Combinators for Single Source Context-Free Path Querying

Mikhail Nikilukin
Inria Paris-Rocquencourt
Rocquencourt, France
trovato@corporation.com

Ekaterina Verbitskaia
The Thørväld Group
Hekla, Iceland
larst@affiliation.org

Semyon Grigorev
Rajiv Gandhi University
Doimukh, Arunachal Pradesh, India
larst@affiliation.org

ABSTRACT

A clear and well-documented LaTeX document is presented as an article formatted for publication by ACM in a conference proceedings or journal publication. Based on the "acmart" document class, this article presents and explains many of the common variations, as well as many of the formatting elements an author may use in the preparation of the documentation of their work.

CCS CONCEPTS

• **Computer systems organization → Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

datasets, neural networks, gaze detection, text tagging

ACM Reference Format:
Mikhail Nikilukin, Ekaterina Verbitskaia, and Semyon Grigorev. 2018. Combinators for Single Source Context-Free Path Querying. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Context-Free path querying (CFPQ) is an actively developed area in graph datatbase analysis.

CFPQ is widely used for static code analysis.

Languages for language-constrained queryes specification. CfSparql and proposal for Cypher.

Integration with general purpose programming language. Typing [1].

Combinators [2].

Single source scenario. Instead of traditional all pairs. Some of algorithms inheritantly calculate only all pairs reachability.

In this paper we make the following contributions.

- Introduce example and explain how to use combinators for CFPQ.
- Evaluate single source. We explore some basic properties of real-world dataset, such as lengths of paths and number of reachable vertices.

Woodstock '18, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

2 MOTIVATING EXAMPLE

In this section we introduce a simple problem of graph analysis which can be solved by using CFPQ.

First of all, we introduce a simple graph to be analyzed.

3 COMBINATORS FOR CONTEXT-FREE PATH QUERYING

In this section we demonstrate main features of combinators in the context of context-free path querying and integration with general-purpose programming languages. To do it we sove the problem which we state in the previous section.

3.1 Compositionality

same generation query

3.2 Type Safety

Static type chacking

3.3 User-Defined Actions

Additional computations

3.4 IDE Support

Screens!!!!

4 EVALUATION

We evaluate Meerkat.Graph on single source context-free path querying scenario. For evaluation we use Neo4j graph databese which was run on PC with the folloeing configuration.

- CPU
- RAM
- OS
- JVM

Neo4j is integreted into application !!!!

Dataset contains two real-world RDFs: Geospecies which contains information about biological hierrarchy[1] and Enzime which is a part of UniProt database[2]. Detailed description of these graphs is presented in table 1. Note, that graphs was loaded into database fully, not only edges which laballed by relations used inqueryes.

Queries for evaluation are versions of same-generation query — classical context-free query which is useful for hierarchy analysis. We equip queryes with user-defined actions for end verties and unique path counting. To demonstarte power of combinators, we use the function !!! defied above to create queries.

[1] https://old.datahub.io/dataset/geospecies. Access date: 12.11.2019.
[2] Protein sequences data base: https://www.uniprot.org/. RDFs with data are available here: ftp://ftp.uniprot.org/pub/databases/uniprot/current_release/rdf. Access date: 12.11.2019

Graph	#Vertices	#Edges	#NT	#BT
Enzime				
Geospecies				

Table 1: Details of graphs

For each graph and each query we run this query form each vertex from graph and measure elapsed time and required memory by using !!! tool.

Enzime RDF querying. We evaluate two queryes: Q_1 — same genaration over !!!! relation

```
def sameGen(brs) =
  reduceChoice(
    brs.map {case (lbr, rbr) =>
      lbr ~ syn(sameGen(brs).?) ~ rbr})
```

and Q_2 — same generation over !!!

```
def sameGen(brs) =
  reduceChoice(
    brs.map {case (lbr, rbr) =>
      lbr ~ syn(sameGen(brs).?) ~ rbr})
```

.

Results of evaluation are presented in figures 1 and 2.

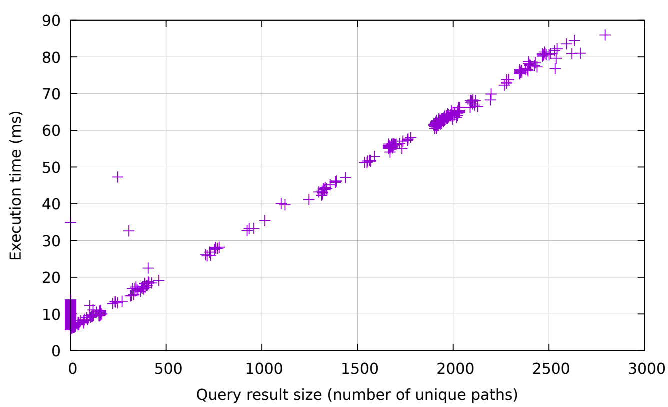

Figure 1: Query execution time for Enzime dataset and queryes Q_1 and Q_2

Figure 1 shows dependency of query evaluation time on query answer size in terms of number of uniqie paths. First of all, we can see that evaluation time is linear on answer size. Also we can see, that time which requred to evaluate query for one specific vertex is relatively small. In our case it is less than 90ms.

Figure 2 shows dependency of memory requred to evaluate qurey on query answer size in terms of number of uniqie paths.

Geospecies RDF querying.

Here we can see !!!!

Finally, we can conclude that confext-free path querying in single source scenario can be efficiently evaluated by using !!!. While all pairs scenario is still hard [?], single source scenarion, which is useful for manual or interactive data analysis, can be !!!

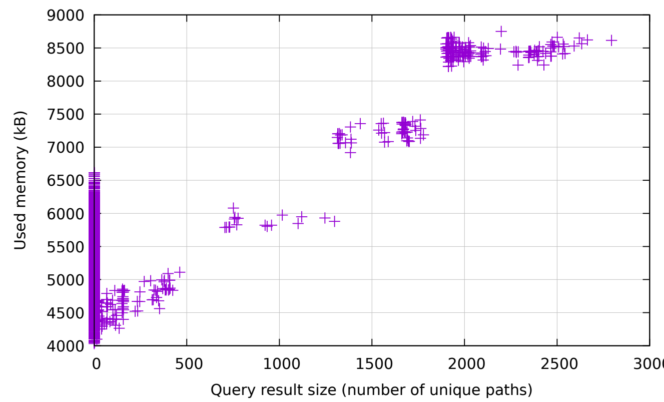

Figure 2: Query required memory for Enzime dataset and queryes Q_1 and Q_2

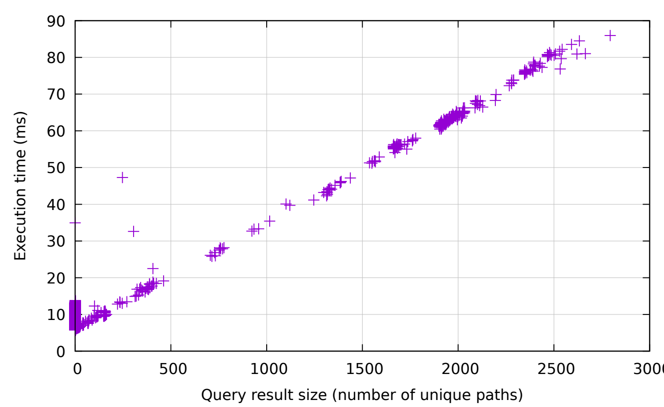

Figure 3: Query execution time for Enzime dataset and queryes Q_3 and Q_4

5 CONCLUSION AND FUTURE WORK

We show that single-source context-free path querying can be !!!

One of important direction of the future reserach is to optimize performance of proposed solution. One of possible solution here is deep integration with Neo4j infrastructure to utilize cache system, etc.

Improve combinators library.

Create set of query templates.

ACKNOWLEDGMENTS

To Robert, for the bagels and explaining CMYK and color spaces.

REFERENCES

[1] Norbert Tausch, Michael Philippsen, and Josef Adersberger. 2011. A Statically Typed Query Language for Property Graphs. In *Proceedings of the 15th Symposium on International Database Engineering & Applications* (Lisboa, Portugal) *(IDEAS '11)*. Association for Computing Machinery, New York, NY, USA, 219–225. https://doi.org/10.1145/2076623.2076653

[2] Ekaterina Verbitskaia, Ilya Kirillov, Ilya Nozkin, and Semyon Grigorev. 2018. Parser Combinators for Context-Free Path Querying. In *Proceedings of the 9th ACM SIGPLAN International Symposium on Scala* (St. Louis, MO, USA) *(Scala 2018)*.

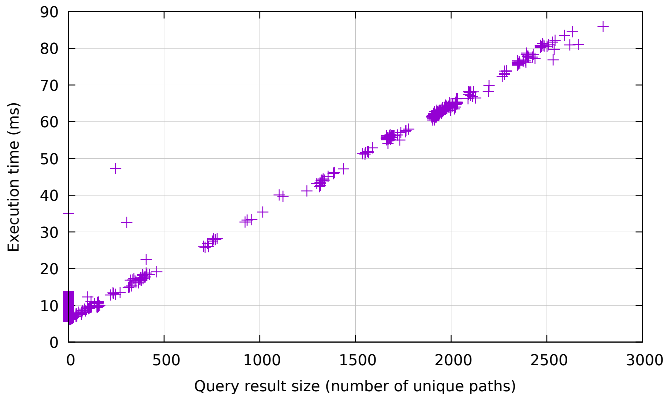

Figure 4: Query execution time for Enzime dataset and queryes Q_3 and Q_4

Association for Computing Machinery, New York, NY, USA, 13–23. https://doi.org/10.1145/3241653.3241655