6
Atributos

Para buscar o sentimento de um texto automaticamente precisamos encontrar boas pistas ao longo do discurso. Uma grande variedade de palavras e expressões possui conotação positiva ou negativa, o que podem nos dar essas pistas de sentimento, porém dependendo do contexto elas podem significar justamente o oposto. Por exemplo, a palavra "subiu"normalmente possui uma conotação positiva no mercado financeiro, porém o exemplo abaixo mostra uma conotação oposta.

"A dívida da empresa subiu."

Somente o dicionário de palavras não é suficiente para representar o sentimento do texto, por isso devemos utilizar padrões de linguagem que generalizem mais e sejam capazes de capturar o contexto.

Existem 3 tipos de atributos que são utilizados em estudos anteriores de analise de sentimento: atributos sintáticos, semânticos e estruturais (13, 21, 06, 01). Os mais utilizados são os sintáticos e semânticos, entre eles podemos citar n-gramas de palavras, anotação morfossintática (Part-Of-Speech - POS tags) e pontuação. Adicionalmente, atributos sintáticos incluem padrões de frases como por exemplo n-gramas de etiquetas morfossintáticas . Fei (06) mostra que padrões como "n+adj"(substantivo seguido de um adjetivo negativo) normalmente expressavam um sentimento negativo. Wiebe (21) cria n-gramas de padrões de frases, como por exemplo "U-adj + como-prep"é usado para todos os bigramas contendo uma única ocorrência de um adjetivo seguido da preposição "como". Abbasi (01) utiliza n-gramas de palavras, de etiquetas morfossintáticas, letras e dígitos, e mostra que essas informações são importantes para capturar distinções de linguagem e melhorar a acurácia.

Cada atributo novo criado é associado a um identificador, correspondendo a uma coordenada do vetor passado para o SVM. Em cada coordenada do vetor, é atribuído o valor 1 se aquele atributo existe naquele documento e zero caso contrário.

6.1 Atributos N-Gramas

Um n-grama é uma sequência de n itens dentro de uma frase. Os itens podem ser palavras, letras, sílabas, classificação gramatical das palavras, ou qualquer outra base. Um n-grama de tamanho 1 é chamado de unigrama, de tamanho de 2, de bigrama, de tamanho 3 é chamado de trigrama, de 4 em diante é n-grama. Para uma sequência de palavras, por exemplo "Ações da Petrobras sobem", um bigrama de palavras seria: "# Ações", "Ações da", "da Petrobras", "Petrobras sobem", "sobem #".

Abbasi (01) propõe a utilização de n-gramas para diferentes bases como palavras, letras, dígitos e etiquetas morfossintáticas. Ele mostra que esse tipo de atributo é bastante informativo e importante para capturar estilos de linguagem de textos positivos e negativos.

Nesse trabalho utilizamos unigramas, bigramas e trigramas de palavras, etiquetas morfossintáticas e etiquetas de chunks, que será descrito no próximo item. Pesquisas anteriores (12, 21) mostram que a utilização de 4-gramas ou mais é redundante, causando ruído e diminuição da eficiência do experimento, por isso descartamos esse tipo de atributo.

6.2 Processamento de Linguagem Natural

Processamento de Linguagem Natural (PLN) é uma área de pesquisa que busca aprimorar a interação entre homem e máquina. Seu principal objetivo é interpretar e gerar linguagem natural (e.g português, inglês, espanhol, etc). Isso significa reconhecer o contexto, fazer análise sintática, semântica, léxica e morfológica, e até mesmo criar resumos e extrair informações.

Durante os experimentos os seguintes atributos lingüísticos foram utilizados para a classificação de sentimento: Anotação Morfossintática e Chunks.

Anotação Morfossintática consiste em atribuir aos itens lexicais de um texto etiquetas que identifiquem sua classe de palavras, gênero, número e classes gramaticais.

Exemplo:

"Eu não gostei desse filme"

Etiqueta Morfossintática:

[Eu-Pronome] [não-Advérbio] [gostei-verbo] [desse-preposição] [filme-substantivo]

Classificação de chunks consiste em dividir o texto em estruturas sintaticamente relacionadas. Essas estruturas não podem ser sobrepostas. Nessa divisão, cada palavra só deve pertencer a um único chunk e as palavras sintaticamente relacionadas devem pertencer ao mesmo chunk.

Exemplo:

[Eu-BNP] [não-O] [gostei-BVP] [de-BPP] [esse-BNP filme-INP]

Nesse exemplo os chunks estão separados por colchetes. Nesse caso "Eu"inicia um chunk nominal (*Begin Noun Phrase*). A negação "não"não pertence a chunk nenhum. O verbo "gostei"inicia um chunk verbal (*Begin Verb Phrase*). A preposição "desse"é quebrada em duas: "de", que inicia um chunk preposicional (*Begin Prepositional Phrase*), e "esse"que inicia um chunk nominal. Finalmente, o substantivo "filme"está contido no último chunk nominal (*Inside Noun Phrase*).

Em contraste com muitos trabalhos que utilizam basicamente a relação verbo-substantivo, nós consideramos todas as classes morfossintáticas e suas relações, além da etiqueta de chunks.

O processamento dos textos para a obtenção desses atributos lingüísticos foi feito pelo FEXT (10), sistema desenvolvido pelo laboratório LEARN da PUC-Rio. O FEXT utiliza o algoritmo de aprendizado ETL (*Entropy Guided Transformation Learning*) (05), uma estratégia que combina as vantagens de árvore de decisão com o TBL (*Transformation Based Learning*). As informações de PLN foram usadas de diferentes maneiras, de modo à melhor capturar o contexto.

A primeira informação útil que podemos tirar das etiquetas PLN é uma informação estrutural, a freqüência das etiquetas. Em seus experimentos, Abbasi (01) considera apenas a freqüência das etiquetas morfossintáticas.

Em seguida, realizamos experimentos utilizando a presença de bigramas e trigramas apenas das etiquetas. Por exemplo, (Pron+Adv+Verb) é um trigrama de etiqueta morfossintática, se em alguma parte do documento aparecem três palavras consecutivas com essas classificações, então esse atributo trigrama é uma coordenada no vetor do SVM com valor "1". Esse tipo de atributo é proposto por Abbasi, e novamente ele utiliza apenas bigramas e trigramas de etiquetas morfossintáticas.

Finalmente, a forma mais usual de se utilizar a informação das etiquetas PLN é criando um par palavra_etiqueta. Pang (13) utiliza apenas a etiqueta morfossintática e não obtém bons resultados. Wiebe (21) utiliza unigramas, bigramas e trigramas do par palavras_etiqueta Morfossintática. Por exemplo, (Eu_Pron + não_Adv + gosto_Verb) é um trigrama formado

pelo pronome "eu"seguido do advérbio "não"seguido do verbo "gosto". Nesse trabalho utilizamos tanto etiquetas morfossintáticas quanto etiquetas de chunks. Dessa maneira para cada palavra geramos 3 pares: palavra_EtiquMorf, palavra_EtiquChk, palavra_EtiquMorf_EtiquChk.

6.3 Atributos Estruturais

Estudos anteriores tem se focado mais em atributos sintáticos e semânticos. O uso de atributos estruturais como distribuição do tamanho de palavra, riqueza de vocabulário, freqüência de caracteres especiais, não tem sido muito utilizado. Pang (12) mostra que muitos atributos não são intuitivamente informativos a principio, porém se mostram de grande importância. Atributos estruturais podem revelar padrões latentes e melhorar a performance do classificador.

Wiebe (21) mede a eficiência em utilizar palavras únicas (que só ocorrem uma única vez) para classificar um texto como objetivo/ subjetivo e mostra que em textos subjetivos existem muito mais ocorrências de palavras únicas, indicando que quando estamos dando opinião somos mais criativos no discurso. A utilização desse tipo de atributos não se mostra muito eficiente para classificação de sentimento para críticas de cinema ou produto, porém Abbasi (01) mostra que esses marcadores de estilo melhoram a classificação de textos da Web pois eles são ricos em variações estruturais. Todos os atributos estruturais utilizados nesse trabalho foram propostos por Abbasi com resultados positivos na classificação de sentimento.

Nossa lista de atributos estruturais é formada por 15 categorias diferente: frequência de palavras com 1 a 20 letras; frequência de letras; frequência de dígitos; ocorrência de pontuação (!, ?, :,...) C; riqueza de vocabulário; tamanho médio das palavras; presença de palavras funcionais (depois, então, nenhum, ...) D; nº de palavras pequenas (até 4 letras); nº de palavras que aparecem apenas 1 vez; nº de palavras que aparecem apenas 2 vezes; nº de letras no exemplo; nº de palavras no exemplo; nº de sentenças; nº médio de letras por sentença; nº médio de palavras por sentença;

6.4 Caminhamento na árvore de decisão

Uma árvore de decisão é um classificador na estrutura de uma árvore. Trata-se de um modelo prático de uma função recursiva que determina o valor de uma variável e, baseando-se neste valor, executa uma ação. Esta ação pode ser a escolha de outra variável ou a saída.

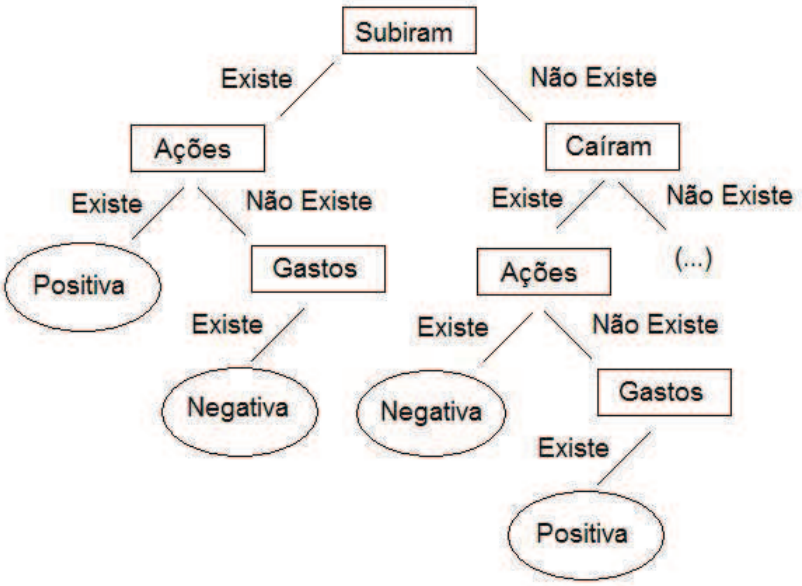

Figura 6.1: Exemplo de Árvore de decisão

Nesse trabalho não estamos utilizando a árvore de decisão como classificador do problema, e sim como gerador de novas informações. Antes das notícias serem encaminhadas para o SVM, geramos uma árvore de decisão. A partir dessa árvore, criamos novos atributos que são incorporados aos vetores e depois passados para o SVM.

Para gerar a árvore utilizamos o algoritmo já pronto do C4.5 (15). Cada nó da árvore é um atributo e as arestas a presença ou ausência do mesmo. As folhas são a resposta da classificação.

Com os caminhamentos na árvore de decisão criamos novos atributos. Cada passo do caminho é um novo atributo. Em cada documento do corpus adicionamos a informação se esses caminhos existem ou não. Por isso a informação das folhas não é usada, ou seja, não nos importa a classificação final dada pela árvore.

Para entender melhor o que significa criar novos atributos através dos caminhamentos, observe o exemplo de uma árvore criada na figura 6.1.

Temos a árvore de decisão montada pelo C4.5. Cada nó da árvore é representado por uma a palavra do dicionário, e cada aresta indica o resultado do teste se aquela palavra existe ou não. Se a palavra existir devemos tomar o ramo da esquerda e verificar se a próxima palavra indicada pela árvore existe ou não e assim sucessivamente. O benefício de utilizar uma árvore de decisão é que ela coloca no topo os atributos mais importantes.

A tabela 6.4 mostra os novos atributos criados pelo exemplo acima.

Subiram(Existe)+Ações(Existe)
Subiram(Existe)+Ações(NãoExiste)
Subiram(Existe)+Ações(NãoExiste)+Gastos(Existe)
Subiram(NãoExiste)+Cairam(Existe)
Subiram(NãoExiste)+Cairam(Existe)+Ações(Existe)
Subiram(NãoExiste)+Cairam(Existe)+Ações(NãoExiste)+Gastos(Existe)

Tabela 6.1: Novos atributos gerados pela Árvore de Decisão