Trabalho Prático 2 - Processamento de Linguagem Natural

Hugo Araujo de Sousa

Processamento de Linguagem Natural (2017/2)
Departamento de Ciência da Computação
Universidade Federal de Minas Gerais (UFMG)

hugosousa@dcc.ufmg.br

Resumo. *O objetivo desse trabalho é estudar a tarefa de Part-of-Speech (POS) tagging para a Língua Portuguesa. Para isso, utilizamos o corpus Mac-Morpho e comparamos o desempenho de dois modelos preditivos, um paramétrico e outro não-paramétrico.*

1. INTRODUÇÃO

Em Processamento de Linguagem Natural, uma vez que o principal objetivo é fazer com que os computadores entendam as linguagens naturais usadas pelos seres humanos, muitas vezes torna-se necessário reduzir a complexidade dessa tarefa ao quebrá-la em tarefas intermediárias. Uma dessas tarefas, que é particularmente útil na área de **parsing**, é a tarefa de **Part-of-speech tagging**, que se trata da identificação das classes gramaticais de cada uma das palavras presentes um corpus [Manning and Schütze 1999]. Esse problema não trata-se apenas da criação de um banco de dados que contenha a classe de cada palavra, uma vez que uma mesma palavra pode estar associada a múltiplas classes de acordo com seu contexto e posição em uma sentença.

Uma das abordagens que podem usadas para resolver esse problema é utilizar algum algoritmo de aprendizado de máquina supervisionado. Nesse trabalho, serão utilizados dois algoritmos de aprendizado para resolver o problema de POS tagging a fim de verificar e comparar a precisão e desempenho de cada um.

2. MODELAGEM

Dentro dos métodos de aprendizado de máquina supervisionada, existem os paramétricos e os não-paramétricos. A diferença entre os dois é que, nos métodos paramétricos, assume-se que os dados se organizam em algum modelo e então encontra-se valores apropriados do modelo a partir dos exemplos. Para abordar o problema de POS tagging, vamos utilizar um algoritmo de cada categoria, sendo **Naive Bayes** [McCallum and Nigam 1998] o paramétrico e o classificador de **Support Vector Machines (SVM)** [Hearst et al. 1998] o não-paramétrico.

Uma vez determinados os algoritmos a serem utilizados no processo de aprendizado, é necessário discutir o conjunto de dados de entrada e o mapeamento desses dados para a entrada dos algoritmos.

2.1. Corpus de Entrada

O conjunto de dados utilizado no trabalho como entrada dos algoritmos de aprendizado é o corpus Mac-Morpho [Aluísio et al. 2003]. O Mac-Morpho é um corpus de textos

escritos em Português Brasileiro, anotados com as classes gramaticais de cada palavra presente. Há, disponíveis para download gratuito, as seções de treinamento, validação e teste do corpus, que representam 76%, 4% e 20% do total do corpus, respectivamente.

Na página do corpus online [1] é possível fazer o download do mesmo, juntamente com o manual das anotações, que descreve todas as classes gramaticais utilizadas no corpus.

2.2. Extração de Features

Com o corpus de entrada em mãos, o próximo passo foi determinar como mapear esses dados para serem alimentados aos algoritmos de aprendizado de máquina. Para isso, a decisão foi trabalhar com features das palavras. Essas features são, como o nome indica, características obtidas através de cada palavra em si, além do seu contexto na sentença em que se encontra, isto é, posição absoluta e relativa às classes gramaticais. A lista das features utilizadas no trabalho é mostrada na Tabela 1.

Feature	Descrição
word	A própria palavra em si.
is_first	Booleano que indica se a palavra é a primeira da sentença.
is_last	Booleano que indica se a palavra é a primeira da sentença.
is_capitalized	Booleano que indica se a palavra começa com uma letra maiúscula.
is_all_caps	Booleano que indica se a palavra somente contém letras maiúsculas.
is_all_lower	Booleano que indica se a palavra contém somente letras minúsculas.
prefix-1	String com o primeiro caractere da palavra.
prefix-2	String com os dois primeiros caracteres da palavra.
prefix-3	String com os três primeiros caracteres da palavra.
suffix-1	String com o último caractere da palavra.
suffix-2	String com os dois últimos caracteres da palavra.
suffix-3	String com os três últimos caracteres da palavra.
prev_tag	String que representa a classe gramatical da palavra anterior à palavra atual.
next_tag	String que representa a classe gramatical da palavra seguinte à palavra atual.
has_hyphen	Booleano que indica se a palavra possui hífen.
is_numeric	Booleano que indica se a palavra é um número (dígitos).

Tabela 1. Features de palavras utilizadas no trabalho.

3. IMPLEMENTAÇÃO

Linguagem Bibliotecas Decisões de implementação Como executar

4. RESULTADOS

Saída do programa Análise

[1] `http://nilc.icmc.usp.br/macmorpho`

5. CONCLUSÃO

6. REFERÊNCIAS

[Aluísio et al. 2003] Aluísio, S., Pelizzoni, J., Marchi, A. R., de Oliveira, L., Manenti, R., and Marquiafável, V. (2003). *An Account of the Challenge of Tagging a Reference Corpus for Brazilian Portuguese*, pages 110–117. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Hearst et al. 1998] Hearst, M., Dumais, S., Osman, E., Platt, J., and Scholkopf, B. (1998). Support vector machines. *Intelligent Systems and their Applications, IEEE*, 13(4):18 –28.

[Manning and Schütze 1999] Manning, C. D. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, USA.

[McCallum and Nigam 1998] McCallum, A. and Nigam, K. (1998). A comparison of event models for naive bayes text classification. In *IN AAAI-98 WORKSHOP ON LEARNING FOR TEXT CATEGORIZATION*, pages 41–48. AAAI Press.