森羅2022 Wikipedia構造化プロジェクト 文書分類タスク 成果報告

UNICORN株式会社　小川 晃

自己紹介

小川 晃

所属： UNICORN株式会社 エンジニア Div.

専攻分野： 自然言語処理（情報学修士修了）

主な業務内容（一例）：

　インターネット広告にて入札するキーワードの自動選定

自己ベストを記録した条件

BERT fine-tuning時 (文書分類)

- モデル: rinna/japanese-roberta-base + 1BiLSTM
  (+ dropout [rate=0.2])
- 訓練データ: Wikipedia記事全体
- 最大文長: 256
- 学習率: 1e-5
- 訓練エポック数: 1 epochs
- バッチサイズ: 32 (訓練), 64 (テスト)
- 訓練 : 開発 = 9 : 1

(その他の条件はデフォルトと同一）

今回試みたアプローチの概要

1. 事前学習モデルとしてRoBERTaモデルを利用
2. BiLSTM層(+ Dropout層)をRoBERTaモデルの最終層に追加
3. 訓練データをWikipedia一部記事ではなく全体を利用(学習時間の増大のため、訓練epoch数を1に制限)

リーダーボードでの結果

デモ
- モデル: tohoku/bert-base
- 訓練データ: Wikipedia一部記事
- 学習率: 5e-5
- 訓練エポック数: 4
- 訓練バッチサイズ: 16

スコア : 87.2

今回の提出モデル
- モデル: rinna/roberta-base
  + 1 BiLSTM
- 訓練データ: Wikipedia記事全体
- 学習率: 1e-5
- 訓練エポック数: 1
- 訓練バッチサイズ: 32

スコア : 92.1

スコアが向上

要因分析1：RoBERTa-baseモデルの活用

デフォルト

モデル:

tohoku/bert-base

- 最大文長: 256
- 学習率: 5e-5
- 訓練エポック数: 50
- 訓練バッチサイズ: 16

スコア： 88.6

RoBERTa

- モデル:

  rinna/roberta-base
- 最大文長: 256
- 学習率: 5e-5
- 訓練エポック数: 50
- 訓練バッチサイズ: 16

スコア： 88.6

スコアに変化なし

↑ 同規模の学習データによる事前学習モデルでは性能に大きな差はない？

要因分析2： LSTM層(+Dropput層)の追加

RoBERTa

- モデル:
  rinna/roberta-base
- 最大文長: 256
- 学習率: 5e-5
- 訓練エポック数: 50
- 訓練バッチサイズ: 16

スコア： 88.6

+LSTM

- モデル:
  rinna/roberta-base
  +1 BiLSTM
  (+ dropout [rate=0.20])
- 最大文長: 256
- 学習率: 5e-5
- 訓練エポック数: 50
- 訓練バッチサイズ: 16

スコア： 88.8

スコアがわずかに向上
↑モデルのスケールアップがパフォーマンス向上に
寄与したと推測

要因分析3：より大規模な学習データの活用

Wiki-small (5,000記事)

- モデル:
  rinna/roberta-base
  + 1 BiLSTM
- 学習データ:
  Wikipedia一部記事
- 学習率: 1e-5
- 訓練エポック数: 50
- 訓練バッチサイズ: 32

スコア：77.3

Wiki-all

- モデル:
  rinna/roberta-base
  + 1 BiLSTM
- 学習データ:
  Wikipedia記事全体
- 学習率: 1e-5
- 訓練エポック数: 1
- 訓練バッチサイズ: 32

スコア：92.1

スコアが向上

↑訓練データサイズの増加により、
より多様な記事にモデルが対応可能になったと推測

Extra. 利用モデルの変更 & 各種パラメータの調整

Extra（本提出後の試行）

- モデル: rinna/roberta-base+2 BiLSTM
  (+ dropout [rate=0.25])
- 訓練データ: Wikipedia記事全体
- 学習率: 5e-5
- 訓練エポック数: 2
- 訓練バッチサイズ: 8

スコア : 94.2

スコアが向上
↑より適切なモデルの構成やパラメータの最適化により、
モデルのパフォーマンスが向上したと推測

まとめ

1. BERTモデルのスケールアップ

2. 学習データサイズの拡大

により、分類タスクにおけるモデルの性能向上を達成することができた