SIGNATE × TECH OCEAN Student Cup 2025 3rd Place Solution

Team SIM (shun_aichi, wally0593, ichito, コーシー, undertaker)

目次

01 チーム紹介

02 試したこと

03 最終解法

04 まとめ

01 チーム紹介

shun_aichi（リーダー）

所属：豊橋技術科学大学

X：hobby_NLP

github：MaekobaShun

wally0593

所属：名古屋大学

X：melo_atc

github：kabetani-yusei

ichito

所属：長岡技術科学大学

X：ichitoink

github：ichito-watanabe

コーシー

所属：慶應義塾大学

X：Cauchy76

undertaker

所属：明治大学

X：ai_hazi

github：undertaker_del

02 試したこと

× Embeddingのみでの推論

・PrivateScore 0.25
・物語の複雑な融合は
線形演算では表現できず
精度が低迷

× 全候補の一括LLM推論

・PrivateScore ~0.54
・情報過多により
モデルがコンテキストを見失い
スコアが頭打ち

× 長文分割・RLM

AIが支持を無視して
回答を捏造するなど
制御不能となり断念

× 英語翻訳

・PrivateScore 0.49
・Publicスコアが悪化したため
採用を見送り

× perplexityで背景知識拡張

・PrivateScore 0.58
・Geminiでは性能向上したが、
Opus4.6では低下
判断が難しく、最終解法への
組み込みは見送り

✓ 候補絞り込み＆推論

・PrivateScore 0.59
・候補の絞り込みで対象を減らし、
LLMによる1件ずつの推論で
深く検討するアプローチを採用

03 最終解法① ベースパイプライン

候補の絞り込み

Embeddingにより
対象候補を
10〜20件に絞り込む

LLMによる1件ずつの推論

Gemini 3 Flash等を使用し、
厳選された候補で 1 件ずつ予測

アンサンブル

複数モデルの推論結果を統合

03 最終解法② 候補の絞り込み

埋め込み表現の作成

gemini-embedding-001 を用いて
base_storiesとfiction_storiesの
すべての作品を埋め込みベクトルに変換

使用したパラメータ
・出力次元: 3072
・タスクタイプ: SEMANTIC_SIMILARITY

ペア順序並び替え

・base_stories の全作品ペアを作成
・それぞれの埋め込みベクトルの和を取り
ペアベクトルを生成
・各 fiction_story と
ペアベクトルとのコサイン類似度を計算する

候補絞り込み

・ペアの順位に基づいて重み付けを行う
・各作品のスコアを集計しランキングを作成
上位10〜20作品を最終候補として抽出

03 最終解法③ 1件ずつの推論で使用したプロンプト

手順を明示する

回答の再現性を上げ，問題を分解して考える構造を作った

根拠を引用

なせその作品を選んだのかを説明させることによって，モデルの判断を人間が検証でき，プロンプトの良し悪しを判断しやすい

自己検証

推論の妥当性を自己検証させるプロセスを導入することによって，推論の整合性をチェックさせる

あなたは「混合作品（2作品合成）の元ネタ同定」専門の分析器です。
次のルールを厳守してください。
【絶対ルール】
- 各Fictionは Candidates から「ちょうど2作品」だけで構成される。
- Candidates 以外の作品を選ぶのは禁止。
- 出力は必ず2つのIDのみ（1つ/3つ/同一ID2回は禁止）。
- 直感で決めない。本文の具体描写とCandidates の具体描写で照合して決める。

【やり方（必須）】
Step 0) 先に“混合の証拠”を探す
- Fiction本文から「明らかに別系統の要素」を2系統見つけ、A系/B系としてメモする。
  例）舞台/ジャンル/目的/能力/組織/時代/テック/宗教/儀式/口調など。
Step 1) 要素抽出（最大6）
- Fictionから特徴的要素を最大6個、短く列挙する。
- 各要素の末尾に[A] または [B] を必ず付ける（どちら由来か決め打ちする）。
Step 2) 候補の当たり付け（比較）
- Candidates を見て、[A] を最も説明できる候補を1つ選ぶ（ID_A）。
- Candidates を見て、[B] を最も説明できる候補を1つ選ぶ（ID_B）。
- 「似てるから」ではなく、要素と一致する箇所がある候補を優先。
Step 3) 根拠（引用は厳格に）
- Fiction本文から根拠引用を各1〜2個だけ抜き出す（短いフレーズ）。
- その引用がCandidates のどの要素（タイトルあらすじの内容）に対応するかを一言で書く。
  ※Candidatesの文章を長引用するのは禁止（要点だけ言語化）。
Step 4) 最終検証（入れ替え条件）
- [A]要素の大半がID_Aで説明でき、[B]要素の大半がID_Bで説明できるか確認。
- 片方の候補で両方の要素を無理やり説明している場合は失格→ 入れ替えて再選定。

【出力形式（厳守）】
Fiction {ID}: {ID_A}, {ID_B}
根拠A:
-「Fiction引用」→ Candidates側の対応点（短く）
根拠B:
-「Fiction引用」→ Candidates側の対応点（短く）
※余計な前置き/まとめ/雑談は禁止。これだけ。

03 最終解法④ アンサンブルと上書き

多数決アンサンブル

複数モデルで予測
同率 1 位の場合は、
Publicスコアが高いモデルの予測を優先

モデルA:（作品X, 作品Y）
モデルB:（作品Y, 作品Z）
モデルC:（作品Y, 作品Z）
採用:（作品Y, 作品Z）

強固な予測による「上書き」

手堅い予測は絶対の自信ありとみなし、
アンサンブルのベース結果を
局所的に上書き（置換）

1. few-shot
8回推論中、6回以上一致した予測
2. Opus 4.6
7回推論中、7回全て一致した予測

04 まとめ

コンペ概要

「実作品が混ざった架空のあらすじ」から元ネタ作品を特定するタスク

解法の全体パイプライン

50候補

→ 候補絞り込み

→ LLMで 1 件ずつ予測

→ 多数決アンサンブル

→ 一定以上一致したもののみ上書き

ポイント

・候補絞り込み＋推論

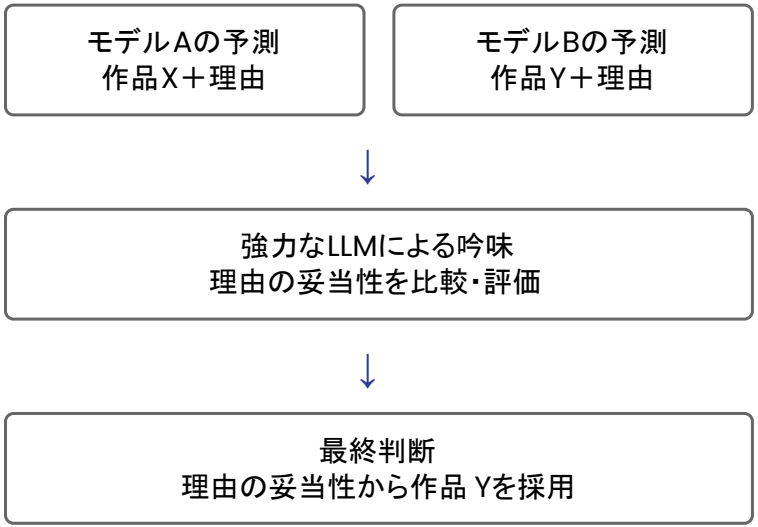
・一致したもののみ置換

試せなかったアイデア

LLMアンサンブルの実装

単純なルールベースの多数決で終わるのではなく、 LLM自身に
複数の「予測結果」「推論の根拠」
を提示し、最終的な答えを議論・選択をさせる手法

・各モデルの推論過程を評価層の LLMが吟味
・多数決では拾えない、少数の鋭い推論を救う
・時間が足りず見送り



楽しいコンペをありがとうございました！！