FUNDAÇÃO GETULIO VARGAS

ESCOLA DE MATEMÁTICA APLICADA - FGV/EMAp

CURSO DE GRADUAÇÃO EM MATEMÁTICA APLICADA

Semântica Computacional para Textos Normativos

por

Guilherme Paulino Passos

Rio de Janeiro

2015

FUNDAÇÃO GETULIO VARGAS

FUNDAÇÃO GETULIO VARGAS

ESCOLA DE MATEMÁTICA APLICADA - FGV/EMAp

CURSO DE GRADUAÇÃO EM MATEMÁTICA APLICADA

Semântica Computacional para Textos Normativos

”Declaro ser o único autor do presente projeto de monografia que refere-se ao plano de trabalho a ser executado para continuidade da monografia e ressalto que não recorri a qualquer forma de colaboração ou auxílio de terceiros para realizá-lo a não ser nos casos e para os fins autorizados pelo professor orientador”

Guilherme Paulino Passos

Orientador: Alexandre Rademaker

Rio de Janeiro

2015

GUILHERME PAULINO PASSOS

Semântica Computacional para Textos Normativos

"Projeto de Monografia apresentado à Escola de Matemática Aplicada - FGV/EMAp como requisito parcial para continuidade ao trabalho de monografia."

Aprovado em ___ de _____ de _____ .

Grau atribuido ao Projeto de Monografia: ___ .

Professor Orientador: Alexandre Rademaker

Escola de Matemática Aplicada

Fundação Getulio Vargas

Sumário

1 Introdução

1.1 Processamento de Linguagem Natural

O estudo computacional da linguagem, conhecido como Processamento de Linguagem Natural (PLN ou, pela sigla em inglês, NLP) ou Lingüística Computacional, é um campo de intenso desenvolvimento nas últimas décadas, tendo fortes impactos na tecnologia atual. Exemplos bem sucedidos são a Siri, um assistente do sistema operacional iOS que interage com o usuário utilizando linguagem natural; serviços de tradução automática, como o do Google, que apresentam constante melhora; e também diversas empresas relacionadas a inteligência de marketing ou empresarial (*marketing intelligence* e *business intelligence*) destinadas a fazer análise de dados a partir de textos em linguagem natural, isto é, textos escritos por humanos para se comunicar com outros humanos, usando uma linguagem desenvolvida naturalmente (e não uma linguagem artificial).

O uso de modelos matemáticos de diferentes formas e tradições foi um passo essencial para o desenvolvimento da ciência, bem como para a levar o conhecimento adquirido a aplicações. Historicamente, ocorreu uma tensão (ou, ao menos, um distanciamento) entre dois paradigmas em NLP: o simbólico e o probabilístico. Tal divisão existiu de modo particularmente notável do fim da década de 50 ao fim da de 60. Desta época, do paradigma simbólico participaram o trabalho de Noam Chomsky em linguagens formais e sintaxe gerativa, o trabalho de lingüistas e cientistas da computação em algoritmos de análise sintática (*parsing*), bem como os da área de inteligência artificial, como sistemas baseados em lógica formal e correspondência de padrões (*pattern matching*), influenciados pelo famoso *Logic Theorist* de Allen Newell, Herbert Simon e Cliff Shaw, um exemplo de sistema baseado em lógica e raciocínio automático. Na tradição estocástica, dois exemplos são o trabalho de Bledson e Browning de um sistema bayesiano para reconhecimento ótico de caracteres, bem como o uso de métodos bayesianos por Mosteller e Wallace para atribuir autoria de trechos d'*O Federalista*. Já nas décadas de 70 e 80, houve grande desenvolvimento de algoritmos de reconhecimento de fala, como o uso de Cadeias Ocultas de Markov. (Jurafsky and Martin, 2009, pp.10-11) A tensão entre as abordagens mostra sinais ainda hoje.

A lingüística possui diversas sub-áreas. Algumas delas são: a morfologia (o estudo da formação e composição de palavras), a sintaxe (o estudo de como as palavras se com-

binam para formar orações e frases), a semântica (o estudo do significado) e a pragmática (o estudo de como o contexto influencia no significado). Nesse trabalho, olharemos particularmente para a semântica.

1.2 Textos Normativos

A palavra "norma" não é daquelas de significado mais claro. Entretanto, explicações de seu sentido normalmente recorrem às idéias de regra, comando, obrigação, dever ou, de modo mais geral, a alguma orientação para acreditar, agir ou sentir.[1] Desse modo, podemos dizer que textos normativos são textos cujo conteúdo é normativo, ou que tratam de normas. Não faltam exemplos de textos normativos em nosso cotidiano: contratos, acordos, promessas, ordens, textos que expressam críticas ou padrões de corretude (moral, estética), decisões judiciais, leis, etc.

A análise computacional desse tipo de texto busca o desenvolvimento de ferramentas úteis para os meios e práticas que se relacionam fortemente com normas. Um exemplo claro é o meio jurídico, o qual acreditamos que ainda usufrui muito pouco de avanços tecnológicos atuais. Exemplos de tarefas para os quais se espera que a análise semântica possa ser útil são a verificação de *compliance*, a de consistência entre leis, a adequação de contratos a outros documentos, a comparação entre decisões judiciais, etc.

1.3 Semântica Computacional

No estudo computacional da semântica, uma idéia central é a de que é possível capturar o significado de expressões de linguagem natural a partir de estruturas formais. Esta área é conhecida por *semântica formal.* A idéia é relacionar estruturas lingüísticas com conhecimento sobre o mundo, que é representado de alguma maneira. São todas questões da semântica formal: a escolha de qual o modo de representar, quais as propriedades da representação e como associar palavras e frases a estruturas. O uso de estruturas formais tem utilidade para lingüistas por permitir que discutam significado de modo mais rigoroso, menos ambíguo. Esta tradição deriva diretamente dos trabalhos de Richard Montague. (Blackburn and Bos, 2005, p. xii)

[1] Para um clássico da análise filosófica sobre normas, bem como um texto de grande importância para a lógica deôntica, veja von Wright (1963)

Entretanto, um modo de expandir essa análise é caminhar em direção à chamada *semântica computacional* que, de modo sucinto, é a área que busca realizar as tarefas da semântica formal por uso de um computador. Isso expande a utilidade de modelos formais para além da análise por um humano. As representações formais tornam possível que um computador consiga acessar o significado e trabalhar com ele, o utilizando para finalidade distintas. Em especial, para a atividade de *inferência*, isto é, tornar explícita informação que estava implícita. Portanto, são objetivos centrais da área a automatização de construção de representações a partir de textos em linguagem natural, bem como a automatização da extração de inferências a partir de representações formais já feitas.

1.4 Implicação textual

Um problema atual motivador para a semântica computacional é o de *implicação textual* (*textual entailment*). Dados dois fragmentos de texto, a tarefa é reconhecer se o significado de um pode ser inferido a partir do significado do outro. Mais especificamento, dado um par de expressões textuais — T, o texto base, e H, a hipótese — dizemos que T acarreta H se o significado de H pode ser inferido do significado de T, de acordo com o que seria tipicamente interpretado por falantes da língua. (Dagan et al., 2006, p.1)

Dois exemplos são:

Texto	Hipótese	Implicação Textual
Sessões no Clube Caverna pagaram aos Beatles £15 à noite e £5 na hora do almoço.	Os Beatles tocaram no Clube Caverna na hora do almoço.	Verdadeiro
A American Airlines começou a demitir centenas de comissários de bordo na terça-feira após um juiz ter rejeitado a proposta da União de bloquear as perdas de empregos.	A American Airlines chamará de volta centenas de comissários de bordo para aumentar o número de vôos que opera.	Falso

2 Desenvolvimento da Abordagem

No trabalho desenvolvido até agora, seguimos o desenvolvimento de Blackburn and Bos (2005).

2.1 Lógica de Primeira Ordem

Um modo de representação que possui diversas propriedades desejáveis é o uso de uma linguagem lógica como, por exemplo, *lógica de primeira ordem* (abreviado como *FOL*, de *First Order Logic*). Jurafsky and Martin (2009) apresentam como propriedades interessantes para representações: verificabilidade, representações não ambíguas, existência de uma forma canônica, capacidade de inferência e uso de variáveis e expressividade. Todas estas são possuídas pela lógica de primeira ordem.

2.2 Cálculo Lambda

O princípio da composicionalidade é aquele segundo o qual o significado de uma expressão complexa é uma função do significado das partes que a constituem. Tal princípio possui raízes em Gottlob Frege, na filosofia da linguagem. (Blackburn and Bos, 2005, p.94) Não é, contudo, sem controvérsia.[2] Entretanto, além de ser intuitivamente plausível, é possível construir sistemas interessantes a partir dele. É de se notar que o princípio não nos diz *como* é essa função, de modo a construir o conteúdo semântico da expressão complexa. A abordagem estudada é a de que ela pode ser montada a partir da *estrutura sintática* da expressão. Portanto, a análise sintática das expressões é elemento importante para nossa análise semântica. Apesar disso, o enfoque desse trabalho será na semântica, de modo que um modelo sintático simples será adotado. Aqui, será usado o modelo de Gramática de Cláusulas Definidas, de modo que a sintaxe pode ser modelada como uma gramática livre de contexto.

O princípio da composicionalidade sugere e se relaciona de modo natural com uma estrutura que reflita o conceito de composição entre expressões. Com efeito, o exemplo a ser seguido aqui é o formalismo do *cálculo lambda*. Este é uma extensão das expressões

[2]Para um exemplo de crítica a tal princípio, destacando exemplos em que uma mesma palavra, ao ser combinada com outras, gera um significado distinto que não pode ser (ao menos claramente) explicado apenas pelas partes, vide Manning and Schütze (1999, pp.110, 151).

de primeira ordem, inserindo os símbolos λ (acompanhado de uma variável), representando uma abstração, e @, que representa uma aplicação, isto é, uma substituição de uma variável abstraída por uma outra expressão. Isso é melhor compreendido com um exemplo simples. Uma expressão de lambda calculus é:

$$\lambda x.matar(caim,x)$$

Essa expressão abstrai sobre o objeto x. Podemos transformá-la em uma verdadeira expressão de primeira ordem aplicando *abel* a ela. Isto é:

$$\lambda x.matar(caim,x)@abel$$

Estamos aplicando *abel* sobre a abstração em x. Assim, retirandos o prefixo $\lambda x.$ e substituímos todas as ocorrências de x por $abel$. Assim, temos:

$$matar(caim,abel)$$

Essa operação é chamada *redução beta*, ou *conversão beta*. É interessante fazer um novo exemplo para mostrar as operações podem ser mais complexas. Cada etapa representa um passo da redução beta.

$$\lambda x.(x@abel)@\lambda y.homem(y)$$
$$\lambda y.homem(y)@abel$$
$$homem(abel)$$

Em uma beta redução do formato $\mathcal{F}@\mathcal{A}$, $\mathcal{F}$ é dito o *funtor* e $\mathcal{A}$ é o *argumento*. Vale notar que o símbolo usado na variável sendo abstraída é irrelevante. Assim, $\lambda x.homem(x)$ é equivalente a $\lambda y.homem(y)$. Tais expressões são ditas *alfa-equivalentes*.

Relativa a isso, outra operação relevante é a de *conversão alfa*. Se aplicarmos a conversão beta do modo que descrevemos, teremos problemas de captura de variável. Desse modo, a conversão alfa a uma expressão $\mathcal{E}$ é a operação de achar uma expressão alfa-equivalente a $\mathcal{E}$, usando apenas variáveis novas. Mais especificamente, antes de realizar uma redução beta, aplica-se a conversão alfa ao funtor, de modo que este não tenha nenhuma variável ligada que seja representada pelo mesmo símbolo que uma variável no argumento (seja a variável do funtor ligada pelo prefixo λ, seja por algum dos quantificadores de primeira ordem $\forall$ ou $\exists$).

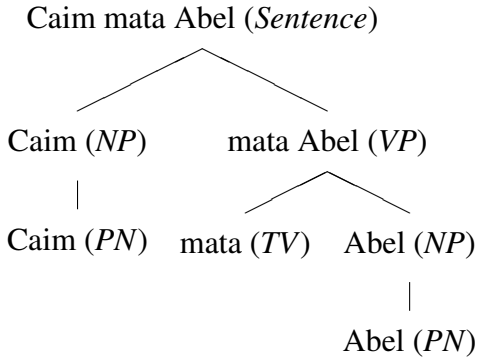
Agora vejamos como o cálculo lambda pode ser usado para representar o princípio da composicionalidade, ligando sintaxe a semântica. Suponha que tenhamos uma gramática que pode ser representada pela seguinte gramática de cláusulas definidas:

$$s \rightarrowtail np, vp \qquad vp \rightarrowtail tv, np$$
$$np \rightarrowtail pn \qquad tv \rightarrowtail [mata]$$
$$pn \rightarrowtail [Caim] \qquad pn \rightarrowtail [Abel]$$

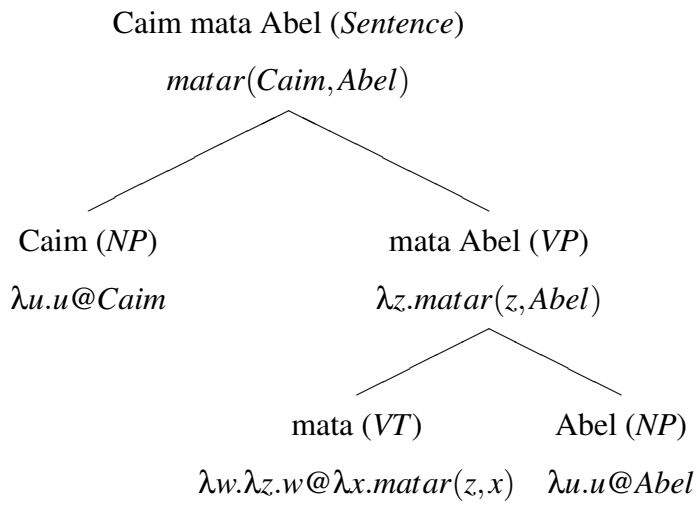
Aqui, *s* representa frase (*sentence*); *np*, sintagma nominal (*noun phrase*), *vp*, sintagma verbal (*verbal phrase*); *pn*, nome próprio (*proper name*) e *tv*, verbo transitivo (*transitive verb*). Podemos mostrar que a frase "*Caim mata Abel*" é gramatical, para essa estrutura. Sendo *s* o símbolo de início:

$$s \Rightarrow np\ vp \Rightarrow pn\ vp \Rightarrow Caim\ vp \Rightarrow Caim\ tv\ np \Rightarrow$$
$$Caim\ mata\ np \Rightarrow Caim\ mata\ pn \Rightarrow Caim\ mata\ Abel$$

Na forma de árvore de *parsing*, temos:



Agora, o que nos importa é associar a cada folha dessa árvore uma expressão de cálculo lambda, de modo que, subindo a árvore, o nó pai seja uma aplicação de um de seus filhos no outro. Assim é para o exemplo abaixo:



A partir daí, são desenvolvidas técnicas para lidar com certos tipos de ambigüidade: as chamadas ambigüidades de escopo. Em apertada síntese, são ambigüidades relativas à posição dos quantificadores na representação lógica. Os métodos aqui usados são baseados em armazenamentos, que a cada expressão na construção semântica (e na decomposição sintática), atribui não um único significado possível (isso é, uma única representação), mas um conjunto de significados.

3 Próximos Passos

Até então, estamos estudando o que já foi desenvolvido para semântica formal. Entretanto, uma série de perguntas emergem: Existem ferramentas análogas para o português? Por exemplo, existe uma representação gramatical do português como uma linguagem livre de contexto? Podemos usar representações gramaticais mais expressivas do que gramáticas livres de contexto mantendo os métodos aprendidos? Além de representação gramatical, é preciso de conhecimento léxico. Existem ferramentas já disponíveis para isso no caso do português? Uma vez que a capacidade de fazer boas inferências depende de uma boa represesentação de (bastante) conhecimento prévio, qual seria um bom banco para usarmos? O bom uso integrado dessas ferramentas produz bons resultados para o problema da inferência textual?

Além disso, nos resta integrar o aprendido com o caso particular dos textos normativos. Qual é um bom modo de representar o conteúdo normativo? Quais as fraquezas e os sucessos de sistemas de lógica deôntica disponíveis? Podemos integrar alguma dessas com a abordagem da semântica computacional para produzir boas inferências em sistemas normativos?

Para além da representação semântica de frases individualmente, uma continuação do estudo de semântica computacional levaria às representações de discurso. A teoria de representação de discurso (*Discourse Representation Theory*) é uma proposta que expande da semântica à pragmática, permitindo que o significado seja extraído com base também em contexto. Dois livros que apresentam tal teoria são: Blackburn and Bos (Não publicado) e van Eijck and Unger (2010).

4 Referências

Patrick Blackburn and Johan Bos. *Representation and Inference for Natural Language. A First Course in Computational Semantics*. CSLI, 2005.

Patrick Blackburn and Johan Bos. *Working with Discourse Representation Theory. An Advanced Course in Computational Semantics*. Não publicado. URL `http://www.let.rug.nl/bos/comsem/book2.html`.

Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In *Proceedings of the First International Conference on Machine Learning Challenges: Evaluating Predictive Uncertainty Visual Object Classification, and Recognizing Textual Entailment*, MLCW'05, pages 177–190, Berlin, Heidelberg, 2006. Springer-Verlag. ISBN 3-540-33427-0, 978-3-540-33427-9. doi: 10.1007/11736790_9. URL `http://dx.doi.org/10.1007/11736790_9`.

Daniel Jurafsky and James H. Martin. *Speech and Language Processing (2nd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2009. ISBN 0131873210.

Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, USA, 1999. ISBN 0-262-13360-1.

Jan van Eijck and Christina Unger. *Computational Semantics with Functional Programming*. Cambridge University Press, New York, NY, USA, 1st edition, 2010. ISBN 0521760305, 9780521760300.

Georg Henrik von Wright. *Norm and action : a logical enquiry*. Routledge & Kegan Paul ; Humanities Press London : New York, 1963.