

Outline Storming GPT Ver2.0

■はじめに

Outline Storming GPT は、
Open AI が提供する Chat GPT API (<https://api.openai.com/v1/chat/completions>) を呼び出すヘルパーアプリです。

OpenAI の Web サイトで事前に Auth キー（API キー）を発行してください。

History の記録や再生、改変をサポートしていますので、
Try ChatGPT では難しかった入力条件の再調整や会話途中での条件振りといった事が簡単に行えます。

■動作環境

Windows 10 / Windows11

.NET 6

Display 1280x720 以上の解像度

要インターネット環境

■利用しているパッケージ

- Csv (2.0.93)

<https://www.nuget.org/packages/Csv>




- Microsoft.AspNetCore.Http.Extensions (2.2.0)

<https://www.nuget.org/packages/Microsoft.AspNetCore.Http.Extensions>

- Microsoft.Extensions.Http (7.0.0)

<https://www.nuget.org/packages/Microsoft.Extensions.Http/8.0.0-preview.2.23128.3>

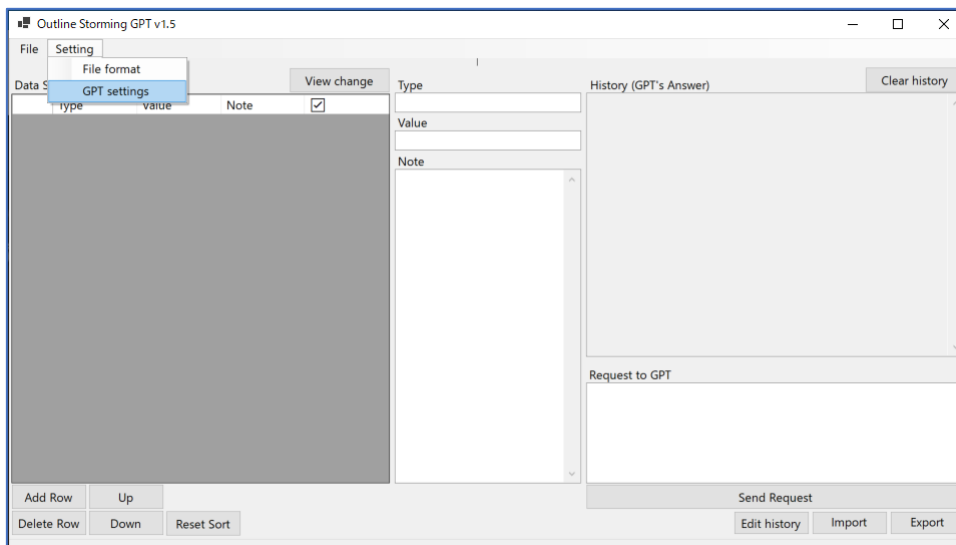
■使い方

名前	更新日時	種類	サイズ
 OutlineStorming.dll.config	2023/04/01 18:37	XML Configuration Fi...	2 KB
 OutlineStorming.exe	2023/04/01 18:59	アプリケーション	1,322 KB
 OutlineStorming.pdb	2023/04/01 18:59	Program Debug Data...	26 KB

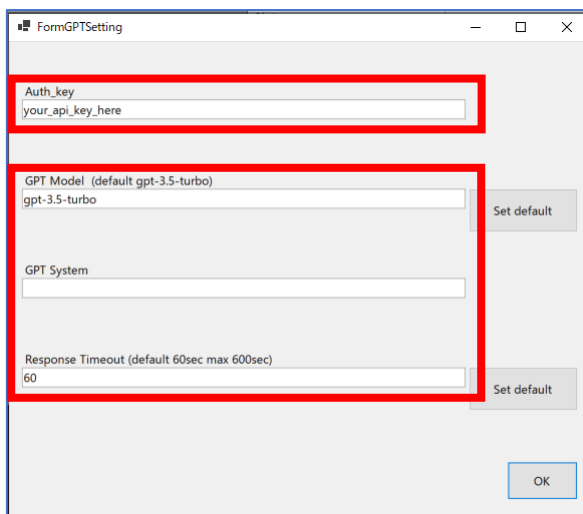
OutlineStorming.exe を起動します。

■初期設定

起動後の画面から、[Setting]→[GPT settings]を選択します。



Auth_key (Open AI から取得した API key)を入力します。



Model は、デフォルトでは gpt-3.5-turbo が設定されています。
GPT4 や他のモデルが利用可能な方は、適宜変更してください。

System は、GPT がどのように振る舞うかの設定です。
(常に英語で答える、～～のように振る舞うなど。)

Response Timeout は API の応答待ち時間です。
標準では、60 秒となっていますが、少し長めに設定した方が良いかもしれません。

■ データセットの入力

Type	Value	Note	View change
システム	レベル	プレイヤーのレベ	<input checked="" type="checkbox"/>
ゲームの目的	ゲームの目的	プレイヤーの目	<input checked="" type="checkbox"/>
武器	武器	3つの数字の	<input checked="" type="checkbox"/>
武器	たけざお	1052 初期所	<input checked="" type="checkbox"/>
武器	こんぼう	60304 [店] ラ	<input checked="" type="checkbox"/>
武器	どうのつるぎ	1809010 [店]	<input checked="" type="checkbox"/>
武器	てつのおの	56028015 [<input checked="" type="checkbox"/>
武器	はがねのつるぎ	150075020 [<input checked="" type="checkbox"/>
武器	ほのおのつるぎ	9800490028	<input checked="" type="checkbox"/>
武器	トトのつるぎ	--40 [宝] 竜	<input checked="" type="checkbox"/>
防具	防具	三つの数字の	<input checked="" type="checkbox"/>
防具	ぬののふく	20102 [店] ラ	<input checked="" type="checkbox"/>
防具	かわのふく	70354 [店] ラ	<input checked="" type="checkbox"/>
防具	くさりかたばら	30015010 [<input checked="" type="checkbox"/>
防具	てつのよろい	100050016 [<input checked="" type="checkbox"/>
防具	はがねのよろい	3000150024 [<input checked="" type="checkbox"/>
防具	まほうのよろい	7700385024	<input checked="" type="checkbox"/>
防具	トトのよろい	--28呪文・炎	<input checked="" type="checkbox"/>

まずは、前提となるデータセットを入力します。

Add Row ボタンで行が追加できますので、データセットを入力します。

(表に直接入力はいけません。中央のパネルで行います。)

Type, Value, Note は明確なルールがある訳ではありません。

自由に使い分けてください。

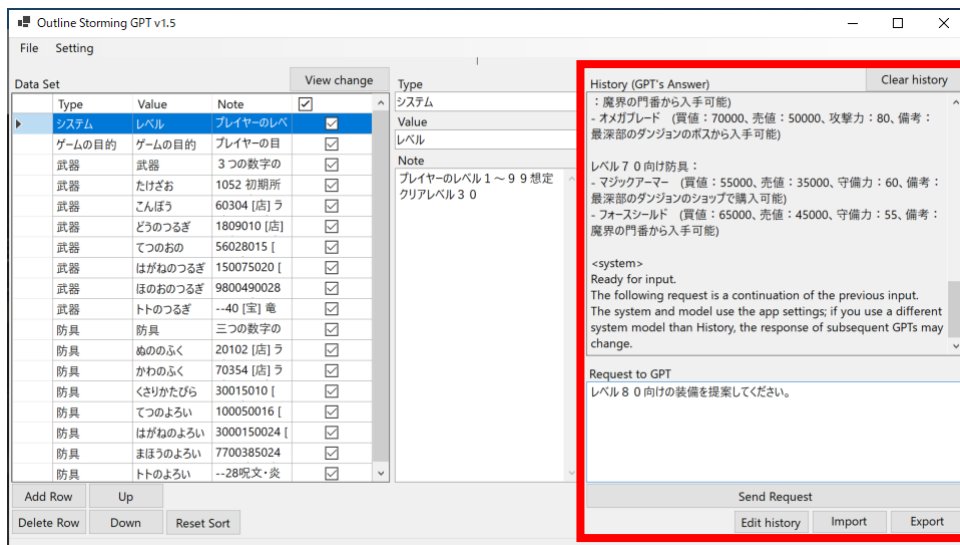
データセットは、CSV 形式のデータを読み込むことも可能です。

読込：[File]→[Open CSV]

保存：[File]→[Save CSV]

文字コードや区切り文字はポップアップに従い組み合わせを設定してください。

■ Chat GPT へのデータ送信



Request to GPT にリクエストを入力します。

[Send Request]ボタンをクリックすると、Chat GPT に対してメッセージが送信されます。
送信したメッセージと、Chat GPT からのレスポンスは、History 欄に追記されていきます。

Chat GPT の仕様上、送信したメッセージは History として毎回送信されます。

gpt-4 では、かなり大きな token が利用可能ですが、

gpt-3.5-turbo の場合、4000token 程度が送信上限となっています。

(各 Message 要素も 1000 文字以内となっています。)

GPT との通信を繰り返し、History が長くなった場合は、

[Clear history]で会話をやり直すか、

[Edit history]で不要な応答を削除してください。

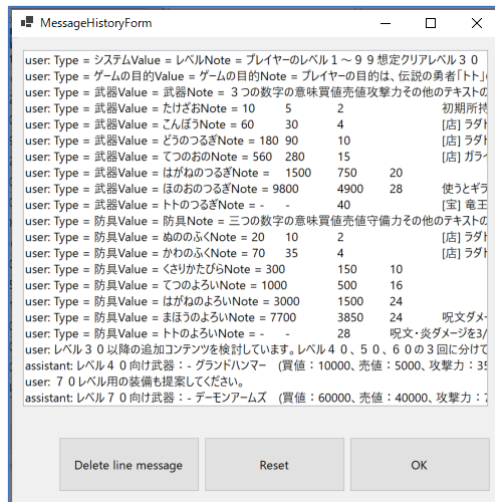
History は、画面右下の[Import]から読み込み、[Export]で保存する事が出来ます。

CSV (データセット) もメッセージとして保持されていますので、[Import]を行うと、データセットも併せて更新されます。

※次ページで History の更新方法を記載していますが(整合性が崩れてしまう為)メッセージの中身やデータセットの中身を変更する事は出来ません。

■不要なメッセージの削除（Edit history）

[Edit history] ボタンをクリックすると、次のようなメッセージリストが表示されます。



MessageHistoryForm

user: Type = システムValue = レベルNote = プレイヤーのレベル 1 ~ 9 想定クリアレベル 3 0
user: Type = ゲームの目的Value = ゲームの目的Note = プレイヤーの目的は、伝説の勇者「トト」
user: Type = 武器Value = 武器Note = 3つの数字の意味買値売値攻撃力その他のテキストの
user: Type = 武器Value = たげおNote = 10 5 2 初期所持
user: Type = 武器Value = こんぼうNote = 60 30 4 [店] ラダ
user: Type = 武器Value = どうのつるぎNote = 180 90 10 [店] ラダ
user: Type = 武器Value = てつのおのNote = 560 280 15 [店] ガラ
user: Type = 武器Value = はがねのつるぎNote = 1500 750 20
user: Type = 武器Value = ほのおのつるぎNote = 9800 4900 28 使うとギラ
user: Type = 武器Value = トトのつるぎNote = - - 40 [宝] 竜王
user: Type = 防具Value = 防具Note = 3つの数字の意味買値売値守備力その他のテキストの
user: Type = 防具Value = めのふくNote = 20 10 2 [店] ラダ
user: Type = 防具Value = かわのふくNote = 70 35 4 [店] ラダ
user: Type = 防具Value = くらかたびらNote = 300 150 10
user: Type = 防具Value = てつよろいNote = 1000 500 16
user: Type = 防具Value = はがねよろいNote = 3000 1500 24
user: Type = 防具Value = まほうよろいNote = 7700 3850 24 呪文・炎ダメージを3/
user: Type = 防具Value = トトよろいNote = - - 28 呪文・炎ダメージを3/
assistant: レベル 3 0 以降の追加コンテンツを検討しています。レベル 4 0、5 0、6 0 の 3 回に分けて
assistant: レベル 4 0 向け武器： - グランドハンマー （買値：10000、売値：5000、攻撃力：35
user: 7 0 レベル用の装備も提案してください。
assistant: レベル 7 0 向け武器： - デモンアームズ （買値：60000、売値：40000、攻撃力：70000、

Delete line message Reset OK

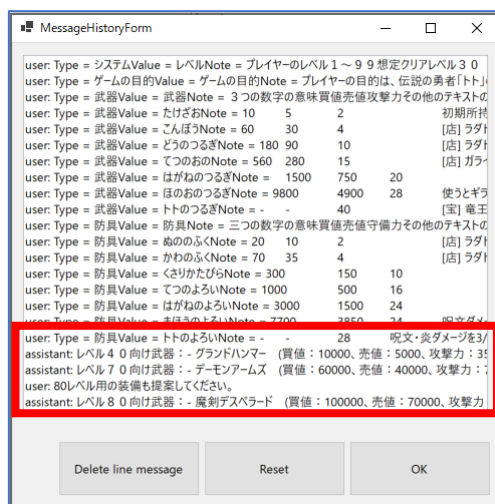
メッセージを選択して[Delete line message]をクリックする事でメッセージが削除されます。

[assistant]は GPT の回答となっており、基本的には直前の[user]に対するレスポンスです。

基本的にはペアで削除する事をお勧めしますが、必ずしもペアである必要はありません。

次に行う Request が、上記の例のような定型化した回答を期待している場合は、[assistant]のみ残しておけば同じような同じ形式で回答を引き出す事が出来ます。

下の画面は、[user]メッセージのみ削除してリクエストを投げた時の History です。



MessageHistoryForm

user: Type = システムValue = レベルNote = プレイヤーのレベル 1 ~ 9 想定クリアレベル 3 0
user: Type = ゲームの目的Value = ゲームの目的Note = プレイヤーの目的は、伝説の勇者「トト」
user: Type = 武器Value = 武器Note = 3つの数字の意味買値売値攻撃力その他のテキストの
user: Type = 武器Value = たげおNote = 10 5 2 初期所持
user: Type = 武器Value = こんぼうNote = 60 30 4 [店] ラダ
user: Type = 武器Value = どうのつるぎNote = 180 90 10 [店] ラダ
user: Type = 武器Value = てつのおのNote = 560 280 15 [店] ガラ
user: Type = 武器Value = はがねのつるぎNote = 1500 750 20
user: Type = 武器Value = ほのおのつるぎNote = 9800 4900 28 使うとギラ
user: Type = 武器Value = トトのつるぎNote = - - 40 [宝] 竜王
user: Type = 防具Value = 防具Note = 3つの数字の意味買値売値守備力その他のテキストの
user: Type = 防具Value = めのふくNote = 20 10 2 [店] ラダ
user: Type = 防具Value = かわのふくNote = 70 35 4 [店] ラダ
user: Type = 防具Value = くらかたびらNote = 300 150 10
user: Type = 防具Value = てつよろいNote = 1000 500 16
user: Type = 防具Value = はがねよろいNote = 3000 1500 24
user: Type = 防具Value = まほうよろいNote = 7700 3850 24 呪文・炎ダメージを3/
assistant: レベル 3 0 以降の追加コンテンツを検討しています。レベル 4 0、5 0、6 0 の 3 回に分けて
assistant: レベル 4 0 向け武器： - グランドハンマー （買値：10000、売値：5000、攻撃力：35
assistant: レベル 7 0 向け武器： - デモンアームズ （買値：60000、売値：40000、攻撃力：70000、
user: 80レベル用の装備も提案してください。
assistant: レベル 8 0 向け武器： - 魔剣デスベアード （買値：100000、売値：70000、攻撃力：70000、

Delete line message Reset OK

■DeepL API の呼び出し

FormDeepLSetting

DeepL API ☒ Free ☐ Pro

Auth_key
your_api_key_here

User Language: Japanese ☐ Enable Auto

GPT's Language: English (British) ☐ Enable Auto

Response Timeout (default 60sec max 600sec)
60

Translation target (Dataset translation is not yet supported.)
☐ Dataset ☒ Request ☐ Response

DeepL API による翻訳を行う為のオプションです。

チェックを入れると、DeepL の呼び出しを行います。

[Request]… ユーザーが入力したメッセージを GPT's Language の言語に翻訳します。

[Response]… GPT からの応答を、User Language の言語に翻訳します。

[DeepL API Free/Pro]…いずれもサポートしています。が、検証は Pro のみです。

[Auth_key]…DeepL API サイトで発行した API-key を入力してください。

[User Language]…ユーザーが使用する言語を指定してください。

[GPT's Language]…実際に GPT に送信する際の言語を指定してください。

[Enable Auto]…翻訳元言語を自動で検出する場合はチェックを入れてください。

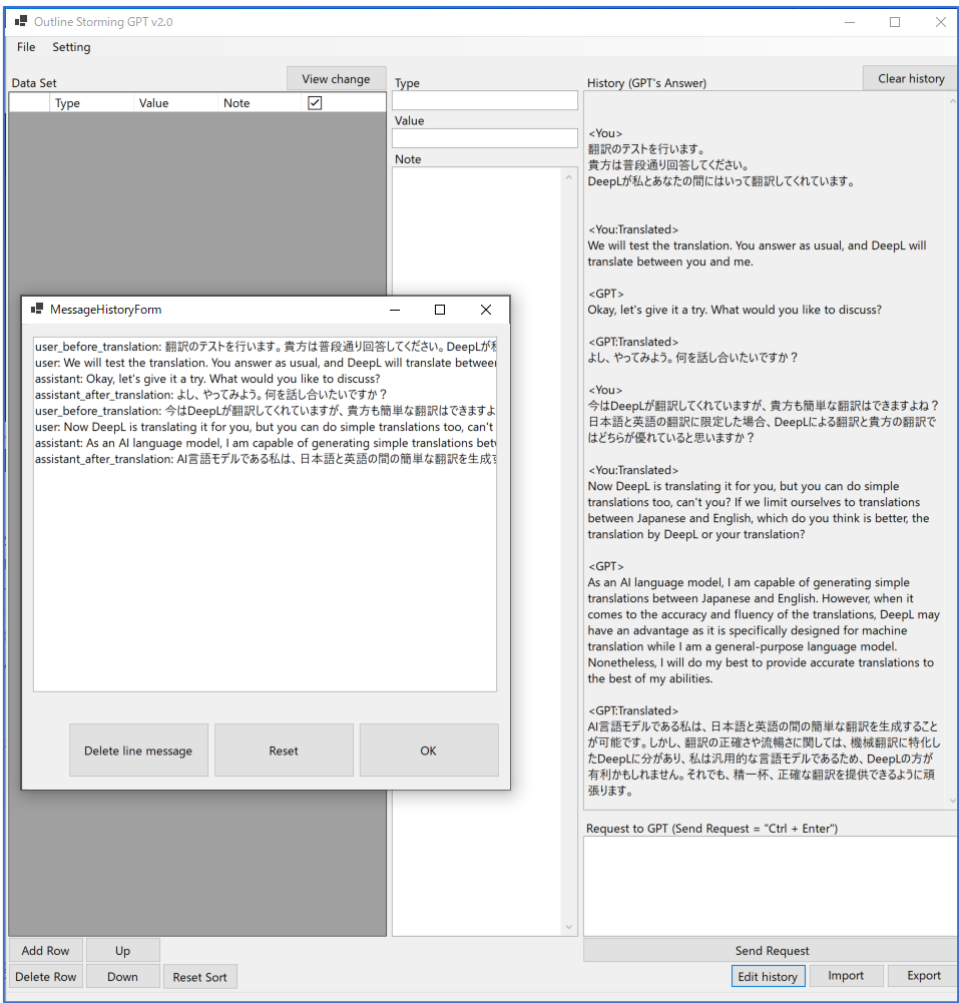
通常は OFF で問題ありません。

[Set default]…DeepL API の Request Timeout を初期値（60 秒）に設定します。

実際の翻訳例は次のページの通りです。

翻訳した結果も併せて Message History に記録されます。
具体的には次のような関係となります。

- [user_before_translation]…ユーザーが入力したメッセージ
- [user]…DeepL が翻訳し、実際に GPT に送信されるメッセージ
- [assistant]…GPT からの応答メッセージ
- [assistant_after_translation]…GPT からの応答メッセージを DeepL が翻訳したもの



解説は以上となります。