wantedly / machine-learning-round-table

[2019/06/26] Machine Learning 輪講 #7

New issue

Open agatan opened this issue 7 days ago · 5 comments

agatan commented 7 days ago Member

Why

Machine Learning 輪講は最新の技術や論文を追うことで、エンジニアが「技術で解決できること」のレベルをあげていくことを目的にした会です。

prev. #6

What

話したいことがある人はここにコメントしましょう！
面白いものを見つけた時点でとりあえず話すという宣言だけでもしましょう！

MizuTakeuchi commented 2 days ago

FYI
Pydata.Tokyoにご参加された町岡さんが19時より参加希望です。
@agatan

1

Hayashi-Yudai commented yesterday

ArcFace: Additive Angular Margin Loss for Deep Face Recognition

https://arxiv.org/pdf/1801.07698.pdf

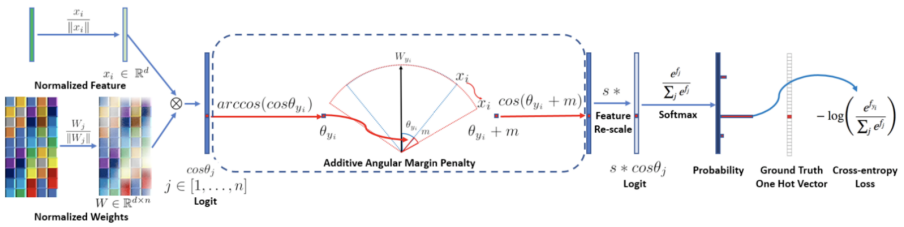
- 顔認証で各クラス同士の間のマージンが大きくなるように学習するための損失関数の提案

従来の方法

- softmax classifier
  - 分類するクラス数に線形にtransformation matrixが巨大化
  - 汎化性能が不十分
- triplet-loss
  - 3つ組の数が巨大なデータセットでは爆発的に増大
  - semi-hard sampleを見つけるのが難しい
- Sphereface
  - angular marginを導入したが、lossを計算するのに近似が必要でtrainingが不安定になりがち

ArcFace

backboneから出てきた出力xと、重みパラメータWをそれぞれL2ノルムで正規化
-> この２つの行列積を取ると、得られたベクトルの各成分はxとWのそれぞれの行のcosになる。
-> 正解ラベルの成分だけ角度をmだけ大きくするペナルティを加えることでより明確にクラス分けを実現する



Assignees
No one assigned

Labels
None yet

Projects
None yet

Milestone
No milestone

4 participants

agatan commented 10 hours ago • edited Author Member

Do Neural Dialog Systems Use the Conversation History Effectively? An Empirical Study

- https://anchor.fm/lnlp-ninja で紹介されていて、興味をもったので読んでみた。
- https://arxiv.org/abs/1906.01603

ACL 2019 の short paper で、対話応答システムがちゃんと対話履歴の情報を使えているのか調べた論文。対話履歴の情報にノイズをのせ、それによるデグレの量を調べている。（大きく落ちるなら対話履歴をちゃんと理解している、落ちないなら対話履歴にかかわらずそれっぽい応答を生成しているだけ）

ノイズは「対話履歴の順番をシャッフル / 逆順にする」「一文内の単語順をいじる」「特定の品詞の単語をマスクする」など。

結果

Models	Test PPL	Only Last	Shuf	Rev	Drop First	Drop Last	Word Drop	Verb Drop	Noun Drop	Word Shuf	Word Rev
		Utterance level perturbations ($\Delta PPL_{[\sigma]}$)					Word level perturbations ($\Delta PPL_{[\sigma]}$)				
DailyDialog											
seq2seq_lstm	$32.90_{[1.40]}$	$1.70_{[0.41]}$	$\mathbf{3.35}_{[0.38]}$	$\mathbf{4.04}_{[0.28]}$	$0.13_{[0.04]}$	$\mathbf{5.08}_{[0.79]}$	$1.58_{[0.15]}$	$0.87_{[0.08]}$	$1.06_{[0.28]}$	$\mathbf{3.37}_{[0.33]}$	$3.10_{[0.45]}$
seq2seq_lstm_att	$29.65_{[1.10]}$	$\mathbf{4.76}_{[0.39]}$	$2.54_{[0.24]}$	$3.31_{[0.49]}$	$\mathbf{0.32}_{[0.03]}$	$4.84_{[0.42]}$	$\mathbf{2.03}_{[0.25]}$	$\mathbf{1.37}_{[0.29]}$	$2.22_{[0.22]}$	$2.82_{[0.31]}$	$\mathbf{3.29}_{[0.25]}$
transformer	$\mathbf{28.73}_{[1.30]}$	$3.28_{[1.37]}$	$0.82_{[0.40]}$	$1.25_{[0.62]}$	$0.27_{[0.19]}$	$2.43_{[0.83]}$	$1.20_{[0.69]}$	$0.63_{[0.17]}$	$\mathbf{2.60}_{[0.98]}$	$0.15_{[0.08]}$	$0.26_{[0.18]}$
Persona Chat											
seq2seq_lstm	$43.24_{[0.99]}$	$3.27_{[0.13]}$	$6.29_{[0.48]}$	$\mathbf{13.11}_{[1.22]}$	$0.47_{[0.21]}$	$\mathbf{6.10}_{[0.46]}$	$1.81_{[0.25]}$	$0.68_{[0.19]}$	$0.75_{[0.15]}$	$1.29_{[0.17]}$	$1.95_{[0.20]}$
seq2seq_lstm_att	$42.90_{[1.76]}$	$\mathbf{4.44}_{[0.81]}$	$\mathbf{6.70}_{[0.67]}$	$11.61_{[0.75]}$	$\mathbf{2.99}_{[2.24]}$	$5.58_{[0.45]}$	$\mathbf{2.47}_{[0.67]}$	$\mathbf{1.11}_{[0.27]}$	$\mathbf{1.20}_{[0.23]}$	$\mathbf{2.03}_{[0.46]}$	$\mathbf{2.39}_{[0.31]}$
transformer	$\mathbf{40.78}_{[0.31]}$	$1.90_{[0.08]}$	$1.22_{[0.22]}$	$1.41_{[0.54]}$	$-0.1_{[0.07]}$	$1.59_{[0.39]}$	$0.54_{[0.08]}$	$0.40_{[0.00]}$	$0.32_{[0.18]}$	$0.01_{[0.01]}$	$0.00_{[0.06]}$
MutualFriends											
seq2seq_lstm	$14.17_{[0.29]}$	$1.44_{[0.86]}$	$\mathbf{1.42}_{[0.25]}$	$1.24_{[0.34]}$	$0.00_{[0.00]}$	$0.76_{[0.10]}$	$0.28_{[0.11]}$	$0.00_{[0.03]}$	$0.61_{[0.39]}$	$0.31_{[0.25]}$	$0.56_{[0.39]}$
seq2seq_lstm_att	$\mathbf{10.60}_{[0.21]}$	$\mathbf{32.13}_{[4.08]}$	$1.24_{[0.19]}$	$1.06_{[0.24]}$	$0.08_{[0.03]}$	$\mathbf{1.35}_{[0.15]}$	$\mathbf{1.56}_{[0.20]}$	$0.15_{[0.07]}$	$\mathbf{3.28}_{[0.38]}$	$\mathbf{2.35}_{[0.22]}$	$\mathbf{4.59}_{[0.46]}$
transformer	$10.63_{[0.03]}$	$20.11_{[0.67]}$	$1.06_{[0.16]}$	$\mathbf{1.62}_{[0.44]}$	$\mathbf{0.12}_{[0.03]}$	$0.81_{[0.09]}$	$0.75_{[0.05]}$	$\mathbf{0.16}_{[0.02]}$	$1.50_{[0.12]}$	$0.07_{[0.01]}$	$0.13_{[0.04]}$
bAbi dailog: Task5											
seq2seq_lstm	$1.28_{[0.02]}$	$1.31_{[0.50]}$	$\mathbf{43.61}_{[15.9]}$	$\mathbf{40.99}_{[9.38]}$	$0.00_{[0.00]}$	$4.28_{[1.90]}$	$0.38_{[0.11]}$	$0.01_{[0.00]}$	$0.10_{[0.06]}$	$0.09_{[0.02]}$	$0.42_{[0.38]}$
seq2seq_lstm_att	$\mathbf{1.06}_{[0.02]}$	$\mathbf{9.14}_{[1.28]}$	$41.21_{[8.03]}$	$34.32_{[10.7]}$	$0.00_{[0.00]}$	$\mathbf{6.75}_{[1.86]}$	$\mathbf{0.64}_{[0.07]}$	$0.03_{[0.03]}$	$0.22_{[0.04]}$	$\mathbf{0.25}_{[0.01]}$	$\mathbf{1.10}_{[0.80]}$
transformer	$1.07_{[0.00]}$	$4.06_{[0.33]}$	$0.38_{[0.02]}$	$0.62_{[0.02]}$	$0.00_{[0.00]}$	$0.21_{[0.02]}$	$0.36_{[0.02]}$	$\mathbf{0.25}_{[0.06]}$	$\mathbf{0.37}_{[0.06]}$	$0.00_{[0.00]}$	$0.00_{[0.00]}$

Table 2: Model performance across multiple datasets and sensitivity to different perturbations. Columns 1 & 2 report the test set perplexity (without perturbations) of different models. Columns 3-12 report the **increase** in perplexity when models are subjected to different perturbations. The mean (μ) and standard deviation [σ] across 5 runs are reported. The *Only Last* column presents models with **only** the last utterance from the dialog history. The model that exhibits the highest sensitivity (higher the better) to a particular perturbation on a dataset is in bold. *seq2seq_lstm_att* are the most sensitive models **24/40** times, while transformers are the least with **6/40** times.

- Transformer ベースのモデルは順番の変化に鈍感
- Attention を使うモデルは、 `Only last` のときに大きくパフォーマンスが変化する = vanilla seq2seq と比べて対話初期のコンテキストを参照できている

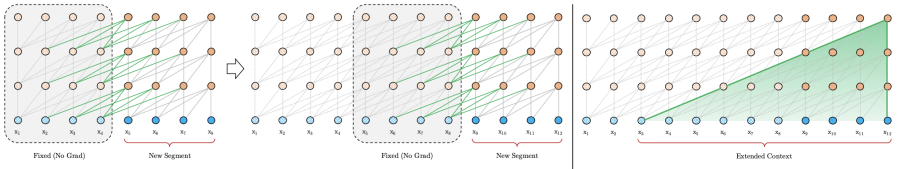
agatan commented 10 hours ago • edited Author Member

Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context

BERT を超えるモデルが出てきた [1906.08237] XLNet: Generalized Autoregressive Pretraining for Language Understanding
XLNet は Transformer-XL ベースらしいので、Transformer-XL のほうから読んでみる。XL は extra long らしい。
（紛らわしいことに facebookresearch も BERT を超えた XLM というのを出していた。こっちは cross lingual の XL かな？ https://arxiv.org/abs/1901.07291 ）

- https://arxiv.org/abs/1901.02860
- https://ai.googleblog.com/2019/01/transformer-xl-unleashing-potential-of.html

Transformer は fixed length context しか考慮できない。segment 間の接続はない上に文節などを考慮せず segment にわけるので、言語モデルに必要な情報がそもそも入力されていないという状況が起こりうる。
そこで前の segment の context を reuse することで transformer にも再帰構造を導入しようというのが Transformer-XL



(a) Training phase. (b) Evaluation phase.

Figure 2: Illustration of the Transformer-XL model with a segment length 4.

Relative Positional Encodings

ナイーブにやると positional な情報を使えないのでパフォーマンスが落ちる。
（Transformer は positional encoding によって各 token が何単語目なのかを伝えていたが、単純に transformer-xl にそれを適用しても segment N と segment N + 1 の間で positional encoding が同じ値になってしまう）
そこで、相対的な位置情報を元に attention score を決定する relative positional encodings を導入している。（ここなんでこれでうまくいくのかよくわからなかった。気持ちは伝わった。）
直感的には「位置による attention 」と「token の意味による attention 」（とその相互作用項）を分離して考える + 絶対位置による embedding をやめて相対位置による embedding にしている？

matchbou commented 3 hours ago

https://github.com/matchbou/Etc20190626