

Raciocínio Probabilístico Tratável em um Fragmento da Lógica de Descrição

MAC0215 - Atividade Curricular em Pesquisa

Andrew Ijano Lopes

Orientador: Marcelo Finger

Instituto de Matemática e Estatística da Universidade de São Paulo

Introdução

As lógicas de descrição são uma família de linguagens formais para a representação do conhecimento. Dentre as várias propostas existentes, a lógica \mathcal{EL}^{++} se mostrou uma das lógicas de descrição mais expressivas cuja complexidade de inferência é decidível em tempo polinomial. [3] Porém, ao adicionar probabilidade às capacidades de inferência seu problema de decisão se torna NP-Completo. [4]

Com um estudo mais recente, está sendo proposto que, ao restringir a expressividade do \mathcal{EL}^{++} , removendo a conjunção, é possível construir uma lógica de descrição probabilística tratável.

Nesse projeto, será usado a linguagem *Graphic \mathcal{EL}^{++}* (\mathcal{GEL}^{++}), uma restrição da linguagem \mathcal{EL}^{++} , removendo as conjunções.

$\text{Fever} \sqsubseteq \text{Symptom}$	$\{\text{john}\} \sqsubseteq \text{Patient}$
$\text{FeverAndRash} \sqsubseteq \text{Fever}$	$\{\text{s1}\} \sqsubseteq \text{FeverAndRash}$
$\text{Dengue} \sqsubseteq \text{Disease}$	$\{\text{john}\} \sqsubseteq \exists \text{hasSymptom}.\{\text{s1}\}$
$\text{Symptom} \sqsubseteq \exists \text{hasCause}.\text{Disease}$	$\text{DenguePatient} \equiv \exists \text{suspectOf}.\text{Dengue}$
$\text{Patient} \sqsubseteq \exists \text{hasSymptom}.\text{Symptom}$	
$\text{hasSymptom} \circ \text{hasCause} \sqsubseteq \text{suspectOf}$	

Figura 1: Exemplo da representação de um caso médico com uma ontologia em \mathcal{GEL}^{++} .

Objetivos

O objetivo da pesquisa é realizar estudos sobre diversos conceitos para conseguir propor um algoritmo de satisfatibilidade, estudar sua complexidade e implementar esse algoritmo tratável para um fragmento de uma lógica de descrição probabilística.

Para o escopo dessa disciplina, o projeto abrange a implementação do início do algoritmo sem a inclusão de probabilidades: leitura da ontologia e a resolução do problema de satisfatibilidade máxima ($\mathcal{GEL}^{++}\text{-MaxSAT}$). Esse algoritmo será aplicado na linguagem \mathcal{GEL}^{++} .

Metodologia

Na primeira etapa do projeto, foi implementado a identificação e leitura de uma ontologia na linguagem \mathcal{GEL}^{++} , codificada no formato OWL. Esse processo envolve observar se a ontologia se enquadra na expressividade do \mathcal{GEL}^{++} e, em caso afirmativo, analisar cada componente, gerando o grafo correspondente. Para essa etapa, foi utilizada a biblioteca OWL API [2], em Java.

Em seguida, foi implementado uma interface entre o programa de leitura, em Java, com o algoritmo de satisfatibilidade, em C++, usando a *Java Native Interface* (JNI) [1]. Além disso, foi implementada a estrutura de dados equivalente ao grafo em Java para armazenar a ontologia e realizar o algoritmo.

Ainda, foi feito o início do algoritmo de cortes mínimos. Um segundo modelo de grafos, apenas com pesos nas arestas, foi implementado e está sendo estudado as aplicações das técnicas de fluxo máximo e corte mínimo no grafo. Disso, a implementação do $\mathcal{GEL}^{++}\text{-MaxSATSolver}$, algoritmo que resolve o problema de satisfatibilidade máxima, é facilmente produzida.

Discussões

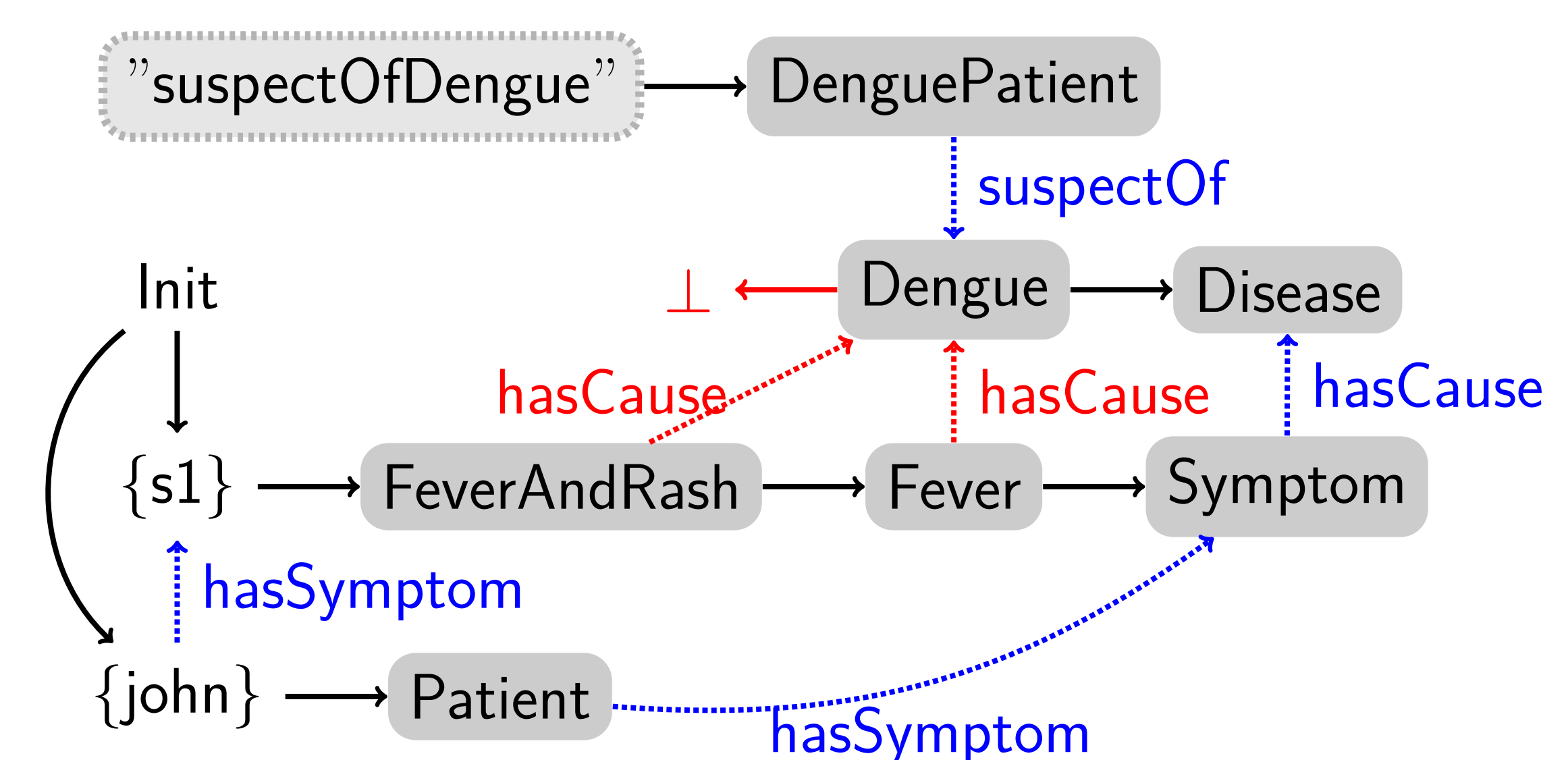


Figura 2: Exemplo da representação de uma ontologia em \mathcal{GEL}^{++} em grafo. Arcos contínuos correspondem à subsumção e os rótulos de arcos pontilhados são propriedades. Numa implementação com probabilidades, os arcos vermelhos indicam que o arco tem probabilidade.

Como citado nos objetivos do projeto, essa pesquisa aplicou métodos para leitura de ontologias em \mathcal{GEL}^{++} , implementou estruturas de dados para armazenar ontologias dessa família de linguagens e resolver problemas de corte mínimo em grafos com pesos. Dentre os objetivos previstos, a resolução do problema de $\mathcal{GEL}^{++}\text{-MaxSAT}$ ainda está em andamento e está sendo implementada junto com o problema de corte mínimo.

Durante o desenvolvimento do projeto, foi despendido um tempo muito maior que o esperado na etapa de leitura da linguagem. Uma das maiores problemáticas dessa parte foi a falta de documentação atualizada sobre a OWL API, aumentando o tempo de aprendizagem no uso da biblioteca.

Para fora do escopo da disciplina, o projeto ainda visa o estudo da representação de probabilidades em lógicas de descrição e a implementação de um algoritmo de satisfatibilidade para o \mathcal{GEL}^{++} probabilístico utilizando o $\mathcal{GEL}^{++}\text{-MaxSATSolver}$.

Referências

- [1] Jni. <https://docs.oracle.com/javase/8/docs/technotes/guides/jni/>.
- [2] Site do owl api. <http://owlcs.github.io/owlapi/>.
- [3] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the EL envelope. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence, IJCAI'05*, pages 364–369, 2005.
- [4] Marcelo Finger. Extending \mathcal{EL}^{++} with linear constraints on the probability of axioms. In Carsten Lutz, Uli Sattler, Cesare Tinelli, Anni-Yasmin Turhan, and Frank Wolter, editors, *Description Logic, Theory Combination, and All That*, volume LLNCS 11560 of *Theoretical Computer Science and General Issues*. Springer International Publishing, 2019.