Лабораторная 4 - text

github classroom task
Порядок сдачи и защиты лабы описан тут
Дедлайн указан в таблице с баллами

Данные

Соберите корпус текстов

Можно использовать датасет из 1 лабы, если там есть тексты и таргет для них. Можете спарсить доп данные с сайта. Либо можете выбрать другой сайт и собрать интересные текстовые данные, в том числе через api.

Идеи датасетов

Использовать именно эти - уже не так интересно. Но для вдохновения подойдет

- комментарии/посты в соц сети
  Внутри каждой соцсети бесконечно много сообществ, внутри каждого есть дихотомия, которую можно научиться выделять автоматически. Например склонность к какой-то позиции: плоскости земли/поддержки общественного движения/отношению к определенному хобби/etc. Или наличие свойства в тексте: токсичность/эмоциональность (от негативной до позитивной)/etc. Можно выделять какие слова склонили детектор в сторону токсичности например. Или соответствие текста нормам конфуцианской морали
- языковая модель текстов видео/песен
  то же, что про комменты. Ютуб генерирует субтитры автоматически, тексты песен давно разобраны
- отзывы о чем угодно
  можно кластеризовать отзывы об инфо цыганских курсах и понять что в них волнует людей больше всего. Можно создать статистику про отзывы из delivery club про любимый ресторан. Можете исследовать отзывы о любимой игре в steam или сообщения из чатов во время матча
- твиты
  можно исследовать как изменилось общественное мнение о любом вопросе: скачать твиты по ключевому слову и изучить sentiment/кластеризацию
- описания
  можно по тексту об одежде предсказать ее рейтинг или по описанию или количество лайков на посте в паблике, где несколько тысяч постов

Задача

Можете взять любую NLP задачу:

- Просто
  - классификация
  - регрессия
  - выделение эмоций (sentiment/toxic detection/etc)

- Средне:
    - моделирование языка
    - генерация
    - machine translation
    - topic modelling
- Сложно:
    - ответы на вопросы
    - диалоговые модели
    - суммаризация

Балл за лабу не зависит от задачи

Задание

1. Соберите датасет текстов
2. (если в данных не было таргета) Создайте таргет
   Для классификации это метки классов. Для диалоговой модели - ответы на ввод пользователя. Используйте LLM. Напишите подробный промпт с описанием задачи. Разметьте весь датасет или достаточную для обучения модели часть
3. Реализуйте текстовую модель
   Это может быть простая DL модель: nn.Embedding + nn.LSTM + nn.Linear слои, аналогичная разбиралась на лекции. Слоев может быть больше, если так реализуете. Может быть сложная предобученная модель, в устройстве которой вы разобрались + nn.Linear слои. Может быть и любая другая **нейросетевая** архитектура: seq2seq/transformer/etc
4. Обучите модель на размеченном датасете
5. Измерьте качество модели
   а) Посчитайте метрики на train и test
   б) (опционально, но интересно) создайте эмбеддинги текстов: используйте выход предпоследнего Linear слоя. Следующий слой модели предсказывает по этому эмбеддингу ответ, поэтому в эмбеддинге на этом слое собрана вся полезная информация. Используйте метод снижения размерности, чтобы отобразить тексты на 2д плоскости. Цветом укажите целевую переменную
   в) (опционально, но полезно) используйте tensorboard для логирования метрик во время обучения модели

Рекомендации

- **не стесняйтесь задавать вопросы в чат курса**
- рекомендуемый размер датасета - больше 1000 текстов. Если будет меньше, то алгоритм может плохо обучиться

Не обязательны, но рекомендуются python библиотеки:

- pytorch
- tensorflow
- youtokentome для byte-pair encoding

Популярные замечания

- используйте конфиг файл для констант
- не коммитить .idea
- функции длиннее 20 строк - плохо. Их можно декомпозировать
- не показано, что модель не переобучена. Сравните метрики на train и test
- в модели нет dropout/batchnorm слоев. Она быстро переобучается из-за этого
- слишком большой размер словаря (>10k) из-за чего модель крайне медленно учится. Используйте byte-pair encoding
- В архитектуре Embedding + Lstm + Linear у последней части всего один линейный слой. Этого мало для качественного извлечения признаков из результатов обработки текста. Добавьте несколько слоев, перемежая с активациями и dropout, как показано на лекции про feed-forward сети

Теория

В каждом вопросе про метрики, формулы и тп нужно знать как это рассчитывается и почему формулы именно такие. Зубрить сами формулы не нужно

В вопросах про модели подразумевается: Как устроена, как обучается, какие гиперпараметры, какие требования к данным и к чему уязвима, какие сценарии применения

Преподаватель может задать доп вопрос об использованном вами термине, чтобы понять, что вы понимаете его смысл

1. Что такое эмбеддинг? Является ли строка датасета эмбеддингом? является ли выход линейного слоя нейросети эмбеддингом? Объяснимость и интерпретируемость моделей - в чем разница?
2. Проблемы кодирования слов: one-hot, stop-слова, формы слов, n-gramms
3. Модель bag of words. Чем cbow отличается от skip-gram
4. Модель tf-idf. Какого размера вектор возвращает, какие данные хранит, как обучается, в чем минусы по сравнению с RNN
5. Что такое корпус, что такое словарь, как размер словаря влияет на алгоритмы. Как уменьшить размер словаря, на что это повлияет
6. Что такое byte pair encoding и как он сокращает размер словаря до минимума, но все равно способен кодировать даже те слова, которые не видел при сборке словаря
7. Что такое рекуррентная ячейка (RNN), почему обрабатывает последовательности любой длины, сколько рекуррентных ячеек в рекуррентной сети, какие векторы принимает и возвращает
8. LSTM ячейка: чем отличается от простой RNN, за что отвечают гейты, как борется со взрывом/затуханием градиентов
9. Как с помощью RNN генерировать текст слово за словом
10. Модель (тип моделей) sequence2sequence. Какие задачи может решать, из чего состоят encoder и decoder
11. Attention. Зачем нужен, геометрический смысл

12. Attention. Как получается вес(=важность) эмбеддинга слова по отношению к другому слову?
13. Attention. Меняется ли размер эмбеддинга слова после применения attention-слоя? Меняется ли сам вектор? Как?
14. Как устроена модель GPT-3
15. Какие сейчас существуют LLM? Чем LLM отличается от простых языковых моделей? Сколько весов(=параметров) у GPT-3.5 и у GPT-4? Что означает размер контекста у LLM?
16. Beam search. Зачем нужен, как работает
17. Метрики BLEU (Bilingual Evaluation Understudy), Perplexity, WER (Word Error Rate). Зачем нужны и как считаются

FAQ

1. Что из pytorch использовать
   torch.nn.Embeddings для кодирования слов, torch.nn.LSTM для обработки последовательности эмбеддингов, torch.nn.Linear для предсказания слова/класса/etc (зависит от задачи). Не забудьте про слои активации, batchnorm, dropout

2. Как предобработать текст
   Зависит от токенизатора и задачи. Рекомендуется использовать bpe-токенизацию из youtokentome. При ней токены (=строки) в разном регистре будут считаться разными, поэтому стоит привести к lowercase. Знаки препинания эта библиотека отделит даже если они вплотную к слову. Если токенизация - это деление по пробелу, то токен с восклицательным знаком (без пробела) выделяется как отдельное слово, из-за чего словарь разрастается напрасно