Conjunctive Path Querying by Matrix Multiplication

Rustam Azimov
Saint Petersburg State University
St. Petersburg, Russia
rustam.azimov19021995@gmail.com

Semyon Grigorev
Saint Petersburg State University
St. Petersburg, Russia
Semen.Grigorev@jetbrains.com

ABSTRACT

1 INTRODUCTION

2 PRELIMINARIES

In this section, we introduce the basic notions used throughout the paper.

Let Σ be a finite set of edge labels. Define an *edge-labeled directed graph* as a tuple $D = (V, E)$ with a set of nodes V and a directed edge-relation $E \subseteq V \times \Sigma \times V$. For a path π in a graph D we denote the unique word obtained by concatenating the labels of the edges along the path π as $l(\pi)$. Also, we write $n\pi m$ to indicate that a path π starts at node $n \in V$ and ends at node $m \in V$.

Similar to the case of the context-free grammars, we deviate from the usual definition of a conjunctive grammar in the *binary normal form* [2] by not including a special start non-terminal, which will be specified in the queries to the graph. Since every conjunctive grammar can be transformed into an equivalent one in the binary normal form [2] and checking that an empty string is in the language is trivial, then it is sufficient to only consider grammars of the following type. A *conjunctive grammar* is 3-tuple $G = (N, \Sigma, P)$ where N is a finite set of non-terminals, Σ is a finite set of terminals, and P is a finite set of productions of the following forms:

- $A \rightarrow B_1C_1 \,\&\ldots\&\, B_mC_m$, for $m \geq 1$, $A, B_i, C_i \in N$,
- $A \rightarrow x$, for $A \in N$ and $x \in \Sigma$.

For conjunctive grammars we also use the conventional notation $A \xrightarrow{*} w$ to denote that the string $w \in \Sigma^*$ can be derived from a non-terminal A by some sequence of applying the production rules from P. The relation $\rightarrow$ is defined as follows:

- Using a rule $A \rightarrow B_1C_1 \,\&\ldots\&\, B_mC_m \in P$, any atomic subterm A of any term can be rewritten by the subterm $(B_1C_1 \,\&\ldots\&\, B_mC_m)$:
$$\ldots A \ldots \rightarrow \ldots (B_1C_1 \,\&\ldots\&\, B_mC_m) \ldots$$
- A conjunction of several identical strings in Σ^* can be rewritten by one such string: for every $w \in \Sigma^*$,
$$\ldots (w \,\&\ldots\&\, w) \ldots \rightarrow \ldots w \ldots$$

The *language* of a conjunctive grammar $G = (N, \Sigma, P)$ with respect to a start non-terminal $S \in N$ is defined by $L(G_S) = \{w \in \Sigma^* \mid S \xrightarrow{*} w\}$.

For a given graph $D = (V, E)$ and a conjunctive grammar $G = (N, \Sigma, P)$, we define *conjunctive relations* $R_A \subseteq V \times V$, for every $A \in N$, such that $R_A = \{(n, m) \mid \exists n\pi m \,(l(\pi) \in L(G_A))\}$.

We define a *conjunctive matrix multiplication*, $a \circ b = c$, where a and b are matrices of the suitable size that have subsets of N as elements, as $c_{i,j} = \{A \mid \exists (A \rightarrow B_1C_1 \,\&\ldots\&\, B_mC_m) \in P$ such that $(B_k, C_k) \in d_{i,j}\}$, where $d_{i,j} = \bigcup_{k=1}^{n} a_{i,k} \times b_{k,j}$.

We define the *conjunctive transitive closure* of a square matrix a as $a^{conj} = a^{(1)} \cup a^{(2)} \cup \cdots$ where $a^{(i)} = a^{(i-1)} \cup (a^{(i-1)} \circ a^{(i-1)})$, $i \geq 2$ and $a^{(1)} = a$.

3 RELATED WORKS

4 CONJUNCTIVE PATH QUERYING BY THE CALCULATION OF TRANSITIVE CLOSURE

Since the query evaluation using the relational query semantics and conjunctive grammars is undecidable problem [1] then we propose an algorithm that calculates the over-approximation of all conjunctive relations R_A.

4.1 Reducing conjunctive path querying to transitive closure

In this section, we show how the over-approximation of all conjunctive relations R_A can be calculated by computing the transitive closure.

Let $G = (N, \Sigma, P)$ be a conjunctive grammar and $D = (V, E)$ be a graph. We number the nodes of the graph D from 0 to $(|V| - 1)$ and we associate the nodes with their numbers. We initialize $|V| \times |V|$ matrix b with $\emptyset$. Further, for every i and j we set $b_{i,j} = \{A_k \mid ((i, x, j) \in E) \wedge ((A_k \rightarrow x) \in P)\}$. Finally, we compute the conjunctive transitive closure $b^{conj} = b^{(1)} \cup b^{(2)} \cup \ldots$ where $b^{(i)} = b^{(i-1)} \cup (b^{(i-1)} \circ b^{(i-1)})$, $i \geq 2$ and $b^{(1)} = b$. For the conjunctive transitive closure b^{conj}, the following statements holds.

LEMMA 4.1. *Let $D = (V, E)$ be a graph, let $G = (N, \Sigma, P)$ be a conjunctive grammar. Then for any i, j and for any non-terminal $A \in N$, if $(i, j) \in R_A$ and $i\pi j$, such that there is a derivation tree according to the string $l(\pi)$ and a conjunctive grammar $G_A = (N, \Sigma, P, A)$ of the height $h \leq k$ then $A \in b^{(k)}_{i,j}$.*

PROOF. (Proof by Induction)

Basis: Show that the statement of the lemma holds for $k = 1$. For any i, j and for any non-terminal $A \in N$, if $(i, j) \in R_A$ and $i\pi j$, such that there is a derivation tree according to the string $l(\pi)$ and a conjunctive grammar $G_A = (N, \Sigma, P, A)$ of the height $h \leq 1$ then there is edge e from node i to node j and $(A \rightarrow x) \in P$ where $x = l(\pi)$. Therefore $A \in b^{(1)}_{i,j}$ and it has been shown that the statement of the lemma holds for $k = 1$.

Inductive step: Assume that the statement of the lemma holds for any $k \leq (p - 1)$ and show that it also holds for $k = p$ where $p \geq 2$. Let $(i, j) \in R_A$ and $i\pi j$, such that there is a derivation tree according to the string $l(\pi)$ and a conjunctive grammar $G_A = (N, \Sigma, P, A)$ of the height $h \leq p$.

Let $h < p$. Then by the inductive hypothesis $A \in b^{(p-1)}_{i,j}$. Since $b^{(p)} = b^{(p-1)} \cup (b^{(p-1)} \circ b^{(p-1)})$ then $A \in b^{(p)}_{i,j}$ and the statement of the lemma holds for $k = p$.

Let $h = p$. Let $A \rightarrow B_1C_1 \,\&\ldots\&\, B_mC_m$ be the rule corresponding to the root of the derivation tree from the assumption

 Published in Proceedings of the 21st International Conference on Extending Database Technology (EDBT), March 26-29, 2018, ISBN 978-3-89318-078-3 on OpenProceedings.org.

of the lemma. Therefore the heights of all subtrees corresponding to non-terminals $B_1, C_1, \ldots B_m, C_m$ are less than p. Then by the inductive hypothesis $B_x \in b^{(p-1)}_{i,t_x}$ and $C_x \in b^{(p-1)}_{t_x,j}$, for $x = 1 \ldots m$ and $t_x \in V$. Let d be a matrix that have subsets of $N \times N$ as elements, where $d_{i,j} = \bigcup_{t=1}^{n} b^{(p-1)}_{i,t} \times b^{(p-1)}_{t,j}$. Therefore $(B_x, C_x) \in d_{i,j}$, for $x = 1 \ldots m$. Since $b^{(p)} = b^{(p-1)} \cup (b^{(p-1)} \circ b^{(p-1)})$ and $(b^{(p-1)} \circ b^{(p-1)})_{i,j} = \{A \mid \exists (A \to B_1C_1 \& \ldots \& B_mC_m) \in P$ such that $(B_k, C_k) \in d_{i,j}\}$ then $A \in b^{(p)}_{i,j}$ and the statement of the lemma holds for $k = p$. This completes the proof of the lemma. ◻

Theorem 1. *Let $D = (V, E)$ be a graph and let $G = (N, \Sigma, P)$ be a conjunctive grammar. Then for any i, j and for any non-terminal $A \in N$, if $(i, j) \in R_A$ then $A \in b^{conj}_{i,j}$.*

Proof. By the lemma 4.1, if $(i, j) \in R_A$ then $A \in b^{(k)}_{i,j}$ for some k, such that $i\pi j$ with a derivation tree according to the string $l(\pi)$ and a conjunctive grammar $G_A = (N, \Sigma, P, A)$ of the height $h \leq k$. Since the matrix $b^{conj} = b^{(1)} \cup b^{(2)} \cup \cdots$, then for any i, j and for any non-terminal $A \in N$, if $A \in b^{(k)}_{i,j}$ for some $k \geq 1$ then $A \in b^{conj}_{i,j}$. Therefore, if $(i, j) \in R_A$ then $A \in b^{conj}_{i,j}$. This completes the proof of the theorem. ◻

Thus, we show how the over-approximation of all conjunctive relations R_A can be calculated by computing the conjunctive transitive closure b^{conj} of the matrix b.

4.2 The algorithm

In this section we introduce an algorithm for calculating the conjunctive transitive closure b^{conj} which was discussed in Section 4.1.

The following algorithm takes on input a graph $D = (V, E)$ and a conjunctive grammar $G = (N, \Sigma, P)$.

Algorithm 1 Conjunctive recognizer for graphs

```
1:  function conjunctiveGraphParsing(D, G)
2:    n ← a number of nodes in D
3:    E ← the directed edge-relation from D
4:    P ← a set of production rules in G
5:    T ← a matrix n × n in which each element is ∅
6:    for all (i, x, j) ∈ E do                ▷ Matrix initialization
7:      T_{i,j} ← T_{i,j} ∪ {A | (A → x) ∈ P}
8:    while matrix T is changing do
9:      T ← T ∪ (T ∘ T)               ▷ Transitive closure calculation
10:   return T
```

Similar to the case of the context-free grammars we can show that the Algorithm 1 terminates in a finite number of steps. Since each element of the matrix T contains no more than $|N|$ non-terminals, then total number of non-terminals in the matrix T does not exceed $|V|^2|N|$. Therefore, the following theorem holds.

Theorem 2. *Let $D = (V, E)$ be a graph and let $G = (N, \Sigma, P)$ be a conjunctive grammar. Algorithm 1 terminates in a finite number of steps.*

Proof. It is sufficient to show, that the operation in line **9** of the Algorithm 1 changes the matrix T only finite number of times. Since this operation can only add non-terminals to some elements of the matrix T, but not remove them, it can change the matrix T no more than $|V|^2|N|$ times. ◻

5 EVALUATION

6 CONCLUSION AND FUTURE WORK

ACKNOWLEDGMENTS

We are grateful to ... for their careful reading, pointing out some mistakes, and invaluable suggestions. This work is supported by grant from JetBrains Research.

REFERENCES

[1] J. Hellings. 2014. Conjunctive context-free path queries. (2014).

[2] Alexander Okhotin. 2013. Conjunctive and Boolean grammars: the true general case of the context-free grammars. *Computer Science Review* 9 (2013), 27–59.