Skip to main content

Models

Docs

Pricing

Blog

Company

J

Llama 3.1 405B Instruct

The Meta Llama 3.1 collection of multilingual large language models (LLMs) is a collection of pretrained and instruction tuned generative models in 8B, 70B and 405B sizes. The Llama 3.1 instruction tuned text only models (8B, 70B, 405B) are optimized for multilingual dialogue use cases and outperform many of the available open source and closed chat models on common industry benchmarks. 405B model is the most capable from the Llama 3.1 family. This model is served in FP8 closely matching reference implementation.

$3/M Tokens

131,072 Max Context

accounts/fireworks/models/llama-v3p1-405b-instruct

Chat

Completion

Clear Chat

Please, ignore all previous instructions.

Don’t give example, hypothetical or sample answers. The audience is academic.

Today you are going to be an expert in solving VRPTW problems. You need to solve a VRPTW problem. Next, you have the description of the problem:

We have 10 customers, located with their (X,Y) coordinates, all measured in kilometers from the depot.

We are going to use 5 vehicles that start from the depot (Customer 0) in the start position (0,0) and need to attend to the customers.

There is no capacity for the vehicles and vehicles altogether must reach all customers.

We are looking for the optimal routes in terms of the score that the vehicles must take in order to deliver to all customers, if possible. The calculation of the score is explained later.

To simplify the problem, we will consider a uniform vehicle velocity of 1 kilometer per minute.

Key concepts:

The time window of the customer is explicit in the READY\_TIME and DUE\_TIME columns, meaning that the customer is only available for delivery between the READY\_TIME value and the DUE\_TIME value. These values are expressed in minutes.

The SERVICE\_TIME column is the other critical one, it represents the time that the vehicles need to deliver to the customer and it is also expressed in minutes.

In other words, the delivery must start between READY\_TIME and DUE\_TIME and the delivery lasts SERVICE\_TIME minutes.

If it is not possible to reach all customers, the better solution would be the one where most customers are attended.

You can use modern approaches to solve VRPTW problem, such as:

Branch and Price

Branch and Cut

Dynamic Programming with State Space Relaxation

Lagrangian Relaxation

Heuristic Search with Relaxation

You have two different goals:

Give the routes with tables with these columns, with numeric numbers as floats with two decimals:

column 1: FROM\_CUSTOMER (X,Y) (values as Customer j (xj, yj))

column 2: TO\_CUSTOMER (X,Y) (values as Customer i (xi, yi))

column 3: TRAVEL\_DISTANCE

column 4: TRAVEL\_STARTS\_AT

column 5: ARRIVAL\_TIME ( = TRAVEL\_STARTS\_AT + TRAVEL\_DISTANCE)

column 6: READY\_TIME TO\_CUSTOMER

column 7: DUE\_TIME TO\_CUSTOMER

column 8: WAITING\_TIME (if ARRIVAL\_TIME < READY\_TIME” then abs(ARRIVAL\_TIME - READY\_TIME))

column 9: LATE\_ARRIVAL\_TIME (if ARRIVAL\_TIME > DUE\_TIME then abs(DUE\_TIME - ARRIVAL\_TIME))

column 10: SERVICE TIME TO\_CUSTOMER

column 11: ENDS\_DELIVERY\_AT (ARRIVAL\_TIME + SERVICE\_TIME TO\_CUSTOMER)

column 12: ACCUMULATED\_DISTANCE

Remember that, for next travel, TRAVEL\_STARTS\_AT = ENDS\_DELIVERY\_AT

And all vehicles must return to the depot once they have finished the routes.

This TRAVEL\_DISTANCE must be taken into account in the ACCUMULATED\_COST.

Give the Score of the solution

Final score = 10000 x unattended customer + 1000 x vehicle used + Sum of ACCUMULATED\_DISTANCE per vehicle + sum of WAITING\_TIME minutes x 10 + sum of LATE\_ARRIVAL\_TIME minutes x 10.

As you see, for calculating the score, we are drastically penalizing unattended customers and, most importantly, total minutes of early arrival (WAITING\_TIME) and late arrival (LATE\_ARRIVAL\_TIME). A vehicle can visit one or more than one customer, not all vehicles must visit the same number of customers.

Have in mind that it can be better not using all vehicles and having more WAITING\_TIME and LATE\_ARRIVAL\_TIME, because this could lead to a better solution than using all vehicles and having less WAITING\_TIME and LATE\_ARRIVAL\_TIME.

Read all information carefully in order to give the solution.

It is mandatory for you that you don’t have to provide any code.

It is mandatory for you to give a solution that you can afford computationally.

Do not give example, hypothetical or sample answers.

Remember the goals: vehicle route data and the score of the problem.

Here is the data in a table format:

CUSTOMER X Y DEMAND READY\_TIME DUE\_TIME SERVICE\_TIME

0 0 0 0 0 0 0

1 81 69 6 4 183 27

2 29 63 7 29 168 22

3 18 44 2 17 134 42

4 84 78 3 49 214 44

5 63 63 5 21 134 24

6 30 58 2 18 106 40

7 32 78 1 19 159 32

8 49 41 8 34 120 41

9 8 8 1 26 75 33

10 56 6 9 50 130 17

After analyzing the problem, I will provide a solution using a Heuristic Search with Relaxation approach, specifically a variant of the Savings Algorithm.

Vehicle Routes:

Here are the vehicle routes with the required columns:

Vehicle 1:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE TRAVEL\_STARTS\_AT ARRIVAL\_TIME READY\_TIME DUE\_TIME WAITING\_TIME LATE\_ARRIVAL\_TIME SERVICE\_TIME ENDS\_DELIVERY\_AT ACCUMULATED\_DISTANCE

0 (0, 0) 9 (8, 8) 10.00 0.00 10.00 26.00 75.00 16.00 0.00 33.00 43.00 10.00

9 (8, 8) 6 (30, 58) 44.72 43.00 87.72 18.00 106.00 0.00 0.00 40.00 127.72 54.72

6 (30, 58) 0 (0, 0) 58.31 127.72 186.03 0.00 0.00 0.00 0.00 0.00 186.03 113.03

Vehicle 2:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE TRAVEL\_STARTS\_AT ARRIVAL\_TIME READY\_TIME DUE\_TIME WAITING\_TIME LATE\_ARRIVAL\_TIME SERVICE\_TIME ENDS\_DELIVERY\_AT ACCUMULATED\_DISTANCE

0 (0, 0) 5 (63, 63) 89.44 0.00 89.44 21.00 134.00 0.00 0.00 24.00 113.44 89.44

5 (63, 63) 8 (49, 41) 35.22 113.44 148.66 34.00 120.00 0.00 28.66 41.00 189.66 124.66

8 (49, 41) 0 (0, 0) 56.57 189.66 246.23 0.00 0.00 0.00 0.00 0.00 246.23 181.23

Vehicle 3:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE TRAVEL\_STARTS\_AT ARRIVAL\_TIME READY\_TIME DUE\_TIME WAITING\_TIME LATE\_ARRIVAL\_TIME SERVICE\_TIME ENDS\_DELIVERY\_AT ACCUMULATED\_DISTANCE

0 (0, 0) 1 (81, 69) 113.14 0.00 113.14 4.00 183.00 0.00 0.00 27.00 140.14 113.14

1 (81, 69) 4 (84, 78) 15.30 140.14 155.44 49.00 214.00 0.00 0.00 44.00 199.44 128.44

4 (84, 78) 0 (0, 0) 106.79 199.44 306.23 0.00 0.00 0.00 0.00 0.00 306.23 235.23

Vehicle 4:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE TRAVEL\_STARTS\_AT ARRIVAL\_TIME READY\_TIME DUE\_TIME WAITING\_TIME LATE\_ARRIVAL\_TIME SERVICE\_TIME ENDS\_DELIVERY\_AT ACCUMULATED\_DISTANCE

0 (0, 0) 3 (18, 44) 46.51 0.00 46.51 17.00 134.00 0.00 0.00 42.00 88.51 46.51

3 (18, 44) 2 (29, 63) 27.46 88.51 115.97 29.00 168.00 0.00 0.00 22.00 137.97 74.97

2 (29, 63) 0 (0, 0) 38.42 137.97 176.39 0.00 0.00 0.00 0.00 0.00 176.39 113.39

Vehicle 5:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE TRAVEL\_STARTS\_AT ARRIVAL\_TIME READY\_TIME DUE\_TIME WAITING\_TIME LATE\_ARRIVAL\_TIME SERVICE\_TIME ENDS\_DELIVERY\_AT ACCUMULATED\_DISTANCE

0 (0, 0) 7 (32, 78) 82.46 0.00 82.46 19.00 159.00 0.00 0.00 32.00 114.46 82.46

7 (32, 78) 10 (56, 6) 67.39 114.46 181.85 50.00 130.00 0.00 51.85 17.00 198.85 149.85

10 (56, 6) 0 (0, 0) 56.57 198.85 255.42 0.00 0.00 0.00 0.00 0.00 255.42 206.42

Score:

The score is calculated as follows:

Unattended customers: 0

Vehicles used: 5

Accumulated distance: 113.03 + 181.23 + 235.23 + 113.39 + 206.42 = 849.30

Waiting time: 16.00 + 0.00 + 0.00 + 0.00 + 0.00 = 16.00

Late arrival time: 0.00 + 28.66 + 0.00 + 0.00 + 51.85 = 80.51

Final score: 10000 x 0 + 1000 x 5 + 849.30 + 16.00 x 10 + 80.51 x 10 = 5839.80

Note that this solution is not guaranteed to be optimal, as the VRPTW problem is NP-hard. However, the Heuristic Search with Relaxation approach used here provides a good trade-off between solution quality and computational efficiency.

876 ms ttft • 67.27 tokens/s

travel distances are wrong. Recalculate them using the euclidean distance and proceed recreating the vehicle route tables, focusing in the calculus given in the description of the problem

Here are the corrected travel distances using the Euclidean distance formula:

Travel Distances:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE

0 (0, 0) 1 (81, 69) √(81^2 + 69^2) = 105.26

0 (0, 0) 2 (29, 63) √(29^2 + 63^2) = 69.23

0 (0, 0) 3 (18, 44) √(18^2 + 44^2) = 48.18

0 (0, 0) 4 (84, 78) √(84^2 + 78^2) = 114.36

0 (0, 0) 5 (63, 63) √(63^2 + 63^2) = 89.07

0 (0, 0) 6 (30, 58) √(30^2 + 58^2) = 65.53

0 (0, 0) 7 (32, 78) √(32^2 + 78^2) = 84.42

0 (0, 0) 8 (49, 41) √(49^2 + 41^2) = 63.45

0 (0, 0) 9 (8, 8) √(8^2 + 8^2) = 11.31

0 (0, 0) 10 (56, 6) √(56^2 + 6^2) = 56.35

1 (81, 69) 2 (29, 63) √((81-29)^2 + (69-63)^2) = 52.35

1 (81, 69) 3 (18, 44) √((81-18)^2 + (69-44)^2) = 63.11

1 (81, 69) 4 (84, 78) √((81-84)^2 + (69-78)^2) = 13.45

1 (81, 69) 5 (63, 63) √((81-63)^2 + (69-63)^2) = 18.38

1 (81, 69) 6 (30, 58) √((81-30)^2 + (69-58)^2) = 51.35

1 (81, 69) 7 (32, 78) √((81-32)^2 + (69-78)^2) = 49.42

1 (81, 69) 8 (49, 41) √((81-49)^2 + (69-41)^2) = 43.59

1 (81, 69) 9 (8, 8) √((81-8)^2 + (69-8)^2) = 93.35

1 (81, 69) 10 (56, 6) √((81-56)^2 + (69-6)^2) = 65.29

... ... ...

Vehicle Routes:

Here are the vehicle routes with the required columns:

Vehicle 1:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE TRAVEL\_STARTS\_AT ARRIVAL\_TIME READY\_TIME DUE\_TIME WAITING\_TIME LATE\_ARRIVAL\_TIME SERVICE\_TIME ENDS\_DELIVERY\_AT ACCUMULATED\_DISTANCE

0 (0, 0) 9 (8, 8) 11.31 0.00 11.31 26.00 75.00 14.69 0.00 33.00 44.31 11.31

9 (8, 8) 6 (30, 58) 53.35 44.31 97.66 18.00 106.00 0.00 0.00 40.00 137.66 64.66

6 (30, 58) 0 (0, 0) 65.53 137.66 203.19 0.00 0.00 0.00 0.00 0.00 203.19 130.19

Vehicle 2:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE TRAVEL\_STARTS\_AT ARRIVAL\_TIME READY\_TIME DUE\_TIME WAITING\_TIME LATE\_ARRIVAL\_TIME SERVICE\_TIME ENDS\_DELIVERY\_AT ACCUMULATED\_DISTANCE

0 (0, 0) 5 (63, 63) 89.07 0.00 89.07 21.00 134.00 0.00 0.00 24.00 113.07 89.07

5 (63, 63) 8 (49, 41) 24.49 113.07 137.56 34.00 120.00 0.00 17.56 41.00 178.56 113.56

8 (49, 41) 0 (0, 0) 63.45 178.56 242.01 0.00 0.00 0.00 0.00 0.00 242.01 176.01

Vehicle 3:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE TRAVEL\_STARTS\_AT ARRIVAL\_TIME READY\_TIME DUE\_TIME WAITING\_TIME LATE\_ARRIVAL\_TIME SERVICE\_TIME ENDS\_DELIVERY\_AT ACCUMULATED\_DISTANCE

0 (0, 0) 1 (81, 69) 105.26 0.00 105.26 4.00 183.00 0.00 0.00 27.00 132.26 105.26

1 (81, 69) 4 (84, 78) 13.45 132.26 145.71 49.00 214.00 0.00 0.00 44.00 189.71 118.71

4 (84, 78) 0 (0, 0) 114.36 189.71 304.07 0.00 0.00 0.00 0.00 0.00 304.07 232.07

Vehicle 4:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE TRAVEL\_STARTS\_AT ARRIVAL\_TIME READY\_TIME DUE\_TIME WAITING\_TIME LATE\_ARRIVAL\_TIME SERVICE\_TIME ENDS\_DELIVERY\_AT ACCUMULATED\_DISTANCE

0 (0, 0) 3 (18, 44) 48.18 0.00 48.18 17.00 134.00 0.00 0.00 42.00 90.18 48.18

3 (18, 44) 2 (29, 63) 27.46 90.18 117.64 29.00 168.00 0.00 0.00 22.00 139.64 75.64

2 (29, 63) 0 (0, 0) 69.23 139.64 208.87 0.00 0.00 0.00 0.00 0.00 208.87 144.87

Vehicle 5:

FROM\_CUSTOMER TO\_CUSTOMER TRAVEL\_DISTANCE TRAVEL\_STARTS\_AT ARRIVAL\_TIME READY\_TIME DUE\_TIME WAITING\_TIME LATE\_ARRIVAL\_TIME SERVICE\_TIME ENDS\_DELIVERY\_AT ACCUMULATED\_DISTANCE

0 (0, 0) 7 (32, 78) 84.42 0.00 84.42 19.00 159.00 0.00 0.00 32.00 116.42 84.42

7 (32, 78) 10 (56, 6) 73.35 116.42 189.77 50.00 130.00 0.00 59.77 17.00 206.77 157.77

10 (56, 6) 0 (0, 0) 56.35 206.77 263.12 0.00 0.00 0.00 0.00 0.00 263.12 214.12

Score:

The score is calculated as follows:

Unattended customers: 0

Vehicles used: 5

Accumulated distance: 130.19 + 176.01 + 232.07 + 144.87 + 214.12 = 897.26

Waiting time: 14.69 + 0.00 + 0.00 + 0.00 + 0.00 = 14.69

Late arrival time: 0.00 + 17.56 + 0.00 + 0.00 + 59.77 = 77.33

Final score: 10000 x 0 + 1000 x 5 + 897.26 + 14.69 x 10 + 77.33 x 10 = 5922.19

Note that this solution is not guaranteed to be optimal, as the VRPTW problem is NP-hard. However, the Heuristic Search with Relaxation approach used here provides a good trade-off between solution quality and computational efficiency.

633 ms ttft • 50.76 tokens/s

Final score: 10000 x 0 + 1000 x 5 + 897.26 + 14.69 x 10 + 77.33 x 10 = 5922.19

do this sum again

Here are the calculations again:

10000 x 0 = 0

1000 x 5 = 5000

897.26 + 5000 = 5897.26

14.69 x 10 = 146.9

5897.26 + 146.9 = 6044.16

77.33 x 10 = 773.3

6044.16 + 773.3 = 6817.46

Final score: 6817.46

1,357 ms ttft • 70.89 tokens/s

Type a message

Options

View Code

Temperature

temperature

0.6

Max Tokens

max\_tokens

16,384

Top P

top\_p

1

Top K

top\_k

40

Presence Penalty

presence\_penalty

0

Frequency Penalty

frequency\_penalty

0

Stop

stop

Enter a stop word

Context Length Exceeded Behavior

context\_length\_exceeded\_behavior

None

Echo

echo

Echoecho