



# Mini-Project 1

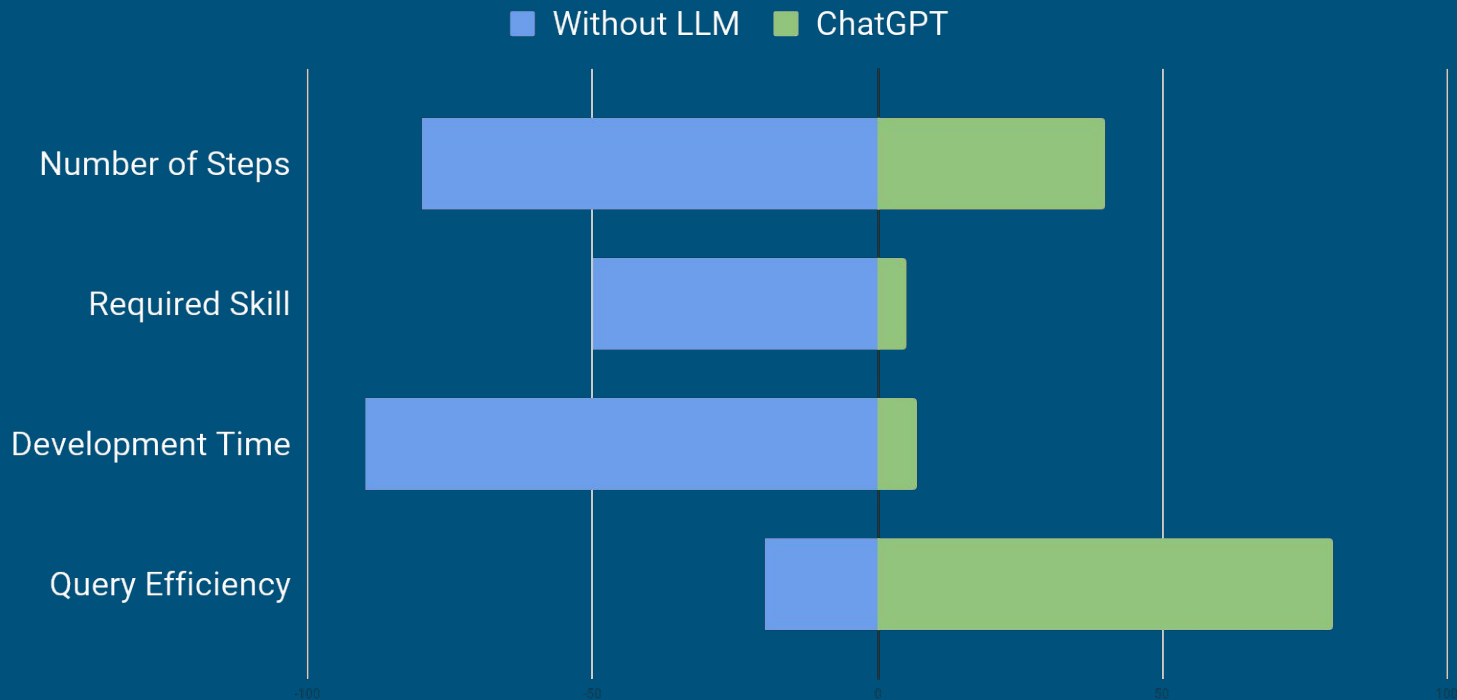


ASE 2024

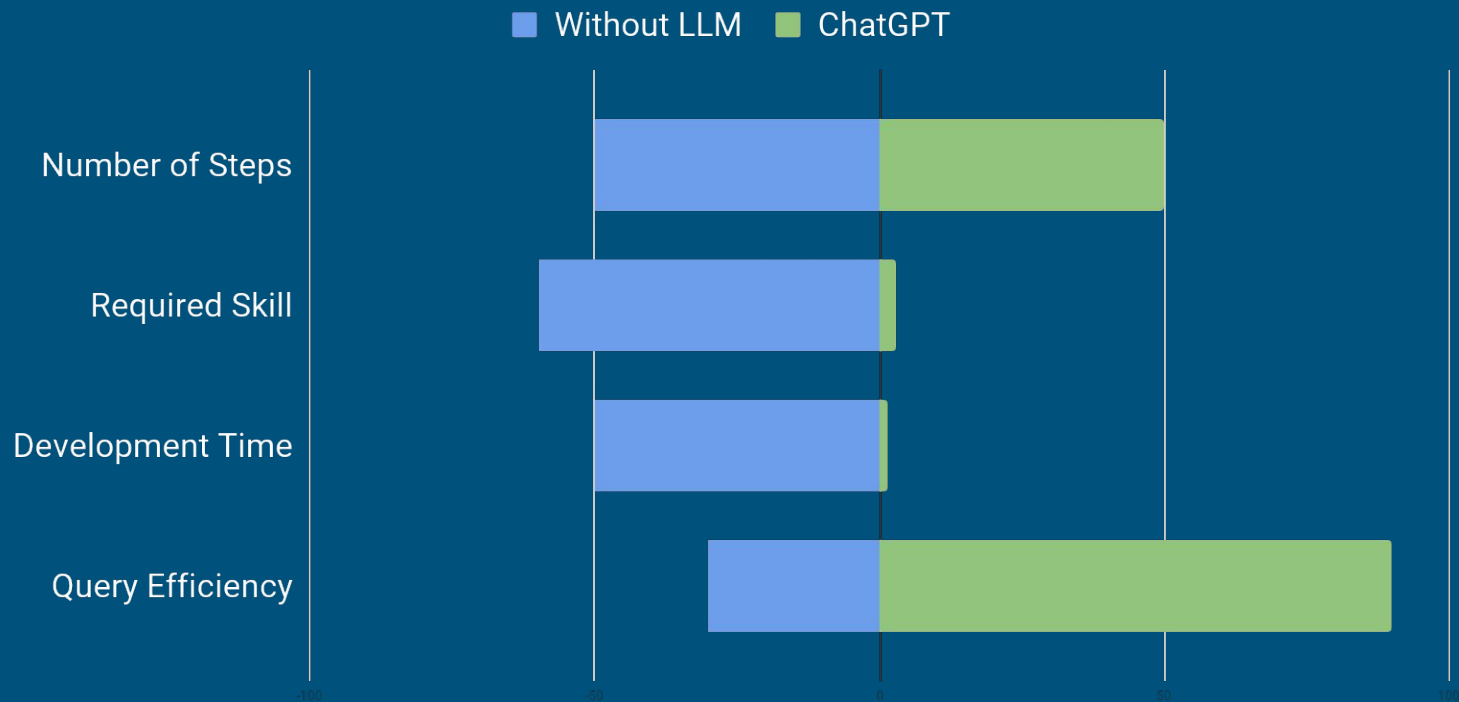
Paul Chevelev · Silvan Verhoeven



# Task 1 - Create initial Query



# Task 2 - Extend Query



# Task 3 - Execution Plan

## PostgreSQL

Sort (cost=148005.40..148005.52 rows=48 width=96) (actual time=2974.874..2994.351 rows=9 loops=1)  
Sort Key: (((f.lc.foreign\_likes)::numeric / ((count(\*))::numeric)) DESC  
Sort Method: quicksort Memory: 25kB  
CTE creator\_friends  
-> Seq Scan on knows (cost=0.00..461.46 rows=28146 width=16) (actual time=0.030..1.994 rows=28146 loops=1)  
CTE foreign\_likes  
-> Merge Anti Join (cost=117656.97..126482.50 rows=82080 width=24) (actual time=1730.061..2444.677 rows=3439 loops=1)  
Merge Cond: ((l.l.personid = creator\_friends.friend\_id) AND (m\_1.m.creatorid = creator\_friends.creator\_id))  
-> Sort (cost=30678.45..30952.05 rows=109440 width=32) (actual time=331.802..365.637 rows=109440 loops=1)  
Sort Key: l.l.personid, m\_1.m.creatorid  
Sort Method: external merge Disk: 4512kB  
-> Hash Join (cost=12908.74..18896.94 rows=109440 width=32) (actual time=192.460..278.215 rows=109440 loops=1)  
Hash Cond: (l.l.messageid = m\_1.m.messageid)  
-> Seq Scan on likes l (cost=0.00..1798.40 rows=109440 width=24) (actual time=0.017..12.857 rows=109440 loops=1)  
-> Hash (cost=7923.44..7923.44 rows=286744 width=16) (actual time=191.210..191.211 rows=286744 loops=1)  
Buckets: 262144 Batches: 4 Memory Usage: 5425kB  
-> Seq Scan on message m\_1 (cost=0.00..7923.44

## ChatGPT

CTE Scan on message\_likes (cost=1000.00..3000.00 rows=2000 width=16)  
-> GroupAggregate (cost=1000.00..2500.00 rows=2000 width=16)  
-> Seq Scan on likes (cost=0.00..500.00 rows=100000 width=8)  
Filter: (COUNT(\*) >= 20)  
  
CTE Scan on creator\_friends (cost=500.00..1500.00 rows=15000 width=8)  
-> Seq Scan on knows (cost=0.00..400.00 rows=30000 width=8)  
  
CTE Scan on friends\_of\_friends (cost=600.00..1800.00 rows=10000 width=8)  
-> Hash Join (cost=200.00..800.00 rows=10000 width=8)  
-> Seq Scan on knows (cost=0.00..400.00 rows=30000 width=8)  
-> Hash (cost=200.00..200.00 rows=15000 width=8)  
-> CTE Scan on creator\_friends (cost=0.00..500.00 rows=15000 width=8)  
  
CTE Scan on extended\_network (cost=1500.00..2500.00 rows=20000 width=8)  
-> Append (cost=500.00..1500.00 rows=20000 width=8)  
-> Seq Scan on creator\_friends (cost=0.00..400.00 rows=15000 width=8)  
-> Seq Scan on friends\_of\_friends (cost=0.00..500.00 rows=10000 width=8)

# Task 3 - Execution Plan

## PostgreSQL

...

Filter: (m\_length > 100)

Rows Removed by Filter: 89766

-> Hash (cost=2434.92..2434.92

rows=2382 width=16) (actual time=149.618..149.619

rows=1597 loops=3)

**Buckets: 4096 Batches: 1 Memory**

**Usage: 107kB**

-> HashAggregate

(cost=2345.60..2434.92 rows=2382 width=16) (actual

**time=135.373..149.366 rows=1597 loops=3)**

Group Key: likes.l\_messageid

Filter: (count(\*) >= 20)

**Batches: 1 Memory Usage:**

**1425kB**

Rows Removed by Filter: 11523

**Worker 0: Batches: 1 Memory**

**Usage: 1425kB**

**Worker 1: Batches: 1 Memory**

**Usage: 1425kB**

-> Seq Scan on likes

## ChatGPT

...

CTE Scan on filtered\_messages (cost=3000.00..7000.00

rows=5000 width=40)

-> Hash Join (cost=2500.00..6000.00 rows=5000

width=40)

-> Hash Join (cost=1500.00..3500.00 rows=5000

width=40)

-> Seq Scan on filtered\_messages (cost=0.00..3000.00

rows=5000 width=16)

-> Seq Scan on message (cost=0.00..2000.00

rows=10000 width=16)

-> Seq Scan on person (cost=0.00..300.00 rows=5000

width=16)

Final Output Sort (cost=5000.00..5200.00 rows=5000

width=40)

-> Sort (cost=5000.00..5200.00 rows=5000 width=40)

-> Hash Join (cost=2500.00..6000.00 rows=5000

width=40)

-> Hash Join (cost=1500.00..3500.00 rows=5000

# Task 3 - Execution Plan

PostgreSQL

...

Planning Time: 2.709 ms

JIT:

Functions: 122

Options: Inlining false, Optimization false, Expressions true,  
Deforming true

Timing: Generation 4.139 ms, Inlining 0.000 ms,  
Optimization 1.992 ms, Emission 177.034 ms, Total 183.166  
ms

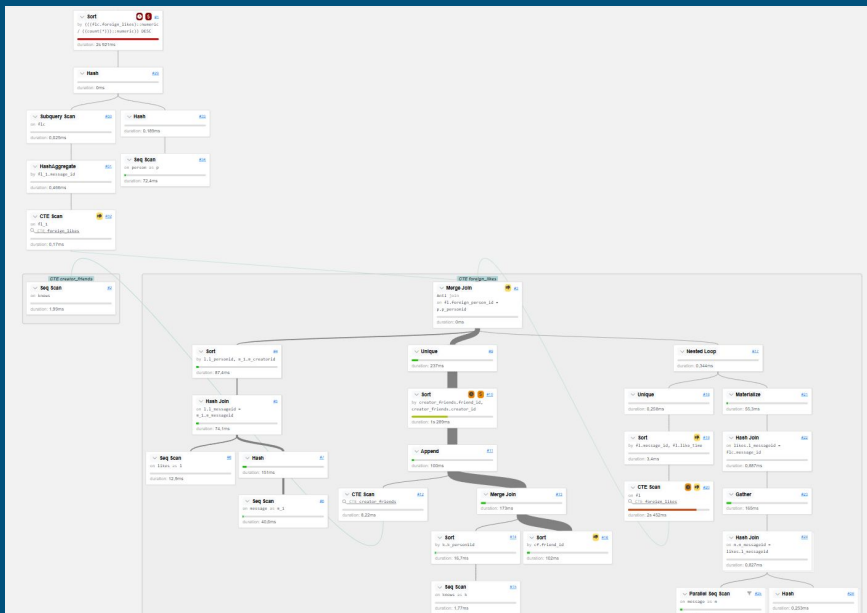
Execution Time: 3084.699 ms  
(82 rows)

ChatGPT

...

# Task 3 - Execution Plan

## PostgreSQL



## ChatGPT

