Уменьшение размерности пространства обучаемых параметров в задаче адаптации к домену

Анна Ремизова
научный руководитель: к.ф.-м.н. А.В. Грабовой

МФТИ

18/05/2024

Содержание

1. Введение
2. Предложенный метод
3. Вычислительный эксперимент

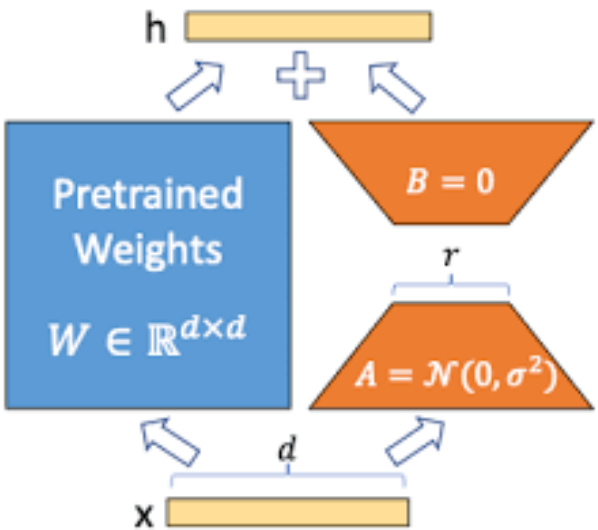
Введение

Цели работы. Исследуются методы снижения пространства обучаемых параметров при помощи сингулярного разложения матриц, а также коорректность применения изучаемых методов к задаче классификации текстов.

Методы исследования. Применяется низкоранговое разложение (англ. Low Rank Adaptation) к матрицам параметров в больших языковых моделях. Используются статистические методы оценки функции минимизации функции потерь, а также свойтва матриц для доказательства применимости предложенного метода к задаче классификации.

Введение

Метод, рассмотренный в данной работе — низкоранговое разложение (англ. Low Rank Adaptation) [1].



Постановка задачи

Для задачи классификации текстов:

$$f_\theta : \hat{V} \to [N_c], \tag{1}$$

где f_θ —модель трансформер, $\hat{V} \subset V^*$; V — словарь токенов и V^* — его замыкание или множество всех последовательностей над V, $[N_c]$ — множество классов. Таким образом, модель отображает текст из $\hat{V}$ в класс из $[N_c]$.

Постановка задачи

При дообучении модель инициализируется предварительно обученными весами Φ_0 и обновляется до $\Phi_0 + \Delta\Phi$, где $\Delta\Phi$ — набор дообучаемых параметров. При использовании LoRA $\Delta\Phi$ задается набором параметров Θ намного меньшего размера, чем Φ_0 : $\Delta\Phi = \Theta$, где $\mid \Theta \mid \ll \mid \Phi_0 \mid$ и задача минимизации функции потерь имеет вид:

$$\begin{aligned} \min_{\Theta} \left(- \sum_{X_i \in \hat{V} \subset V^*} \sum_{c_i \in [N_c]} \log \left(P_{\Phi_0 + \Theta} \left(c_i \mid X_i \right) \right) \right) = \\ = \max_{\Theta} \sum_{X_i \in \hat{V} \subset V^*} \sum_{c_i \in [N_c]} \log \left(P_{\Phi_0 + \Theta} \left(c_i \mid X_i \right) \right). \end{aligned} \tag{2}$$

LoRA адаптер

В данной работе LoRA применяется к задаче классификации. Структура обновления весов при использовании LoRA адаптера описана в таблице 1,

Fine tuning	LoRA fine tuning
$W_{upd} = W + \Delta W$ $\hat{y} = xW_{upd} = x(W + \Delta W)$ $\hat{y} = xW + x\Delta W$	$W_{upd} = W + AB$ $\hat{y} = xW_{upd} = x(W + AB)$ $\hat{y} = xW + xAB$

Таблица 1: Структура обновления весов при использовании LoRA адаптера

где $W \in \mathbb{R}^{d \times k}$ — предобученные веса, $\Delta W \in \mathbb{R}^{d \times k}$ — матрица обновленных весов. ΔW приближается с помощью метода LoRA произведением $A \cdot B$, где $A \in \mathbb{R}^{d \times r}$, $B \in \mathbb{R}^{r \times k}$ и r — гиперпараметр ранга. Здесь $A \sim \mathcal{N}(0, \sigma^2)$ и $B = [0]_{r \times k}$.

Состоятельность предложенной модели

Сходимость модели трансформер без использования LoRA была доказана в работе [2]. Доказательство приведено для задачи классификации:

Theorem

Будем считать, что:
1) Задана модель с набором параметров Θ^*, генерирующая эмпирическое распределение данных $P_{model}(\cdot, \Theta^*)$, которое аппроксимирует истинное распределение данных P_{true} с минимальным расхождением по KL-дивергенции:

$$\exists \Theta^* : \Theta^* = \arg\min_{\Theta} D_{KL}(P_{true} \,||\, P_{model}(\cdot, \Theta)), \tag{3}$$

Состоятельность предложенной модели

Theorem

2) При увеличении размера выборки $\hat{V}$ эмпирическое распределение данных $P_{model}(\cdot, \Theta)$ приближается к истинному распределению, генерирующему данные.
3) Функция ошибки $\mathscr{L}(\theta)$ — непрерывная, дифференцируемая.
Где

$$\mathscr{L}(\Theta) = -\sum_{X_i \in \hat{V} \subset V^*} \sum_{c_i \in [N_c]} \log\left(P_{\Phi_0 + \Delta\Phi(\Theta)}\left(c_i \mid X_i\right)\right). \quad (4)$$

Тогда минимизация функции потерь $\mathscr{L}(\Theta)$ приводит к состоятельной оценке истинного распределения, порождающего данные. Т.е.:

$$\lim_{|\hat{V}| \to \infty} \arg\min_{\Theta} \mathscr{L}(\Theta) = \Theta^*. \quad (5)$$

О применимости LoRA к задаче классификации

Note

Для решения задачи классификации с помощью BERT требуется не более чем дополнительный $\mathrm{softmax}$ слой после BERT:

$$p(c \mid \mathbf{x}) = \mathrm{softmax}(W^T \mathbf{x})$$
$$\hat{\mathbf{y}} = \mathrm{softmax}\left(W^T \mathbf{x}\right) = \frac{\exp\left(W^T \mathbf{x}\right)}{\sum_{i=1}^{k} \exp\left(W^T \mathbf{x}\right)_i}, \tag{6}$$

где $\mathbf{x}$ — это выходной результат последнего слоя BERT, а W — матрица весов.

О применимости LoRA к задаче классификации

Theorem

В рамках задачи классификации, при заданных условиях:
1) Модель семейства BERT с дополнительным слоем

$$\hat{\mathbf{y}} = \text{softmax}\left(W_{upd}^T \mathbf{x}\right) = \frac{\exp\left(W_{upd}^T \mathbf{x}\right)}{\sum_{i=1}^{k} \exp\left(W_{upd}^T \mathbf{x}\right)_i}, \tag{7}$$

где

$$W_{upd} = \underset{(d\times k)}{W} + \underset{(d\times k)}{\Delta W}, \tag{8}$$

и x —это выходной результат BERT, W — матрица весов, ΔW — матрица обновленных весов.

О применимости LoRA к задаче классификации

Theorem

2) Данная модель BERT без дополнительного слоя также корректно работает с аппроксимацией

$$\underset{(d\times k)}{\Delta W} = \underset{(d\times r)}{A} \times \underset{(r\times k)}{B}, \tag{9}$$

3) Выполняется теорема о состоятельности предложенной модели.

Тогда можно утверждать, что при (9) *заданная модель BERT с дополнительным слоем гарантирует корректную выходную матрицу.*

Данные

Открытый исходный датасет для мультиклассовой классификации текстов, написанных человеком и различными языковыми моделями. Представлено 4 класса: ChatGPT, Davinci, Cohere, Humans. Всего в датасете 47327 текстов с разметкой по классам. Средняя длина текста по всему датасету — 400 слов, средняя длина текстов в зависимости от класса представлена в таблице 3. Средняя длина слова — 5 символов. Вес каждого класса — ,безразмерная величина, показывающая насколько несбалансированна выборка и к каким классам применять большие веса. Статистика по весам классов приведена в таблице 2.

Данные

имя класса	**вес, б/р**
chatGPT	0.986
cohere	1.043
davinci	0.986
human	0.986

Таблица 2: Вес каждого класса

имя класса	**длина текста, слова**
chatGPT	362
cohere	279
davinci	343
human	607

Таблица 3: Средняя длина текста

Предобученная модель DRoBERTa-base, мультиклассовая классификация.

После обучения для оценки использовались матрица ошибок и метрики точности, полноты и F1-меры, результаты представлены в таблице 4. Для визуализации ошибки использовалась матрица несоответствий (англ. Confusion matrix), для данного эксперимента результаты приведены в таблице 5.

время обучения: 4041.3188 секунд

имя класса	precision	recall	f1-score
chatGPT	1.000	0.993	0.997
cohere	0.963	0.999	0.981
davinci	0.986	0.996	0.991
human	0.991	0.952	0.971

Таблица 4: Метрики качетва DRoBERTa-base

Предобученная модель DRoBERTa-base, мультиклассовая классификация.

истинные метки \ предсказаные метки	chatGPT	Cohere	Davinci	Human
chatGPT	**0.993**	0.002	0.0	0.005
Cohere	0.0	**0.999**	0.0	0.001
Davinci	0.0	0.001	**0.996**	0.003
Human	0.0	0.035	0.013	**0.952**

Таблица 5: Confusion matrix, DRoBERTa-base

Предобученная модель DRoBERTa-base & LoRA, мультиклассовая классификация.

Только 0.828% параметров обучаются при использовании LoRA. Предположим, что обучится такая модель гораздо быстрее. Гипотеза подтвердилась экспериментально, что отображено в таблице 6. Матрица несоответсвий для данноого эксперимента представлена в таблице 7.

время обучения: 3210.977 секунд
trainable params: 685828, all: 82807304 || trainable%: 0.8282

model	precision	recall	f1-score
chatGPT	0.997	0.786	0.879
cohere	0.667	0.940	0.780
davinci	0.703	0.971	0.816
human	0.717	0.317	0.440

Таблица 6: Метрики качетва DRoBERTa-base & LoRA

Предобученная модель DRoBERTa-base & LoRA, мультиклассовая классификация.

истинные метки \ предсказаные метки	chatGPT	Cohere	Davinci	Human
chatGPT	**0.79**	0.01	0.08	0.12
Cohere	0.0	**0.94**	0.06	0.003
Davinci	0.001	0.03	**0.98**	0.0
Human	0.002	0.43	0.25	**0.32**

Таблица 7: Confusion matrix, DRoBERTa-base & LoRA

Три независимые модели DRoBERTa-base & LoRA, бинарная классификация.

ChatGPT vs Human

Эксперимент, представленный здесь, аналогичен предыдущему, но модель решает задачу бинарной классификации. Результаты представлены в таблице 8.

время обучения: 1633.8114 секунд

model	precision	recall	f1-score
chatGPT	1.000	0.891	0.942
human	0.902	1.000	0.950

Таблица 8: Метрики качетва DRoBERTa-base & LoRA, chatGPT vs Human

Три независимые модели DRoBERTa-base & LoRA, бинарная классификация.

Cohere vs Human

Результат эксперимента представлен в таблице 9.

время обучения: 1583.556 секунд

model	precision	recall	f1-score
cohere	0.999	0.837	0.911
human	0.853	0.999	0.920

Таблица 9: Метрики качетва DRoBERTa-base & LoRA, Cohere vs Human

Три независимые модели DRoBERTa-base & LoRA, бинарная классификация.

Davinci vs Human

Результат эксперимента представлен в таблице 10.

время обучения: 1632.395 секунд

model	precision	recall	f1-score
davinci	0.996	0.851	0.918
human	0.870	0.997	0.929

Таблица 10: Метрики качетва DRoBERTa-base & LoRA, Davinci vs Human

Выводы

Если “усреднить” показатели трех моделей эксперимента, то можно заметить улучшение качества по сравнению с метриками качетва DRoBERTa-base & LoRA для мультиклассовой классификации, таблица 11, также показатели сравнимы с показателями метрик до применения LoRA, таблица 12.

model	**precision**	**recall**	**f1-score**
chatGPT	1.000	0.891	0.942
cohere	0.999	0.837	0.911
davinci	0.996	0.851	0.918
human	0.875	0.999	0.933

Таблица 11: Метрики качетва DRoBERTa-base & LoRA, бинарные классификаторы

Выводы

model	precision	recall	f1-score
chatGPT	1.000	0.993	0.997
cohere	0.963	0.999	0.981
davinci	0.986	0.996	0.991
human	0.991	0.952	0.971

Таблица 12: Метрики качетва DRoBERTa-base, мультиклассовая классификация

Показатели precision выросли у всех классов, кроме human, в то время как у этого класса выросла метрика recall. Суммарно, качество классификации выросло, не потеряв во времени обучения, по сравнению с предобученной моделью DRoBERTa-base. И сильно выиграло в качестве у модели DRoBERTa-base & LoRA, но проиграв ей во времени обучения.

Вывод эксперимента

При решении задачи мультиклассовой классификации предложенная модель BERT LoRA тратит меньше ресурсов, чем модель без использования LoRA, но метрики качества падают. Однако, при решении тремя одинаковыми независимыми моделями задачи бинарной классификации с последующим усреднением метрики качетва растут, а использование ресурсов — нет. Таким образом, в данной работе теоритически и экспериментально доказана состоятельность и эффективность предложенного метода. 24 / 25

Выносится на защиту

1. Сформулирована и доказана теорема о применимости LoRA адаптера к задаче классификации текстов
2. Переформулирована теорема о состоятельности модели для модели с использованием LoRA адаптера
3. Проведен эксперимент и доказана эффективность предложенного метода

Библиография

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen.
Lora: Low-rank adaptation of large language models.
arXiv preprint arXiv:2106.09685, 2021.

Minhyeok Lee.
A mathematical investigation of hallucination and creativity in gpt models.
Mathematics, 11(10):2320, 2023.