Structural Dominance Analysis UML Diagram

StructuralScoreCalculator
+calculate(patterns): float

DominancePatternDetector
+detect(turns, metrics): dict

（会話構造性スコア算出クラス）
役割
- 検出された支配パターンを用いて会話構造上の支配性を数値化
- スコアを 0～100 に正規化

メソッドの役目
calculate(patterns) → float
- StructuralConfig で定義された重みを使用
- UI で変更された支配性評価基準を即時反映
- 「構造的にどれだけ強いか」を連続値で表現

（支配構造パターン検出クラス）
役割
- 会話全体から構造的な圧力・支配のパターンを検出
- 単発の強い発話ではなく継続的・構造的な優位性を評価対象とする

メソッドの役目
detect(turns, metrics) → dict
- 以下のような構造的特徴を検出
  - 一方的な発話継続
  - 反論機会の欠如
  - 話題支配
- ルールベース要素を多く含む

AnalysisScore

（分析共通出力DTO）
役割
- A・S・L の出力形式を完全に統一
- 統合ロジック（HSICalculator）が中身を知らずに扱えるようにする

各属性の役目
model_type ・ A／S／L のどれかを識別
score ・会話構造上の支配性の強さ（0～100）
confidence ・このスコアがどれだけ信頼できるか ・構造分析における誤検知対策の要
features_ui ・人が読める説明用特徴量 ・「なぜ支配的と判断されたか」の説明用
features_internal ・内部検証・研究用データ ・UIには出さない

（会話構造性分析サービス）
役割
- 会話構造性分析（S）の処理全体を統括
- 会話ログ（文字起こし）を入力として支配性・優位性・圧力構造を定量化
- A・L と同様に analyze() だけを公開する独立サービス

メソッドの役目
analyze(input: AnalysisInput) → AnalysisScore
- 会話構造性分析の唯一の外部公開メソッド
- 以下の処理を順序制御のみで実行
  a. 会話のターン分割
  b. 発話構造の解析
  c. 支配的パターンの検出
  d. スコア算出
  e. confidence 算出
  f. AnalysisScore への格納
- HSICalculator からは内部構造を一切知られずに利用される

StructuralConfig
+dominance_weights
+threshold_settings
+confidence_weights

（会話構造性分析設定クラス）
役割
- 会話構造性分析に関するすべての可変評価基準を管理
- UI と分析ロジックの橋渡し役

主な属性の役目
- dominance_weights
  - 各支配指標の寄与度
- threshold_settings
  - 支配性判断の境界条件
- confidence_weights
  - 信頼度算出用係数

StructuralDominanceService
+analyze(input: AnalysisInput): AnalysisScore

（信頼度評価クラス）
役割
- 会話構造性スコアの信頼性・再現性を評価
- 短時間会話・データ不足による誤検知を抑制

メソッドの役目
evaluate(patterns, rule_hits, total_rules) → float
- 以下を組み合わせて confidence を算出
  - パターンの一貫性
  - ルール一致率
  - 会話量の十分性
- confidence を 0.0～1.0 で返却

ConfidenceEvaluator
+evaluate(patterns, rule_hits, total_rules): float

（発話権分析クラス）
役割
- 会話の中で誰がどれだけ会話を支配しているかを解析
- 発話回数・発話長・割り込み頻度などを数値化

メソッドの役目
analyze(turns) → dict
- 以下のような指標を算出
  - 発話比率
  - 平均発話長
  - 割り込み回数
- features_internal 用の定量データを生成

TurnTakingAnalyzer
+analyze(turns): dict

（会話構造性分析設定クラス）
役割
- 他のクラスやメソッドでできたものを保管する
- dictが乱立するのを防ぐ
- S拡張時に破綻するのを防ぐ

ConversationTurn

（会話分割クラス）
役割
- 文字起こしデータを「会話ターン（誰が・いつ話したか）」単位に分割
- 会話構造分析の前提となる基盤処理

メソッドの役目
segment(transcript) → List<ConversationTurn>
- 発話者・発話順・発話長をもとにターンを生成
- 出力は以降の分析すべてで共有される

ConversationSegmenter
+segment(transcript): List~ConversationTurn~

AnalysisInput

flow / depends / consumes patterns / output / uses / depends on turns / consumes patterns / input / first step / produces / depends