Iliad Documentation - AI API

Summary

Iliad is a REST API for accessing large language model resources at AbbVie.

AI - Chat

POST /api/v1/chat/gpt-3.5-turbo

> *Create a chat completion with gpt-3.5-turbo.*

Request JSON Object:messages (list[Message]) – Conversation historytemperature (float) – Model temperature (optional)Request Headers:X-API-Key – The Iliad API keyX-User-Token – End-user's AD token (optional)Status Codes:200 OK – Successful response400 Bad Request – Bad request401 Unauthorized – Bad end-user token422 Unprocessable Entity – Poorly-formatted request

Example:

```
import requests

resp = requests.post(
    url=BASE_URL + "/api/v1/chat/gpt-3.5-turbo",
    json={"messages": [{"role": "user", "content": "what's two plus two?"}]},
    headers={"X-API-Key": os.environ.get("ILIAD_API_KEY")}
)
resp.raise_for_status()
print(resp.json())
```

Summary

Batch Analyze (dataframe)

POST /api/v1/analyze/batch/dataframe

Analyze one or more dataframes in batches.To be used for longer spreadsheets where the token count is expected to exceed the specified model's context window.If the token count exceeds the context window limit, analyze/batch/dataframe will break the dataframe(s) into mini-batches, and maps the original prompt to each mini-batch for chat completion.Supports multi-sheet dataframes.

Form Parameters:files – One or more uploaded files to be analyzed.prompt – Instructions or questions for the language model.model – Which model to analyze the document. Choose from "gpt-4-32k", "claude-3-sonnet", or "gpt-3.5-turbo-16k". (Optional. Default: gpt-4-32k) analyze_by_row – (Boolean) Analyze dataframes by row, providing an additional column consisting of the LLM's response for each row. (Optional. Default: false)return_rag – (Boolean) If True, perform Retrieval-Augmented Generation. Returns an array of tuples consisting of the LLM's response and the segment extracted from the corresponding batch with the highest similarity score. (Optional. Default: false)return_dataframes – (Boolean) Include the entire augmented dataframe in the response. Cannot be set to True if analyze_by_row is false. (Optional. Default: false) return_harmonized_response – (Boolean) If True, combines chat replies from all batches, and feeds the result to the LLM to generate a single "harmonized_response". (Optional. Default: false)harmonization_prompt – (String) Prompt applied to aggregated chat replies to be returned as "harmonized_response". Defaults to prompt if None. (Optional. Default: None)

Request Headers:X-API-Key – The Iliad API keyX-User-Token – End-user's AD token (optional)

Status Codes:200 OK – Successful response401 Unauthorized – Bad end-user token415 Unsupported Media Type – Invalid file type uploaded422 Unprocessable Entity – Poorly-formatted request

Example request:

```
import requests

url = BASE_URL + "/api/v1/analyze/batch/dataframe"
data={
    "prompt": "What's all this then?",
    "analyze_by_row": True,
    "return_dataframes": True,
}
files = [("files", open("/path/to/train_sm.csv", "rb"))]
headers = {"X-API-Key": os.environ.get("ILIAD_API_KEY")}
resp = requests.post(url=url, data=data, headers=headers, files=files)
resp.raise_for_status()
print(resp.json())
```

Batch Analyze (document)¶

POST /api/v1/analyze/batch/document

Analyze one or more documents in batches.To be used for longer documents where the token count is expected to exceed the specified model's context window.If the token count exceeds the context window limit, analyze/batch/document will break the document(s) into mini-batches, and maps the original prompt to each mini-batch for chat completion.

Form Parameters:files – One or more uploaded files to be analyzed.prompt – Instructions or questions for the language model.model – Which model to analyze the document. Choose from "gpt-4-32k", "claude-3-sonnet", or "gpt-3.5-turbo-16k". (Optional. Default: gpt-4-32k)return_rag – (Boolean) If True, perform Retrieval-Augmented Generation. Returns an array of tuples consisting of the LLM's response the the segment extracted from the corresponding batch with the highest similarity score. (Optional. Default: false)return_harmonized_response – (Boolean) If True, combines chat replies from all batches, and feeds the result to the LLM to generate a single "harmonized_response". (Optional. Default: false) harmonization_prompt – (String) Prompt applied to aggregated chat replies to be returned as "harmonized_response". Defaults to prompt if None. (Optional. Default: None)
Request Headers:X-API-Key – The Iliad API keyX-User-Token – End-user's AD token (optional)
Status Codes:200 OK – Successful response401 Unauthorized – Bad end-user token415 Unsupported Media Type – Invalid file type uploaded422 Unprocessable Entity – Poorly-formatted request

Example request:

```
import requests

url = BASE_URL + "/api/v1/analyze/batch/document"
payload = {"prompt": "Summarize"}
files = [("files", open("/path/to/2305.14564.pdf", "rb"))]
data={
    "prompt": "What's all this then?",
    "return_rag": False,
    "return_harmonized_response": True,
}
headers = {"X-API-Key": os.environ.get("ILIAD_API_KEY")}
resp = requests.post(url=url, data=data, headers=headers, files=files)
resp.raise_for_status()
print(resp.json())
```