

進捗報告

1 今週やったこと

- Docker の勉強
- 先行研究 Polyvore dataset + Bi-LSTM [1] のコードの実行
- その先行研究の論文の熟読

2 Docker を使って実際に動かしてみた

コードや Dockerfile を全て github に上げてくれていたので、改めてトレーニングし (1 日以上かかりました...), 以下の三つのタスクも解いてみた. (Docker は寺内大先生に教えてもらって勉強しました. めっちゃ時間かかりました.)

- 穴埋め問題 ←私がずっとやっていた問題
- ファッションアイテムの互換性予測 (いわば与えられたコーディネートに点数をつける)
- コーディネート生成
 - 与えられた複数のアイテムに合うように生成
 - 与えられた画像+したいイメージのテキスト情報から生成
 - 複数のテキスト情報だけで生成

前回のゼミで言っていた献立 GA と組み合わせて使うには, この 2 つめのコーディネートに点数をつけるタスクである.

$$E_f(F; \Theta_f) = -\frac{1}{N} \sum_{t=1}^N \log \Pr(x_{t+1} | x_1, \dots, x_t; \Theta_f) \quad (1)$$

$$\Pr(x_{t+1} | x_1, \dots, x_t; \Theta_f) = \frac{\exp(h_t x_{t+1})}{\sum_{x \in X} \exp(h_t x)} \quad (2)$$

$$\min_{\Theta} \sum_F (F_f(F; \Theta_f) + E_b(F; \Theta_b)) + E_e(\Theta_e) \quad (3)$$

(1) 式は forward LSTM の目的関数で, backward LSTM の目的関数も同じように表される. (3) 式は先行研究で提案されている全体の目的関数である. 最終項はテキストと画像のジョイント空間埋め込み学習の損失関数である.

互換性予測は (3) 式を使用して求められている. テストセットからファッションアイテムをランダムに選択することにより, 互換性のない 4,000 の服を作成してこのタスクが解かれている. 図 1 は, この手法の結果を示している. 図ではこのように 0 から 1 に正規化された点数が書かれているのだが, この点数の出し方が一切示されていない. 実際にコードを実行しても forward 方向の loss と backward 方向の loss が最終的に出力されるだけでここからどうやってこのような点数にするのでしょうか???

参考文献

- [1] Xintong Han, Zuxuan Wu, Yu-Gang Jiang, and Larry S Davis. Learning fashion compatibility with bidirectional lstms. In *ACM Multimedia*, 2017.

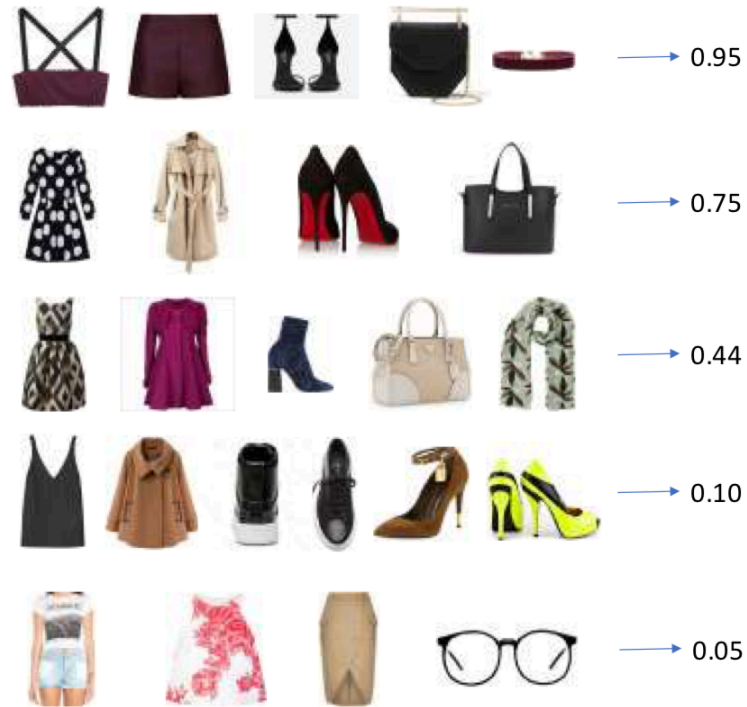


Figure 5: Results of our method on the fashion outfit compatibility prediction task. Scores are normalized to be between 0 and 1 for better visualization.

图 1: compatibility prediction task