第2章 Pythonプログラムを自動修正するライブラリ
Wolverineを使う

ご購入はこちら

自然言語モデルとシステムの対話によるプログラム自動生成

新谷 俊了

● 人間の判断を不要とする自動生成システムが欲しい

OpenAIのChatGPTの登場でプログラム・アシスト機能や文章の要約など，さまざまなタスクを自然言語モデルに実行させる試みがなされています．現在，その多くのシステムは自然言語モデルと人間との対話をベースとしています．基本的には，人間の要望に対して自然言語モデルが生成してきた回答を人間が判断し，要望に沿っていれば承認し採用するという流れです．

自然言語モデルの回答は完全ではないため，回答を人間が確認することは安全確認の意味でとても重要です．しかし，人間の判断を不要とするようにシステムを構築できれば，負荷はよりいっそう軽減されます．

1つの方法としては，自然言語モデル自体に自問自答させる方法が考えられます．例えば2つの自然言語モデルを用意し，一方は人間の代理として判断を行い，もう一方は通常のアシスタントとして振る舞わせれば，人間の判断は不要になります．このような試みは例えばCAMEL(1)などのライブラリで実験的に実装されています．

● 生成物に対する判断基準は必要

一方で，自然言語モデルだけで閉じてしまうと，人間の望むゴールを自然言語モデルが含んでいない場合，そこに到達することは難しくなります．例えば，モデルが学習していない未来の情報などが判断に必要になる場合には，望む回答を得ることは難しくなると考えられます．そのような自然言語モデル単体で回答が出せない場合には，外部の情報を収集してくることが必要となります．

2023年4月ごろに提案されたAuto-GPT(2)というフレームワークでは，自然言語モデルに検索などの外部プログラムを使うことを提案させ，それを実際に実行し，人間はゴールのみを与えることで自動的にタスクを完了させる手法が考案されて話題になりました．この中で検索結果などを反映するケースは，自然言語モデルと機械的なシステムを対話させているとも考えられます．機械的なシステムの部分が自然言語モデルの回答を評価することができれば，自然言語モデルは与えられた評価に対して新たな回答を生成する役割を担うだけです．人間が都度，判断をする必要がなくなります．

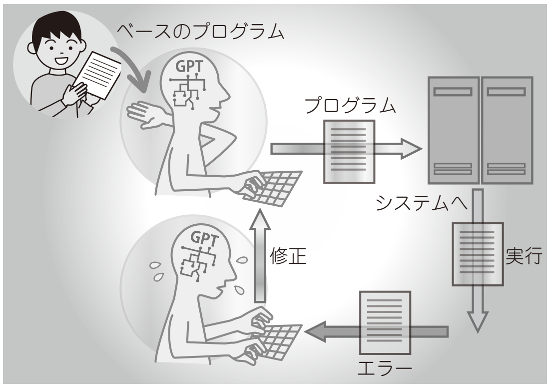

図1　ChatGPTが自分でエラーを修正して動くプログラムを作ってくれる

● トライすること

今回は**図1**のように，自然言語モデルにプログラムを生成させ，実際にそのプログラムを実行し，エラーを客観的な評価として自然言語に与える方法を試してみます．2023年3月ごろにWolverine(3)というPythonのライブラリが公開されており，OpenAIでAPIを使用できるアカウントがあれば，簡単に上記の方法によるプログラム生成を試すことができます．

本稿では，次の2点にトライします．

1, Pythonライブラリ Wolverineの紹介とプログラム修正の実例（Wolverineはcommit hash 81143a3のものを利用する）
2, Wolverineの仕組みを踏襲して自然言語モデルに数値最適化問題を解かせる

● 自動生成，自動実行ならではの注意点

自然言語モデルとシステムを対話させるという仕組み上，試行回数に制限を導入しないと無限にリクエストが送られてしまう可能性があります．さらに，プロ

◆参考文献◆

（1）CAMEL：Communicative Agents for “Mind” Exploration of Large Scale Language Model Society．arXiv．`https://arxiv.org/abs/2303.17760`