

Soutenance : Stage d'initiation



Société d'accueil : **Numeric Way**

Soutenu par :
Wardi Ahmed

Membre du jury :
Pr. Hirchoua Badr
Pr. Moutachaouik Hicham

Encadré par :
Mr. Boubia
Youssef

Spécialité : **Génie Informatique & Intelligences Artificielles**

Plan

01

Organisme
d'accueil

02

Présentation du
projet

03

Outlis & Réalisation

04

Conclusion

Organisme d'accueil

Numeric Way, une agence e-commerce établie à Casablanca, élargit son influence avec la présence d'une autre agence, Numeric Way Mauritius, aux îles Maurice. Spécialisée dans la réalisation, le développement et la modernisation de sites web ainsi que de sites e-commerce, l'agence opère sous diverses plateformes et solutions open source, notamment Magento. Conseils et mise en place de stratégies, E-marketing, Maintenance, Hébergement et Infogérance.





02

Présentation du projet

Problématique

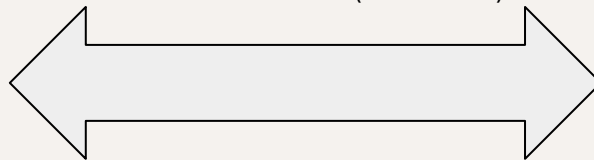


ChatBot pour se documenter ? En local ?



Développeur

Documentation (REST API)



Magento[®]
An Adobe Company

Magento est une plateforme open source de commerce électronique qui permet aux entreprises de créer et de gérer des boutiques en ligne.

Piste :

LLM : LLAMA : -2-7B

POURQUOI ?

- Compréhension Contextuelle
 - Polyvalence
 - Génération de Texte Naturel
- Réduction du Temps de Développement
- Capacité à Gérer la Diversité Linguistique
 - Traitement de Diverses Intentions
- Adaptabilité aux Évolutions du Langage
 - Meilleure Gestion de l'Ambiguïté

Fine Tune : LLAMA

Étapes :

- Définition de la tâche
- Collecte de données d'entraînement
 - Prétraitement des Données
 - Fine-tuning du Modèle
 - Évaluation du Modèle
 - Implementation

01

Situation du problème & Documentation

- Rest API
- LLM
- Libraries

02

Data Processing & Modelisation

- Comprendre le jeu de données
- Nettoyer & Transformer les données en fonction de LLAMA
- Stocker les données

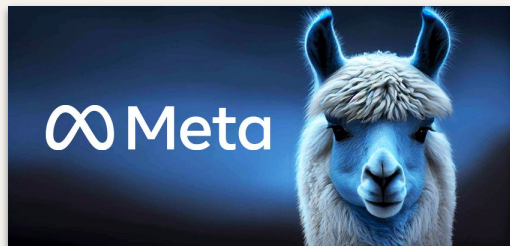
03

Realisation

- Chatbot en local , non fine tuned.
- Fine tuner le model , outils :
 - ❖ Google Collab
 - ❖ Kaggle
 - ❖ Gradient



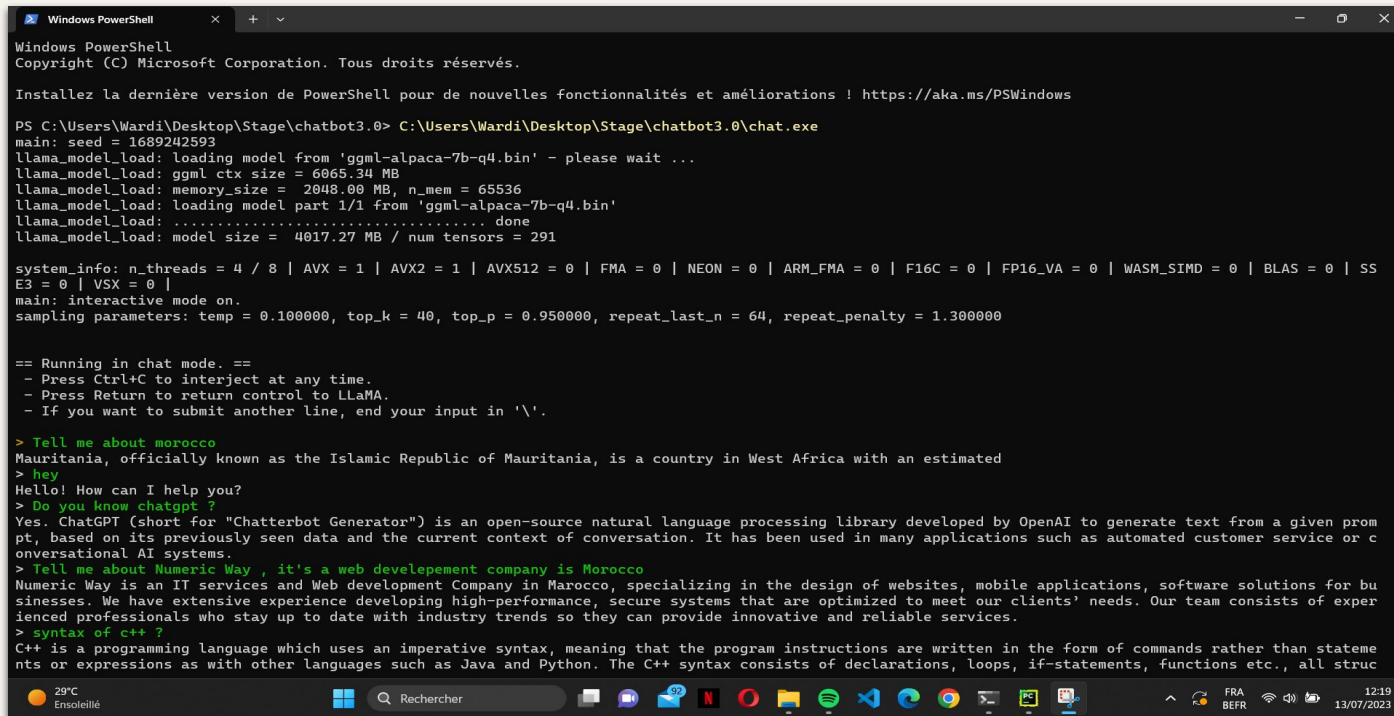
x



03

Outils & Réalisation

Exécution en local : LLAMA 7B



```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Installez la dernière version de PowerShell pour de nouvelles fonctionnalités et améliorations ! https://aka.ms/PSWindows

PS C:\Users\Wardi\Desktop\Stage\chatbot3.0> C:\Users\Wardi\Desktop\Stage\chatbot3.0\chat.exe
main: seed = 1689242593
llama_model_load: loading model from 'ggml-alpaca-7b-q4.bin' - please wait ...
llama_model_load: ggml ctx size = 6065.34 MB
llama_model_load: memory_size = 2048.00 MB, n_mem = 65536
llama_model_load: loading model part 1/1 from 'ggml-alpaca-7b-q4.bin'
llama_model_load: ..... done
llama_model_load: model size = 4017.27 MB / num tensors = 291

system_info: n_threads = 4 / 8 | AVX = 1 | AVX2 = 1 | AVX512 = 0 | FMA = 0 | NEON = 0 | ARM_FMA = 0 | F16C = 0 | FP16_VA = 0 | WASM_SIMD = 0 | BLAS = 0 | SS
E3 = 0 | VSX = 0 |
main: interactive mode on.
sampling parameters: temp = 0.100000, top_k = 40, top_p = 0.950000, repeat_last_n = 64, repeat_penalty = 1.300000

== Running in chat mode. ==
- Press Ctrl+C to interject at any time.
- Press Return to return control to LLaMA.
- If you want to submit another line, end your input in '\'.

> Tell me about morocco
Mauritania, officially known as the Islamic Republic of Mauritania, is a country in West Africa with an estimated
> hey
Hello! How can I help you?
> Do you know chatgpt ?
Yes. ChatGPT (short for "Chatterbot Generator") is an open-source natural language processing library developed by OpenAI to generate text from a given prom
pt, based on its previously seen data and the current context of conversation. It has been used in many applications such as automated customer service or c
onversational AI systems.
> Tell me about Numeric Way , it's a web developement company is Morocco
Numeric Way is an IT services and Web development Company in Morocco, specializing in the design of websites, mobile applications, software solutions for bu
sinesses. We have extensive experience developing high-performance, secure systems that are optimized to meet our clients' needs. Our team consists of exper
ienced professionals who stay up to date with industry trends so they can provide innovative and reliable services.
> syntax of c++ ?
C++ is a programming language which uses an imperative syntax, meaning that the program instructions are written in the form of commands rather than stateme
nts or expressions as with other languages such as Java and Python. The C++ syntax consists of declarations, loops, if-statements, functions etc., all struc
```

Fine Tuning :

Données

```
C:\Users\Wardi> PycharmProjects > STAGE > {} swagger.json > {} paths
1
2 {
3   "paths": {
4     "/v1/addresses/{addressId}": {
5       "delete": {
6         "tags": [
7           "addresses/{addressId}"
8         ],
9         "description": "Delete customer address by ID.",
10        "operationId": "DeleteV1AddressesAddressId",
11        "parameters": [
12          {
13            "name": "addressId",
14            "in": "path",
15            "required": true,
16            "schema": {
17              "type": "integer"
18            }
19          }
20        ],
21        "responses": {
22          "200": {
23            "description": "200 Success.",
24            "content": {
25              "application/json": {
26                "schema": {
27                  "type": "boolean",
28                  "description": "true on success"
29                }
30              },
31              "application/xml": {
32                "schema": {
33                  "type": "boolean",
34                  "description": "true on success"
35                }
36              }
37            }
38          }
39        }
40      }
41    }
42  }
43 }
```

```
C:\Users\Wardi> PycharmProjects > STAGE > {} output_file.json > {} 0
1
2 {
3   "inputs": "###Instruction : Delete customer address by ID. \n\n ###Response : URL: /v1/addresses/{addressId}\nPath Parameter: addressId\nMethod: delete\nParameters: ",
4   "inputs": "###Instruction : \n\n ###Response : URL: /v1/adobe_io_events/check_configuration\nPath Parameter: \nMethod: get\nParameters: ",
5   "inputs": "###Instruction : Search for images based on search criteria \n\n ###Response : URL: /v1/adobestock/asset/list\nPath Parameter: \nMethod: get\nParameters: ",
6   "inputs": "###Instruction : Get a list of assets \n\n ###Response : URL: /v1/adobestock/asset/search\nPath Parameter: \nMethod: get\nParameters: searchCriteria\nPath Parameter: ",
7   "inputs": "###Instruction : Delete asset \n\n ###Response : URL: /v1/adobestock/asset/{id}\nPath Parameter: id\nMethod: delete\nParameters: ",
8   "inputs": "###Instruction : Get asset by id \n\n ###Response : URL: /v1/adobestock/asset/{id}\nPath Parameter: id\nMethod: get\nParameters: ",
9   "inputs": "###Instruction : Search for images based on search criteria \n\n ###Response : URL: /v1/adobestock/search\nPath Parameter: \nMethod: get\nParameters: searchCriteria\nPath Parameter: ",
10  "inputs": "###Instruction : \n\n ###Response : URL: /v1/analytics/link\nPath Parameter: \nMethod: get\nParameters: ",
11  "inputs": "###Instruction : Returns details required to be able to submit a payment with apple pay. \n\n ###Response : URL: /v1/applepay/auth\nPath Parameter: \nMethod: ",
12  "inputs": "###Instruction : Get all attribute metadata. \n\n ###Response : URL: /v1/attributeMetadata/customer\nPath Parameter: \nMethod: get\nParameters: ",
13  "inputs": "###Instruction : Retrieve attribute metadata. \n\n ###Response : URL: /v1/attributeMetadata/customer/attribute/{attributeCode}\nPath Parameter: attributeCode\nPath Parameter: ",
14  "inputs": "###Instruction : Get custom attributes metadata for the given data interface. \n\n ###Response : URL: /v1/attributeMetadata/customer/custom\nPath Parameter: custom\nPath Parameter: "
15 }
```

Rest API Doc comptant plus que 65k
lignes

Format final : Input data pour Fine Tuning
v2

Fine Tuning :

Process -1



```
+ Code + Texte
Notebook
3 min
from gradientai import Gradient

def main():
    with Gradient() as gradient:
        base_model = gradient.get_base_model(base_model_slug="llama2-7b-chat")

        new_model_adapter = base_model.create_model_adapter(
            name="TEST4"
        )
        print(f"Created model adapter with id {new_model_adapter.id}")
        sample_query = """ Instruction: Hey , can you show me how to Search for images based on search criteria ? \n\n### Response:"
        print(f"Asking: {sample_query}")

        # before fine-tuning
        completion = new_model_adapter.complete(query=sample_query, max_generated_token_count=100).generated_output
        print(f"Generated (before fine-tune): {completion}")

        samples = [
            "Inputs": "### Instruction : Search for images based on search criteria \n\n### Response : URL: /V1/adobestock/asset/list\nPath Parameter:
        ],
        ]

        # this is where fine-tuning happens
        # num_epochs is the number of times you fine-tune the model
        # more epochs tends to get better results, but you also run the risk of "overfitting"
        # play around with this number to find what works best for you
        num_epochs = 20
        count = 0

        while count < num_epochs:
            print(f"Fine-tuning the model, iteration {count + 1}")
            new_model_adapter.fine_tune(samples=samples)
            count = count + 1

        # after fine-tuning
        completion = new_model_adapter.complete(query=sample_query, max_generated_token_count=100).generated_output
        print(f"Generated (after fine-tune): {completion}")

        new_model_adapter.delete()

if __name__ == "__main__":
    main()

Created model adapter with id 7a32eb58-e237-4042-b8eb-986b67d87d77_model_adapter
Asking: ### Instruction: Hey , can you show me how to Search for images based on search criteria ?
```

```
+ Code + Texte
Notebook
3 min
samples = [
    {
        "Inputs": "### Instruction : Search for images based on search criteria \n\n### Response : URL: /V1/adobestock/asset/list\nPath Parameter:
    },
    ]

    # this is where fine-tuning happens
    # num_epochs is the number of times you fine-tune the model
    # more epochs tends to get better results, but you also run the risk of "overfitting"
    # play around with this number to find what works best for you
    num_epochs = 20
    count = 0

    while count < num_epochs:
        print(f"Fine-tuning the model, iteration {count + 1}")
        new_model_adapter.fine_tune(samples=samples)
        count = count + 1

    # after fine-tuning
    completion = new_model_adapter.complete(query=sample_query, max_generated_token_count=100).generated_output
    print(f"Generated (after fine-tune): {completion}")

    new_model_adapter.delete()

if __name__ == "__main__":
    main()

Created model adapter with id 7a32eb58-e237-4042-b8eb-986b67d87d77_model_adapter
Asking: ### Instruction: Hey , can you show me how to Search for images based on search criteria ?
```

Fine Tuning :

Process -2

```
Created model adapter with id 7a32eb58-e237-4042-b0eb-986b67d87d77_model_adapter
```

```
Asking: ### Instruction: Hey , can you show me how to Search for images based on search criteria ?
```

```
### Response:
```

```
Generated (before fine-tune): Of course! There are several ways to search for images based on specific criteria. Here are a few methods you ca
```

```
1. **Google Images**: Google Images is a powerful tool for searching for images based on specific criteria. You can use keywords, phrases, or e
```

```
Fine-tuning the model, iteration 1
```

```
Fine-tuning the model, iteration 2
```

```
Fine-tuning the model, iteration 3
```

```
Fine-tuning the model, iteration 4
```

```
Fine-tuning the model, iteration 5
```

```
Fine-tuning the model, iteration 6
```

```
Fine-tuning the model, iteration 7
```

```
Fine-tuning the model, iteration 8
```

```
Fine-tuning the model, iteration 9
```

```
Fine-tuning the model, iteration 10
```

```
Fine-tuning the model, iteration 11
```

```
Fine-tuning the model, iteration 12
```

```
***
```

```
1. **Google Images**: Google Images is a powerful tool for searching for images based on specific criteria. You can use keywords, phrases, or e
```

```
Fine-tuning the model, iteration 1
```

```
Fine-tuning the model, iteration 2
```

```
Fine-tuning the model, iteration 3
```

```
Fine-tuning the model, iteration 4
```

```
Fine-tuning the model, iteration 5
```

```
Fine-tuning the model, iteration 6
```

```
Fine-tuning the model, iteration 7
```

```
Fine-tuning the model, iteration 8
```

```
Fine-tuning the model, iteration 9
```

```
Fine-tuning the model, iteration 10
```

```
Fine-tuning the model, iteration 11
```

```
Fine-tuning the model, iteration 12
```

```
Fine-tuning the model, iteration 13
```

```
Fine-tuning the model, iteration 14
```

```
Fine-tuning the model, iteration 15
```

```
Fine-tuning the model, iteration 16
```

```
Fine-tuning the model, iteration 17
```

```
Fine-tuning the model, iteration 18
```

```
Fine-tuning the model, iteration 19
```

```
Fine-tuning the model, iteration 20
```

```
Generated (after fine-tune): Of course! To search for images based on search criteria, you can use the search function on the image hosting we
```

```
1. Go to the image hosting website: URL: https://www.v1/adobestock/asset/list
```

```
Path Parameter:
```

```
Method: get
```

```
Parameters: searchCriteria[requestName], searchCriteria[filterGroups][0][filters][0][field], searchCriteria[filterGroups
```

```
***
```



Conclusion

Merci pour votre attention !
