

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

- текст → токены → BERT → предсказание



- 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

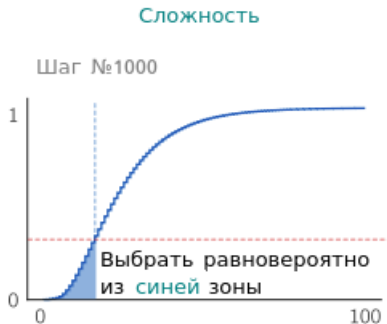
- требуемое время: 1-2 дня

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

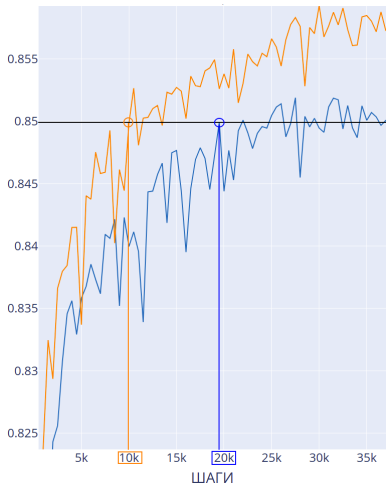
- ① сортируем данные по сложности (длина)
- ② в течение T шагов (рассмотрим t -й шаг)
 - вычисляем $c(t) \in [0, 1]$
 - формируем пакет данных маленького размера из множества $c(t)$ **легких** примеров
 - шаг обучения



Platanios et al., Competence-based Curriculum Learning for Neural Machine Translation, 2019

Метод сравнения алгоритмов обучения

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



метрика	классификация	перевод	предобучение	NLU
длина		✓		
языковая ¹				
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)
- большинство работ улучшают **качество** модели, но не **скорость** обучения
- не рассмотрен случай с **шумными** тренировочными данными

¹Sluis et al. (2010) показали слабую корреляцию с реальной сложностью [текста](#)

Цель: ускорить обучение языковой модели BERT с помощью обучения по плану за счет применения улучшенной метрики сложности текстовых данных на задачах классификации и предобучения

Задачи:

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

Поиск метрик

❶ база

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

❷ информационный поиск

- **tf-idf**

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

- EE, TSE

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



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

Асимптотика	Время
$\mathcal{O}^*(2^n), \mathcal{O}(n^2)$	> 1 мес.
$\mathcal{O}(n)$	< 4ч.

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

❹ модельная (MLM-loss)

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

❺ среднее число токенов в слове (TPW)

- **статистики**

- ❶ длина \rightarrow число текстов с такой длиной
 - ❷ $(i, x_i) \rightarrow$ число текстов, где $t_i = x_i$
 - ❸ $(x_i) \rightarrow$ число текстов, где x_i является последним токеном
 - ❹ $(i, x_{i-1}, x_i) \rightarrow$ число текстов, где на $(i-1)$ -й позиции стоит x_{i-1} , а на i -й позиции стоит x_i
 - ❺ $x_i \rightarrow$ число текстов, в которых есть x_i
- сбор статистик в параллельном режиме (разделение по данным)

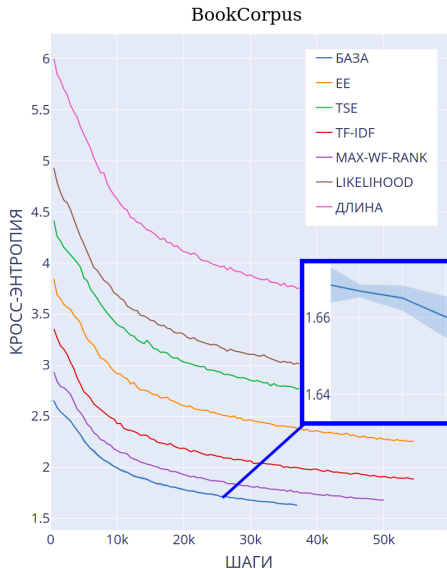
Режим	Время
1 CPU	≈ 2 недели
5 CPU	$\approx 1-2$ дня
20 CPU	< 6 ч.

Итого:

- предложены подходы, покрывающие широкое множество метрик
- предложены алгоритмы, вычисляющие метрики за пренебрежимо маленькое время ($< 8\%$ от времени обучения)

Сравнение метрик. Предобучение

- Данные:
Wikipedia, BookCorpus
- Обучение по плану проигрывает базовому решению от 2 до 5 раз
- 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 от 2 до 12 раз



Сравнение метрик. Предобучение

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

Корпус данных	BookCorpus (2.00)					
семплер	CB	DB	Hyp	SS	SM	min loss
max wf rk	∞	17.5k	16.5k	16.5k	27k	1.58
TF-IDF	∞	34k	35k	37.5k	∞	1.84
база	9.5k					1.58
	BookCorpus (3.50)					
EE	∞	4k	3.5k	4.5k	9.5k	2.25
TSE	∞	9k	9k	8.5k	18k	2.60
правдоподобие	∞	13.5k	13.5k	15.5k	50k	2.83
длина	∞	50.5k	∞	-	-	3.45

- (\pm) лучшая метрика – максимальный ранг слова ($(-)$ замедляет в 2 раза $(+)$ без потери качества)
- $(-)$ обучение по плану замедляет обучение от 2 до 5 раз и ухудшает качество модели

Сравнение метрик. Классификация

$\max \Delta \leq 3k$

Корпус данных	sentiment140 (85.5%)					
семплер	CB	DB	Hyp	SS	SM	Точность
длина	112.5k	20k	19k	-	-	86.2%
TF-IDF	115.5k	21.5k	19.5k	16.5k	22k	86.7%
TSE	95.5k	16.5k	20.5k	21.5k	18k	86.8%
EE	59k	19.3k	23k	20k	19k	86.7%
max wf rk	70k	18.5k	19.5k	17k	19k	86.7%
правдоподобие	112k	17.5k	21.5k	17.5k	21.5k	86.7%
MLM-loss	59.5k	21k	23.5k	?	?	86.1%
база	17.5k					87%

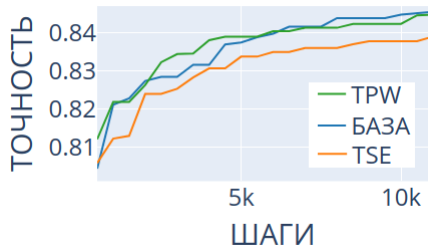
- (+) лучшая конфигурация (TF-IDF+SS) ускоряет обучение до 3% в среднем
- (-) длина и MLM-loss уменьшают точность модели на 0.6%
- (-) нет значительного ускорения обучения

Влияние метрик на скорость обучения. Шум

- $p \sim U[0, 0.4]$ – уровень шума (доля слов с шумом)
 - **искусственная метрика**
- виды шума:
 - 1 вставки
 - 2 удаления
 - 3 **клавиатурный**



метрика "уровень шума" ускоряет обучение на старте до 2.5 раз



метрика TPW ускоряет обучение на старте до 2 раз

Влияние метрик на скорость обучения. Шум

- Корреляция Спирмена
- Свойство метрики TPW: сортирует корпус данных по уровню шума
- Модель сначала учит чистые примеры, плавно переходя к шумным
- (+) получили ускорение обучения в **2 раза** на первых 10% всего обучения на шумном корпусе данных

- 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 ($\leq 8\%$ от времени обучения)
- ❸ Предобучение
 - (+) есть строгий порядок метрик
 - (−) длина – худшая метрика на предобучении (замедляет обучения до 12 раз, уменьшает качество модели)
 - максимальный ранг слова – лучшая метрика на предобучении ((−)замедляет в 2 раза (+) **без потери качества**)
- ❹ Классификация
 - (+) лучшая конфигурация (TF-IDF+SS) ускоряет обучение до 3% в среднем на классификации
 - (−) длина и MLM-loss уменьшают точность модели на 0.6%
 - (+) метрика TPW ускоряет обучение до **2 раз** на первых 10% всего обучения на шумном корпусе данных
 - (+) Выявлено фильтрующее свойство TPW

- Ay, N., Olbrich, E., Bertschinger, N., & Jost, J. (2006, August). A unifying framework for complexity measures of finite systems. In Proceedings of ECCS (Vol. 6).
- Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009, June). Curriculum learning. In Proceedings of the 26th annual international conference on machine learning (pp. 41-48).
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. arXiv preprint arXiv:2005.14165.
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

- Hacoen, G., & Weinshall, D. (2019, May). On the power of curriculum learning in training deep networks. In International Conference on Machine Learning (pp. 2535-2544). PMLR.
- Kocmi, T., & Bojar, O. (2017). Curriculum learning and minibatch bucketing in neural machine translation. arXiv preprint arXiv:1707.09533.
- Kurdi, M. Z. (2020). Text Complexity Classification Based on Linguistic Information: Application to Intelligent Tutoring of ESL. arXiv preprint arXiv:2001.01863.
- Mermer, M. N., & Amasyali, M. F. (2017). Scalable Curriculum Learning for Artificial Neural Networks. IPSI BGD TRANSACTIONS ON INTERNET RESEARCH, 13(2).

- Narasimhan, S., Narasimhan, V. A. P. B. S., Karch, G., Rao, R., Huang, J., Zhang, Y., Ginsburg, B., Chitale, P., Sreenivas, S., Mandava, S., Ginsburg, B., Forster, C., Mani, R., & Kersten, K. (2020, October 13). NVIDIA Clocks World's Fastest BERT Training Time and Largest Transformer Based Model, Paving Path For Advanced Conversational AI. NVIDIA Developer Blog. <https://developer.nvidia.com/blog/training-bert-with-gpus/>
- Platanios, E. A., Stretcu, O., Neubig, G., Poczos, B., & Mitchell, T. M. (2019). Competence-based curriculum learning for neural machine translation. arXiv preprint arXiv:1903.09848.
- Sajjad, H., Dalvi, F., Durrani, N., & Nakov, P. (2020). Poor Man's BERT: Smaller and Faster Transformer Models. arXiv preprint arXiv:2004.03844.

- Shen, S., Dong, Z., Ye, J., Ma, L., Yao, Z., Gholami, A., Mahoney, M. W., & Keutzer, K. (2020). Q-BERT: Hessian Based Ultra Low Precision Quantization of BERT. Proceedings of the AAAI Conference on Artificial Intelligence, 34(05), 8815–8821. <https://doi.org/10.1609/aaai.v34i05.6409>
- van der Sluis, F., & van den Broek, E. L. (2010, August). Using complexity measures in information retrieval. In Proceedings of the third symposium on information interaction in context (pp. 383-388).
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. arXiv preprint arXiv:1706.03762.

- Xu, B., Zhang, L., Mao, Z., Wang, Q., Xie, H., & Zhang, Y. (2020). Curriculum Learning for Natural Language Understanding. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 6095–6104.
<https://doi.org/10.18653/v1/2020.acl-main.542>
- Zhang, X., Kumar, G., Khayrallah, H., Murray, K., Gwinnup, J., Martindale, M. J., ... & Carpuat, M. (2018). An empirical exploration of curriculum learning for neural machine translation. arXiv preprint arXiv:1811.00739.

Дополнительно: Поиск метрик

метрика	формула
Мультиинформация	$\sum_{v \in V} H_p(X_v) - H_p(X_V)$
Избыточная энтропия (ЕЕ)	$\left[\sum_{v \in V} H(X_{V \setminus \{v\}}) \right] - (n - 1)H(X_V)$
TSE	$\sum_{k=1}^{n-1} \frac{k}{n} C^{(k)}(X_V), \text{ где}$ $C^{(k)}(X_V) =$ $\frac{n}{k \binom{n}{k}} \sum_{A \subseteq V, A =k} H(X_A) - H(X_V)$
Переходная информация	:(

$$V = \{1, \dots, n\}, X_V = (X_1, \dots, X_n)$$

Nihat Ay et al., A **Unifying** Framework for Complexity Measures of Finite Systems, 2006

Дополнительно: Адаптация ЕЕ и TSE под задачи обработки языка

1 Образование совместной случайной величины

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

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

↓

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

2 Вычисление энтропии

$$H(\mu) = \sum_{i=1}^n H(\mu_i | \mu_1, \mu_2, \dots, \mu_{i-1}) = \sum_{i=1}^n H(\mu_i | \mu_{i-L}, \dots, \mu_{i-1})$$

3 $L = 1$

$$H(\mu) = H(\mu_1) + H(\mu_2 | \mu_1) + \dots + H(\mu_i | \mu_{i-1}) + \dots + H(\mu_n | \mu_{n-1})$$

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

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

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

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

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

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

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

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