita sobre um texto retirado do site do Greenpeace. Como esperado o documento é classificado como tendo polaridade negativa.

Figura 23: Análise de co-ocorrência de palavras presentes nos documentos do corpus

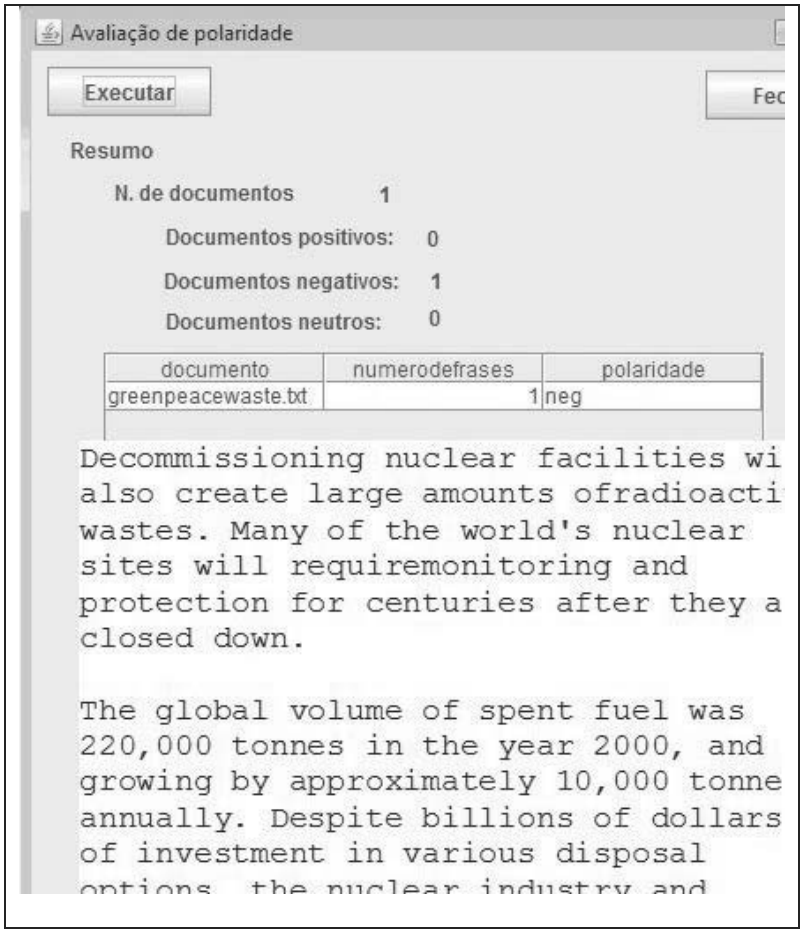

Figura 24: Cálculo da polaridade de um texto extraído do site do Greenpeace.

Para uma melhor compreensão de que *features* lingüísticas e algoritmos que deveriam ser escolhidos, além de se testar o desempenho dos algoritmos PAUM versus SVM foram feitos vários experimentos executados sobre os corpora do domínio nuclear e o corpus adquirido do *Twitter*.

Várias features lingüísticas foram testadas. Na maioria dos testes manteve-se a *feature sent_size* a qual é utilizada no exemplo de classificação de textos no manual do software GATE.

As avaliações com uma rara exceção foram feitas se utilizando unigramas. Embora tenha se optado por utilizar unigrama, algumas *features* lingüísticas utilizadas foram definidas com várias informações lingüísticas tais como POS, presentes em *tokens* que circundavam o *token* corrente sendo analisado.

O objetivo de tal configuração foi possibilitar aos classificadores considerarem no processo de aprendizagem a presença de negadores e intensificadores que podem

alterar o sentido do *token* sendo analisado.

A seguir são apresentados os resultados utilizando-se os algoritmos SVM e PAUM com um conjunto de *features* diferentes.

Na Tabela 14 estão listados os resultados dos classificadores de subjetividade usados no primeiro passo de processamento para o cálculo da polaridade dos documentos.

O melhor resultado foi obtido quando foi considerada a utilização de bigramas. O classificador SVM apresentou os melhores resultados comparado com o classificador PAUM.

Os resultados apresentados por YAOYONG *et al.* (2002) não puderam ser comparados diretamente com os resultados obtidos neste trabalho, pois foram obtidos para categorização de documentos e não para análise de sentimentos.

Contudo os resultados obtidos neste trabalho podem ser comparados com os resultados obtidos por YAOYONG *et al.* (2002) quando eles consideraram para treinamento todas as categorias para classificação.

Algoritmo	n-grama	Features	Precision	Recall	F1
SVM	1	sent_size	0.70617044	0.70617044	0.70617044
PAUM	1	sent_size	0.50538254	0.50538254	0.5048944
SVM	2	sent_size	0.79115665	0.6973555	0.6973555
SVM	1	spertus	0.7252693	0.7252693	0.7252693
PAUM	1	spertus	0.501231	0.5009794	0.5011052

Tabela14: Resultados dos classificadores de subjetividade

A Tabela 15 apresenta a comparação entre os classificadores de polaridade utilizando-se os algoritmos SVM e PAUM. Neste exemplo constatou-se uma proximidade dos resultados entre os dois classificadores como era de se esperar. Nota-se também um pequeno ganho quando se utilizando as *features* SPERTUS com o algoritmo SVM produzindo o melhor resultado.

Algoritmo	n-grama	Features	Precision	Recall	F1
SVM	1	sent_size	0.7946027	0.7946027	0.7946027
PAUM	1	sent_size	0.78284216	0.78284216	0.78284216
SVM	1	Lookup	0.79115665	0.79115665	0.79115665
SVM	1	Spertus	0.7957587	0.7957587	0.7957587
PAUM	1	Lookup	0.7909393	0.7904676	0.7907033
PAUM	1	Spertus	0.78335154	0.78288656	0.7831189

Tabela 15: Resultados dos classificadores de polaridade

Para se obter os resultados conseguidos, a estratégia apresentada neste trabalho necessita da análise do corpus que representa o domínio. Esta análise pode ser parcial quando da falta de exemplos. Neste caso, a utilização das técnicas de adaptação de domínio apresentadas anteriormente devem ser utilizadas. Os resultados óbitos pelos módulos de identificação de co-ocorrência devem servir de apoio aos dados de entrada, para a criação do léxico semântico referente ao domínio sendo modelado.

7 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho tem como principal contribuição estabelecer uma estratégia para se realizar análise de sentimentos baseada em domínio.

Para validar tal estratégia foi desenvolvido um ambiente computacional, denominado JULGAR, totalmente configurável por domínio, o qual permite que sejam escolhidos de acordo com a aplicação: inclusão de classificadores treinados externamente, léxicos semânticos, lista de “stopwords”, listas com palavras com polaridade positiva e negativa referente ao domínio e configurações de tipos de termos a serem analisados e processados.

O sistema JULGAR pode rodar em várias plataformas por ter sido desenvolvido utilizando a linguagem Java e devido ao padrão de arquitetura adotado em que ele foi projetado, o qual facilita o seu transporte em parte ou no todo para outro ambiente computacional.

O sistema JULGAR proporciona análise de conteúdo dos textos, cujas informações calculadas podem servir de *feedback* para o auxílio da construção dos léxicos semânticos e conseqüentemente da melhoria dos classificadores para outros domínios.

Outra contribuição relevante foi o desenvolvimento de um corpus nuclear anotado o qual pode servir de base para adaptação para novos domínios correlatos. O esquema de anotação proposto por WILSON (2007) foi testado e poderá ser utilizado no processo de anotação de novos domínios.

Integrado a esta metodologia foi usado os programas computacionais AUTOSLOG-TS e BASILISK que proporcionam a obtenção de padrões de extração e novos termos para o domínio que se quer analisar.

A estratégia proposta neste trabalho foi testada com sucesso quando a ferramenta foi configurada para atender outro domínio, no caso um conjunto de mensagens extraídas do *Twitter*. Para tal só foi necessário treinar os classificadores externamente e a incorporação das informações do novo domínio sendo analisado.

Adicionalmente a estas contribuições já citadas, outra contribuição importante desenvolvida foi a adaptação das regras propostas por SPERTUS (1997) e a inclusão de novas regras para o domínio Nuclear e para o domínio *Twitter*, as quais proporcionaram um melhor entendimento dos textos e em um melhor desempenho dos

classificadores usados.

Como sugestão de trabalhos futuros pode-se citar a migração do sistema JULGAR para o ambiente Web.

A inclusão de mais um item de menu para a criação de um corpus em tempo real utilizando as APIs disponibilizadas pelo o Google ou por outros mecanismos de busca em função do domínio sendo escolhido.

A criação de um componente para o ambiente GATE similar ao componente *Real-time Corpus Controller (*interrompe o processamento de uma página em função do tempo necessário para processar a página), o qual iria capturar ou não uma página em função da presença ou ausência de certas características pré-definidas.

Como exemplo de tal funcionalidade seria a possibilidade de configurar a interrupção ou não do processamento de uma página caso ela contivesse ou não certas palavras no seu título ou no seu início.

A integração do algoritmo BASILISK ao sistema JULGAR para este usar as informações de conteúdo provenientes do módulo de análise de conteúdo como ponto de partida para o treino de extração de padrões de extração.

Criação de um módulo automático de extração de regras de domínio no formato da linguagem JAPE para serem usados diretamente na criação de anotações do domínio.

Avaliar a utilização de lógica *Fuzzy* em conjunto com a linguagem JAPE para aprimorar o processo de anotação. Atualmente já existe o componente *Fuzzy Believer*[1] o qual permite modelar que tipos de declarações devem ou não ser aceitas.

Por fim, difundir as potencialidades e características de processamento de linguagem natural presentes no ambiente computacional GATE para sua utilização em várias áreas.

[1] http://www.semanticsoftware.info/category/project/fuzzy-believer

8 REFERÊNCIAS BIBLIOGRÁFICAS

AUE, A. and GAMON, M., 2005, "Customizing Sentiment Classifiers to New Domains: a Case Study". In: *Proceedings of International Conference on Recent Advances in Natural Language Processing (RANLP)*, [7 p.], Borovets, BG, 21-23 Sep.

BARTLETT J., and ALBRIGHT R., 2008, "Coming to a Theater Near You! Sentiment Classification Techniques Using SAS® Text Miner", SAS Global Forum 2008, SAS Institute Inc., [9 p.], Cary, NC, USA.

CHEN, B. et al, 2009, "Extracting discriminative concepts for domain adaptation in text mining". In: *Proceedings of the Fifteenth ACM SIGKDD International Conference On Knowledge Discovery and Data Mining*, pp.179-188, New York, NY, USA.

CHOI, Y. and CARDIE, C., 2008, "Learning with Compositional Semantics as Structural Inference for Subsentential Sentiment Analysis", In: *Proceedings of Conference on Empirical Methods in Natural Language Processing*, pp.793-801, Waikiki, Honolulu, Hawaii, USA, 25-27 Oct.

CHOI, Y. and CARDIE,C, 2009, "Adapting a Polarity Lexicon using Integer Linear Programming for Domain-Specific Sentiment Classification", In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pp.590-598, Singapore, 6-7 Aug.

COSTA-FONT, J., RUDISILL, C. and MOSSIALOS, E., 2008, "Attitudes as Expression of Knowledge and "Political Anchoring": The Case of Nuclear Power in the United Kingdom", Risk Analysis, vol. 28, n.5, pp. 1273 – 1287

CUNNINGHAM, H. et al.,2010, *Developing Language Processing Components with GATE Version 5 (a User Guide)*, 2010. Disponível em: <http://gate.ac.uk/sale/tao/split.html> Acesso em: 10 nov. 2010, 13:53:30.

CURRAN, J., MURPHY, T. e SCHOLZ, B. 2006, "Minimizing semantic drift with Mutual Exclusion Bootstrapping", In: *Proceedings of the Conference of the Pacific Association for Computational Linguistic*s, pp. 172–180, Melbourne, Australia, 19–21 Sep.

DAI, W. et al., 2007, "Co-clustering based classification for out-of-domain documents", In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, San Jose, California, USA, 12-15 Aug.

GO, A., HUANG, L., and BHAYANI, R., 2009, *Twitter sentiment analysis*. Final Projects from CS224N for Spring 2008/2009 at The Stanford Natural Language

Processing Group.

Disponível em: <http://nlp.stanford.edu/courses/cs224n/2009/fp/3.pdf>. Acesso em: 16 Nov. 2010: 13:02

GO, A., HUANG, L., and BHAYANI, R., Twitter sentiment classification using Distance Supervision. Disponível em:

<http://www.stanford.edu/~alecmgo/papers/TwitterDistantSupervision09.pdf> Acesso em: 16 nov. 2010, 12:56

GUPTA, R., e SARAWAGI, S., 2008, "Domain adaptation of Information Extraction Models", ACM SIGMOD, Volume 37, Issue 4, pp. 35-40. Vancouver, Canadá, 9-12 Jun.

HAYKIN, S., 2001, *Redes Neurais Princípios e Práticas*. 2. ed. Porto Alegre, Bookman.

HSU, C., CHANG, C. and LIN, C. (2009), *A Practical Guide to Support Vector Classification*. Disponível em: http://www.csie.ntu.edu.tw/~cjlin/libsvm/. Acesso em: 16 Nov. 2010, 13:10

JUNIOR, J., BARROSO, A. et al., 2007, "News and its influence on the viability of nuclear power plants deployment – a modified epidemiological model for news generation", In: *Proceedings 2007 International Nuclear Atlantic Conference – INAC 2007*, [p.7] Santos, SP, Brazil September 30 to Oct. 5, 2007.

KAMPS, J., et al, 2004, "Using WordNet to measure semantic orientation of adjectives", In: *Proceedings of the 4th International Conference on Language Resources and Evaluation*, v. IV pp.1115–1118, Lisbon, PT.

KEEFE T. e KOPRINSKA, I., 2009, "Feature Selection and Weighting Methods in Sentiment Analysis", In: *Proceedings of the 14th Australasian Document Computing Symposium*, pp.67-74, Sydney, Australia, 4 Dec.

KONCHADY, M., 2008, *Building Search Applications: Lucene, LingPipe, and Gate*, Mustru Publishing, 3112 Bradford Wood Court, Oakton, VA 22124

KUGO, A. et al, 2005a, "Risk Communication System for High Level Radioactive Waste Disposal", In: *Proceedings of GLOBAL 2005*, n. 334, [p.6] Tsukuba,Japan, Oct.

KUGO, A. et al, 2005b, "Text Mining Analysis of Public Comments Regarding High Level Radioactive Waste Disposal", *Journal of Nuclear Science and Technology*,

Vol.42, N.9., pp.755-767.

KUGO, A. et al, 2008, “Study on risk communication by using Web system for the social consensus toward HLW final disposal”, *Progress in Nuclear Energy*, vol. 50, n., pp. 700 – 708

LI, Y., BONTECHEVA, K., CUNNINGHAM, H., 2007, "Experiments of Opinion Analysis on Two Corpora MPQA and NTCIR-6", In: Proceedings of the Sixth NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access, pp. 323 – 329, Tokyo, Japan, 15-18 May.

McDONALD, R. et al, 2007, “Structured Models for Fine-to-Coarse Sentiment Analysis”, In: *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 432–439, Prague, Czech Republic, Jun.

McINTOSH, T. e CURRAN, J., 2006, “Weighted Mutual Exclusion Bootstrapping for Domain Independent Lexicon and Template Acquisition”, In: Proceedings of the Australasian Language Technology Workshop, [p.9], Hobart, Australia, 30 Nov. -1 Dec.

MULLEN, T. e COLLIER, N., 2004 ,” Sentiment analysis using support vector machines with diverse information sources”, In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP),* pp. 412–418, July.

NAIRN, R., CONDORAVDI, C., KARTTUNEN, L., 2006, “Computing relative polarity for textual inference”, In: *Workshop Proceedings of ICoS-5, Inference in Computational Semantics*, [10 p.], Kuala Lumpur, 2-5 Aug.

OGURI, P., 2006, Aprendizado de máquina para o Problema de Sentiment Classification, Tese de M.Sc., PUC, RJ/Brasil.

PAK, A. and PAROUBEK, P., 2010, “Twitter as a corpus for sentiment analysis and opinion mining.”, In: *Proceedings of the Seventh Conference on International Language Resources and Evaluation*, pp. 1320-1326, 19-21 May.

PANG, B. et al, 2002, “Thumbs up? sentiment classification using machine learning techniques”, In: *Proceedings of the ACL-02 Conference on Empirical methods in natural language processing*, pp. 79-86, Philadelphia, PA, USA, July.

PANG, B., LEE, L, 2004, “A Sentimental Education: Sentiment analysis using subjectivity summarization based on minimum cuts”, In: *Proceedings of ACL-04, 42nd Meeting of the Association for Computational Linguistics,* pp. 271-278, Barcelona, ES, July.

POLANYI, L. and ZAENEN, A., 2004, “Contextual Valence Shifters”, In: *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text*, pp.106-111. Stanford University, CA, USA, 25-29 July.

RIBEIRO, J., BARROSO, A., IMAKUMA, K., 2007, “The Communication of the Value and Public Acceptance of Nuclear Plants”, In: *Proceedings of ICAPP 2007*, pp. 13 - 18, France, May.

RILOFF, E. e JONES, R., 1999, “Learning Dictionaries for Information Extraction by Multi-Level Boostrapping”, In: *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pp. 474-479, Orlando, Florida, USA , 18–22 July.

RILOFF, E. e PHILLIPS, W., 2004, *An Introduction to the Sundance and AutoSlog System*, In: Technical Report UUCS-04-015, School of Computing, University of Utah, Salt Lake City, UT 84112 USA.

RILOFF, E. e WIEBE, J., 2003, “Learning extraction patterns for subjective expressions”, In: Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP’03), pp.105-112, Sapporo, Japan. July 11-12.

RILOFF, E. e WIEBE, J., WILSON, T., 2003, “Learning Subjective Nouns using Extraction Pattern Bootstrapping”, In: Proceedings of the Seventh Conference on Natural Language Learning (CoNLL-2003), pp. 25-32, Edmonton, Canada, May 31 and June 1

SEGARAN, T., 2008, *Inteligência Coletiva*, edição Brasil Alta Books

SILVA,B. et al, 2006, “Methods and Tools for Encoding the WordNet.Br Sentences, Concept, Glosses and Conceptual-Semantic Relations”, PROPOR 2006, *Lecture Notes in Computer Science 2006*, Volume 3960/200, pp. 120-130.

SPERTUS, E., 1997, “Smokey: Automatic Recognition of Hostile Messages”, In: Proceedings of Innovative Applications of artificial Intelligence (IAAI), pp. 1058-1065, Providence, Rhode Island, 27-31 July.

THAKKER, D., OSMAN,T. e LAKIN, P., *GATE JAPE Grammar Tutorial*, 2009, Disponível em: <http://gate.ac.uk/sale/thakker-jape-tutorial/GATE%20JAPE%20manual.pdf> Acesso em: 11 nov. 2010, 14:40

THELEN, M., RILOFF, E., 2002, “A Bootstrapping Method for Learning Semantic Lexicons using Extraction Pattern Contexts”, In: *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, [8 p.], Philadelphia, USA, 6-7 July.

TURNEY,P., 2002, “Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews”, In: *Proceedings of ACL-02, 40th Annual Meeting of the Association for Computational Linguistics*, pp. 417–424, Philadelphia, US. July.

WHITELAW, C., GARG, N., AND ARGAMON, S., 2005, “Using Appraisal Groups for Sentiment Analysis”, In: *Proceedings of CIKM-05, 14th ACM International Conference on Information and Knowledge Management*, pp. 625–631, Bremen, DE, 31 Oct - 5 Nov.

WIEBE, J., 2002, *Instructions for Annotating Opinions in Newspaper Articles*, In: Technical Report TR-02-101, University of Pittsburgh, Pittsburgh, PA.

WIEBE, J., RILOFF, E., 2005, “Creating subjective and objective sentence classifiers from unannotated texts”, In: *Proceedings of CICLing* - 2005, [12 p.], Mexico City, Mexico, 13-19 Feb.

WIEBE, J., WILSON, T. AND CARDIE, C., 2003, “Annotating Expressions of Opinions and Emotions in Language”, In: Proceedings of EMNLP 2003, [50 p.], Saporo, Japan, 11-12 July.

WILCOCK, G., 2009, Introduction to Linguistic Annotation and Text Analytics, Morgan & Claypoll Publishers.

WILSON, T. and WIEBE, and HOFFMANN, P., 2005, “Recognizing Contextual Polarity of Phrase-Level Sentiment Analysis In: *Proceedings of HLT/EMNLP*, [8p.], Vancouver, Canada, 6-8 Oct.

WILSON, T. and WIEBE, J., 2003, “Annotating Opinions in the World Press”, In: *Proceedings of the ACL SIGDIAL-03*, pp. 13 - 22, [S.l.]

WILSON, T. and WIEBE, J., 2005, “Annotating Attributions and Private States”, In: *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, pp. 53-60, NJ, USA, June.

WILSON, T., 2006, *Instructions for Annotating Attitudes Types and Targets*, Disponível em: <http://homepages.inf.ed.ac.uk/twilson/attitude-instructions.pdf> Acesso em: 17 Nov. 2010, 09:05

WILSON, T., 2007, Fine-Grained Subjectivity and Sentiment Analysis: Recognizing the Intensity, Polarity, and Attitudes of Private States, D.Sc. Dissertation, University of Pittsburgh, Pittsburgh/USA.

YAOYONG, L., *et al.*, 2002,”The Perceptron Algorithm with Uneven Margins”, In:

Proceedings of ICML 2002. Sydney, NSW, Australia, 8-12 July.

YE Q, LIN B. e LI Y., 2005, “Sentiment Classification for Chinese Reviews: A Comparison between SVM and Semantic Approaches”, In: *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, pp. 909-915, Guangzhou*, China, 18-21 August.

APÊNDICE A

A ferramenta desenvolvida, daqui em diante chamada de Sistema JULGAR foi desenvolvida em Java com o objetivo de implementar e validar vários conceitos e estratégias que surgiram no decorrer deste trabalho de tese.

Dentre as principais características do sistema JULGAR pode-se citar:

- Utilizar classificadores de subjetividade e polaridade desenvolvidos externamente;
- Utilizar listas de palavras e expressões próprias ao domínio sendo analisado;
- Permitir a análise de conteúdo pela freqüência das palavras;
- Seleção do tipo de agrupamento para a confecção de um dendrograma.

A Figura 25 mostra de forma resumida a lógica de construção deste sistema. Como primeiro passo um corpus é anotado no ambiente GATE através da aplicação de um dos vários recursos de processamento lingüístico. Estes são usados para treinar um classificador segundo os parâmetros definidos no arquivo de configuração.

Após ser validado o classificador é salvo utilizando uma opção do ambiente GATE e este é incorporado ao sistema JULGAR.

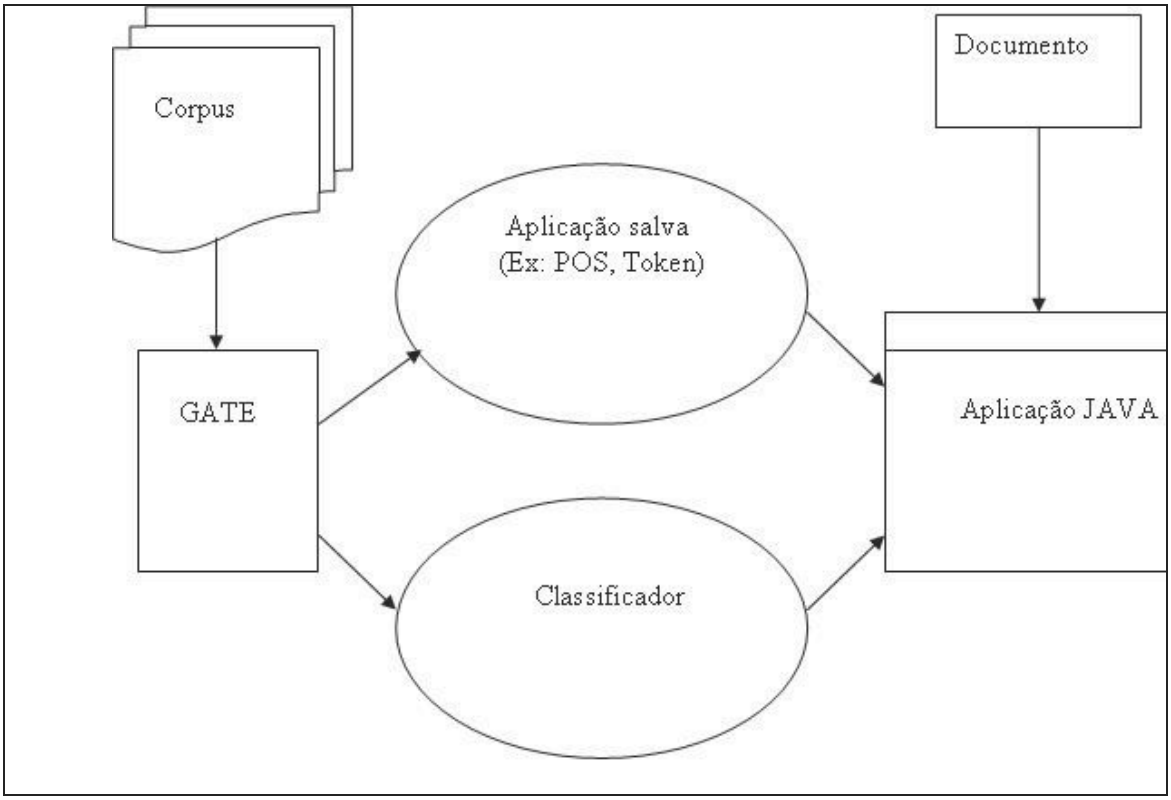

Figura 25: Fluxo de construção de uma aplicação utilizando aplicações desenvolvidas dentro do ambiente do software GATE.

O sistema JULGAR foi construído utilizando o padrão de projeto MVC o qual permitirá sua portabilidade para outros ambientes, como por exemplo, a Web.

Outro ponto importante considerado na sua construção foi de ele ser rapidamente configurável para analisar os sentimentos expressos e conteúdo de textos extraídos de outros domínios.

Para tal, basta incluir as listas de palavras e expressões relativas ao domínio sendo analisado e treinar os classificadores de subjetividade e polaridade utilizando o ambiente GATE.

As novas listas têm que ser incluídas no subdiretório *plugins/ANNIE/resources/gazetteer* do ambiente GATE e os nomes delas incluídos no arquivo *lists.def* que fica também neste diretório.

Após serem salvos os classificadores podem ser importados facilmente para dentro do sistema JULGAR, bastando indicar a posição deles através da opção configuração do menu do sistema.

Quando necessário, dependendo do domínio, podem ser criadas e

incorporadas novas regras para o tratamento de como os sentimentos são expressos no domínio e incluí-las no arquivo de regras *spertus.jape*.

Estas regras têm que ser codificadas na linguagem JAPE respeitando as convenções estabelecidas (ver THAKKER, 2009).

A Figura 26 apresenta a tela inicial do sistema JULGAR que disponibiliza via menu várias opções para a análise de sentimentos.

Figura 26: tela inicial do sistema Julgar

As opções com suas descrições são apresentadas a seguir:

A) Opção: Arquivo

Permite o usuário definir um novo corpus ou abrir um corpus criado anteriormente.

B) Opção: Editar

Permite atualizar um corpus já salvo.

C) Opção: Executar

Esta opção tem as principais funções executadas pelo o sistema, são elas (Figura 27):

- Análise de conteúdo;
- Análise de sentimentos;
- Categoria e
- Similaridade de documentos.

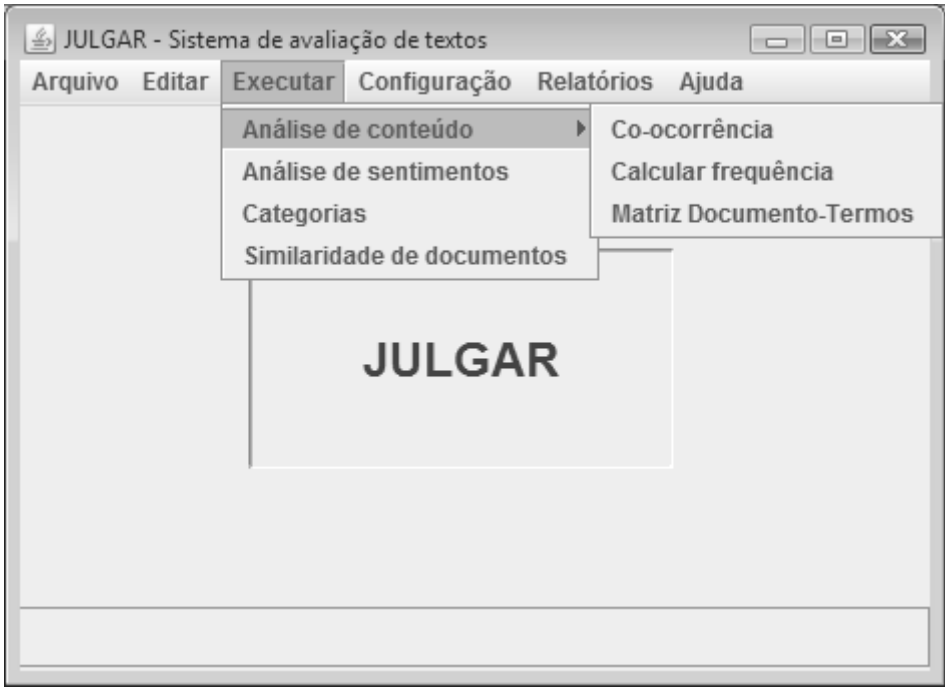

Figura 27: Opção executar do sistema JULGAR.

c.1)O item Análise de conteúdo é subdividido em :

1. Co-ocorrência

Indica a co-ocorrência de uma palavra com outras palavras presentes no corpus. O usuário entra com uma palavra e como resposta é mostrada para ele não só a palavra que co-ocorre com a palavra dada, mas também o número de documentos em que estas palavras co-ocorrem. É permitido criar expressões com mais de uma palavra para o cálculo de co-ocorrência das palavras no corpus.

2. Calcular freqüência

Apresenta uma tabela que relaciona cada documento a um *token* e apresenta as suas freqüências: Freqüência do termo, Freqüência máxima (maior freqüência de um *token* dentro do documento) e o TF*IDF relativo ao *token*.

3. Matriz Documento-Termos

Esta opção apresenta uma tabela que relaciona os documentos aos *tokens* apresentando as suas freqüências (TF*IDF).

c.2) Análise de sentimentos

Calcula os sentimentos expressos nos documentos. Como resultado apresenta o número de documentos com polaridade positiva, com polaridade negativa e neutra.

c.3) Categoria

Uma categoria é definida através de um arquivo texto contendo vários itens. Estes itens podem ser, por exemplo, os nomes de marcas de carros. O que esta opção faz é apresentar as palavras presentes no corpus que são mais relacionadas com cada item de uma categoria.

As categorias são definidas através da opção configuração do menu e colocadas no subdiretório categoria do sistema JULGAR com a extensão de arquivo ".cat".

d) Similaridade de documentos.

Apresenta um dendrograma relacionando os documentos do corpus por similaridade. Permite através do menu configuração selecionar o tipo de *link* usado para o agrupamento (completo ou simples) e o tipo de modelo usado para o cálculo de proximidade. Nesta versão só está implementado o cálculo de proximidade cosseno.

D) Opção: Configuração (Figura 28)

A opção de configuração está dividida em cinco abas. A primeira aba permite o usuário tratar do que vai ser processado. Pode entre outras coisas processar ou não *stop words*, estabelecer o número mínimo de caracteres que um *token* deve ter para ser considerado para análise ou se as palavras vão ou não ser reduzidas a sua raiz.

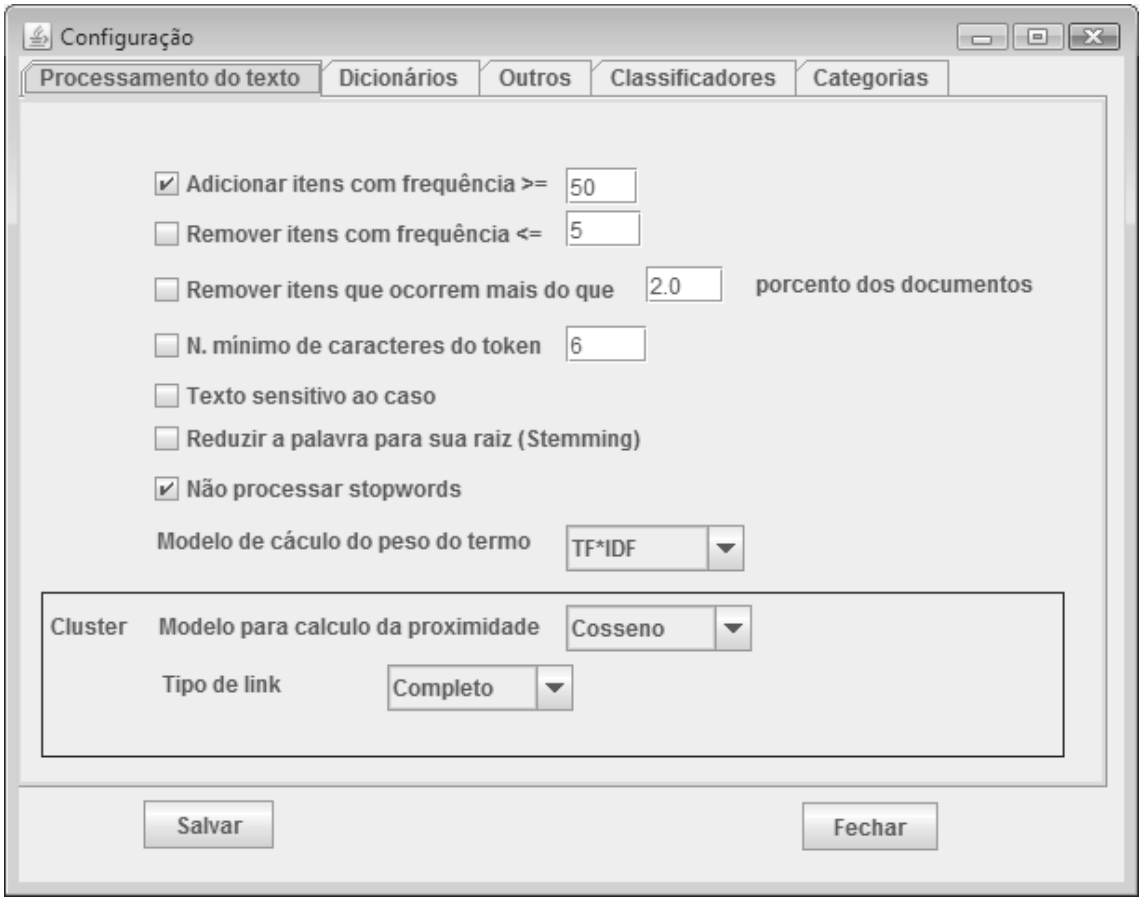

Figura 28: Tela de configuração

As demais abas permitem definir os locais das listas (dicionários) utilizadas, escolher a categoria à ser analisada e o local dos classificadores.

E) Opção: Relatórios \ Resumo

Apresenta um relatório resumido da análise de sentimentos adicionado com o nome das regras codificadas no arquivo *spertus.jape* que foram encontradas no corpus. Neste caso, se algum documento contiver um ou mais palavrões, será listado o nome da regra que identifica este tipo de palavra.

APÊNDICE B

A seguir são apresentadas algumas regras desenvolvidas utilizando a linguagem JAPE para anotar e processar anotações dos textos:

As regras são da forma LHS/RHS e são aplicadas as anotações criadas pela aplicação de um ou mais componentes de processamento lingüístico segundo o parâmetro definido na opção *Control*. O parâmetro *Input* define sobre quais anotações a regra atuará.

```
//------------------------------------------------------------------

Phase: spertus
Input: Sentence Lookup Token
Options: control=appelt
Debug=true
Macro: PLIC
(
  {Token.kind==punctuation, Token.string=="\""} |
   {Token.kind==punctuation, Token.string=="'"}
)

//------------------------------------------------------------------
Rule:socalled
Priority:50
(
  (  // you .... miffed - 100%
      {Token.string==~"[Yy]ou"}
      ({Token})[1,5]
      {Token.string=="miffed"}
  ) |
  (  // as yourself
      (
        {Token.string=="as"} |
        {Token.string=="like"}
      )
      (
        {Token.string=="yourself"} |
```

```
        {Token.string=="yourselves"}
      )
  ) |

 (
    (   {Token.string=="you"}
        {Token.string=="so"}
        {Token.string=="called"}
)
 |(
{Token.string=="you"}
{Token.string=="so-called"}
)
  ) |


 (  // get a life
    (
      {Token.string==~"[Gg]et"}
      {Token.string=="a"}
      {Token.string=="life"}
    )
 )|
     (
      {Token.string=="you have a right"} |
      {Token.string=="your right"} |
      {Token.string=="bash"}
     )

):rotulo
-->
:rotulo.spertus={rule="socalled"}

//--------------------------------------------------------------
Rule:negativewordthis
Priority:30
(
```

```
 (
   {Token.string=="this"}
   ({Token})[0,4]
   {Lookup.majorType==word,Lookup.minorType==negativo}
 ) |
 (
   {Lookup.majorType==word,Lookup.minorType==negativo}
   ({Token})[0,4]
   {Token.string=="this"}
 )
):rotulo
-->
:rotulo.spertus={rule="negativeword-this"}


//--------------------------------------------------------------
Rule:SecondPerson
Priority:20
(
    (
     (
       {Token.string=="you"} |
       {Token.string=="your"} |
       {Token.string=="yourself"}
     )
     (
      ({Lookup.majorType==word, Lookup.minorType==negativo})
     )
     (
        {Token.string==","}
        ({Lookup.majorType==word, Lookup.minorType==negativo})
     )*
  ) |
  (
      // 'service' of yours
     PLIC
     {Token.category==NN}
```

```
    PLIC
    {Token.category==IN}
    (
      {Token.string=="you"} |
      {Token.string=="yours"} |
      {Token.string=="yourself"}
    )
  )
): rotulo
-->
:rotulo.spertus={rule="Second-Person"}

//------------------------------------------------------------
Rule: apposition
Priority:10
(
  {Token.string==~"[Yy]ou"}
 (
  {Token.category==NNS, !Token.string=="folks", !Token.string=="guys"} |
  {Token.category==NN, !Token.string=="folks", !Token.string=="guys"}
 )
): rotulo
-->
:rotulo.spertus={rule="apposition"}

//------------------------------------------------------------
Rule:Profanity
// What the f... is your problem?
(
 ( {Lookup.majorType==word, Lookup.minorType==negativo}
  {Token.category==VBZ}
  {Token.category==PRP}//possessive pronoun
 ) |
 (
    {Token.string=="you"}
    ({Lookup.majorType==word, Lookup.minorType==negativo})[1,4]
```

```
 )
):rotulo
-->
:rotulo.spertus={rule="profanity"}

//------------------------------------------------------------
Rule:Imperative
Priority:40
(
  (
    {Token.string=~"[Tt]ake"}
    {Token.string=="care"}
  ) |
  (
     (
       {Token.string=="you"} |
       {Token.string=="yours"}
     )
     {Token.string=="ilk"}
  )
):rotulo
-->
:rotulo.spertus={rule="Imperative"}



//------------------------------------------------------------
Rule:dominiospecific
Priority:50
(
  {Lookup.majorType==dominio ,Lookup.minorType==negativo}

):rotulo
-->
:rotulo.spertus={rule="dominio-negativo"}
//------------------------------------------------------------
Rule:badwords
```

```
Priority:50
(
  {Lookup.majorType==word ,Lookup.minorType==negativo}

):rotulo
-->
:rotulo.spertus={rule="bad-words"}

//------------------------------------------------------------
Rule:laugh
Priority:5
(
   ({Token.string==~"[Hh]a"})+
):rotulo
-->
:rotulo.spertus={rule="laugh"}
//-----------------------------------------
// Ex: Weeeee
Rule:twitter
Priority:50
(
  (
    {Token.string=~"[a-z]+a{3,8}[a-z]+"}|
    {Token.string=~"[a-z]+e{3,8}[a-z]+"}|
    {Token.string=~"[a-z]+i{3,8}[a-z]+"}|
    {Token.string=~"[a-z]+o{3,8}[a-z]+"}|
    {Token.string=~"[a-z]+u{3,8}[a-z]+"}
  ) |
  (
     ({Token.kind==punctuation, !Token.string=~"[\\.]" })[4,8]
  )

):rotulo
-->
:rotulo.spertus={rule="twiteer"}
```

APÊNDICE C

GLOSSÁRIO

Tipos de análises

Semântica	Define uma classe para a palavra Ex: CIDADE, CEP
Morfológica	Ex: Maria comprou um carro. Maria: substantivo próprio, comprou: verbo, um: artigo, carro: substantivo comum
Sintática	Ex: Maria comprou um carro. Maria: sujeito, comprou: núcleo do predicado verbal (comprou um carro), um: adjunto adnominal, carro: núcleo do objeto direto (um carro)
Léxica	Ex: Palavra, pontuação, número

Termos mais freqüentes

Afinidade semântica	A afinidade semântica de um EP com uma categoria semântica é a medida de sua tendência para extrair NPs pertencentes a uma categoria semântica.
Análise de sentimentos	É a tarefa de identificar opiniões, emoções e avaliações positivas ou negativas
Análise de subjetividade	Determinar se uma sentença e positiva ou negativa. É a tarefa de identificar quando um private state está sendo expresso e identificar os atributos deste private state
Appraisal theory	Teoria da linguagem avaliativa é a linguagem usada para

	se expressar opiniões, um aspecto importante da linguagem.
Bag of Words (BoW)	O BoW, saco-de-palavras, é um modelo de linguagem unigrama
Bigramas	São grupos de dois tokens, é um caso especial de n-grama
Content word	Uma palavra tal qual um substantivo, verbo, adjetivo, que possui um significado léxico estável.
Co-ocorrência	É a propriedade temporal de duas coisas acontecerem ao mesmo tempo.; o intervalo que determina a coincidência pode ser ajustáveis. Em poucas palavras, termos, stems e conceitos que coincidem mais freqüentemente tendem a ser relacionados. É uma associação semântica entre dois termos. Uma diferença entre co-ocorrência e collocation é que collocation (colocação) refere-se a elementos que gramaticalmente estão abrigados em uma determinada ordem e co-ocorrência seria mais geral no qual certas palavras são utilizadas em um mesmo contexto. Por exemplo, quando ouvimos o termo “aloha” nós pensamos imediatamente no Hawaii.
F-measure ou *F1 score*	É a média harmônica de *precision* e *recall*.
Frase ou sentença	Frase é todo enunciado lingüístico capaz de transmitir uma idéia. A frase se define pelo propósito de comunicação, e não pela sua extensão. A frase pode ser composta por uma palavra ou conjunto de palavras que constitui um enunciado de sentido completo. Frase é a menor unidade da comunicação lingüística. Tem como características básicas: 1. A apresentação de um sentido ou significado completo

	2. Não há um padrão definido de frase; contudo, podemos identificá-la em três tipos distintos de construção:
F-score	É a medida da acuracidade do teste: 2* (p*r) /(p+r)
Function word	São palavras que servem para expressar relações com outras palavras na sentença ou especificam a atitude ou humor do orador (pronomes, conjunções, etc)
Hiperonímia (genérico)	É a palavra que dá a idéia do todo
Hiponímia (específico)	É a palavra que indica cada parte ou cada item de um todo
Lemmatization	É o processo de determinar a forma básica da palavra . O processo envolve determinar o PoS da palavra na frase, o que requer o conhecimento da gramática da língua. Ex: Better good
Mutual information	Mede quanto uma variável randômica diz a respeito de outra. Pode ser pensada como uma redução de incerteza sobre uma variável tendo o conhecimento de outra. Quando o valor é zero significa que as variáveis são independentes.
Oração, Sentence (clause)	Oração é todo enunciado lingüístico que tem o núcleo como o verbo, apresentando, desta maneira e na maioria das vezes, “termos essenciais da oração, sujeito e predicado”.
Orientação semântica (*Semantic Orientation*)	É um método utilizado para a avaliação das características das palavras. Classifica as palavras com bom e ruim., e então contabiliza o escore total de palavras boas e ruins para o texto. Ela pode apresentar direção de orientação positiva (elogio) ou negativa (crítica), ou um grau (leve ou forte). Uma revisão pode ser classificada baseada na média da orientação semântica das frases que contêm advérbios ou adjetivos.

Pairwise comparasion	É o processo de comparar entidades em pares para julgar qual de cada par é o preferível ou se tem uma quantidade maior de alguma propriedade
Penn Treebank POS Tags	Padrão de tags http://www.cis.upenn.edu/~treebank/home.html
Período (*sentence*)	Unidade lingüística composta por uma ou mais orações. Tem como características básicas: A apresentação de um sentido ou significado completo Encerrar-se por meio de certos símbolos de pontuação (excluem-se a vírgula e o ponto-e-vírgula).
Pointwise Mutual Information (PMI-IR)	É uma medida de associação utilizada em IR, é definida como o log do desvio entre a freqüência observada de um bigram e a probabilidade deste bigram se eles fossem independentes. Pontuação(escolha1)= $\log_2\left(\frac{p(problema \cap escolha1)}{p(problema)\,p(escolha1)}\right)$ Como a probabilidade do problema é a mesma para todas as escolhas ela pode ser suprimida. O log também devido a ele crescer monoticamente. Temos então uma probabilidade condicional do problema dado a escolha1. Turney usa 4 modelos para calcular as pontuações.
Polissemia	É a propriedade que uma mesma palavra tem de apresentar vários significados.
POS *Tag*	Técnica de processamento de linguagem natural
POS *Tagging*	É uma tarefa de rotular cada palavra em uma sentença com sua POS apropriada
Precision	Mede quantos dos itens que o sistema identificou são realmente corretos sem considerar que o sistema falhou de recuperar itens corretos;

Private state (PS)	É um termo geral usado para expressar opiniões, emoções e especulações. São estados mentais ou emocionais. Estes estados não estão abertos para observação ou verificação objetiva.
Recall	Mede quantos dos itens que deveriam ter sido realmente identificados foram identificados sem considerar quantas identificações espúrias foram feitas. Um recall de 55% indica que 55% das sentenças subjetivas recuperadas contêm pelo menos um padrão subjetivo
Saco-de-palavras	Ver *Bag of Words*
Sentença objetiva	Utilizada para transmitir uma informação objetiva ou factual
Sentença subjetiva	Usada para transmitir a opinião, avaliações, emoções ou especulações.
Sintagma **(***phrase*)	Sintagma é uma unidade formada por uma ou várias palavras que, juntas, desempenham uma função na frase. Para isto buscamos o elemento núcleo e classificamos o sintagma. Ex: pela cidade, núcleo=pela, preposição por + ela.
Stemming	É o processo de reduzir as palavras para seu stem (parte da palavra deixada após a remoção de seus afixos), ou seja, é a parte da palavra que é comum a todas as suas formas inflexionais (gênero, número, tempo ou pessoa). É parecido com a *lemmatization* mas só opera na palavra. Ex: *walking walk walk*
token	É uma única palavra, mais também pode ser também uma seqüencia de caracteres, uma frase, um endereço de e-mail, um link URL ou um acrônimo.
word n-gram	São um conjunto de N palavras contínuas extraídas de uma frase.