Método de pontos interiores para estimar os parâmetros de um modelo probabilístico usando o *corpus Thyco Brahe*

Esther S. Mamián Lopez[1], **Aurelio Ribeiro Leite Oliveira**[1]

[1]Inst. de Matemática, Estatística e Computação Científica – Universidade Estadual de Campinas (UNICAMP)
Rua Sérgio Buarque de Holanda, 651– 13083-859– Campinas – SP – Brazil

esmamian@gmail.com, aurelio@ime.unicamp.br

Abstract. *Statistical methods research for natural language processing and other important applications, have been presenting fast growth in the recent years. In this work, we propose a primal-dual interior point method for trainning stochastic context free grammar. For that purpose, we use a Portuguese based corpus Tycho Brahe [IEL-UNICAMP and IME-USP].*

Resumo. *Nos anos recentes têm acontecido um importante crescimento no interesse sobre os métodos estatísticos para processamento de linguagens e as diferentes aplicações que se derivam. Propomos um método de pontos interiores barreira logarítmica primal-dual para abordar o problema de atribuir valores ótimos de probabilidade às regras de uma gramática probabilística livre de contexto (GPLC) baseados no corpus da linguagem portuguesa Tycho Brahe [IEL-UNICAMP and IME-USP].*

1. Introdução

Dentre os modelos estatísticos para modelar as linguagens naturais estão as gramáticas probabilísticas livres de contexto, que na sua forma mais simples podem ser decompostas numa parte estrutural e numa parte estocástica. Abordamos o problema de modelar uma linguagem natural através de uma GPLC treinando a mesma, ou seja, encontrar as probabilidades ótimas associadas às regras da gramática [Manning and Schutze. 2003]. Este processo é realizado em base a um *corpus* que contém sentenças da linguagem que desejamos modelar. Para o processo do treino precisamos de uma função critério da amostra[1] e um marco para otimizá-la: usamos a função de máxima verossimilhança da amostra e o método de pontos interiores barreira logarítmica primal-dual, respectivamente.

Assim, o problema de treinamento de uma gramática é resumido como o problema de otimização descrito a seguir:

$$\begin{array}{ll} \text{maximizar} & f(x) \\ \text{sujeito a} & \sum\limits_{x_i \in \Psi_A} x_i = 1, \quad \forall\ \Psi_A : A \in \Sigma \\ & 0 \leq x_i \leq 1, \qquad i = 1, ..., |P| \end{array} \tag{1}$$

onde $f(x) = Pr(\Omega|G_p)$ é a função de verossimilhança que, depende da amostra Ω e da GPLC G_p. Esta função é um polinômio nas variáveis x_i, $i = 1...|P|$, sendo que

[1]Usamos a palavra *corpus* e amostra indistintamente para nos referir ao conjunto de sentenças de uma linguagem natural que estamos desejando modelar.

x_i correspondem aos valores de probabilidade associados às regras de G_p. Denotamos por Ψ_A como os valores de probabilidade associados às regras que possuem o mesmo antecedente. Denotamos por Σ o conjunto finito de símbolos não terminais, e P como o conjunto finito das regras da G_p.

2. Metodologia

Uma vez definido o problema (1), implementamos o método barreira logarítmica primal-dual baseados na formulação para problemas não lineares, descrita em [El-Bakry et al. 1996]. Este método foi implementado na linguagem C++, versão 4.8.4. O *corpus* usado é o *Tycho Brahe*, para o português do Brasil. Para analisar a viabilidade da proposta extraímos do *corpus* algumas GPLCs de diferentes tamanhos e as treinamos usando o método barreira logarítmica primal-dual. Seguem os detalhes das gramáticas extraídas para o treinamento e algumas discussões associadas às implementações.

2.1. Tamanho dos problemas

Trabalhamos com seis sub-problemas detalhados na Tabelas 1 e 2.

	Nro símbolos não terminais m	Nro símbolos terminais n	Nro de regras $\|P\|$	Símbolo inicial
Sub-Gramática 1	5	2.500	12.580	S
Sub-Gramática 2	7	2.500	17.752	S
Sub-Gramática 3	9	2.500	23.076	S

Tabela 1. Características da gramática para cada sub-problema do 1 até 3.

	Nro símbolos não terminais m	Nro símbolos terminais n	Nro de regras $\|P\|$	Símbolo inicial
Sub-Gramática 4	3	5.500	16.512	S
Sub-Gramática 5	3	7.500	22.512	S
Sub-Gramática 6	3	7.500	22.512	S

Tabela 2. Características da gramática para cada sub-problema do 4 até 6.

2.2. Cálculo da função objetivo $f(x)$

Note que, o cálculo da função objetivo $f(x)$ deve ser feita a cada iteração do método de pontos interiores barreira logarítmica primal-dual. E, dado que a função objetivo depende das probabilidades das sentenças do *corpus*, observe que não é eficiente calcular a probabilidade de uma sentença como a soma das probabilidades de todas suas possíveis árvores sintáticas. Para isso, implementamos um método baseado no método Cocke-Younger-Kassami, desenvolvido em [López 2018] para calcular o valor da função objetivo sem comprometer as capacidades físicas das máquinas, e utilizamos a biblioteca Open Multi-processing (OpenMP 2.5) para melhorar os tempos computacionais deste método.

2.3. Cálculo do gradiente e da Hessiana de $f(x)$

Os valores do gradiente e da Hessiana devem ser calculados a cada iteração. Usamos o método das diferenças finitas para calculá-lo, pois a expressão das derivadas é muito cara.

2.4. Resolução do sistema linear

Além do custo computacional para calcular o valor da função objetivo e suas derivadas a cada iteração, o trabalho computacional dos métodos de pontos interiores também é dominado pela resolução de um sistema linear [Gondzio 2012]. Portanto uma forma eficiente de solução desses sistemas lineares é indispensável para resolver problemas de grande porte.

A matriz de coeficientes do sistema [Gondzio 2012] é uma matriz indefinida, mal condicionada, esparsa, portanto usamos um método iterativo precondicionado: o método do gradiente bi-conjugado estabilizado para matrizes esparsas, com precondicionador baseado nas entradas da diagonal [Saad 2003].

2.5. Reescalonamento do problema

Uma grande dificuldade é quando, no método de pontos interiores barreira logarítmica primal-dual, os diferentes valores das variáveis com as quais estamos trabalhando atingem valores pequenos causando *underflow*, levando a problemas estabilidade e arredondamento [Trefethen and Bau III 1997, Ruggiero and da Rocha Lopes. 1997]. Dada a ordem do polinômio do problema, os principais valores que devemos analisar e acompanhar são a função de verossimilhança, o valor do gradiente e o valor da Hessiana. Portanto, nossa proposta é manter os valores dessas grandezas e os cálculos que os envolvem controlados, tanto para diminuir os erros de arredondamento, como para evitar problemas de *underflow*. Assim, quando for preciso, multiplicamos a função objetivo por uma constante $C \gg 0$. Basicamente estamos re-escalando os valores numéricos para resolver estes inconvenientes.

3. Resultados e discussões

Nas tabelas a seguir apresentamos os resultados obtidos para as seis sub-gramáticas detalhadas nas Tabelas 1 e 2. Apresentamos o tamanho da amostra usada, o número de iterações e o tempo que utilizou o método até convergir, assim como o valor da constante C. Para as primeiras três sub-gramáticas (ver Tabela 1) usamos uma amostra de tamanho 67, com comprimento das sentenças entre quatro e cinco. Para o segundo grupo de sub-gramáticas (ver Tabela 2) usamos amostras de diferentes tamanhos e sentenças de comprimento entre quatro e quinze.

	$\|\Omega\|$	Iterações até convergir	Tempo	Vlr da constante C
Sub-Problema 1	67	9	34m45,830s	10^{20}
Sub-Problema 2	67	9	335m13,625s	10^{15}
Sub-Problema 3	67	9	4558m27,323s	10^{10}

Tabela 3. Resultados obtidos para o primeiro grupo de sub-problemas.

Da Tabela 3 podemos evidenciar que, uma vez obtida a convergência para um problema com m símbolos não terminais, podemos aumentar este valor e utilizando a estratégia para controlar problemas de *underflow*, atingimos a convergência do método.

A Tabela 4 apresenta um aumento do tamanho da amostra e do comprimento das sentenças da mesma, havendo um maior tamanho dos problemas. O método atinge convergência e

	$\|\Omega\|$	Iterações até convergir	Tempo	Vlr da constante C
Sub-Problema 4	161	9	127m19,235s	10^{25}
Sub-Problema 5	388	9	407m19,538s	10^{25}
Sub-Problema 6	310	9	9519m2,029s	10^{15}

Tabela 4. Resultados obtidos para o segundo grupo de sub-problemas.

como no caso do primeiro grupo de testes, o acompanhamento garante que não se gerem problemas de *underflow*, isto basicamente é controlado com o parâmetro C.

4. Conclusão

O método de pontos interiores mostrou-se um método viável para estimar as gramáticas probabilísticas livres do contexto. Um dos detalhes relevantes é que o número de iterações até atingir convergência é mantido em nove iterações, isto quer dizer que o método é estável para diferentes tamanhos tanto da gramática como do *corpus*. Isto é importante porque uma das maiores desvantagens do Método *Inside-Outside* [Baker 1979] em aplicações práticas é o elevado número de iterações requeridas para atingir a convergência.

O método de pontos interiores implementado é robusto e bem comportado na presença de novos dados, sendo uma proposta que sugere continuar fazendo testes aumentando o tamanho da amostra, até conseguir modelar uma linguagem natural.

Referências

Baker, J. K. (1979). Trainable grammars for speech recognition. *The Journal of the Acoustical Society of America*, 65(S1):S132–S132.

El-Bakry, A., Tapia, R., Tsuchiya, T., and Zhang, Y. (1996). On the formulation and theory of the newton point-pnterior for nonlinear programming. *Journal of Optimization Theory and Applications*.

Gondzio, J. (2012). Interior point methods 25 years later. *European Journal of Operational Research*, 218(3):587–601.

IEL-UNICAMP and IME-USP. *Corpus anotado do Português histórico Tycho Brahe*. http://www.tycho.iel.unicamp.br/corpus/index.html, acessado em 2017.

López, E. S. M. (2018). *Método de pontos interiores para estimar os parâmetros de uma gramática probabilística livre do contexto.* Disertação doutorado, Universidade Estadual de Campinas. Instituto de matemática, Estatística e Computação Científica.

Manning, C. D. and Schutze., H. (2003). *Foundations of statistical natural language processing*. Cambridge, MA : MIT.

Ruggiero, M. A. G. and da Rocha Lopes., V. L. (1997). *Cálculo numérico. Aspectos teóricos e computacionais.* São Paulo, SP : Makron.

Saad, Y. (2003). *Iterative Methos for Sparse Linear Systems*. SIAM Publications, SIAM, Philadelphia, PA, USA.

Trefethen, L. N. and Bau III, D. (1997). *Numerical linear algebra*. Siam.