【別添資料1】 本研究で構築したシステムの詳細

本研究で構築したシステムは下記の5つの処理ステップからなる。

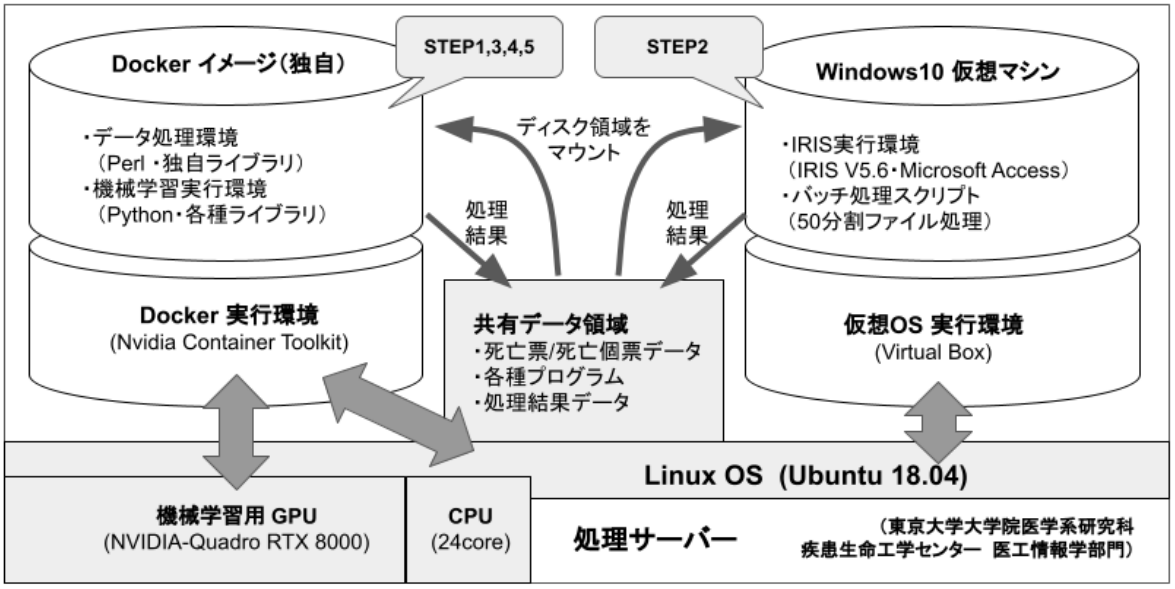
STEP1: 死亡票・死亡個票からの **IRIS** 入力用データの作成
STEP2: **IRIS** での仮原死因確定処理
STEP3: **IRIS** 処理結果の解析
STEP4: 機械学習用データセットの作成
STEP5: 各種機械学習での仮原死因変更有無予測モデルの構築

上記の処理はLinuxサーバ (Ubuntu 18.04) 上のDocker環境にて、本研究にて構築された Docker イメージを用いて行った。Docker環境が用意されているサーバであれば、同イメージを用いることで同様の処理がどのマシンでも可能である。

また STEP2 の部分だけは IRISを動作させる必要上 Windows 環境が必要であったため、Linux サーバー上に仮想OS環境 (Virtual Box) を構築し、その上で動作する Windows 10 仮想マシンにて処理を行った。同一マシンにて処理が完結するため、共有フォルダを介して、データのやり取りが可能であり、1台のサーバーにて完結するシステムである。

共有データ領域にあるプログラムをDockerイメージに梱包して含める構成も可能であり、この場合は実行させるための十分なスペックを持つマシンが有れば、同Dockerイメージを移植することで処理環境だけをどこでも再現することができ、可搬性が高い。一方、IRIS実行環境を含むWindows10仮想マシンの方も移動が可能であるが、OSライセンス (Windows) の問題から自由に共有することは難しい。

下記に構成図を示す。



（図1：システムの概要と処理サーバー内の構成）

以下各ステップでの処理の詳細を示す。

【STEP1】死亡票・死亡個票からの IRIS 入力用データの作成　(Linux サーバ)

■ 01_突合DB (ユニークキー) の作成

- [INPUT]
  - 統計法33上に基づき提供を受けた死亡票・死亡個票 (2015～2020: 6年分)
- [処理]
  - 死亡個票からは「処理年月、届出地、事件簿番号」
    死亡票からは「調査年、提出年月、届出地、事件簿番号」を用いて、両者の結合処理（JOIN）を行う (結果を「突合**DB**」と称する)
  - 「処理年月、届出地番号、事件簿番号」の組み合わせを各死亡案件のキーとした際に、複数回存在するものが存在（ヒアリングの結果、早期提出、また事件簿番号が9999までカウントアップするとまた0に戻る仕様になっていることに起因するもの）するため、これらは処理対象から除外
  - 結果として「処理年月、届出地番号、事件簿番号」の組み合わせで一意に定まるレコードのみを抽出、「突合**DB**（ユニークキー）」と称する。
  - これは、突合**DB**の **98.84%** に相当する。
- [OUTPUT]
  - 突合DB (ユニークキー)　”new_shibo_join.tsv”

■ 02_備考欄前処理

- [INPUT]
  - 突合DB (ユニークキー)
- [処理]
  - 死亡票・死亡個票の各欄からは、入力時の文字数制限により、入り切らない文字列が備考欄に溢れて記入されることがあるが、この「備考欄に溢れた文字列」を多くの正規表現ルールにより、元の然るべき項目へ可能な限り復元して結合する処理
  - 内容は、I欄II欄病名とそれぞれの期間、解剖・手術の詳細、傷害が発生したところ・手段及び状況、その他付言すべき事柄、生後1年未満での病死に関する詳細
  - 元の項目に復元できなかった文字列のみを「備考欄の文字列」として残す
- [OUTPUT]
  - 突合**DB (**ユニークキー・備考欄前処理後**)**
    - “03_new_shibo_join2.tsv”

■ 03_IRIS用入力データ作成

- [INPUT]
  - 突合**DB (**ユニークキー・備考欄前処理後**)**
    - “02_備考欄前処理/03_new_shibo_join2.tsv”
- [処理]
  - 処理年月日が指定された年のもののデータを読み込み
  - 期間表現の修正
  - 標準病名マスター (v5.04) と文字列処理を用いた独自の自動ICD-10コーディングを行う
  - 標準病名マスターのICD-10コードの中で、IRIS では用いられていないICD-10コードを置換する処理
    - 頻度上位のものにつき手動で置換ルールを作成
  - 全てにICD-10コードが当たった死亡票を対象に、生年、没年、性別等をIRISフォーマットに変換
  - IRIS 処理の速度向上のため、50ファイルのIdent, MedCod テーブルに分割して出力

    - 統計情報の出力
        - 「突合DB（ユニークキーのみ）: new_shibo_join2.tsv」の各データ件数のカウント
        - 突合DBで使用された原死因コード、外因コード、母側病態コードの一覧
        - 自動ICDコーディングの結果付与されたICDコードの一覧
- [OUTPUT]
    - IRIS 処理に回せた仮名IDのリスト
        - 03_IRIS用入力データ作成/Result/$year/50分割/ForIris.txt
    - IRIS サーバーへ結果をエクスポート

【STEP2】IRIS での仮原死因確定処理　(Virtual Box 上 仮想 Windows10 マシン)

- **IRIS** バッチ処理（分割ファイル分全て）
    - 50分割されたテキストファイルをAccessDB へインポート
    - コマンドラインでの IRIS バッチ処理
    - 処理結果のテキストデータをLinuxとの共有データ領域へ吐き出し
      (TestIdent1～50)

【STEP3】IRIS 処理結果の解析　(Linuxサーバ)

■ 04_IRIS 処理結果の解析

- [INPUT]
    - Iris 処理用マシンからIris処理結果をインポート
    - Irisの処理結果 (TestIdent1～50.txt) を読み込み
- [処理]
    - Iris処理結果内の不正なセル内改行コード (<CR>) の除去
    - 分割された結果の統合
    - 統計情報の出力（年ごと）（統計情報 / stat${year}.dat)
        - UCcode (8列目) 件数
        - MainInjury (9列目) 件数
        - UC, MainInjury の両方存在
        - IRIS 処理結果 (Final, Initial, Reject) 件数
        - IRIS Reject の内訳件数
- [OUTPUT]
    - IRIS処理結果（TestIdentKekka_${year}.txt）
      （IRIS 処理結果の年別まとめファイル）

■ 05_解析用統合テーブルの作成

- [INPUT]
    - 突合**DB (**ユニークキー・備考欄前処理後**)**
        - 02_備考欄前処理/03_new_shibo_join2.tsv)
    - IRIS へ入力した仮名死亡票ID
        - 03_IRIS用入力データ作成/Result/$year/50分割/ForIris.txt
    - IRIS処理結果
        - 04_IRIS処理結果/TestIdentKekka _${year}.txt
- [処理]

- 死亡票データから各付帯情報の記載の有無テーブルを作成
  - [Script] 1_parseShiboJoin.pl
  - [OUTPUT]
    - Results/Material/${year}/FORIRIS_withCodeHutai.txt
- IRIS処理結果と死亡表データ（突合DBユニークキー）を合わせた「統合テーブル」を作成
  - IRIS の処理結果を読み込み、死亡票由来情報(上記)と突合
  - IRIS による仮原死因コードを修正
    - IRIS では死因符号・外因符号の両方が存在する場合、外因符号を原死因として選択し、国内と逆であるため。
  - [Script] 2_parseIrisOutput.pl
  - [OUTPUT]
    - IRIS処理結果と死亡表データ（突合DBユニークキー）を合わせた「統合テーブル」(50列)
    - Results/${year}/Testkekka_Irekae.txt
- 「統合テーブル」のコード修正
  - IRIS仮原死因コード、国内原死因コードの粒度を合わせるための修正を頻度が多いものを対象に可能な限り行う。
    - IRIS仮原死因コード
      - IRIS側の方が粒度が細かい (桁が多い) ので落とす処理
      - 25ルール
    - 国内原死因コード
      - 国内の方が粒度が細かい (桁が多い) ので落とす処理
        - (例): A048A => A048
      - 原死因には用いない分類のコードの修正
        - (例): C77-C79 => C80
        - 「e-Stat－人口動態統計－分類表－2019年－9死因基本分類表」の「備考」参照
      - 研究期間の途中で準拠するICD-10のバージョンが変更されたことに起因する不一致
        - 平成28年度以前は ICD-10 2003年版
        - 平成29年度以降は ICD-10 2013年版
        - 元データから削除する
      - 397ルール

  - 両者の修正を終えた後、一致/不一致 (0/1) を再度修正
  - [Script] 3_modifyCodes.pl
  - [OUTPUT]
    - **IRIS**処理結果と死亡表データ（突合**DB**ユニークキー）を合わせた「統合テーブル」(50列, コード修正済み)
    - 詳細仕様は「表**C**」参照
    - Results/${year}/BasicTable.txt

- 統計情報の出力
  - 付帯情報の有無 x 仮原死因変更の有無のクロス集計表

- [OUTPUT]
  - **IRIS**処理結果と死亡表データ（突合**DB**ユニークキー・備考欄前処理後）を合わせた「統合テーブル」（**BasicTable)**
    - ${year}/BasicTable.txt

【STEP4】機械学習用データセットの作成　(Linuxサーバ)

■ 06_機械学習用基本データ

- [INPUT]
  - 統合テーブル（Step3で作成）
- [処理]
  - IDに対する共通ベクトルを作成
    - AGE, 性別 (要素数: 2)
    - 使用された ICD10コードに対する数値ベクトル (要素数: 可変)
      - 使用された全ICD10コードを桁数として、使用されたICDコードの所だけ数値が入力されているもの。
      - 桁数は IRIS 出力結果の ACME コードを解析した結果、用いられていた全ICDコードの種類数として別途計算
        - ACMEコードとは、IRIS 内部での修正処理が加わった、I欄・II欄病名に対応するICD10コードセット
        - IRIS が出力する Identテーブルの11列目に該当。
        - IRISが仮原死因コードと選択したものは、必ず「1」
        - その他のコードは「I欄のエ→ア、II欄病名」という優先順位で 0.85, 0.7, 0.55, 0.4, 0.25, 0.1 と「0.15 ずつ減算する形」で傾斜スコアリング。
        - 使用されていないICDコードは「0」。
      - 使用する年数が増えると、出現するICDコードは変化する。
        - 詳細は「表★:各種機械学習手法の結果一覧」を参照
    - 付帯情報の項目の有無 (要素数: 22)
      - TestKekka_syuusei.txt の 27～48列目
  - ID, 共通ベクトル, 仮原死因からの変更の有無(1/0, 正解データ) を組み合わせた学習用基本データを作成
- [OUTPUT]
  - 学習用基本データ
    - RESULT/learningData_${year}.txt
    - 2015, 2016,... 2020 年までの各年データと、2017-2020 年 (ICD-10 2013年版準拠)の通算データを作成

■ 07_付帯情報Embedding

- [INPUT]
  - 学習用基本データ
- [処理]
  - 各種 Embedding 手法により、付帯情報の文字列の内容をベクトル（分散表現）へ変換
  - 学習用基本データと結合し、疎行列表現として出力
- [OUTPUT]
  - 各種Embedding手法によるXGBoost学習用データ
    - BASELINE:
    - TFIDF
    - LSI
    - WORD2VEC
    - DOC2VEC (PV-DM)
    - DOC2VEC (PV-DBOW)

【STEP5】各手法での仮原死因変更有無予測モデルの学習　(Linuxサーバ)

■ 08_機械学習

- STEP4 までで作成された各種Embedding手法による学習用データに対し、勾配ブースティング決定木 (XGBoost) にて「付帯情報を考慮した上で、仮原死因変更有無を予測する」モデルを学習
- BERT による同予測モデルは、別途事前実験にて検証、結果精度がBASELINEを大きく改善しないことから、本実験からは外しているため割愛
    - 詳細は「別添資料2：**BERT**を用いた予測モデルの学習実験」参照
- 結果は「表**F**：各種機械学習手法の結果一覧」参照