АНАЛИЗ ТОНАЛЬНОСТИ ТЕКСТОВ С ПОМОЩЬЮ НЕЙРОННЫХ СЕТЕЙ, ИСПОЛЬЗУЮЩИХ МЕХАНИЗМЫ ВНИМАНИЯ

А.В. Шакель

Белорусский государственный университет, г. Минск;
shakel2000@mail.ru;
науч. рук. – В.И. Отлига, ст. преп.

Задача анализа тональности текстов является важной задачей из сферы автоматической обработки текстов, благодаря ее широкой применимости в практических задачах. Существует множество различных методов и подходов к решению данной задачи, в том числе придуманных и до эры глубокого обучения. В данной работе рассматривается задача бинарной классификации текстовых отзывов о фильмах из базы данных IMDb [1]. В первой части работы проводится сравнение производительности и результатов рекуррентных нейронных сетей с аналогичными архитектурами с использованием механизмов внимания. Во второй части проведено исследование влияния различных приемов при дообучении языковой модели BERT на значения метрик и время обучения сетей.

Ключевые слова: анализ тональности текстов, обработка естественного языка, рекуррентные нейронные сети, механизмы внимания, дообучение модели BERT.

ВВЕДЕНИЕ

В настоящее время, благодаря развитости интернета и огромному числу пользователей по всему миру, ежедневный поток текстовых данных в сети составляет огромные объемы. Очень часто подобная информация - это прямое выражение предпочтений и интересов пользователей посредством естественного языка. Именно поэтому в эпоху больших данных возможность извлекать информацию из текстов без привлечения человеческого труда чрезвычайно важна.

Немалую роль в извлечении информации из текстовых данных несет *анализ тональности*, под которым понимается выявление в тексте эмоционально окрашенной лексики и эмоциональной оценки автором объектов, о которых идет речь в тексте. Для решения многих задач обработки естественного языка, в том числе и для задачи анализа тональности текстов, достаточно часто применяются *рекуррентные нейронные сети.* Тем не менее у таких сетей есть важный недостаток эффект *забывания*, при котором рекуррентная нейронная сеть постепенно теряет информацию из-за последовательной обработки входных данных. Такие существующие модификации рекуррентных нейронных сетей, как *LSTM* (Long Short-Term Memory) и *GRU* (Gated Recurrent Unit), избавлены от этой проблемы лишь частично.

МЕХАНИЗМЫ ВНИМАНИЯ

Для повышения качества работы таких рекуррентных нейронных сетей, как LSTM и GRU, хорошо подходит использования *механизмов внимания* [2]. При классическом использовании рекуррентных нейронных сетей на вход полносвязного нейросетевого классификатора подается их последний вектор-состояние. Используя механизмы внимания можно модифицировать этот вектор-состояние, использовав его как вектор-запрос и посчитав функцию схожести этого вектора со всеми остальными векторами состояний сети. Новый вектор-состояние образуется как линейная комбинация всех состояний сети, взвешенными с коэффициентами, получаемыми с помощью операции *softmax* для посчитанных значений функции схожести. В качестве функции схожести используется скалярное произведение векторов.

Рассматриваемые при сравнении моделей метрики качества – *правильность* (accuracy), *точность* (precision), *полнота* (recall) и *площадь под roc-кривой* (roc-auc). Основная метрика – правильность.

В таблице 1 указаны результаты значений метрик для рекуррентных нейронных архитектур на базе LSTM и GRU, а также их модификаций с использованием механизмов внимания.

Таблица 1

Значения метрик до и после применения механизмов внимания (%)

Модель	Правильность	Точность	Полнота	Площадь под roc-кривой
LSTM	85.2	87.9	81.7	92.8
LSTM + внимание	88.9	86.3	92.4	95.5
GRU	85.0	90.7	78.1	93.5
GRU + внимание	88.8	89.7	87.7	95.1

В таблице 2 приведены временные затраты на обучение аналогичных моделей до сходимости и число потребовавшихся эпох.

Таблица 2

Временные затраты на обучение моделей до и после применения механизмов внимания

Модель	Число эпох	Общее время (мм:сс)
LSTM	5	04:15
LSTM + внимание	2	04:21
GRU	4	03:06
GRU + внимание	5	11:45

ДООБУЧЕНИЕ ЯЗЫКОВОЙ МОДЕЛИ BERT

BERT (Bidirectional Encoder Representations from Transformers)[4] – языковая модель с *трансформерной* [3] архитектурой, с помощью которой можно

эффективно решать различные задачи обработки естественного языка. Для задач классификации у BERT есть специальный токен [CLS], которого после прохождения сети BERT используется в качестве входа для полносвязного нейросетевого классификатора. В данной статье используется модель BERT base, состоящая из слоя представлений и 12 слоев трансформеров.

Обучение во всех экспериментах проводится до сходимости, в течение 1-4 эпох, используемый *оптимизатор* – AdamW с *темпом обучения* $2 \cdot 10^{-5}$. Также исследованы различные подходы при дообучении:

- Разморозка различного числа слоев [5] – часть предобученной модели BERT остается неизменной и обучается только ее часть. Проведено сравнение разморозки только классификатора, классификатора и последнего трансформера, классификатора и 6 последних трансформеров, всей сети
- Использование различной максимальной длины текста – BERT поддерживает максимальную длину текста до 512, проведено сравнение результатов для длин 200 и 512.
- Использование линейного планировщика темпа обучения [6, 7] – темп обучения сначала линейно возрастает в течение 10% всех итераций обучения до стандартного значения, а затем линейно убывает в течение оставшихся итераций обучения.
- Дискриминативное дообучение [6, 7] – использование на данном слою сети тем меньшего темпа обучения, чем глубже данный слой в сети. С каждым логическим слоем темп обучения умножается на 0,95.

В таблице 3 приведены значения метрик для всех обученных моделей. В скобках у каждой модели обозначения следующие: 200/512 – максимальная длина текстов, cls/cls+tr/cls+tr6/full – один из четырех способов разморозки соответственно, scheduler – использование планировщика темпа обучения, discr – использование дискриминативного дообучения).

Таблица 3

Значения метрик для дообученных моделей BERT (%)

Модель	Правильность	Точность	Полнота	Площадь под roc-кривой
BERT (200 + cls)	82.8	82.8	82.7	90.7
BERT (200 + cls+tr)	89.6	89.4	89.9	96.2
BERT (200 + cls+tr6)	90.9	92.7	88.9	97.1
BERT (200 + full)	91.0	91.6	90.3	97.1
BERT (512 + full)	93.7	93.3	94.1	98.3
BERT (512 + full + scheduler)	94.0	93.1	95.0	98.4
BERT (512 + full + scheduler + discr)	94.1	93.7	94.5	98.5

В таблице 4 приведены временные затраты на обучения моделей BERT.

Таблица 4

Временные затраты для дообученных моделей BERT

Модель	Число эпох	Общее время (чч:мм:сс)
BERT (200 + cls)	4	00:21:52
BERT (200 + cls+tr)	4	00:24:13
BERT (200 + cls+tr6)	2	00:19:29
BERT (200 + full)	1	00:14:34
BERT (512 + full)	1	01:09:50
BERT (512 + full + scheduler)	2	01:21:20*
BERT (512 + full + scheduler + discr)	2	02:18:55

Примечание. Модель, помеченная символом *, была обучена на более мощной GPU, время обучения на аналогичной другим моделям GPU должно незначительно отличаться от времени обучения последней модели.

ЗАКЛЮЧЕНИЕ

Исходя из таблицы 1, использование механизмов внимания для обеих рекуррентных архитектур дает существенный прирост по совокупности метрик, снижая лишь значение полноты. Значительный прирост получает метрика правильность. При этом время, затрачиваемое на одну эпоху, значительно возрастает при использовании механизмов внимания, но может также повлечь более быструю сходимость, как в случае с архитектурой LSTM, за счет чего время обучения вырастет незначительно.

Дообучение языковой модели BERT позволяет достичь очень высоких показателей по всем метрикам. Для максимального качества эффективно размораживать и обучать всю сеть целиком, но даже с разморозкой одного трансформера можно получить достаточно высокое качество. Значительный прирост качества дает использование максимальной длины текстов равной 512, однако в таком случае обучение длится в несколько раз дольше, чем для длины 200. Использование дополнительных приемов при дообучении также положительно влияет на значения метрик и позволяет получить итоговое значение правильности в 94,1%.

Библиографические ссылки

1. Maas, Andrew L. and Daly, Raymond E. and Pham, Peter T. and Huang, Dan and Ng, Andrew Y. and Potts, Christopher Learning Word Vectors for Sentiment Analysis - 2011 - С.142-150.

2. D Bahdanau, K Cho, Y Bengio: Neural machine translation by jointly learning to align and translate //arXiv preprint arXiv:1409.0473 – 2014.
3. Vaswani A. et al. Attention is all you need //arXiv preprint arXiv:1706.03762. – 2017.
4. Devlin J. et al. Bert: Pre-training of deep bidirectional transformers for language understanding //arXiv preprint arXiv:1810.04805. – 2018.
5. Lee J., Tang R., Lin J. What would elsa do? freezing layers during transformer fine-tuning //arXiv preprint arXiv:1911.03090. – 2019.
6. Howard J., Ruder S. Universal language model fine-tuning for text classification //arXiv preprint arXiv:1801.06146. – 2018.
7. Sun C. et al. How to fine-tune BERT for text classification? //China National Conference on Chinese Computational Linguistics. – Springer, Cham, 2019. – С. 194-206.