

Done: Overview of system on Agent-Issue Bench

WIP: Github Api OverheadOptimization

crewAIInc/crewAI/issues/1753

crewAIInc/crewAI/pull/1752

Description

When using output_pydantic with sqlite3 memory and the type in the output_pydantic contains a field of type Enum. In this case, an exception will be raised inside of the KickoffTaskOutputsSQLiteStorage in the add function which calls to:

```
json.dumps(output, cls=CrewJSONEncoder)
Claiming: Object of type is not JSON serializable.
```

Steps to Reproduce

1. Create an Enum class:

```
from enum import Enum

class CrewStatus(Enum):
    TODO = "To Do"
    SUCCESSFUL = "Successful"
    FAILED = "Failed"
```

2. Create a crew with a small task and `output_pydantic=CrewStatus`, `memory=True`
3. Run the `crew.kickoff()` and get an exception `Object of type CrewStatus is not JSON serializable.`

Expected behavior

Not getting the exception

Screenshots/Code snippets

```
raise TypeError(f'Object of type {o.__class__.__name__} '
                'TypeError: Object of type CrewStatus is not JSON serializable')
```

```
1  import json
2  from datetime import date, datetime
3  from decimal import Decimal
4  + from enum import Enum
5  from uuid import UUID
6
7  from pydantic import BaseModel

11     def default(self, obj):
12         if isinstance(obj, BaseModel):
13             return self._handle_pydantic_model(obj)
14  + elif isinstance(obj, UUID) or isinstance(obj, Decimal) or isinstance(obj, Enum):
15             return str(obj)
16
17         elif isinstance(obj, datetime) or isinstance(obj, date):
```

Previous System Output: No Agent Issue
After checking: No Agent Issue
Key fix: Allow serialization for Enum obj
Agent Reasoning

The problem described pertains to JSON serialization of Enum types in a specific storage class, which is not related to LLM provider integration, tool invocation, memory mechanisms, LLM operation, workflows, or utilities unique to agent systems.

context

crewAIInc/crewAI/issues/1824

crewAIInc/crewAI/pull/1826

Description

The build for `tiktoken==0.7.0` fails while installing `crewai`

Steps to Reproduce

Run `pip install crewai` or `uv pip install crewai`

Expected behavior

The build for `tiktoken` should not fail

Screenshots/Code snippets

```
> uv pip install crewai
Resolved 220 packages in 378ms
x Failed to build 'tiktoken==0.7.0'
└─ The build backend returned an error
└─ Call to 'setuptools.build_meta.build_wheel' failed (exit status: 1)
```

hint: This usually indicates a problem with the package or the build environment.
help: 'tiktoken' (v0.7.0) was included because 'crewai' (v0.86.0) depends on 'litellm' (v1.56.4) which depends on 'tiktoken'

```
50
51 [project.optional-dependencies]
52 tools = ["crewai-tools>=0.17.0"]
53 + embeddings = [
54 +     "tiktoken~=0.7.0"
55 + ]
56 agentops = ["agentops>=0.3.0"]
57 fastembed = ["fastembed>=0.4.1"]
58 pdfplumber = [
```

Previous System Output: No Agent Issue
After checking: **Agent Issue(just lower ratio)**
Key fix: Name and build the wheel directly
Agent Reasoning

The problem is related to a failed build of the `'tiktoken'` library during installation, which is a dependency issue rather than an issue specific to LLM-based agent systems or their unique components.

add on of this issue

issue
1824

pr
1826

issue closed

pr
1840

115
other
issues

```
1457     assert agent.llm.model == "ollama/llama3.2:3b"
1458     assert agent.llm.base_url == "http://localhost:11434"
1459
1460 +     task = "Respond in 20 words. Which model are you?"
1461     response = agent.llm.call({"role": "user", "content": task})
1462
1463 +     assert response

1473         temperature=0.7,
1474         max_tokens=30,
1475     )
1476 +     messages = [
1477 +         {"role": "user", "content": "Respond in 20 words. Which model are you?"}
1478 +     ]
1479
1480     response = llm.call(messages)
1481
```

🔖 **Spartan-71** added **bug** on Dec 30, 2024

🔗 **devin-ai-integration** added a commit that references this issue on Dec 30, 2024
feat: add tiktoken as explicit dependency and document Rust requireme! ... bb5bed5

🔗 **devin-ai-integration** mentioned this on Dec 30, 2024
fix: add tiktoken as explicit dependency and document Rust requirement #1826

🔗 **joaomdmoura** added a commit that references this issue on Dec 30, 2024
fix: add tiktoken as explicit dependency and document Rust requiremen! ... Verified ba0965e

joaomdmoura on Dec 30, 2024 Collaborator ...

Merged a fix that should address this in main, it will be in the new version later today but you can test directly from main if you want to :D
Also added new instruction to readme about it, great flag **@Spartan-71** , if this doesn't help it lmk and I'll reopen the issue

👍 1

🔒 **joaomdmoura** closed this as **completed** on Dec 30, 2024

🔗 **bhancockio** added 2 commits that reference this issue on Jan 2
fix: add tiktoken as explicit dependency and document Rust requiremen! ... a7b00ce
Trying out timeouts (#1840) ... 4bcc3b5

🔗 **devin-ai-integration** added 2 commits that reference this issue on Feb 9
fix: add tiktoken as explicit dependency and document Rust requiremen! ... 4e575d2
Trying out timeouts (#1840) ... d6acd9c

this is an agent issue

They change the prompt and
eliminate timeout

Josh-XT/AGiXT/issues/1026

Josh-XT/AGiXT/issues/1026

Description

```
agixt-agixt-1 | File "/agixt/extensions/voice_chat.py", line 17, in __init__
agixt-agixt-1 |     if kwargs["USE_STREAMLABS_TTS"].lower() == "true":
agixt-agixt-1 | AttributeError: 'bool' object has no attribute 'lower'
```



Steps to Reproduce the Bug

1. Follow quick start instructions (clone, run shell script, it runs docker containers).
2. Go to the OpenAI Agent to add an API key to the settings.
3. Click save.
4. In future error appears whenever trying to load settings page.

Expected Behavior

Agent should save normally, error should not appear, perhaps there should be a way to reset this file to default?


```
self.voice_prompt = "Voice Chat"
self.tts_command = "Speak with TTS with Streamlabs Text to Speech"
if "USE_STREAMLABS_TTS" in kwargs:
+     if isinstance(kwargs["USE_STREAMLABS_TTS"], bool):
+         if kwargs["USE_STREAMLABS_TTS"]:
+             self.tts_command = "Speak with TTS with Streamlabs Text to Speech"
+         else:
+             if kwargs["USE_STREAMLABS_TTS"].lower() == "true":
+                 self.tts_command = "Speak with TTS with Streamlabs Text to Speech"
if "USE_GTTTS" in kwargs:
+     if isinstance(kwargs["USE_GTTTS"], bool):
+         if kwargs["USE_GTTTS"]:
+             self.tts_command = "Speak with GTTS"
+         else:
+             if kwargs["USE_GTTTS"].lower() == "true":
+                 self.tts_command = "Speak with GTTS"
if "USE_HUGGINGFACE_TTS" in kwargs:
    if (
        kwargs["USE_HUGGINGFACE_TTS"].lower() == "true"
```

Previous System Output: Agent Issue
After checking: Agent Issue
Key fix:
Agent Reasoning

The issue involves a tool configuration error where a boolean value is incorrectly handled as a string, causing an `AttributeError`. This relates to **agent-specific components**, as it affects the behavior of the LLM-based agent system when saving settings.

ModuleNotFoundError: No module named 'griffe.enumerations' #239

Closed #245



aizhweiwei opened on Aug 21, 2024

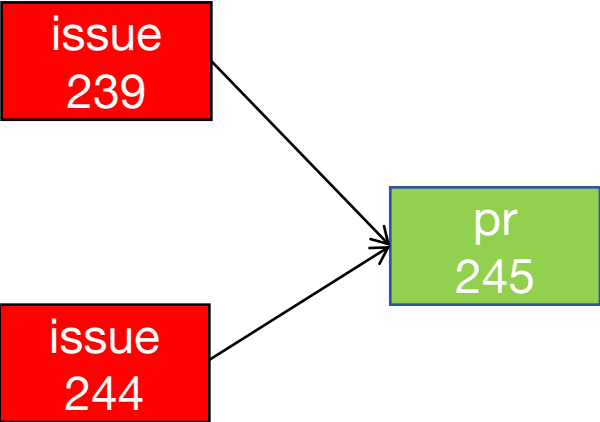
```
(xtuner0121) root@intern-studio-160750:~/puyu/7/agent# streamlit run examples/internlm2_agent_web_demo.py
Collecting usage statistics. To deactivate, set browser.gatherUsageStats to false.

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8502
Network URL: http://192.168.239.222:8502
External URL: http://192.168.239.222:8502

2024-08-21 18:42:56.930 Uncaught app exception
Traceback (most recent call last):
  File "/root/.conda/envs/xtuner0121/lib/python3.10/site-packages/streamlit/runtime/scriptrunner/script_runner.py", line 589, in _run_script
    exec(code, module.__dict__)
  File "/root/puyu/7/agent/examples/internlm2_agent_web_demo.py", line 8, in <module>
    from lagent.actions import ActionExecutor, ArxivSearch, IPythonInterpreter
  File "/root/.conda/envs/xtuner0121/lib/python3.10/site-packages/lagent/__init__.py", line 2, in <module>
    from .actions import * # noqa: F401, F403
  File "/root/.conda/envs/xtuner0121/lib/python3.10/site-packages/lagent/actions/__init__.py", line 3, in <module>
    from .action_executor import ActionExecutor
  File "/root/.conda/envs/xtuner0121/lib/python3.10/site-packages/lagent/actions/action_executor.py", line 4, in <module>
    from .base_action import BaseAction
  File "/root/.conda/envs/xtuner0121/lib/python3.10/site-packages/lagent/actions/base_action.py", line 16, in <module>
    from griffe.enumerations import DocstringSectionKind
ModuleNotFoundError: No module named 'griffe.enumerations'
```

what's happen with this question



4	python-pptx
5	+ timeout_decorator
6	torch
7	+ transformers>=4.34,<=4.40
8	vllm>=0.3.3

2	distro
3	filelock
4	func_timeout
5	+ griffe<1.0
6	json5
7	jsonschema
8	jupyter

13	tiktoken
14	timeout-decorator
15	typing-extensions
16	+ duckduckgo_search==5.3.1b1

Previous System Output: No Agent Issue
After checking: No Agent Issue
Key fix: Fix the code according to griffe’s interface changes

Agent Reasoning

the issue is a generic Python module import error and does not relate to LLM-based agent systems, LLM provider integration, tool invocation, memory mechanisms, or agent-specific workflows.

agentcoinorg/evo.ninja/issues/{504,515,525,594,652}

(main branch actually named dev) -> can't foresee, just discard

504 Max token overflow causing embedding err

```
Action executed:

Web search for 'Tesla financial reports from 2003 to 2023'

The function 'web_search' failed, this is the error:

SyntaxError: Unexpected token 'B', "Body excee"... is not valid JSON

Arguments:

{
  "queries": [
    "Tesla financial reports from 2003 to 2023"
  ]
}
```

504 fix: add related args

```
constructor(
  private tokenizer: Tokenizer,
  private _setCapReached: () => void,
  private modelConfig: {
    model: string,
    maxTokensPerInput: number,
    maxInputsPerRequest: number
  } = DEFAULT_ADA_CONFIG,
) {}
```

515: calling sub agent for simple mission is overhead

```
USER
what can you do

EVO

Prediction:
-> The best next step for the assistant would be to provide a list of capabilities or tasks that it can accomplish. This will help the user understand the assistant's capabilities and determine how it can assist them.

Selected agent:
-> Researcher

Step 1

Action executed:
Web search for 'what can an advanced web information retriever do'
```

515 fix add an clone empty agent for shortcut

```
28   AgentFunctionBase<unknown>[]
29   ]> => {
30     const allAgents: Agent[] = [
31       CsvAnalystAgent,
32       ResearcherAgent,
33       SynthesizerAgent,
34 +   ].map((agentClass) => new agentClass(context.cloneEmpty()));
35
36     const agentsWithPrompts = allAgents.map(agent => {
37       return {
```

my opinion: definitely agent issue

OpenBMB/ChatDev/[issues/465](#)
OpenBMB/ChatDev/[issues/318](#)

no related pr

465



thinkwee on Dec 17, 2024

We have fixed it. Please pull the latest version, see [#466](#)



yswami closed this as completed on Dec 20, 2024

why no pr related: pr is in comment
then we can not get this relation with sql
request, but need ai to read



gpeal closed this as completed in [#5298](#) 4 days ago



pull added a commit that references this issue 4 days ago

[MCP] Render full MCP errors to the model ([openai#5298](#)) ...

what it should look like

318



yu-shaonian on Dec 31, 2023

yes, it lacks something in the Human and Art config file. We will fix it quick



thinkwee closed this as completed on Jan 2, 2024

why do you close it without fixing

OpenBMB/ChatDev/[issues/465](#)
OpenBMB/ChatDev/[issues/318](#)

no related pr

465

TypeError: __init__() got an unexpected keyword argument 'audio' #465

🔒 Closed

```
109     meta_dict: Optional[Dict[str, str]] = None
110     role: str = "user"
111     content: str = ""
112     refusal: str = None
113 +   audio: object = None
```

My opinion: 465 is, 318 is not

318: key missing

I get this when I run run.py

I've reinstalled all the modules, used conda, nothing is helping.

```
Traceback (most recent call last):
  File "/Users/djbritt/Downloads/ai/gamedev/update_12.31.23/ChatDev/run.py", line 102, in <module>
    chat_chain = ChatChain(config_path=config_path,
  File "/Users/djbritt/Downloads/ai/gamedev/update_12.31.23/ChatDev/chatdev/chat_chain.py", line 66, in __init__
    self.web_spider = self.config["web_spider"]
  KeyError: 'web_spider'
```



Fix hard coded `self.update_messages(output_messages[0])` #88

```

3 +
4     while message_counter < max_num_messages:
5
6         user_msgs, user_terminated, user_info = user_agent.step(
7         +         assistant_msg.to_user_chat_message())
8
9         # Condition 1: User terminates the chat
10        if user_terminated:
11
12            break
13
14        user_msg = user_msgs[0]
15        user_agent.update_messages(user_msg)
17 +        print(f"User:\n{user_msg.content}\n")
18
19
20        assistant_msgs, assistant_terminated, assistant_info = (
21        +         assistant_agent.step(user_msg.to_user_chat_message()))
22
23        # Condition 2: Assistant terminates the chat
24        if assistant_terminated:
25
26            break
27
28        assistant_msg = assistant_msgs[0]
29        +         assistant_agent.update_messages(assistant_msg)
30        print(f"Assistant:\n{assistant_msg.content}\n")
31
32
33        # Condition 3: Break if user does not give instruction
34        if user_no_instruct_word not in user_msg.content:

```

Fix hard coded `self.update_messages(output_messages[0])` #88

Closed



lightaime opened on Apr 15, 2023

Member

Assign

No one assigned

Labels

bug

https://github.com/lightaime/camel/blob/ba3ec7603ebc735600e32379df235e627c4141c8/camel/agents/chat_agent.py#L93

only url but no context

Previous System Output: No Agent Issue

After checking: Agent Issue

Key fix: Still hard code, but add logic of msg consumer

Agent Reasoning

The issue description **lacks context** about LLM provider usage, prompt management, memory behaviors, tool invocation, or workflow anomalies. It appears to be a code-specific issue without clear ties to agent system components or behaviors.

microsoft/autogen/issues/5124
microsoft/autogen/issues/5164

GrpcWorkerAgentRuntimeHost fails to start due to "no running event loop" error (Doc and Sample updated required) #5124

Closed Bug #5164

linznin opened on Jan 21 · edited by linznin

EditsContributor

What happened?

While using Distributed Agent Runtime to start `GrpcWorkerAgentRuntimeHost`, an error occurs. Below are the steps to reproduce the issue and the code snippet.

Steps to Reproduce:

1. Use the following code to start `GrpcWorkerAgentRuntimeHost`:

```
def main():
    host = GrpcWorkerAgentRuntimeHost(address="0.0.0.0:50051")
    host.start() # Start a host service in the background.

if __name__ == "__main__":
```

2. Run the script, and the following error appears:

```
Traceback (most recent call last):
  File "d:\Users\10609302\Documents\workspace\ai-agent\ag_event_host.py", line 96, in <module>
    main()
  File "d:\Users\10609302\Documents\workspace\ai-agent\ag_event_host.py", line 93, in main
    host.start() # Start a host service in the background.
  File "C:\Users\10609302\AppData\Local\pypoetry\Cache\virtualenvs\ai-agent-avTrqWV-py3.11\Lib\site-packages\autogen\agent\grpc_worker.py", line 100, in start
    self._serve_task = asyncio.create_task(self._serve())
  File "D:\Program Files\Python311\Lib\asyncio\tasks.py", line 381, in create_task
    loop = events.get_running_loop()
RuntimeError: no running event loop
sys:1: RuntimeWarning: coroutine 'GrpcWorkerAgentRuntimeHost._serve' was never awaited
```

Assignees

No one assigned

Labels

needs-triage

Type

Bug

Projects

No projects

Milestone

0.4.x-py3

No due date

Relationships

None yet

Development

Update Dis platform Sa

Notifications

It's not a bug, you need to run this inside an async function.

```
async def main():
    host = GrpcWorkerAgentRuntimeHost(address="0.0.0.0:50051")
    host.start() # Start a host service in the background.
    await service.stop_when_signal()

if __name__ == "__main__":
    asyncio.run(main())
```

See sample: https://github.com/microsoft/autogen/blob/main/python/samples/core_grpc_worker_runtime/run_host.py

linznin mentioned this on Jan 23

Update Distributed Agent Runtime Cross-platform Sample #5164

Previous System Output: No Agent Issue
After checking: No even an Issue
Key fix: Add a sample

Agent Reasoning

this is not an agent issue. It focuses on improving service shutdown handling for cross-platform compatibility, which does not involve LLM provider integration, tool invocation, memory mechanisms, LLM operation, workflows, or utilities specific to agent systems.

microsoft/autogen/issues/5012

microsoft/autogen/pull/5013/

Hello World code sample in README.md is not properly awaited #5012

ClosedBug#5013

Programmer-RD-AI opened on Jan 12

Contributor

What happened?

The Hello World code sample provided in the README.md file of the repository does not properly await the asynchronous agent.run() method. This results in a RuntimeWarning and improper execution of the sample code.

```
<coroutine object BaseChatAgent.run at 0x75adf0cbb2e0>
/home/ranuga/Jobs/Altrium-PD/Learning/AutoGen/basic.py:7: RuntimeWarning: coroutine 'BaseChatAgent.run' was never awaited
  print(agent.run(task="Say 'Hello World!'"))
RuntimeWarning: Enable tracemalloc to get the object allocation traceback
```

What did you expect to happen?

The code sample should execute correctly without raising a warning, and it should print the result of the task Say 'Hello World!'.

How can we reproduce it (as minimally and precisely as possible)?

1. Clone the repository and follow the setup instructions in the README.md.

2. Copy the Hello World code sample from the README.md.

3. Run the code without modifying it.

Previous System Output: No Agent Issue

After checking: No a code Issue

Key fix: fix the example in readme no change to code

Agent Reasoning

this is not an agent issue. The problem is related to incorrect usage of asynchronous code in the README.md, not specific to LLM provider integration, tool invocation, memory mechanisms, or other agent-specific components.

2 README.md

@@ -34,7 +34,7 @@ from autogen_ext.models.openai import OpenAIChatCompletionClient

34

35

36

37

38

39

40

34

35

36

37

38

39

40

python/packages/autogen-core/docs/src/index.md

@@ -119,7 +119,7 @@ from autogen_ext.models.openai import OpenAIChatCompletionClient

119

120

121

122

123

124

125

119

120

121

122

123

124

125

microsoft/autogen/issues/4733

microsoft/autogen/pull/4734

questcollector opened on Dec 16, 2024

Contributor ...

What happened?

When using user defined functions for code executor with `Alias`, `ImportFromModule`, error emerges.

In `build_python_functions_file` function uses `Set[Import = Union[str, Alias, ImportFromModule]]`,

`autogen_core.code_executor._func_with_reqs.py`

```
Import = Union[str, ImportFromModule, Alias]
...

def build_python_functions_file(
    funcs: Sequence[Union[FunctionWithRequirements[Any, P], Callable[..., Any], FunctionWithRequirementsStr]],
) -> str:
    """meta private"""
    # First collect all global imports
    global_imports: Set[Import] = set()
    for func in funcs:
        if isinstance(func, (FunctionWithRequirements, FunctionWithRequirementsStr)):
            global_imports.update(func.global_imports)
    ...
```

Alias and ImportFromModule cannot be updated to set because they are not hashable.

What did you expect to happen?

user defined functions in code executor does not support Alias, ImportModuleFrom case

TL,DR: Can not put **'Alias'** and **'ImportFromModule'** into dict or set(unhashable)

```
30
31 - @dataclass
32     class Alias:
33         name: str
34         alias: str
35
36
37 - @dataclass
38     class ImportFromModule:
39         module: str
40 -     imports: List[Union[str, Alias]]
```


```
33
34 + @dataclass(frozen=True)
35     class Alias:
36         name: str
37         alias: str
38
39
40 + @dataclass(frozen=True)
41     class ImportFromModule:
42         module: str
43 +     imports: Tuple[Union[str, Alias], ...]
44 +
45 +     ## backward compatibility
46 +     def __init__(
47 +         self,
48 +         module: str,
49 +         imports: Union[Tuple[Union[str, Alias], ...], List[Union[str, Alias]]],
50 +     ):
51 +         object.__setattr__(self, "module", module)
52 +         if isinstance(imports, list):
53 +             object.__setattr__(self, "imports", tuple(imports))
54 +         else:
55 +             object.__setattr__(self, "imports", imports)
56
57
```


Previous System Output: No Agent Issue
After checking: No Agent Issue
Key fix: Make 2 classes frozen after instanced,
or professionally, hashable
Agent Reasoning

this is not an agent issue. The problem is related to the hashability of classes used in a code executor utility, which is not specific to LLM-based agent systems or their unique components or behaviors

microsoft/autogen/issues/3362
microsoft/autogen/pull/3361

[Bug]: TypeError: unhashable type: 'ImportFromModule' for User Defined Functions #3361

 Closed  #3362

 zcipod opened on Aug 16, 2024 · edited by zcipod Edits Contributor ...

Describe the bug

I got this error when I created a user defined function:
`TypeError: unhashable type: 'ImportFromModule'`

Steps to reproduce

My function is like this:

```
@with_requirements(python_packages=["requests", "urllib3"], global_imports=["requests", "os", ImportFromModule])
def search_from_web(search_keyword: str):
    pass
```

Assignees

No one assigned

Labels

No labels

Type

No type

Projects

No projects

Milestone

No milestone

Actually a subset problem of #4733
but this only dirty fix it
but 4733 fix it entirely

```
6 from dataclasses import dataclass, field
7 from importlib.abc import SourceLoader
8 from textwrap import dedent, indent
9 + from typing import Any, Callable, Generic, List, Set, TypeVar, Union
10
11 from typing_extensions import ParamSpec
12
159     funcs: List[Union[FunctionWithRequirements[Any, P], Callable[..., Any], FunctionWithRequirementsStr]]
160 ) -> str:
161     # First collect all global imports
162 + global_imports: Set[str] = set()
163     for func in funcs:
164         if isinstance(func, (FunctionWithRequirements, FunctionWithRequirementsStr)):
165 + global_imports.update(map(_import_to_str, func.global_imports))
166
167 + content = "\n".join(global_imports) + "\n\n"
168
169     for func in funcs:
170         content += _to_code(func) + "\n\n"
```

Previous System Output: No Agent Issue
After checking: No Agent Issue
Key fix: dirty fix with “set” fixing
Agent Reasoning

This issue is related to a `TypeError` due to an unhashable type in a user-defined function, which is not specific to LLM-based agent systems or their unique components like LLM provider integration, tool invocation, or memory mechanisms.

microsoft/autogen/issues/4382
microsoft/autogen/pull/4402

ModuleNotFoundError: No module named 'grpc' #4382

Closed Bug #4402



gagb opened on Nov 26, 2024 Collaborator

What happened?

I ran the weather agents from <https://microsoft.github.io/autogen/dev/user-guide/agentchat-user-guide/quickstart.html>

What did you expect to happen?

I expected the code to run instead I got

```
File "/Users/gaganbansal/workspace/test-autogen/weather.py", line 5, in <module>
  from autogen_agentchat.teams import RoundRobinGroupChat
File "/Users/gaganbansal/miniconda3/envs/test-autogen/lib/python3.10/site-packages/autogen_agentchat/teams/_ir
  from .group_chat._base_group_chat import BaseGroupChat
File "/Users/gaganbansal/miniconda3/envs/test-autogen/lib/python3.10/site-packages/autogen_agentchat/teams/_gr
  from autogen_core.application import SingleThreadedAgentRuntime
File "/Users/gaganbansal/miniconda3/envs/test-autogen/lib/python3.10/site-packages/autogen_core/application/_:
  from ._worker_runtime import WorkerAgentRuntime
File "/Users/gaganbansal/miniconda3/envs/test-autogen/lib/python3.10/site-packages/autogen_core/application/_wc
  from .protos import agent_worker_pb2, agent_worker_pb2_grpc
File "/Users/gaganbansal/miniconda3/envs/test-autogen/lib/python3.10/site-packages/autogen_core/application/pr
  import grpc
ModuleNotFoundError: No module named 'grpc'
```

```
25         "opentelemetry-api~=1.27.0",
26         "asyncio_atexit",
27         "jsonref~=1.1.0",
28 +     "grpcio~=1.62.0", # TODO: update this once we have a stable version.
29 ]
30
31 [project.optional-dependencies]
```

Previous System Output: Agent Issue
After checking: No Agent Issue
Key fix: fix config file

Agent Reasoning

The issue involves a missing library (`grpc`) required for the agent's operation, which falls under "Tool dependency issues" in the taxonomy. This is related to tool invocation in the agent system, making it an agent issue.

microsoft/autogen/issues/1844
microsoft/autogen/pull/4734

[Bug]: overlap parameter in the split_text_to_chunks not used. #1844

Closed



mohamedaliELfegy opened on Mar 3, 2024

Describe the bug

overlap in the split_text_to_chunks function in the autogen.retrieve_utils.split_text_to_chunks , I think it's easy to fix it by adding it in the right place

Steps to reproduce

- 1. go to autogen.retrieve_utils file
- 2. search for the function split_text_to_chunks
- 3. it will be in the parameters but not used with cnt indexing

Assignees

No one assigned

Labels

rag

Type

No type

Projects

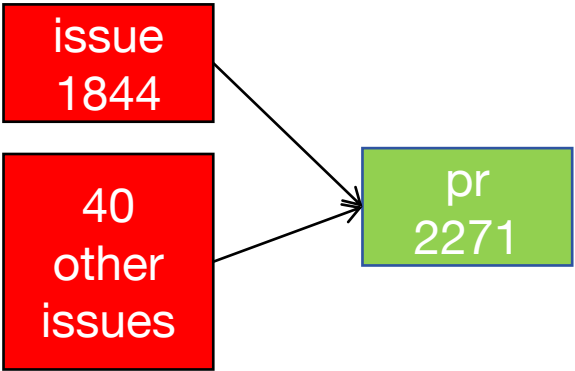
No projects

Previous System Output: No Agent Issue
After checking: Agent Issue
Key fix: make overlap

(overlap helps continuosity, but not bug)

Agent Reasoning

The problem described involves a parameter not being used in a text splitting utility function, which does not relate to LLM provider integration, tool invocation, memory mechanisms, LLM operation, workflows, or utilities unique to agent systems.



Stability Index

Discriminator Stability for Agent-Issue Detection

Scenario. Let a repository contain n issues. We run an LLM-based discriminator m times (e.g. different random seeds, different temperatures, or repeated sampling) to decide for each issue whether it is an *agent issue* (“yes”) or not (“no”).

Notation.

$$\mathcal{I} = \{I_1, I_2, \dots, I_n\}$$

is the set of n issues. For experiment (run) i (with the LLM), denote the set of issues labeled as *agent* by

$$\mathcal{A}_i \subseteq \mathcal{I}, \quad i = 1, \dots, m.$$

Define the intersection and union across runs:

$$A_{\cap} = \bigcap_{i=1}^m \mathcal{A}_i, \quad A_{\cup} = \bigcup_{i=1}^m \mathcal{A}_i.$$

1. Set-based stability (Intersection–Union Ratio)

The set-based stability index for agent detection is

$$S_{\text{set}} = \frac{|A_{\cap}|}{|A_{\cup}|},$$

with the convention $S_{\text{set}} = 1$ if $|A_{\cup}| = 0$.

Interpretation:

- $S_{\text{set}} = 1$: every issue that was ever labeled “agent” was labeled so in *all* runs (perfect agreement among runs on the positive set).
- $S_{\text{set}} = 0$: there is no issue that was labeled “agent” in all runs.

2. Entropy-based stability (per-issue uncertainty)

For each issue I_j ($j = 1, \dots, n$), let

$$f_j = \#\{i : I_j \in \mathcal{A}_i\}$$

be the number of runs that labeled I_j as an agent issue, and define the empirical probability

$$p_j = \frac{f_j}{m} \in \{0, \frac{1}{m}, \dots, 1\}.$$

Treat each issue's label across runs as a Bernoulli variable with empirical success probability p_j . The per-issue binary entropy (in bits) is

$$H_j = -p_j \log_2 p_j - (1 - p_j) \log_2 (1 - p_j),$$

with the usual convention $0 \log_2 0 = 0$.

Define the average entropy across all issues:

$$\overline{H} = \frac{1}{n} \sum_{j=1}^n H_j.$$

Then the entropy-based stability score is

$$S_{\text{entropy}} = 1 - \overline{H}.$$

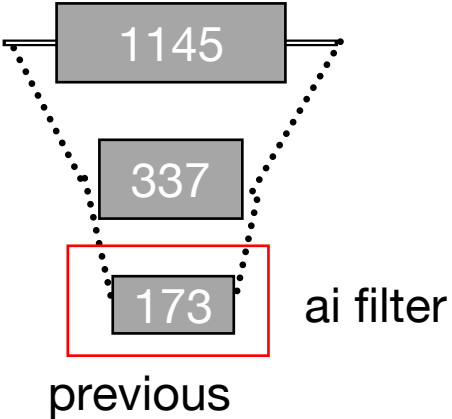
Interpretation:

- $S_{\text{entropy}} = 1$: every issue is deterministically labeled across runs ($p_j \in \{0, 1\}$ for all j).
- $S_{\text{entropy}} = 0$: every issue has maximal uncertainty ($p_j = 0.5$ for all j).

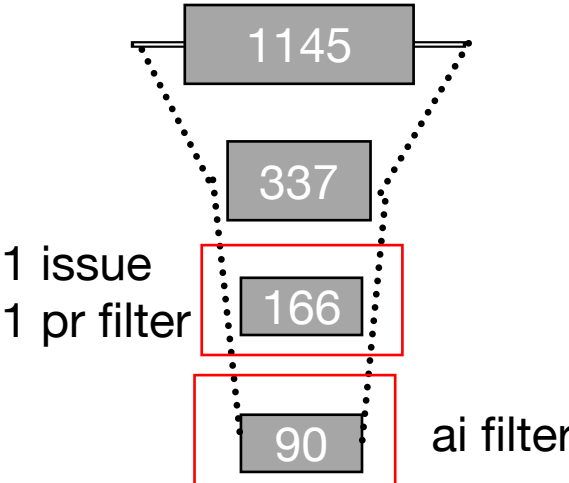
on codex:



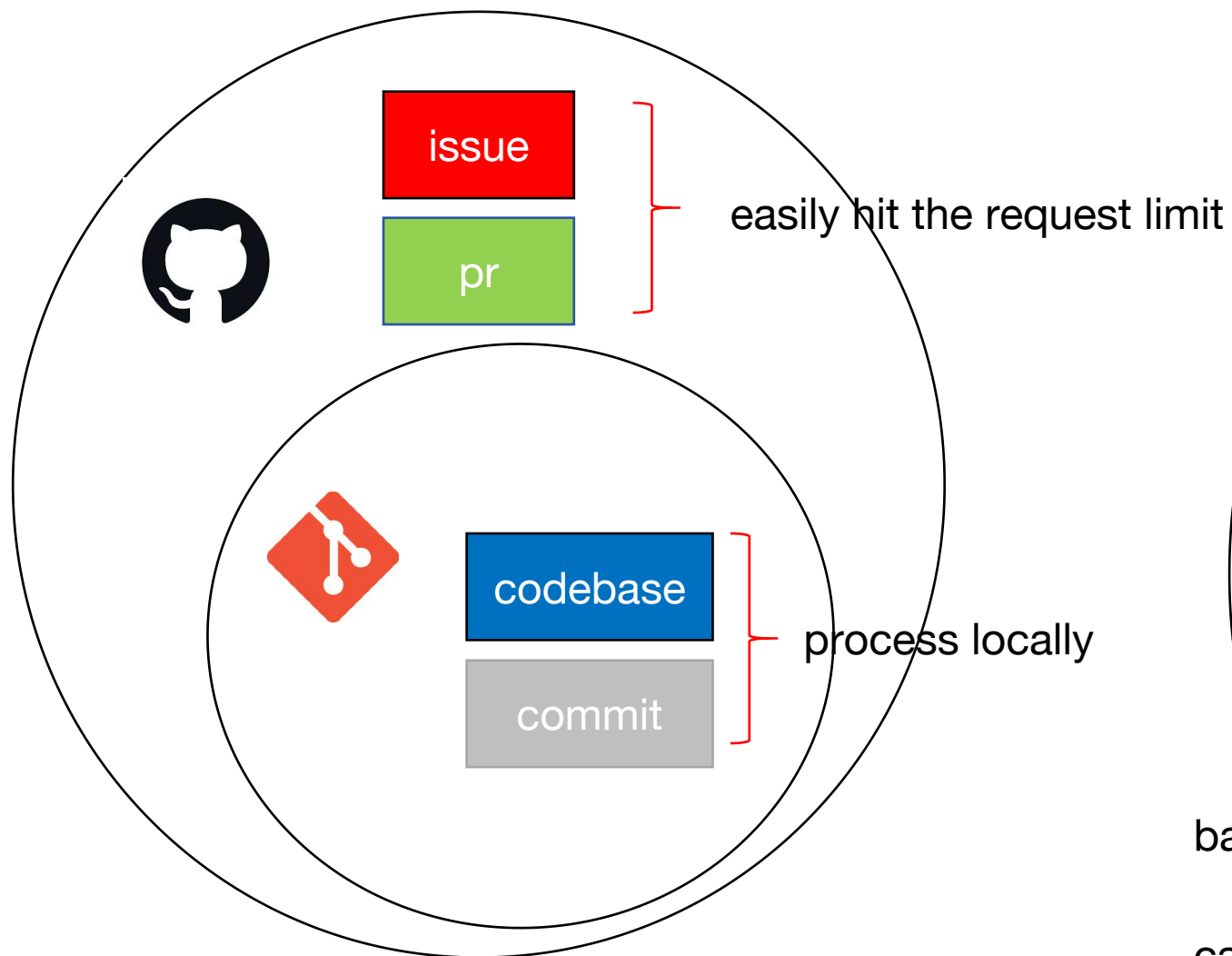
heat map
(only agent issues
and once changed issues)



index	volumn	# agent issue	set-based stability	entropy-based stability
1	166	89	87.2%	90.7%
2	166	90		
3	166	88		
4	167	89		
5	167	90		



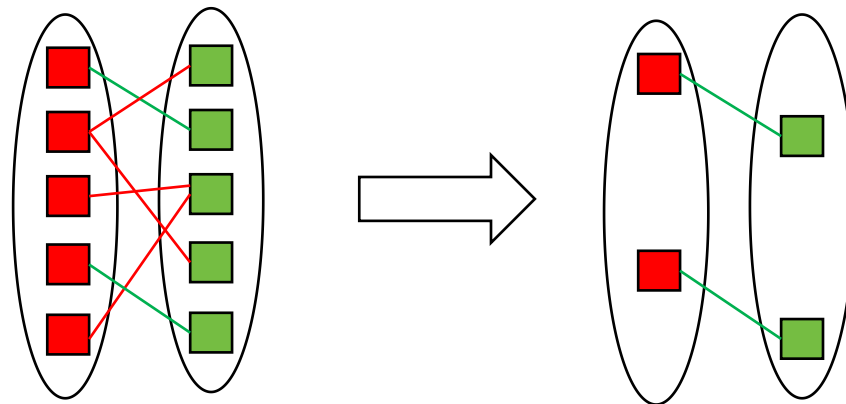
WIP: Github Api OverheadOptimization



1. Crawl the issues and prs
Optimize the QPI(query per issue)
:WIP

2. 1pr 1 issue matching:

bipartite graph



baseline: degree filtering

can try :Hopcroft-Karp

(code implemented,
experiment needed)