Query Embedding Strategy

Connor Henderson

February 2021

1 Sub-graph Generation

Algorithm 1: $get\_subgraphs(\mathcal{G}, n)$

input : $\mathcal{G} = (\mathcal{V}, \mathcal{E})$: Directed acyclic graph from which sub-graphs have to be generated
n : Node which acts as the root of the sub-graph

output : *joins* : A list of tuples representing the join node and the tables involved with that join
terminals : A list representing the tables present in a graph

$n :=$ root
$joins := [\,]$
$terminals := [\,]$

do
  $\mathcal{N}_n := \{\ n' \mid (n, n') \ \in \ \mathcal{E} \ \}$
  $n := n'$
while *len*($\mathcal{N}_n$) == *1*;

$\mathcal{N}_n := \{\ n' \mid \ (n, n') \ \in \ \mathcal{E} \ \}$

if *len*($\mathcal{N}_n$) > 1 **then**
  foreach $n' \in \mathcal{N}_n$ **do**
    $joins', \ terminals' = get\_subgraphs(\mathcal{G}, \ n')$
    $joins := joins \cup$ joins'
    $terminals := terminals \cup$ terminals'
  end
  $joins := joins \cup$ [(n, terminals)]
end

if *len*($\mathcal{N}_n$) $== 0$ **then**
  $terminals := terminals \cup$ [n]
end

return $joins, terminals$

2 Embedding

Algorithm 2: *embedding_process*

input : $\mathbb{G} = \{G_1, G_2, ..., G_n\}$: Set of graphs for which embeddings are desired
D: The maximum number of tables to consider for a sub-graph
δ: number of dimensions in the latent space
ϵ: number of training epochs
α : learning rate
output : $\Phi \in \mathbb{R}^{\|\mathbb{G}\| \times \delta}$: Matrix of graph representation vectors

```
Initialize Φ with random values
for e ← 0 to ϵ do
    foreach G_i ∈ 𝔾 do
        sg_n = get_subgraphs(G_i, 0)
        J(Φ) := −logPr(sg_n | Φ(G_n))
        Φ = Φ − α ∂J/∂Φ
    end
end
return Φ
```