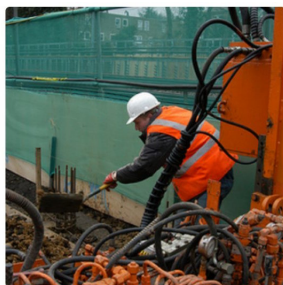


Задание 4. Генерация Описаний для Изображений

Курс: Глубинное обучение, осень 2016



"man in black shirt is playing guitar."



"construction worker in orange safety vest is working on road."



"two young girls are playing with lego toy."



"boy is doing backflip on wakeboard."

Начало выполнения задания: ? декабря

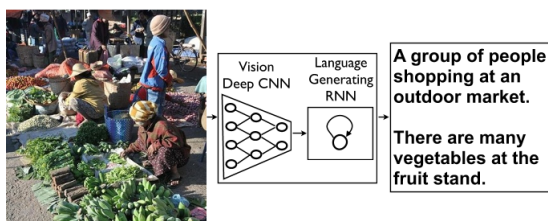
Срок сдачи: ? декабря, 23:59.

Среда для выполнения задания – PYTHON 2.x.

Содержание

| | | |
|---|------------------------|---|
| 1 | Заготовка решения | 1 |
| 2 | Задание | 1 |
| 3 | Вычислительные ресурсы | 2 |
| 4 | Оформление задания | 2 |
| | Список литературы | 3 |

1 Заготовка решения



На семинаре была разобрана заготовка кода, для аннотирования изображений, код доступен по ссылке https://github.com/ars-ashuha/caption_binder. Это пример минимально работающего кода, для генерации описаний. Домашнее задание предполагает лучение выданного кода, но можете и написать всё сами.

2 Задание

Минимально рабочая версия есть, но довести модель до хорошего качества вам предстоит в течении домашнего задания, а проведение экспериментов вам в этом поможет, придется покопаться. Для реализации необходимо использовать библиотеку lasagne.

- Для обучения могут быть использованы выборки COCO, Flickr8k, Flickr30k.

- В качестве метрики качество используйте BLEU-1 score, пример реализации `nltk.align.bleu`.
- Обязательно используйте в сети не детерминированные преобразования, чтобы не генерировать тождественную аннотацию для картинки.
- Используйте для получение дескриптора картинки сеть ResNet, как меняется скорость сходимости, loss модели, качество подписывания? Аккуратно следите за тем, что Вы правильно загрузили сеть, а так-же правильно пред-обрабатываете изображения. Как проверить, что сеть работает верно.
- Инициализируйте Embedding слой векторами слов из модели word2vec, как меняется скорость сходимости, loss модели, качество подписывания?
- Проведите сравнение нескольких различных рекуррентных архитектур, как меняется скорость сходимости, loss модели, качество подписывания? Аккуратно подбирайте метод оптимизации и его параметры.
- Приведите 10+ хороших/плохих аннотаций полученных от вашей сети.
- Реализуйте визуализацию короткого рассказа (10-15 предложений) картинками из обучающей выборки. Опишите ваш метод и приведите пример.

Помимо ipython ноутбука с кодом экспериментов, вам необходимо прислать файл `ic-model.py` в котором реализована функция `load_model` возвращающая функцию аннотирования картинок и размер входного изображения.

```
def load_model(path_to_model):
    """
    path_to_model: path to .pkl file with stored weights
    return:
    - callable function annotate
      - annotate(input_image) = caption
      - annotete should do image prossesing and eval deep features
    - input_image.shape (channels, width, height)
    """
    annotate = lambda x: "empty annotation is here"
    return annotate, (3, 100, 200)
```

При написании отчета, стоит приводить графики loss и качества против числа итераций, на одном графике, для разных моделей, чтобы была возможность сравнить несколько моделей. Так-же не забывайте делать выводы на основе экспериментов.

3 Вычислительные ресурсы

У Amazone есть программа [AWS Educate](#), если Вы студент ВУЗа партнера Amazone (а МГУ как-раз такой), то вам дают грант 100\$ которого в итоге хватает на 100 часов сервера с GPU (`g2.2xlarge`). Мы советуем арендовать сервера в режиме Spot Instances <https://aws.amazon.com/ec2/pricing/>. При выборе системы для установки на сервер, Вы можете найти систему с заранее установленными драйверами и библиотеками theano и lasagne.

4 Оформление задания

Выполненное задание следует отправить письмом по адресу bayesml@gmail.com с заголовком письма

«[ВМК ГО16] Задание 4, Фамилия Имя».

Убедительная просьба присылать выполненное задание только один раз с окончательным вариантом. Также убедительная просьба придерживаться заданной выше спецификации.

Присланный вариант задания должен содержать в себе:

- Текстовый файл в формате PDF с указанием ФИО, содержащий описание всех проведённых исследований.

- Все исходные коды с необходимыми комментариями.
- Ссылку на обученную модель и изображения из примеров, функция `annotate` должна генерировать описания примерно того-же качества.

Список литературы

- [1] Andrej Karpathy, Li Fei-Fei: Deep Visual-Semantic Alignments for Generating Image Descriptions, CVPR 2015
- [2] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, Yoshua Bengio, Show, Attend and Tell: Neural Image Caption Generation with Visual Attention
- [3] Oriol Vinyals, Alexander Toshev, Samy Bengio, Dumitru Erhan Show and Tell: A Neural Image Caption Generator