Brain(s)コンテスト2022夏LT

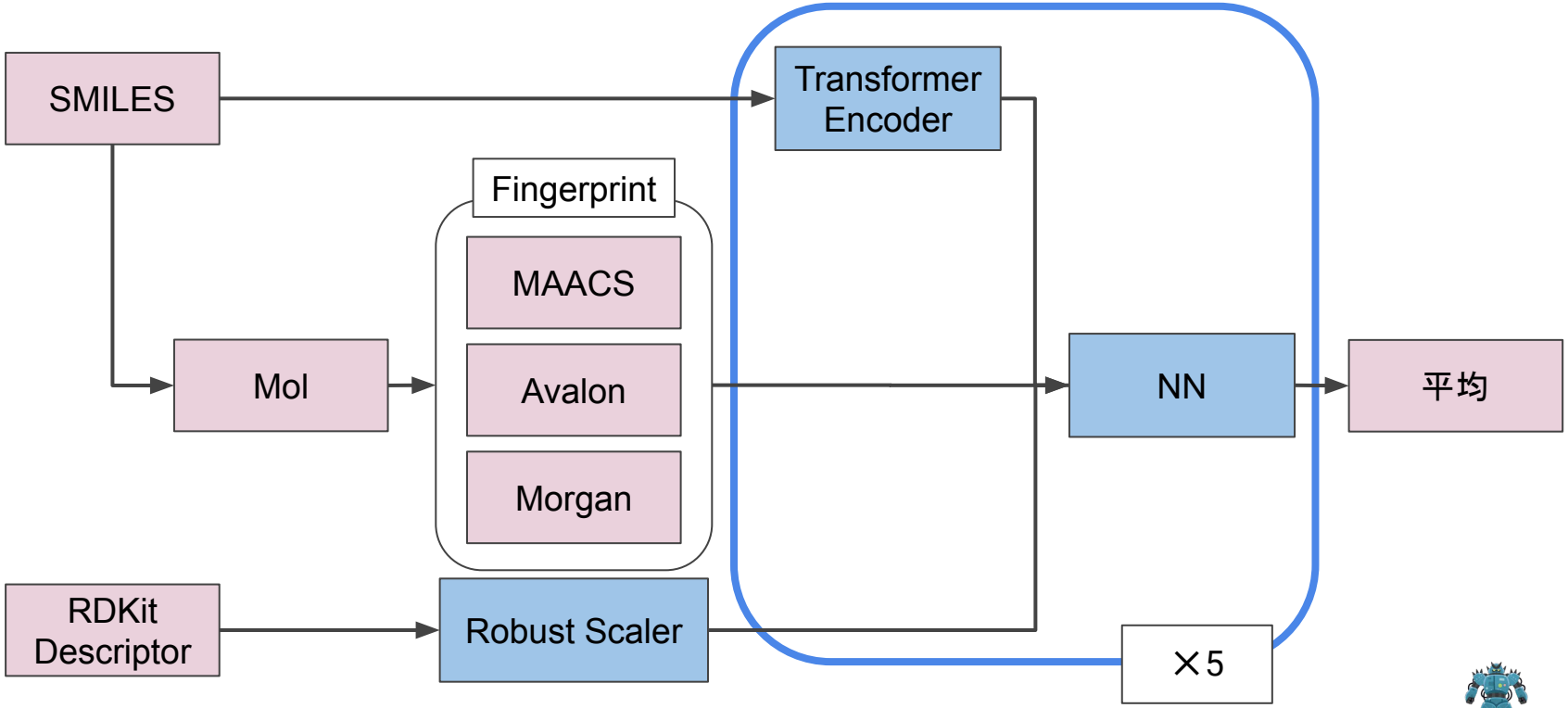
Q2 解法

Q2 解法

SMILESから特徴量作る
の大変そうだなあ…

文字列だし全部Transformerに
任せよう！

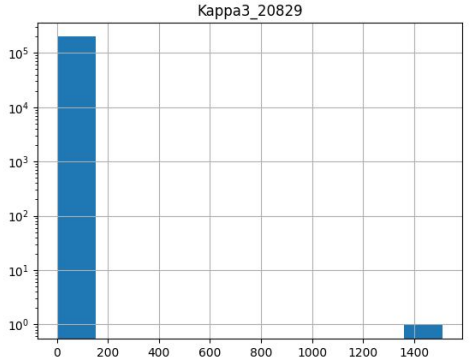
Q2 解法モデル


SMILES
Transformer Encoder
Fingerprint
MAACS
Mol
Avalon
NN
平均
Morgan
RDKit Descriptor
Robust Scaler
×5

Q2 解法

- 工夫点
  - 出力の正規化，Dropoutの除外，正則化
  - 性能に影響したか不明なもの，うまくいかなかったものも含めると他にも色々ある

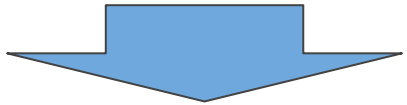
- 苦労した点
  - CVのスコアとLBスコアの乖離
    - 原因は未だによくわからない
    - publicとprivateの乖離はあまりなかった
  - モデルサイズの制約 (10MB以内)



Q3 解法

Q3 解法

- 実験候補として優先順位が高いものは？
  - 良い実験結果が得られているものと近いサンプル
- 実験候補として優先順位が低いものは？
  - 悪い実験結果が得られているものと近いサンプル

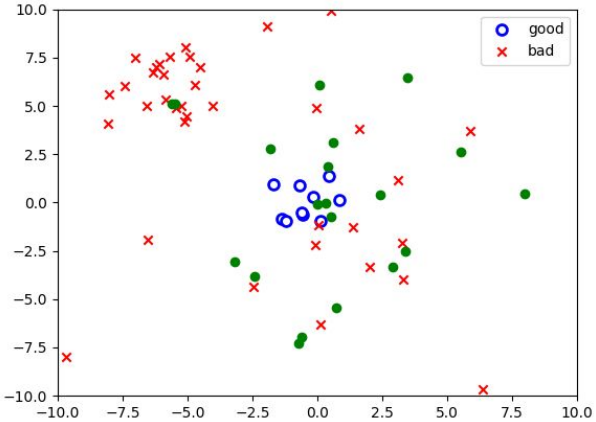


- 良い実験結果に近く，悪い実験結果からは遠いサンプルを優先的に選ぶ
  - 考え方的には対照学習 (SimCLR等)に近い

Q3 解法

- 良い実験結果に近く，悪い実験結果からは遠いサンプルを選ぶ
  - 感覚的に悪い実験結果から離すとスコアが結構上がった

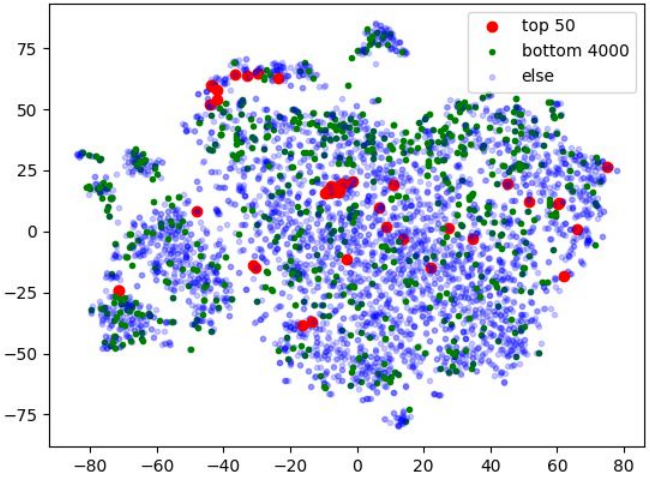
- 距離計算に用いた特徴量
  - PCA + RDKit Descriptor (n = 20)
  - FingerprintのTanimoto係数 (= n(A∧B) / n(A∨B))
    - Morgan
    - MACCS
    - Hashed Atom Pair



Q3 解法　工夫点など

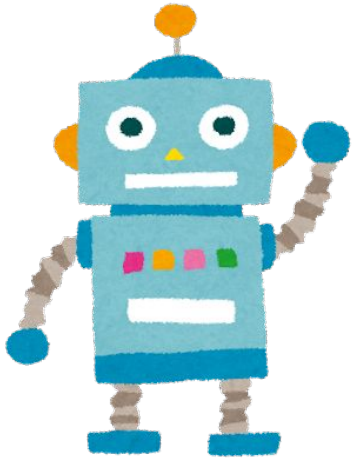
- 確率的なサンプリング
  - 良い結果が空間内の一つの点に必ずしも固まっているとは限らない
  - 選択する実験候補に摂動を与える
  - 良い実験結果が見つからないと始まらない

- 計算量・メモリ使用量の削減
  - 特に計算時間の制約がきつかった
  - 実は提出コードに無駄な処理があることにスライド作成時に気がついた



まとめ・感想

- Q2
  - Transformer+NNのモデル学習を全力で行うことでスコアの向上を目指した
- Q3
  - モデルの学習を行わずに，サンプリング確率などの調整でスコアの向上を目指した



- 必要そうな技術を調べたり，実際にコードを書いて実装する中でいろいろと学ぶことが多かった

どうもありがとうございました！

おまけ：最終提出に付け加えるとしたら何を加えるか？

- Q2
  - データの前処理部分で外れ値や多重共線性を考慮した処理を行う
  - Transformerで学習した特徴量を使って，LightGBMなどの軽量のモデルを大量にスタッキングする

- Q3
  - 距離行列を計算するのでなく，必要最低限のもののみを計算する
  - 余った時間で新たな特徴量を計算する
  - 実験結果のvalueも活用する