Context-Free Path Querying by Matrix Multiplication

Matrix-based algorithm for graph structured data analysis

Rustam Azimov[1], Semyon Grigorev[1],
[1] *Saint Petersburg State University, JetBrains, St. Petersburg, Russia*
E-mails: *st013567@student.spbu.ru, s.v.grigoriev@spbu.ru*

Motivation

Context-free path querying is a technique, which recently gains popularity in many areas, for example, graph databases, bioinformatics, static analysis, etc. In some of these areas, it is often required to query large graphs, and existing algorithms demonstrate a poor performance in this case. The generalization of matrix-based Valiant's context-free language recognition algorithm for graph case is widely considered as a recipe for efficient context-free path querying; however, no progress has been made in this direction so far.

We propose the first generalization of matrix-based Valiant's algorithm for context-free path querying. Our generalization does not deliver a truly sub-cubic worst-case complexity algorithm, whose existence still remains a hard open problem in the area. On the other hand, the utilization of matrix operations (such as matrix multiplication) in the process of context-free path query evaluation makes it possible to efficiently apply a wide class of optimizations and computing techniques, such as GPGPU (General-Purpose computing on Graphics Processing Units), parallel processing, sparse matrix representation, distributed-memory computation, etc. Indeed, the evaluation on a set of conventional benchmarks shows, that our algorithm outperforms the existing ones.

Context-free path querying

Matrix-based approach

References

[1] X. Zhang, Z. Feng, X. Wang, G. Rao, and W. Wu. Context-free path queries on rdf graphs. In *International Semantic Web Conference*, pages 632–648. Springer, 2016.

[2] Semyon Grigorev and Anastasiya Ragozina. Context-free path querying with structural representation of result. *arXiv preprint arXiv:1612.08872*, 2016.

Results

- We propose the matrix-based algorithm for context-free path querying.
- We implemented this algorithm with a number of optimizations and applied these implementations to the navigation query problem for some popular ontologies, taken from [1].
- We also compared the performance of our implementations with the fastest analog from [2].

Performance comparison of CF path querying algorithms

- sCPU (sparse CPU) — an implementation of our algorithm using the CSR format for sparse matrix representation and a CPU for matrix operation calculation. For sparse matrix representation in CSR format we used the Math.Net Numerics package.
- sGPU (sparse GPU) — an implementation of our algorithm using the CSR format for sparse matrix representation and a GPU for matrix operation calculation. For calculations of the matrix operations on a GPU, where matrices represented in a CSR format, we used a wrapper for the CUSPARSE library from the managedCuda library.

Query 1 is based on the grammar G for retrieving the concepts on the same layer. The running time of the algorithms is presented in ms.

Ontology	V	E	#results	GLL	sCPU	sGPU
skos	144	323	810	10	14	12
generations	129	351	2164	19	20	13
travel	131	397	2499	24	22	30
univ-bench	179	413	2540	25	25	15
atom-primitive	291	685	15454	255	92	22
biomedical	341	711	15156	261	113	20
foaf	256	815	4118	39	48	9
people-pets	337	834	9472	89	142	32
funding	778	1480	17634	212	447	36
wine	733	2450	66572	819	797	54
pizza	671	2604	56195	697	430	24
g_1	6224	11840	141072	1926	26957	82
g_2	5864	19600	532576	6246	46809	185
g_3	5368	20832	449560	7014	24967	127

Query 2 is based on the grammar G_S^2 for retrieving concepts on the adjacent layers.

Ontology	V	E	#results	GLL	sCPU	sGPU
skos	144	323	1	1	2	1
generations	129	351	0	1	2	0
travel	131	397	63	1	7	10
univ-bench	179	413	81	11	15	9
atom-primitive	291	685	122	66	9	2
biomedical	341	711	2871	45	91	24
foaf	256	815	10	2	14	3
people-pets	337	834	37	3	38	6
funding	778	1480	1158	23	344	27
wine	733	2450	133	8	179	6
pizza	671	2604	1262	29	258	23
g_1	6224	11840	9264	167	21115	38
g_2	5864	19600	1064	46	10874	21
g_3	5368	20832	10096	393	15736	40

Future Research

- Currently, we are working on the matrix-based algorithm for path querying with conjunctive grammars which have more expressive power, than context-free grammars.
- We want to find new applications for path querying techniques and implement required tools.

Acknowledgments

The research was supported by the Russian Science Foundation grant 18-11-00100 and a grant from JetBrains Research.

Information

All materials available on GitHub: https://github.com/YaccConstructor