

Методы предобработки текстовых данных для ускорения обучения языковых моделей с помощью обучения по плану

Сурков Максим Константинович

Научный руководитель: Ямщиков Иван Павлович

Санкт-Петербургская школа физико-математических и компьютерных наук
НИУ ВШЭ СПб

20 апреля 2021 г.

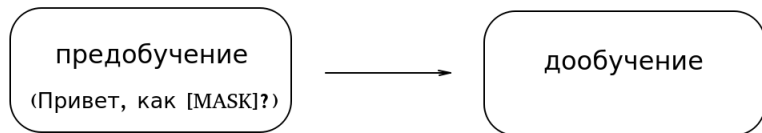
- социальные сети
- голосовые помощники
- переводчики
- чат-боты



- классификация
- машинный перевод
- вопросно-ответные системы

- небольшие языковые модели
- GPT-3
 - очень большая модель
- BERT
 - высокое качество

Мотивация. Обучение языковой модели



- требуемое время: от 1-2 дней до **1-2 недель**

- мировой рекорд: 47 минут с использованием **1472 GPU**

Корпус данных	Размер
Wikipedia	3-600M
BookCorpus	74M

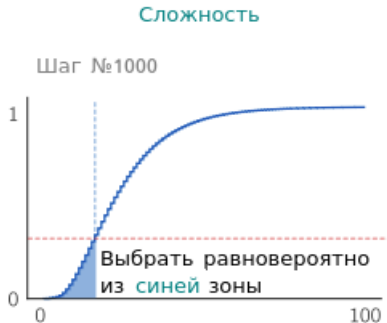
- требуемое время: 1-2 дня

Корпус данных	Размер
HND	600k-2M
s140	1.6M
IWSLT	200-230k
QQP	364k
MNLI	393k

- долго** обучать
- нужно обрабатывать **большие** объемы данных

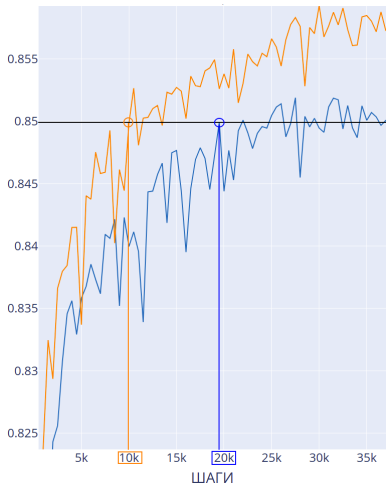
Обучение по плану. Определение

- 1 сортируем данные по сложности (длина)
- 2 в течение T шагов (рассмотрим t -й шаг)
 - вычисляем $c(t) \in [0, 1]$
 - формируем пакет данных маленького размера из множества $c(t)$ **легких** примеров
 - шаг обучения



Метод сравнения алгоритмов обучения

- 1 фиксируем: корпус данных, модель, семплер
- 2 обучаем модель
- 3 фиксируем достаточно большой порог (точность, функция потерь)
- 4 сравниваем графики
- 5 или сравниваем среднее число шагов, необходимое для достижения данного порога



Поле исследований

метрика	классификация	перевод	предобучение	NLU ¹
длина		✓		
языковая ²				
энтропия				
модельная				✓
частота слов		✓		
правдоподобие		✓		
?				

- не изучено влияние обучения по плану на задачах классификации и предобучения
- покрыто узкое множество метрик (длина - лучшая метрика на данный момент)
- нет универсального решения

¹ вопросно-ответная система

²Sluis et al. (2010) показали слабую корреляцию с реальной сложностью текста

Цель: ускорить обучение языковой модели BERT с помощью обучения с расписанием за счет метрики оценки сложности текстовых данных на задачах предобучения и классификации

Задачи:

- 1 Предложить метрики оценки сложности текста
- 2 Реализовать производительные алгоритмы вычисления предложенных метрик на больших наборах данных
- 3 Сравнить найденные метрики
- 4 Исследовать влияние найденных метрик на скорость обучения языковой модели BERT

1 база

- **длина**, вероятность правдоподобия (Platanios et al., 2019)
- самое редкое слово в предложении (Xuan Zhang et al., 2018)

2 информационный поиск

- **tf-idf**

3 теория информации (Nihat Ay et al., 2006)

- EE, TSE

$$T = (t_1, t_2, \dots, t_{i-1}, t_i, \dots, t_n)$$

↓

$$\xi = (\xi_{t_1}^1, \xi_{t_2}^2, \dots, \xi_{t_{i-1}}^{i-1}, \xi_{t_i}^i, \dots, \xi_{t_n}^n)$$

$t_i \rightarrow \xi_{t_i}^i =: \mu_i$ — бинарная случайная величина

4 модельная

- MLM-loss

- EE, TSE
 - сложные математическая формулы
 - $\mathcal{O}^*(2^n), \mathcal{O}(n^2)$ – несравнимо долго \rightarrow алгоритм за $\mathcal{O}(n)$
- максимальный частотный ранг
 - 1 вычисляем частоту каждого слова
 - 2 присваиваем каждому слову позицию в массиве, отсортированном по убыванию частоты
 - 3 сложность предложения – максимальный ранг по всем словам в предложении
- правдоподобие

$$L(T) = - \sum_{i=1}^n \log f(t_i), \text{ где } f(x) - \text{частота слова}$$

- MLM-loss
 - учим BERT на задаче MLM (Пример: "Привет, как [МАСКА]?"), оптимизируя кросс-энтропию
 - сложность = значение кросс-энтропии на данном тексте

- **статистики**

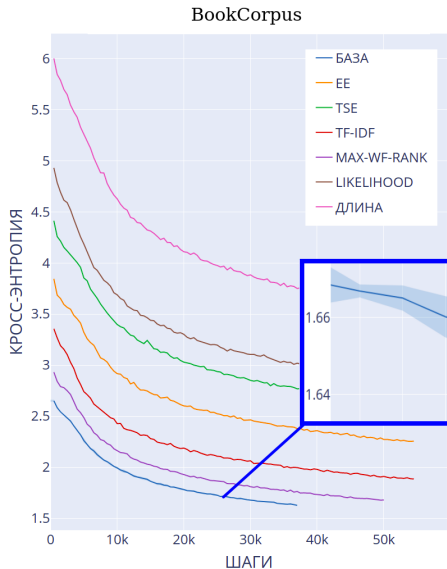
- 1 длина \rightarrow число текстов с такой длиной
 - 2 $(i, x_i) \rightarrow$ число текстов, где $t_i = x_i$
 - 3 $(x_i) \rightarrow$ число текстов, где x_i является последним токеном
 - 4 $(i, x_{i-1}, x_i) \rightarrow$ число текстов, где на $(i - 1)$ -й позиции стоит x_{i-1} , а на i -й позиции стоит x_i
 - 5 $x_i \rightarrow$ число текстов, в которых есть x_i
- сбор статистик в параллельном режиме (разделение по данным)
 - вычисление MLM-loss требует GPU

Итого:

- предложены подходы, покрывающие широкое множество метрик
- предложены алгоритмы, вычисляющие метрики за пренебрежимо малое время (по сравнению со временем обучения)

Сравнение метрик. Предобучение

- Обучение по плану сильно проигрывает базовому решению
- метрики имеют порядок вне зависимости от семплера
 - 1 максимальный ранг слова
 - 2 TF-IDF
 - 3 EE
 - 4 TSE
 - 5 правдоподобие
 - 6 длина
- длина проигрывает остальным метрикам
- семплеры имеют порядок вне зависимости от метрики



Сравнение метрик. Классификация

$$\max \Delta \leq 3k$$

Корпус данных	sentiment140 (85.5%)				
семплер	CB	DB	Нур	SS	SM
длина (86.2%)	112.5k	20k	19k	-	-
TF-IDF (86.7%)	115.5k	21.5k	19.5k	16.5k	22k
TSE (86.8%)	95.5k	16.5k	20.5k	21.5k	18k
EE (86.7%)	59k	16.5k	23k	20k	19k
max wf rk (86.7%)	70k	18.5k	19.5k	17k	19k
правдоподобие (86.7%)	112k	17.5k	21.5k	17.5k	21.5k
MLM-loss (86.1%)	59.5k	21k	23.5k	?	?
база (87%)	18k				

- нет статистически значимой разницы в метриках (искл.: длина, MLM-loss)
- длина ухудшает качество модели
- влияние семплера много больше влияния метрики на скорость обучения

Влияние метрик на скорость обучения

- любая конфигурация обучения по плану проигрывает стандартному алгоритму обучения
- влияние токенизатора: нет
- влияние гиперпараметров обучения: нет
- влияние опыта предобученной модели на обучение по плану: нет
 - замена предобученного BERT-base на случайно инициализированный не приводит к выигрышу обучения по плану
- BERT переобучается на длину: нет (семплы SS, SM)
- итоговое распределение датасета неравномерное: нет (DB, Hup)

- ❶ Предложен широкий спектр метрик оценки сложности текста
 - метрики TSE и EE адаптированы под задачу обработки языка
- ❷ Реализованы алгоритмы подсчета метрик на больших объемах данных
- ❸ Проведено сравнительное исследование метрик
 - длина проигрывает всем
 - предобучение: есть строгий порядок (Wikipedia, BookCorpus)
 - классификация: нет значимых отличий (s140, HND)
 - показано, что влияние метрики зависит от семплера
- ❹ поведение метрик зависит от задачи \Rightarrow не удалось найти универсального решения
- ❺ Не удалось добиться существенного ускорения относительно базового подхода на задачах предобучения и классификации

$$EE(X) = \left[\sum_{v \in V} H(X_{V \setminus \{v\}}) \right] - (n-1)H(X_V) =$$

$$\left[\sum_{i=1}^n H(\mu_1, \dots, \mu_{i-1}, \mu_{i+1}, \dots, \mu_n) \right] - (n-1)H(\mu)$$

- $\mathcal{O}(n^2)$
- $\mathcal{O}(n)$

$$\sum_{i=1}^n H(\mu_1, \dots, \mu_{i-1}, \mu_{i+1}, \dots, \mu_n) =$$

$$= \sum_{i=1}^n H(\mu) - H(\mu_i | \mu_{i-1}) - H(\mu_{i+1} | \mu_i) + H(\mu_{i+1})$$

$$EE(X) = \sum_{i=2}^n H(\mu_i) - H(\mu_i | \mu_{i-1}) = \sum_{i=2}^n I(\mu_{i-1} : \mu_i)$$

$$\sum_{k=1}^{n-1} \frac{k}{n} C^{(k)}(X_V)$$
$$C^{(k)}(X_V) = \frac{n}{k \binom{n}{k}} \sum_{A \subseteq V, |A|=k} H(X_A) - H(X_V) =$$
$$= \frac{n}{k} \left[\frac{1}{\binom{n}{k}} \sum_{A \subseteq V, |A|=k} H(X_A) \right] - H(X_V)$$

Дополнительно: Вычисление TSE

$$\frac{1}{\binom{n}{k}} \sum_{A \subseteq V, |A|=k} H(X_A) = \frac{1}{\binom{n}{k}} \sum_{1 \leq i_1 < i_2 < \dots < i_k \leq n} H(\mu_{i_1}, \mu_{i_2}, \dots, \mu_{i_k})$$

- ❶ $\mathcal{O}^*(2^n)$
- ❷ $\mathcal{O}(n^2)$ - динамическое программирование
- ❸ $\mathcal{O}(n)$

$$\sum_{i=1}^n A_i H(\mu_i) + \sum_{i=2}^n B_i H(\mu_i | \mu_{i-1})$$

$$A_i = \begin{cases} \binom{n-2}{k-1} / \binom{n}{k} = \frac{k(n-k)}{n(n-1)}, & i > 1 \\ \binom{n-1}{k-1} / \binom{n}{k} = \frac{k}{n}, & i = 1 \end{cases}$$

$$B_i = \frac{\binom{n-2}{k-2}}{\binom{n}{k}} = \frac{k(k-1)}{n(n-1)}$$

Результаты. классификация. HND

Корпус данных: Hyperpartisan News Detection

$$\max \Delta \leq 3k$$

Корпус данных семплер	HND (92.9%)				
	CB	DB	Hyp	SS	SM
length (93.7%)	55k	23k	22.5k	-	-
TF-IDF (93.5%)	∞	19.5k	24k	23.5k	33k
TSE (93.8%)	56.5k	21k	23k	22k	31k
EE (93.8%)	71.5k	25.5k	22.5k	19.5k	32.5k
max wf rk (93.6%)	∞	22k	20.5k	22.5k	39k
правдоподобие (93.8%)	∞	20k	24k	20k	30k
база (93.8%)	22k				