卒研に向けて④

アジェンダ

- 先週の課題
- 研究の軸について
- 方向転換について
- 今のお悩み

先週つまってたところ

- 先週の課題
  - 現象：学習すればするほど精度悪化
  - 原因：使用モデルが同じデータセットでファインチューニング済み→過学習
  - 確認：trainデータとtestデータでの精度比較
  - 結果：trainデータ：0.0001、testデータ　7.89とか（元は4.33）
- 対策
  - データセット変更
  - 英語一行手書き文字画像
  - cvlとかいうドイツ語英語のデータセットのみ
    - ドイツ語をデータセットから削除して使用
- 結果
  - 再学習大成功→notion見せる

研究の軸について（1/3）

研究の軸の妥当性確認

- 現状課題
  - 手書き文字認識のエンコーダーの性質から視覚特徴にしか基づいてない
- 研究目的

  そんなモデルに言語認識特徴を付与することで精度向上に寄与できる
- より効果的に示すために
  - 欠損とか損傷加えたほうが研究の妥当性を出せそうだな。。。
  - どんな欠損いれようか

問題提起が適切か確認してみた
→では、現状のモデルの誤検知がどんなものが多いか見てみよう
→エクセルファイル見せる

研究の軸について（2/3）

なぜ？？

→TrOCRのデコーダ部分を確認してみる

```
  (decoder): TrOCRForCausalLM(
    (model): TrOCRDecoderWrapper(
      (decoder): TrOCRDecoder(
        (embed_tokens): TrOCRScaledWordEmbedding(50265, 1024, padding_idx=1)
        (embed_positions): TrOCRLearnedPositionalEmbedding(514, 1024)
        (layernorm_embedding): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
        (layers): ModuleList(
          (0-11): 12 x TrOCRDecoderLayer(
            (self_attn): TrOCRAttention(
              (k_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (v_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (q_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (out_proj): Linear(in_features=1024, out_features=1024, bias=True)
            )
            (activation_fn): GELUActivation()
            (self_attn_layer_norm): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
            (encoder_attn): TrOCRAttention(
              (k_proj): Linear(in_features=768, out_features=1024, bias=True)
              (v_proj): Linear(in_features=768, out_features=1024, bias=True)
              (q_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (out_proj): Linear(in_features=1024, out_features=1024, bias=True)
            )
            (encoder_attn_layer_norm): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
            (fc1): Linear(in_features=1024, out_features=4096, bias=True)
            (fc2): Linear(in_features=4096, out_features=1024, bias=True)
            (final_layer_norm): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
          )
        )
      )
    )
    (output_projection): Linear(in_features=1024, out_features=50265, bias=False)
  )
)' loaded.
```

研究の軸について（2/3）

なぜ？？

→TrOCRのデコーダ部分を確認してみる

```
  (decoder): TrOCRForCausalLM(
    (model): TrOCRDecoderWrapper(
      (decoder): TrOCRDecoder(
        (embed_tokens): TrOCRScaledWordEmbedding(50265, 1024, padding_idx=1)
        (embed_positions): TrOCRLearnedPositionalEmbedding(514, 1024)
        (layernorm_embedding): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
        (layers): ModuleList(
          (0-11): 12 x TrOCRDecoderLayer(
            (self_attn): TrOCRAttention(
              (k_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (v_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (q_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (out_proj): Linear(in_features=1024, out_features=1024, bias=True)
            )
            (activation_fn): GELUActivation()
            (self_attn_layer_norm): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
            (encoder_attn): TrOCRAttention(
              (k_proj): Linear(in_features=768, out_features=1024, bias=True)
              (v_proj): Linear(in_features=768, out_features=1024, bias=True)
              (q_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (out_proj): Linear(in_features=1024, out_features=1024, bias=True)
            )
            (encoder_attn_layer_norm): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
            (fc1): Linear(in_features=1024, out_features=4096, bias=True)
            (fc2): Linear(in_features=4096, out_features=1024, bias=True)
            (final_layer_norm): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
          )
        )
      )
    )
    (output_projection): Linear(in_features=1024, out_features=50265, bias=False)
  )
)' loaded.
```

言語モデルが採用されてる

研究の軸について（3/3）

とは言ったものの

- やってみたい、実装コスト低いしとりあえずまわしてみた

  LLMなし CER: 0.0189

  LLM有り CER: 0.0276

方向転換

- デコーダがあかん
  - どんなデコーダなら有効？？
    - CTCデコーダ → 非自己回帰性だから
    - 音声認識のほうでもCTCにやっていた
  - CTCに置き換えるデメリット
    - 精度悪化の可能性
    - もとのtrocrのencoderに適用できるかあんまわからん
  - なぜデメリットを抱えてCTCにするの？？
    - 推論時の高速性
    - 研究の妥当性(認識にには言語情報が大事)をより鮮明に示すため
  - CTCデコーダで実験回してみていいすか

今のお悩み

- デコーダをどのようにするのか
  - CTCでよいのか
  - ベースラインを自分で作成したモデルでよいのか
- 使用モデル
  - TrOCRは3種類あるどれ使用すべき？（やっぱ理想は全部やるべきか。。。）

Model	Parameters	Total Sentences	Total Tokens	Time	Speed #Sentences	Speed #Tokens
$\text{TrOCR}_{\text{SMALL}}$	62M	2,915	31,081	348.4s	8.37 sentences/s	89.22 tokens/s
$\text{TrOCR}_{\text{BASE}}$	334M	2,915	31,959	633.7s	4.60 sentences/s	50.43 tokens/s
$\text{TrOCR}_{\text{LARGE}}$	558M	2,915	31,966	666.8s	4.37 sentences/s	47.94 tokens/s

今のお悩み

- データベース
  - 勝手にこっちで編集したものを使っていいのか
  - デコーダ変更したら同じデータセットでも過学習しない？？
    - 結構前のデータセット気に入ってる
    - さっき持ってきたデータセットはモデルが予測するのに簡単
- 妥当性を主張する際に使用する欠損損傷データ作成方法
  - もし代表的なものがあれば知りたい
  - なければ自分で調べる