Структурное представление результата поиска путей с контекстно-свободными ограничениями

Семён Григорьев

JetBrains Research, лаборатория языковых инструментов
Санкт-Петербургский государственный университет

21.10.2017

Поиск путей в графах

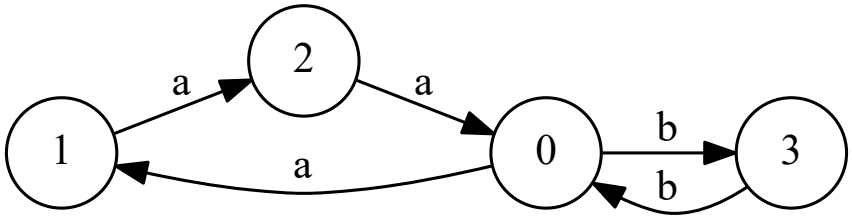
- Анализ графов
  - Запросы к графовым базам данных
  - Анализ сетей (социальных, интернет и т.д.)
- Статический анализ программ
  - Анализ алиасов
  - Taint analysis
  - Статический анализ динамически формируемого кода
- ...

Поиск путей с контекстно-свободными ограничениями

- $\mathbb{G} = (\Sigma, N, P)$ — контекстно-свободная грамматика
- $p = v_0 \xrightarrow{l_0} v_1 \xrightarrow{l_1} \cdots v_{n-1} \xrightarrow{l_{n-1}} v_n$ — путь в графе G
- $w(p) = w(v_0 \xrightarrow{l_0} v_1 \xrightarrow{l_1} \cdots v_{n-1} \xrightarrow{l_{n-1}} v_n) = l_0 l_1 \cdots l_{n-1}$
- $G = (V, E, L)$ — ориентированный граф, $E \subseteq V \times L \times V$, $L \subseteq \Sigma$
- $R = \{p \mid \exists N_i \in N(w(p) \in L(\mathbb{G}, N_i))\}$
  - Стартовый нетерминал можно зафиксировать заранее
  - **Проблема:** множество R может быть бесконечным

Пример

Входной граф



Запрос — грамматика G для языка $L = \{a^n b^n; n \geq 1\}$ с явным выделением середины пути

$$
\begin{aligned}
0: &\quad S \to a\ S\ b \\
1: &\quad S \to Middle \\
2: &\quad Middle \to a\ b
\end{aligned}
$$

Ответ — бесконечное множество путей

- $p_1 = 0 \xrightarrow{a} 1 \xrightarrow{a} 2 \xrightarrow{a} 0 \xrightarrow{b} 3 \xrightarrow{b} 0 \xrightarrow{b} 3$
- $p_2 = 0 \xrightarrow{a} 1 \xrightarrow{a} 2 \xrightarrow{a} 0 \xrightarrow{a} 1 \xrightarrow{a} 2 \xrightarrow{a} 0 \xrightarrow{b} 3 \xrightarrow{b} 0 \xrightarrow{b} 3 \xrightarrow{b} 0 \xrightarrow{b} 3 \xrightarrow{b} 0$
- ...

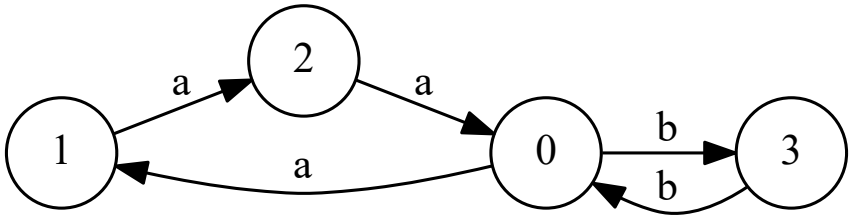
Существующие решения

- В качестве основы используют алгоритм CYK
  - Необходимо преобразовывать грамматику в нормальную форму Хомского
  - Только проверка наличия пути (*Zhang X. et al.* “Context-free path queries on RDF graphs.” 2016)
  - Грамматика в качестве представления результата (*Hellings J.* “Conjunctive context-free path queries.” 2014)

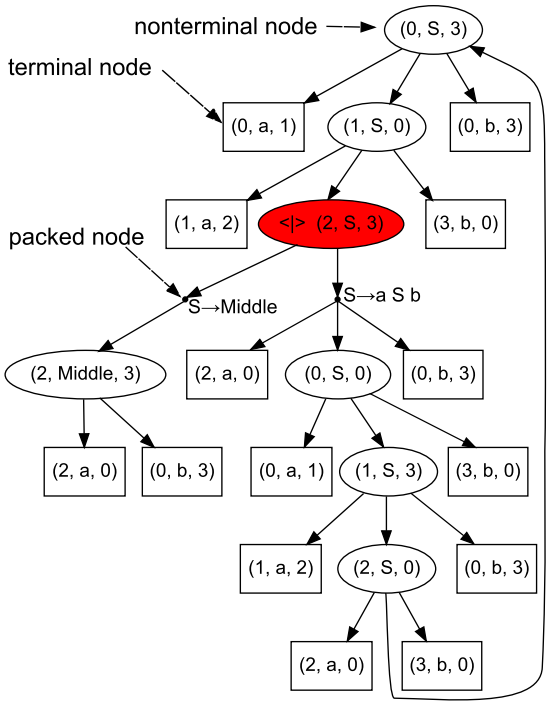
Алгоритм выполнения КС запросов к графам

- Основан на обобщённом LL (Generalized GLL, GLL)
  - *Scott E., Johnstone A.* "GLL parsing"
- Поддерживает произвольные контекстно-свободные граммтики (неоднозначные, леворекурсивные)
- Не требует преобразования грамматики в нормальную форму Хомского
- Строит сжатое представление леса разбора (Sharep Packed Parse Forest, SPPF) — конечное представление бесконечного ответа

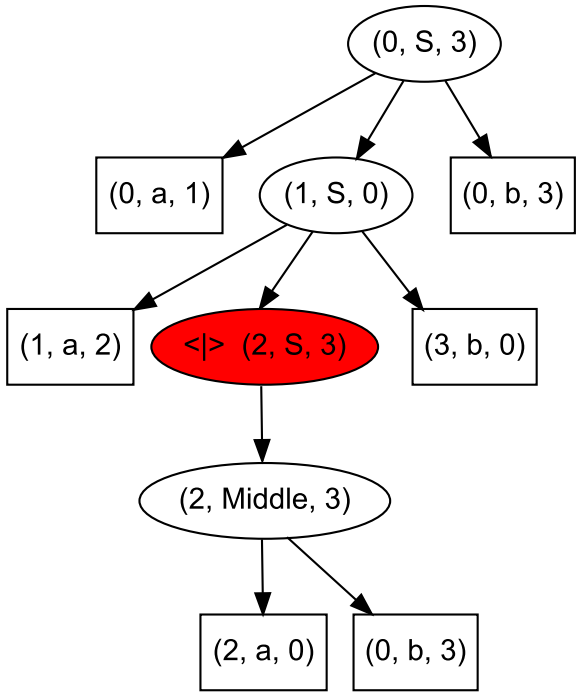
Структурное представление результата запроса



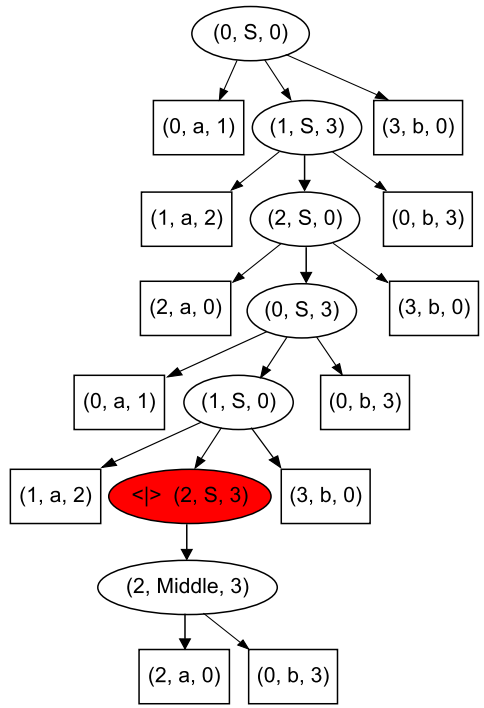
Входной граф



Результат (SPPF)

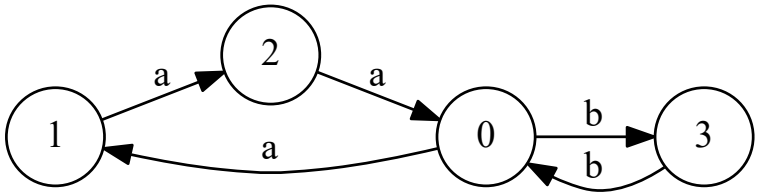


Дерево вывода пути p_1



Дерево вывода пути p_2

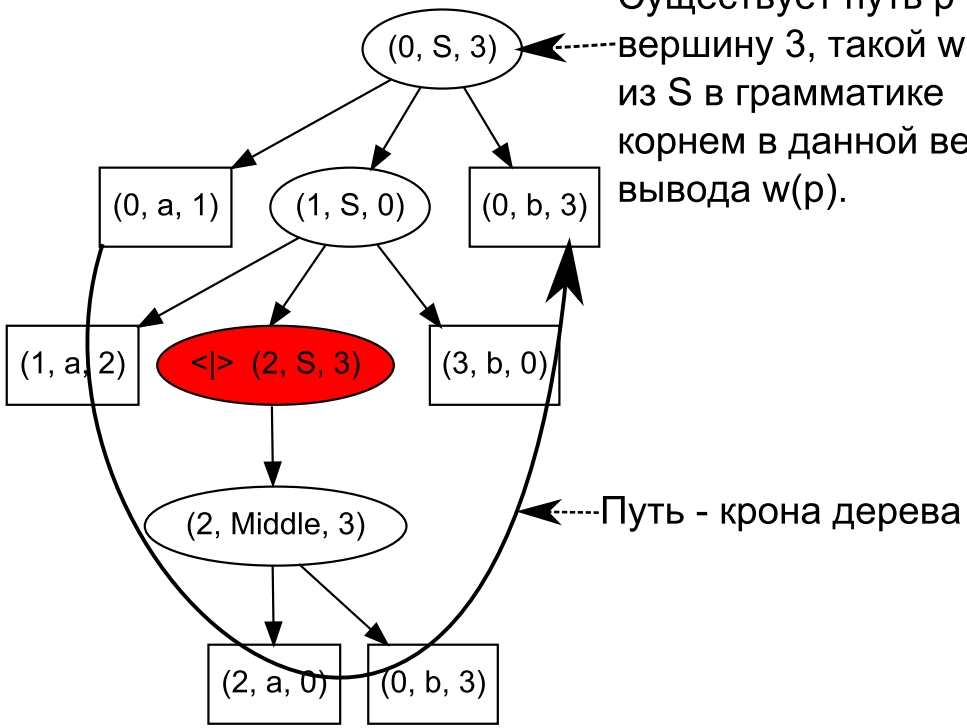
Пример: извлечение путей



$0: \quad S \to a\ S\ b$

$1: \quad S \to Middle$

$2: \quad Middle \to a\ b$



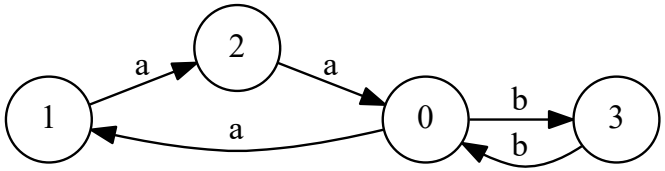
Существует путь p из вершины 0 в вершину 3, такой w(p) что выводима из S в грамматике G. Поддерево с корнем в данной вершине - дерево вывода w(p).

Путь - крона дерева

Путь: $0 \xrightarrow{a} 1 \xrightarrow{a} 2 \xrightarrow{a} 0 \xrightarrow{b} 3 \xrightarrow{b} 0 \xrightarrow{b} 3$

Почему это работает

Замкнутость КС языков относительно пересечения с регуляными



Регулярный язык

$0: \quad S \to a\ S\ b$

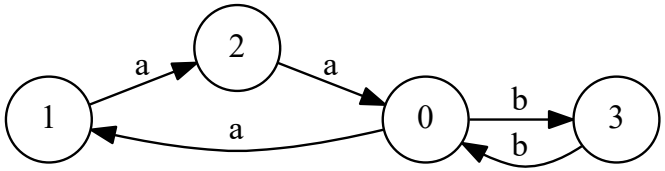
$1: \quad S \to Middle$

$2: \quad Middle \to a\ b$

Контекстно-свободный язык

Почему это работает

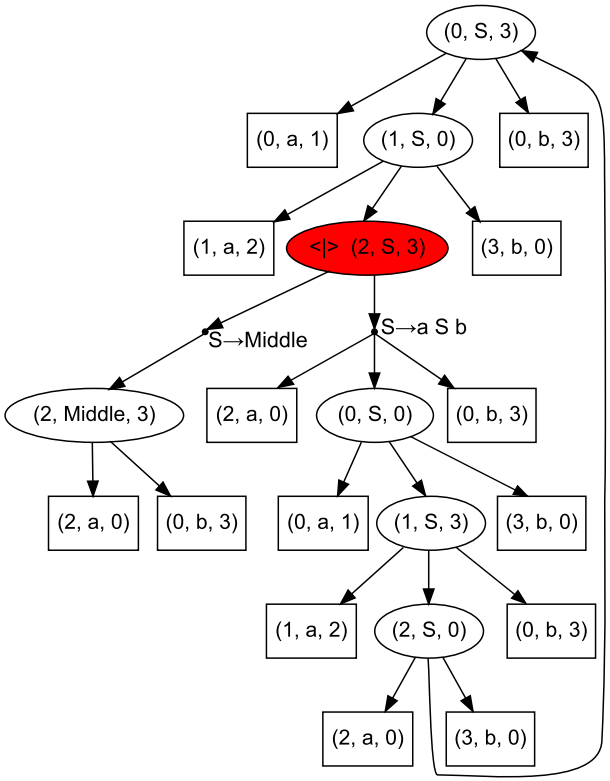
Замкнутость КС языков относительно пересечения с регуляными



Регулярный язык

$$
\begin{aligned}
0: &\quad S \to a\ S\ b \\
1: &\quad S \to Middle \\
2: &\quad Middle \to a\ b
\end{aligned}
$$

Контекстно-свободный язык



$$
\begin{aligned}
(0, S, 3) &\to (0, a, 1)\ (1, S, 0)\ (0, b, 3) \\
(1, S, 0) &\to (1, a, 2)\ (2, S, 3)\ (3, b, 0) \\
(2, S, 3) &\to (2, a, 0)\ (0, S, 0)\ (0, b, 3) \\
(2, S, 3) &\to (2, Middle, 3) \\
(0, S, 0) &\to (0, a, 1)\ (1, S, 3)\ (3, b, 0) \\
(1, S, 3) &\to (1, a, 2)\ (2, S, 0)\ (0, b, 3) \\
(2, S, 0) &\to (2, a, 0)\ (0, S, 3)\ (3, b, 0) \\
(0, Middle, 3) &\to (2, a, 0)\ (0, b, 3)
\end{aligned}
$$

Свойства алгоритма

Пусть на входе граф $M = (V, E, L)$, тогда

- Пространственная сложность предложенного алгоритма $O(|V|^3 + |E|)$
- Временная сложность предложенного алгоритма $O\left(|V|^3 * \max_{v \in V}(deg^+(v))\right)$
- Результирующий SPPF имеет размер $O(|V'|^3 + |E'|)$, где $M' = (V', E', L')$ — подграф M, содержащий только искомые пути

Экспериментальное исследование: запросы

$$
\begin{aligned}
0: &\quad \mathbf{S} \to \mathit{subClassOf}^{-1}\ \mathbf{S}\ \mathit{subClassOf} \\
1: &\quad \mathbf{S} \to \mathit{type}^{-1}\ \mathbf{S}\ \mathit{type} \\
2: &\quad \mathbf{S} \to \mathit{subClassOf}^{-1}\ \mathit{subClassOf} \\
3: &\quad \mathbf{S} \to \mathit{type}^{-1}\ \mathit{type}
\end{aligned}
$$

Грамматика для запроса Query 1

$$
\begin{aligned}
0: &\quad \mathbf{S} \to \mathbf{B}\ \mathit{subClassOf} \\
1: &\quad \mathbf{S} \to \mathit{subClassOf} \\
2: &\quad \mathbf{B} \to \mathit{subClassOf}^{-1}\ \mathbf{B}\ \mathit{subClassOf} \\
3: &\quad \mathbf{B} \to \mathit{subClassOf}^{-1}\ \mathit{subClassOf}
\end{aligned}
$$

Грамматика для запроса Query 2

Экспериментальное исследование: результаты

Ontology	#edg	Query 1			Query 2	
		time CYK[1] (ms)	time (ms)	#result	time (ms)	#result
skos	252	1044	10	810	1	1
generations	273	6091	19	2164	1	0
travel	277	13971	24	2499	1	63
univ-bench	293	20981	25	2540	11	81
people-pets	640	82081	89	9472	3	37
atom-primitive	425	515285	255	15454	66	122
biomedical-measure-primitive	459	420604	261	15156	45	2871
pizza	1980	3233587	697	56195	29	1262
wine	1839	4075319	819	66572	8	133

[1] Zhang, et al. "Context-free path queries on RDF graphs."

Результаты

Разработан алгоритм поиска путей с контекстно-свободными ограничениями в графе

- Строит конечное структурное представление результата
- Пространственная сложность $O(|V|^3 + |E|)$
- Временная сложность $O\left(|V|^3 * \max_{v \in V}(deg^+(v))\right)$
- Достаточно производителен для решения практических задач
- Может применяться для выполнения запросов к графовым БД, решения задач статического анализа кода и других задач анализа графов

Контакты

- Почта: semen.grigorev@jetbrains.com
- GitHub-сообщество YaccConstructor: https://github.com/YaccConstructor