



# ConversationTokenBufferMemory

`ConversationTokenBufferMemory` keeps a buffer of recent interactions in memory, and uses token length rather than number of interactions to determine when to flush interactions.

Let's first walk through how to use the utilities

```
from langchain.memory import ConversationTokenBufferMemory
from langchain.llms import OpenAI

llm = OpenAI()
```

## API Reference:

- `ConversationTokenBufferMemory` from `langchain.memory`
- `OpenAI` from `langchain.llms`

```
memory = ConversationTokenBufferMemory(llm=llm, max_token_limit=10)
memory.save_context({"input": "hi"}, {"output": "whats up"})
memory.save_context({"input": "not much you"}, {"output": "not much"})
```

```
memory.load_memory_variables({})
```

```
{'history': 'Human: not much you\nAI: not much'}
```

We can also get the history as a list of messages (this is useful if you are using this with a chat model).

```
memory = ConversationTokenBufferMemory(
    llm=llm, max_token_limit=10, return_messages=True
)
```

```
memory.save_context({"input": "hi"}, {"output": "whats up"})
memory.save_context({"input": "not much you"}, {"output": "not much"})
```

## Using in a chain

Let's walk through an example, again setting `verbose=True` so we can see the prompt.

```
from langchain.chains import ConversationChain

conversation_with_summary = ConversationChain(
    llm=llm,
    # We set a very low max_token_limit for the purposes of testing.
    memory=ConversationTokenBufferMemory(llm=OpenAI(),
    max_token_limit=60),
    verbose=True,
)
conversation_with_summary.predict(input="Hi, what's up?")
```

### API Reference:

- [ConversationChain](#) from `langchain.chains`

> Entering new ConversationChain chain...

Prompt after formatting:

The following is a friendly conversation between a human and an AI. The AI is talkative and provides lots of specific details from its context. If the AI does not know the answer to a question, it truthfully says it does not know.

Current conversation:

Human: Hi, what's up?

AI:

> Finished chain.

" Hi there! I'm doing great, just enjoying the day. How about you?"

```
conversation_with_summary.predict(input="Just working on writing some
documentation!")
```

> Entering new ConversationChain chain...

Prompt after formatting:

The following is a friendly conversation between a human and an AI. The AI is talkative and provides lots of specific details from its context. If the AI does not know the answer to a question, it truthfully says it does not know.

Current conversation:

Human: Hi, what's up?

AI: Hi there! I'm doing great, just enjoying the day. How about you?

Human: Just working on writing some documentation!

AI:

> Finished chain.

' Sounds like a productive day! What kind of documentation are you writing?'

```
conversation_with_summary.predict(input="For LangChain! Have you heard of
it?")
```

> Entering new ConversationChain chain...

Prompt after formatting:

The following is a friendly conversation between a human and an AI.

The AI is talkative and provides lots of specific details from its context. If the AI does not know the answer to a question, it truthfully says it does not know.

Current conversation:

Human: Hi, what's up?

AI: Hi there! I'm doing great, just enjoying the day. How about you?

Human: Just working on writing some documentation!

AI: Sounds like a productive day! What kind of documentation are you writing?

Human: For LangChain! Have you heard of it?

AI:

> Finished chain.

" Yes, I have heard of LangChain! It is a decentralized language-learning platform that connects native speakers and learners in real time. Is that the documentation you're writing about?"

```
# We can see here that the buffer is updated
conversation_with_summary.predict(
    input="Haha nope, although a lot of people confuse it for that"
)
```

> Entering new ConversationChain chain...

Prompt after formatting:

The following is a friendly conversation between a human and an AI. The AI is talkative and provides lots of specific details from its context. If the AI does not know the answer to a question, it truthfully says it does not know.

Current conversation:

Human: For LangChain! Have you heard of it?

AI: Yes, I have heard of LangChain! It is a decentralized language-learning platform that connects native speakers and learners in real time. Is that the documentation you're writing about?

Human: Haha nope, although a lot of people confuse it for that  
AI:

> Finished chain.

" Oh, I see. Is there another language learning platform you're referring to?"