



# 政策差分グラフ

インタビューAIグランプリ 中間発表

2025-12-07 西尾泰和

# インタビューAIグランプリについて(by安野たかひろ)

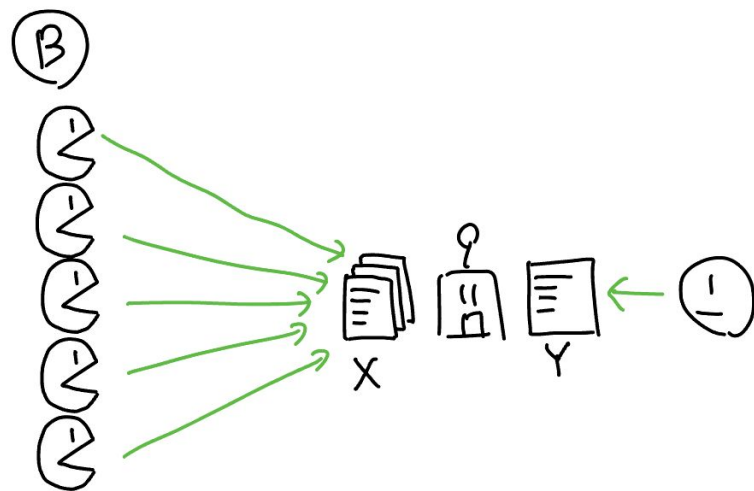
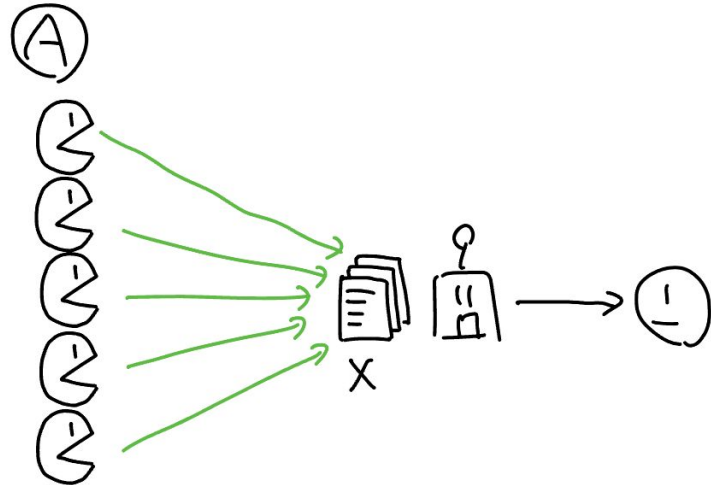
ブロードリスニングを現実の政治に実装するときの大きい課題の1つが、「大量のデータからいかに政策立案者にとって学びのあるデータを抽出するか」

- アンケート調査のような一往復の浅いデータから、より対話的に深いリアクションまで調査するのにAIインタビューという形式が有望そう
- ただし、その非構造化データからは『生ログを読まなければ』なかなか有用な知見が得づらいという課題がある
  - 一般の方向けに必要なレベル感「わかりやすいか」は別として一定できている
- このミッシング・ピースを埋めなければデジタル民主主義の実装は進まないが、逆にここが周りはじめたら、後はゴリゴリ実装していけば良い
- いろんな人のいろんな試行錯誤が行われることが効く領域なので、i-1グランプリという形でやってみることにした
  - 上手くいったらもう少し継続的に実施を検討したい



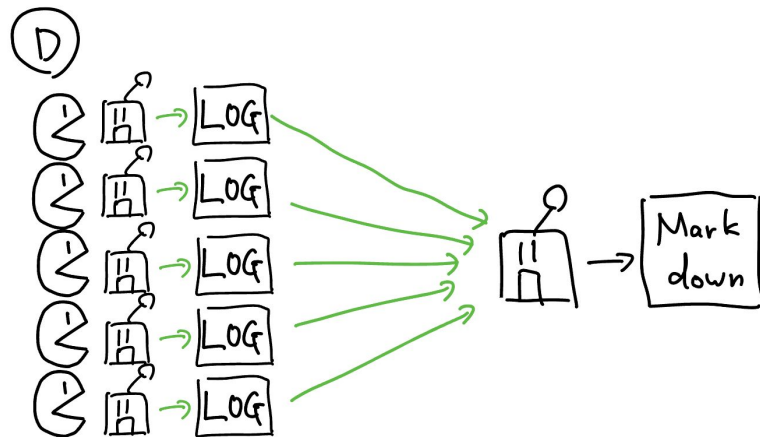
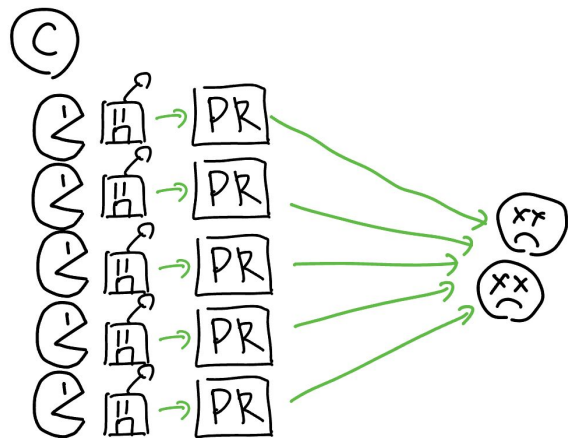
## このグランプリの面白い点

- 通常「**ブロードリスニング**」というと人々からのデータXをAIが要約して意思決定者にレポートを出す(A)
- しかし今回の問題設定では「チームみらいが今考えていること」というデータYが提供されている(B)
- X/Yの**2種類のデータ**が与えられていて**Yをよりよく更新するために Xを使う**
- 振り返ってみるとこれは「**しゃべれるマニフェスト**」と似た構図



# 「しゃべれるマニフェスト」と 何が違うか

- 「しゃべれるマニフェスト」では  
人々と会話したAIがPR(プルリクエスト)を生成  
したが、これはたくさん集まったときに機械的に  
処理することが困難だった(C)
- 今回の問題設定では会話ログを収集し  
それを別のAIシステムが読んで分析し  
Markdownのレポートを作る(D)
- 今回このMarkdownを試行錯誤する中で  
“Yをよりよく更新するために Xを使う”  
から概念上はPRだと気づいた
- つまり「N人の情報を集約してからPRを作るシ  
ステム」と解釈できる



## 実験4: 2つの異なるデータを照らし合わせるメカニズム

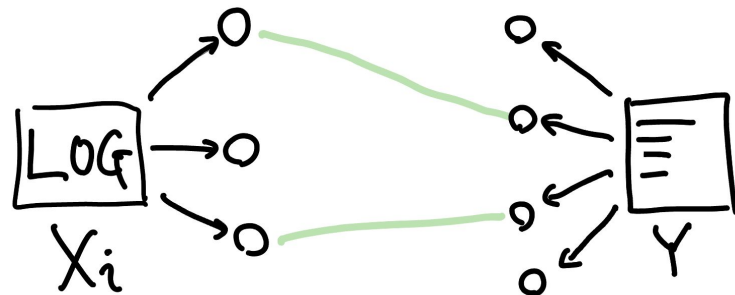
- チャットログから「提案」を抽出
- この提案とbaseline文書の間係を以下から1つ選択:
  - "one\_step\_extension": baselineを一步発展させる
  - "missing\_concern": baselineにない懸念を指摘(見落とし発見)
  - "new\_opportunity": baselineにない新しい活用案・解決策
  - "already\_covered": baselineですでにカバー済み
  - "not\_relevant": baselineのテーマと関連性が薄い
- baselineのどの箇所に関連するか特定

実験004: relation分布 (179件)

relation	件数	割合
one_step_extension	80	44.7%
missing_concern	42	23.5%
already_covered	36	20.1%
new_opportunity	17	9.5%
not_relevant	4	2.2%

## 実験6: グラフ形式で出力

- ここまでログ断片とbaseline行との関係をLLMで抽出して直接Markdownを作っていた
- が、この手前のデータをJSON形式のグラフ構造データとして出力した方がいいのではと考えた



- きっかけ: 人間が読むレポートを直接「LLMをAPI呼び出しするスクリプト」が出力するのではなく、ChatGPT GPT5.1のコンテキスト幅に収まるように情報をダイジェストして出力し、人間用の最終レポートは賢いLLMが出すのが良いのではないか
  - 今回の要件はMarkdownの提出であり、全自動のプログラムを組む必要がないため
- >anno: Claude Codeとかの挙動をみていたりすると、仮説をもとに Agenticに深堀りさせながら追記していくやり方もありそう
  - これにも使えそう(毎回LLMを叩くよりJSONを読む方がAgentは楽)
- 研究的には「中間データをグラフ構造で持った方がいいのでは」という発想は GraphRAGという名前で試行錯誤されているテーマ。今回の実験ではあんまりグラフ構造のメリットを活用していない。今後 GraphRAGを参考にした発展もありそう(出現した専門用語のグラフの活用など)

## 実験8: 関係性の編集操作での定義

- 関係性を「どうであるか」ではなく「どうするべきか」(編集操作)で定義
- challenge(反証) - 3.2%(5件)
  - 編集操作: baseline行の真偽 or 優先度を見直したくなる、削除候補
- refine(限定) - 4.4%(7件)
  - 意味: baseline行に条件・範囲・前提を追加する
  - 編集操作: 「条件Xのとき」にと書き加えたいくなる
- orthogonal(別軸) - 24.1%(38件)
- support(裏付け) - 68.4%(108件)
- 編集緊急度の高い「反証」「限定」  
の数が意外と少ない
  - AIに賛成したり別の話をしたりする人は多いが、反論したり条件を詳細化する人が少ない
  - しかしこれこそ重要な情報では？

alignment軸 (NEW in 実験008-009)

alignment	件数	割合
support	108	68.4%
orthogonal	38	24.1%
refine	7	4.4%
challenge	5	3.2%

# 実験10: 中間提出レポート作成

- 生成したJSONグラフからClaude CodeでGPT5.1 Pro用のプロンプトを作る
- GPT5.1 Proがレポートを作る
- 生成されたレポートがグランプリの「出典ヘリンクせよ」の要件を満たしていなかった  
たのでClaude Codeに修正させてる
  - これは結構大変だった、やりやすいように元データを準備しておくべき。DB上のsession+message IDは長いので、短いIDを振り直してGPTに渡すのが良いと思う

プロンプト 生成成功:  
- 文字数: 14,543文字  
- 推定トークン数: 36.4K tokens (GPT-5.1Proのコンテキストに余裕で収まる)  
- コスト: \$0 (ローカル処理のみ)  
- 処理時間: <1分

プロンプトに含まれる提案

alignment	全件数	プロンプトに含まれる数	含有率
challenge	5件	5件 (全件)	100%
refine	7件	7件 (全件)	100%
orthogonal	38件	15件 (サンプル)	39%
support	108件	10件 (サンプル)	9%
合計	158件	37件	23%



# コスト

>生成するのにかかるAPIコスト・作成や修正にかかる工数

- 細かい実験を繰り返しているので合計に自信がない...
- GPT-4o-mini
  - 実験002(専門用語抽出～セッションフィルタ) \$0.28
    - Phase 1A | 専門用語抽出v2(383セッション)| \$0.11 | 1.2分
    - Phase 1B-C | LLM分類 + 人間レビュー | \$0.05 | 10分(?)
    - Phase 2 | 提案snippet抽出(57セッション → 218提案) | \$0.03 | 1.1分
    - Phase 3 | baseline差分分析(218提案) | \$0.09 | 1.5分
    - 合計 \$0.28 | 14.8分
  - 実験008 alignment軸追加(167件にLLMで付与) \$0.0326
- 定額サブスクリプション:
  - Claude Code (Max 5x) \$100/mo + ChatGPT Pro \$200/mo

余談

# 気になったことをClaude Codeに聞くことができる

- AIが人間からの質問に「IMOってのがあって～」と回答しているログを見かけて、IMOってキーワードがAIの発言由来じゃないかと不安になった
- CCに気軽に聞いたら抽出済みデータからIMOを含む断片が4件あるねと提示

```
> IMOなどへの言及が人間ではなくAIの発言から抽出された可能性を疑っている
```

- 重要な懸念ですね。IMOがユーザーの発言ではなくAIの質問文から抽出されている可能性を確認します。

```
● Bash(cd /Users/nishio/mirai_interview_report_grandprix/experiments/2025-12-03-009_graph_v3_generation/output && python3 << 'EOF'
  import json...)
  | === IMO Terms ===
  | Term: IMO
  | Category: legal
  | ... +24 lines (ctrl+o to expand)
```

- IMOを含む4件のsnippetが見つかりました。これらが本当にユーザー (role: user) の発言か確認します：

```
● Bash(cd /Users/nishio/mirai_interview_report_grandprix && python3 << 'EOF'
  import json...)
  | === Checking message roles for IMO snippets ===

  Session: 9f0e53e3...
  | ... +18 lines (ctrl+o to expand)
```

- 良い結果です！4件すべてが **role: user**(ユーザーの発言) でした。IMOはAIの質問ではなく、実際のユーザーの発言から抽出されています。

- Claude Codeとこういう対話をしながら確認できるのはとてもよい
- (Webシステムを作るよりCCでこういうことがするためのファイル構成を作って政策担当者が直接 CCを使った方がいい可能性すらある)

## 結論

✅ IMOは正しくユーザーの発言から抽出されています

- 4件すべてのIMO snippetが **role: user** のメッセージから抽出
- 専門用語抽出スクリプト ([extract\\_terminology\\_v2.py](#)) は明示的に **role == 'user'** でフィルタ
- AIの質問文は除外されている

## ユーザーの発言例：

- 「従来通り専門家がIMOとの関係の中で決めていく」（専門家が主語）
- 「IMOやICCなどで標準化に積極的に関わり...」（国際機関への関与を提案）
- 「IMOが標準化を進めているのであれば...」（IMOの動向を認識）

これらは明らかに専門知識を持つ参加者の発言であり、抽出は適切です。

## 実験~2: 専門用語フィルタ

from Slack 11/24~25

基本的なアイデアとしては「専門家の意見には専門用語が使われている可能性が高く、まず安いLLMで専門用語の抽出をして、それからその専門用語をたくさん含んでいる順にソートしたら専門的な議論が見つかるはずだ」というものです

### レポートv2生成の仕組み

- 専門用語抽出
- 専門用語を使って専門家っぽいセッションを抽出
- 与えられた現状の思考のデータ(baseline.md)との照らし合わせ→見落としの指摘
- 生成されたレポートが過剰に詳細だったのでそのままGPT5.1 Proに投げて見た目を良くした

# 「専門家フィルタ」について、その後の考察(12/5)

初期の実験で「専門用語の出現頻度をみてフィルターする」という話をしたんだけど、その後いろいろみて考えが少し変わってきた

- A:「専門家」というと学者などをイメージしてしまいがちだが、例えば医療に関する制度を変えようと考えている場合にそういう「いわゆる専門家」の声は既に従来の経路で収集できていて、必要なのは患者やその家族、その患者と直接やりとりしてる医者などの声。「専門家」というより「**当事者**」の方がしっくりくるかも
- B:「いわゆる専門用語」ではない、例えば「韓国」とか「FedEx」とかの単語が当事者性の強いチャットに現れている。こういうのを扱うなら「専門用語」と呼ぶのは適切ではなくて、「**具体性の高さ**」のような別の軸になりそう&単語単位の出現で判断するのは多分不適切でLLMに対する指示を工夫する感じかな
- C:「いわゆる専門用語」の中に特殊な振る舞いをする単語がある。代表例が「ブロックチェーン」で、専門性のある意見も少しあるが、このキーワードを使ってる浅い意見が大量にあってS/N比がとても悪い。この種の**バズワード**化している専門用語はピックアップ条件に入れない方がいいかも
  - (個人的にもAIと言わずに「機械学習」「深層学習」「LLM」などより詳細な言葉を使いがちだし、SNSで「コロナ」をmuteして「COVID19」で情報収集してたし)

# 実験3: 繰り返しアップデート

from Slack 11/26

レポートを一旦出してから人間やGPT5がフィードバックをして修正する手法

- A: スクリプトがレポートを生成する
- B: GPT5.1 Thinkingがダメ出しをする
- C: Claude Codeがそれをみて最終仕上げをする

12/5に安野さんが紹介した論文 "Deep Researcher with Test-Time Diffusion" は「雑にレポートを生成してからその改善サイクルを回す」という手法で、この実験3で手作業で素朴にやったことを機械的にやったと解釈できる。

Deep Researchでは(B)のダメ出しでレポートの弱点を特定した後、Webを検索するタスクが走って情報収集をする。ここをその通り真似しても有益そう。

また「これをもっと調べるべき」という情報が得られてるわけだから、チームみらいの文脈では「AIインタビューのプロンプトに差し込む」とか「過去の参加者から詳しくそうな人に『この点についてもう少しお聞かせ願えないですか』と連絡して追加インタビューをする」という手もある

# 処理規模のサマリー

- 元データ: 383セッション
- 専門家フィルタ後: 57セッション(85%削減)
- 抽出された提案: 218件
- 有効な提案: 167件
- グラフノード: 722 (baseline: 31, proposals: 167, snippets: 445, terms: 79)
- グラフエッジ: 504

# レポート品質と参加感のトレードオフ

- 「専門家フィルタ」的な手法の是非に関する考察
- レポート品質を高める上ではデータを高品質なものだけに絞り込むべき
- 「インタビューに参加した人」は自分の出力を捨てられると残念な気持ち
- これは2種類のユーザの満足がトレードオフの関係になっている
  - 前者(チームみらい)はお金と時間を払っている、後者は時間を払っている
  - 長期的にみて前者が払う金額を上回る効果が得られなければこのサービス自体を閉じることになるからまず前者を十分な品質にするべきだと思う
  - 長期的にこれが有用なシステムになって継続的に繰り返し使われるようになっていけば、運用も改善されて参加者の満足度も上がると思う
- レポートの品質を下げずに、後者の人のがっかりをなるべく減らすべき
  - 貿易業務当事者の声が聞きたいということを事前に明記する方が良かったと思う  
(実験した段階ではそこまで明確に言語化されていなかったなので仕方ないが今後はなるべく )
  - 参加者を限定しないで広く聞く時には「確実に読まれる /活用される」という期待を持たれないように期待値を下げると良さそう
  - 集めてしまった声を無理やりレポートに入れようとするべきではない、レポート品質が低下して本末転倒だから



## future work

- 今回実験2で抽出した断片を使い続けているが、実験8での発見をベースに「後で編集操作で分類すること」を意識した抽出にした方が良さそう
- 「専門家フィルタ」は「当事者フィルタ」に変えた方が良さそう
- 今回の実験~10の結果やコードを参考に、フルスクラッチで新しいデータに適用できるパイプラインを作る、その際に前段の抽出部分ごと変更する