

КЛАССИФИКАЦИЯ ТЕКСТА НА ИЗОБРАЖЕНИЯХ РЕАЛЬНЫХ СЦЕН ПРИ ПОМОЩИ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ

Е.А. Исаев, Е.В. Михайлова
Ю.А. Иванова
Томский политехнический университет
elena.mikky@gmail.com

Введение

В настоящее время в условиях возрастающего объема информации в электронном виде задача классификации текста, то есть определение принадлежности текста к некоторому классу является актуальной.

Классификация текста на изображениях реальных сцен может применяться в задачах автоматической аннотации, поиска информации и навигации.

Автоматическая аннотация изображений - процесс автоматического присвоения системой метаданных в форме заголовка или ключевых слов цифровому изображению.

Задача автоматической аннотации изображений является очень востребованной в последние годы в связи с её потенциальным влиянием на понимание изображений и веб-поиск.

Целью данной работы является настройка сверточной нейронной сети, способной производить классификацию текста на изображениях реальных сцен.

Архитектура СНС

Сверточная нейронная сеть (СНС) обычно представляет собой чередование сверточных слоев (convolution layers), субдискретизирующих слоев (subsampling layers) и при наличии полносвязных слоев (fully-connected layer) на выходе. Все три вида слоев могут чередоваться в произвольном порядке [1].

В сверточном слое нейроны, которые используют одни и те же веса, объединяются в карты признаков (feature maps), а каждый нейрон карты признаков связан с частью нейронов предыдущего слоя. При вычислении сети получается, что каждый нейрон выполняет свертку некоторой области предыдущего слоя (определяемой множеством нейронов, связанных с данным нейроном). Пример архитектуры сверточной нейронной сети представлен на рисунке 1.

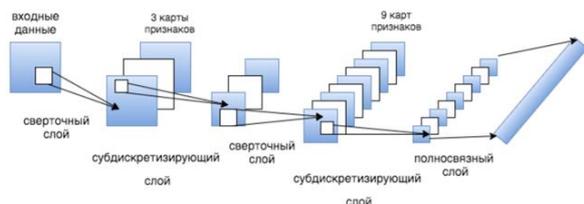


Рис. 1. Архитектура сверточной нейронной сети

Формирование исходной выборки

На изображениях реальных сцен встречается множество текстовой информации.

Для решения поставленной задачи было выделено четыре основных класса:

1. реклама – представлена надписями на различных баннерах, вывеска, рекламных щитах;
2. дорожные знаки – представлены знаками, содержащими текстовую информацию;
3. организации – представлены их названиями, встречающихся на различных вывесках и табличках;
4. улицы – представлены их названиями, находящимися на табличках домов и уличных столбах.

Эксперименты проводились на данных, сформированных вручную [2 – 4]. Текст выборки представлен на английском языке. Тексты имеют различное количество слов, варьирующееся приблизительно в пределах от 1 до 50. Общий размер выборки составляет 1600 текстов. На каждый класс приходится по 400 текстов. Выборка была разделена на тренировочную, составляющую 80% от общего объема, и тестовую – 20%.

Таким образом, на тренировочную выборку приходится 1280 текстов, а на тестовую – 320. Тексты в обоих типах выборок были перемешаны случайным образом.

Формирование модели Word2Vec и обработка выборки

Для того, чтобы представить тексты выборки в цифровом виде использовался подход Word2Vec [5]. Для создания модели Word2Vec была использована реализация технологии из библиотеки Gensim на языке программирования Python [6]. Параметры обучения представлены ниже:

- Архитектура – Skip-gram. Данный алгоритм работает медленнее, чем CBOW, однако лучше работает для относительно редких слов, что является преимуществом, т.к. уникальных слов в выборке, используемой в работе, большое количество.

- Функция активации – Иерархический softmax. Данная функция хорошо подходит для создания модели относительно редких слов. Негативное сэмпирование лучше моделирует более частотные слова.

- Снижение количества часто встречающихся слов: 0.001.

- Размер векторов слов – 300.

- Контекст (как много слов из контекста обучающий алгоритм должен принимать во внимание) – 10.

•Минимальное количество слов (слова, количество появлений которых меньше заданного игнорируется и не добавляется в словарь) – 1.

Единица была выбрана так как в выборке встречаются тексты из одного слова.

На основе имеющихся данных была получена модель, размер словаря которой составил 2443 слова, где каждое слово является вектором размера 300.

Обучение и тестирование

Сверточная нейронная сеть была реализована на языке Python с использованием библиотеки TensorFlow [7].

В ходе экспериментов наилучший результат классификации текстов дала архитектура сверточной нейронной сети, описанная в таблице 1.

На вход нейронной сети на каждой итерации подавалась матрица размера 64×300. Через каждые 20 итераций производилось тестирование, при котором подавалась вся тестовая выборка, размер матрицы которой 240×300. Всего итераций было произведено 2000, что соответствует 100 эпохам обучения. Первоначально веса нейронной сети инициализируются из нормального распределения.

Функция активации на всех слоях кроме последнего – ReLU, на последнем – Softmax. Оптимизатор градиентного спуска – AdamOptimizer, скорость обучения – 0,01.

Таблица 1. Архитектура сверточной нейронной сети

№ слоя	Количество нейронов	Размер ядра свертки	Размер ядра Max-pooling
1	16	5	2
2	16	5	2
3	256	–	–
4	4	–	–

Результаты работы СНС

Результат обучения и тестирования сверточной нейронной сети показан на рисунке 2. Сплошная линия характеризует точность классификации, прерывистая – функцию ошибки, которая минимизируется в процессе обучения.

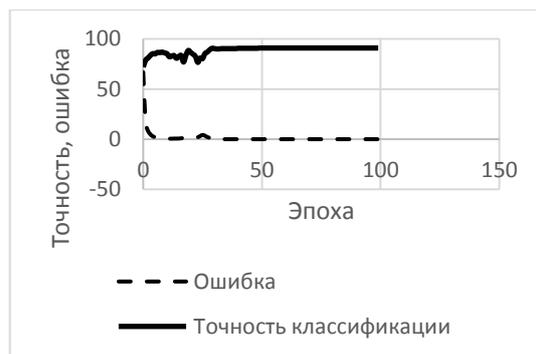


Рис. 2. Динамика точности классификации текстов

Из графика видно, что максимальная точность классификации текста достигла 91%. После 40 эпох сеть перестала обучаться, вследствие предельно малого значения функции ошибки.

Список использованных источников

1. Рысьмятова А.А. Использование сверточных нейронных сетей для задачи классификации текстов – Москва, 2016. – 34 с.
2. Road signs in the United States [Электронный ресурс] / Википедия. – Режим доступа: https://en.wikipedia.org/wiki/Road_signs_in_the_United_States, свободный (дата обращения 17.04.2018).
3. List of Companies [Электронный ресурс] / U.S. securities and exchange commission. – Режим доступа: <https://www.sec.gov/rules/other/4-460list.htm> (дата обращения: 17.04.2018).
4. OpenAddresses data [Электронный ресурс] / OpenAddresses. – Режим доступа: <https://results.openaddresses.io>, свободный (дата обращения: 17.04.2018).
5. Word2Vec [Электронный ресурс] / Википедия. – Режим доступа: <http://ru.wikipedia.org/wiki/Word2Vec>, свободный (дата обращения: 17.04.2018).
6. Немного про word2Vec: полезная теория [Электронный ресурс] / NLPx – Tales of Data Science. – Режим доступа: <http://nlpx.net/language/ru/>, свободный (дата обращения: 17.04.2018).
7. TensorFlow API Documentation [Электронный ресурс]. – Режим доступа: https://www.tensorflow.org/api_docs/, свободный (дата обращения: 17.04.2018).

Работа выполнена в рамках Программы повышения конкурентоспособности ТПУ и при финансовой поддержке РФФИ в рамках научного проекта № 18-08-00977 А