Представление новостных сюжетов с помощью событийных фотографий

© М.М. Постников © Б.В. Добров

Московский государственный университет имени М.В. Ломоносова
факультет вычислительной математики и кибернетики,
Москва, Россия

mihanlg@yandex.ru dobrov_bv@mail.ru

Аннотация. Рассмотрена задача аннотирования новостного сюжета изображениями, ассоциированными с конкретными текстами сюжета. Введено понятие «событийной фотографии», содержащей конкретную информацию, дополняющую текст сюжета. Для решения задачи применены нейронные сети с использованием переноса обучения (Inception v3) для специальной размеченной коллекции из 4114 изображений. Средняя точность полученных результатов составила более 94,7%.

Ключевые слова: событийная фотография, новостные иллюстрации, перенос обучения.

News Stories Representation Using Event Photos

© M.M. Postnikov © B.V. Dobrov

Lomonosov Moscow State University, Faculty of Computational Mathematics and Cybernetics,
Moscow, Russia

mihanlg@yandex.ru dobrov_bv@mail.ru

Abstract. The task of annotating a news story with images associated with specific texts is discussed in the article. The definition of "event photography" containing specific information supplementing text of a story is introduced. Neural networks (Inception v3) are used to solve a task for a special marked collection of 4114 images using the transfer learning method. The average precision of the results is more than 94.7%.

Keywords: event image, news illustration, transfer learning.

1 Введение

Распространение интернета, социальных сетей, развитие носимой электроники, внедрение хороших камер в каждый мобильный телефон – благодаря всем этим факторам, но не ограничиваясь ими, интернет становится главным источником новостей для современного человека, в то время как телевидение и печатные средства массовой информации постепенно уходят на второй план. С развитием технологий растут и скорость распространения информации, и ее количество.

Иллюстрации несут значительную часть информации, иногда даже большую, чем иллюстрируемый текст.

В данной работе рассмотрена задача определения изображений, «полезных» для понимания новостных сообщений, иллюстрируемых ими. Как известно, новостные сообщения прежде всего содержат информацию о некотором событии и должны отвечать на вопрос: «Что произошло?».

Рисунок 1 Пример несобытийной (слева) и событийной (справа) фотографии для новости с заголовком «в Краснодаре ГК СКИФ победил ставропольское «Динамо-Виктор» – 28:23»

Для ответа на общий вопрос обычно требуется ответить на совокупность частных вопросов: «Кто? Где? Когда? Каким образом?» и т. д. Важно научиться отличать полезные изображения от тех, которые не несут важной конкретной информации (см. Рис. 1).

В рамках данной работы *событийной фотографией* будем называть изображение, используемое для иллюстрации новости, для которого выполняются следующие требования:

а) изображение соответствует тексту новостной статьи;

б) изображение является фотографией с места событий или могло бы ей быть (подразумевается событие, описываемое в новостной статье).

Труды XIX Международной конференции «Аналитика и управление данными в областях с интенсивным использованием данных» (DAMDID/ RCDL'2017), Москва, Россия, 10–13 октября 2017 года

Например, фотографии с футбольного матча, места происшествия или встречи глав государств с соответствующими новостными текстами являются событийными. Если же на фотографии изображены логотип, рекламный баннер, вывеска, изображение взято из фотобанка или фотография не соответствует новости, тогда данная иллюстрация не будет считаться событийной.

Задача является актуальной при создании новостных агрегаторов по большому количеству источников.

Большой интерес для исследования представляют социальные сети – огромное количество свидетельств очевидцев в первую очередь публикуется в социальной сети, а затем уже может найти свое отражение в СМИ.

Идея работы состоит в том, чтобы попробовать выделить ключевые объекты на изображении, сопоставить их с текстом и достичь желаемого результата. Для распознавания объектов на изображении использованы сверточные нейронные сети.

В последнее время сверточные нейронные сети получили широкое распространение в обработке и классификации изображений, благодаря чему началась активная разработка фреймворков для удобной работы с ними (tensorflow [15], theano [17], keras [2] и др.), что снизило порог входа для применения данных технологий. Но обучение сложных моделей все еще отнимает значительное количество времени и средств. Например, обучать большую нейронную сеть для классификации на стандартном персональном компьютере без специальных компонентов можно и неделю, и месяц, а то и больше. И даже на мощной системе это занимает значительное количество времени [7, 9, 22].

Существуют подходы, например, метод *переноса обучения* [8], позволяющие существенно снизить временные издержки. Современные нейронные сети обработки изображений являются многослойными, последующие слои комбинируют признаки, выделенные на предыдущих уровнях. Начальные слои ответственны за выделение базовых примитивов изображения, следующие слои – за выделение типовых фигур как комбинаций базовых примитивов и т. д. Соответственно, можно попробовать взять сеть, обученную до некоторого уровня на одних коллекциях изображений, и дообучить ее на собственной коллекции (более подробно см. в разделе 5).

Также в работе исследована возможность улучшения качества выделения событийного изображения к новостному сообщению с использованием текста новости.

2 Постановка задачи

Формальная постановка задачи выглядит следующим образом.

Пусть T – множество новостных текстов, I – множество изображений, а $Y=\{0, 1\}$ – конечное множество оценок. Существует неизвестная целевая зависимость – отображение $F: T \times I \to Y$, значения которой известны только на объектах конечной обучающей выборки $\Omega = T^m \times I^m \times Y^m = \{(t_1, i_1, y_1), \dots, (t_m, i_m, y_m)\}$.

Будем также считать, что задана неотрицательная целочисленная функция потерь $L(\tilde{y}, y)$, которая показывает, насколько отличается предсказанное классификатором значение $\tilde{y}$ от истинного значения.

Обучающую выборку Ω разделим на две непересекающиеся коллекции:

- тренировочную (используется для обучения модели)
$$\Omega_{train} = (t_1, i_1, y_1), \dots, (t_n, i_n, y_n)\ ;$$
- тестовую (используется для оценивания модели)
$$\Omega_{test} = (t_{n+1}, i_{n+1}, y_{n+1}), \dots, (t_{|\Omega|}, i_{|\Omega|}, y_{|\Omega|}).$$

Задача классификации состоит в нахождении функции $F^* = F(t_j, i_j)$:

$$F^* = \operatorname{argmin}_F L(F(t_j, i_j), y_j), (t_j, i_j, y_j) \in \Omega_{test},$$

которая называется *классификатором*. Значение F^* может быть вещественным из диапазона [0;1], его можно считать вероятностью того, что изображение является событийным.

Решение задачи можно рассматривать как решение следующих трех подзадач.

2.1 Детектор объектов

На этапе обработки изображения построим модель, которая принимает на вход изображение, а возвращает вектор вероятностей присутствия определенных объектов на изображении.

Рисунок 2 Пример изображения для детектирования объектов

Например, пусть на вход подается следующее изображение (см. Рис. 2). Пусть рассматривается присутствие следующих классов: *мотоцикл, автомобиль, человек, домашнее растение, велосипед, автобус, поезд, птица, лодка, лошадь, самолет, бутылка, телевизор, кресло, собака, кот, стол, кровать, корова, овца*. Результатом работы детектора может быть следующий вектор вероятностей: [0.9984, 0.4156, 0.0144, 0.006, 0.003, 0.0009, 0.0008, 0.0007, 0.0005, 0.0004, 0.0001, 0.0001, 0, …, 0]. Значение на позиции i представляет собой вероятность присутствия объекта класса i из списка. В данном случае *мотоцикл* присутствует на изображении с вероятностью 99,9%, а *автомобиль* – с вероятностью 41,6%.

2.2 Векторизация текста

Для обработки текста будем обучать модель, которая создает векторное представление новостного текста, поданного на вход классификатору.

2.3 Модель согласованности

Финальный этап в работе классификатора объединяет в себе итоги предыдущих подзадач. Модель, используемая на этом этапе, принимает на вход два вектора – вектор, полученный в результате обработки изображения, и вектор, полученный в результате обработки текста. Промежуточные представления векторов объединяются в единый вектор, а на выходе модели получается число из интервала [0;1] – вероятность того, что входное изображение является событийным для входной новостной статьи.

3 Обзор

Между изображениями и текстовой информацией, которая может быть ассоциирована изображению, существуют достаточно сложные взаимосвязи в зависимости от контекста и решаемых задач.

В настоящей статье рассматривается задача выбора лучшего изображения среди возможных для новостного текста с использованием информации об объектах, которые можно выделить на изображении, а также информации о связи текста с выделенными объектами.

Известны похожие постановки задач для решения проблем описания изображений текстом (Image Caption), поиска изображений по текстовому запросу (Visual Question Answering). Одним из направлений решения задач, возникающих в данных областях, является определение типа события, отображаемого на картинке/фотографии [1]. Для этой цели создаются коллекции изображений [21], аннотируемых либо свободным описанием несколькими экспертами, либо тегами [3].

К сожалению, существует несколько фундаментальных проблем описания изображений текстом. Имеет место существенный семантический разрыв между семантикой изображения и семантикой текста – обычно слишком много деталей опущено. Эксперты, описывающие изображения, могут по-разному их понимать, в том числе в силу разного жизненного опыта. Кто-то видит просто мужчину, а кто-то – известного актера, кто-то видит просто темный диск, а кто-то – грампластинку, и т. д. Кто-то может не описать тот или иной фрагмент изображения, так как он показался ему неинтересным. Существующие иерархии концептов для описания изображений пока существенно неполны.

В результате в работе [18] отмечено, что только 20% проанализированных авторами описаний изображений не содержат ошибок, при этом 26% описаний по мнению авторов не релевантны изображениям.

Отметим также, что часто банки изображений формируются из всех фотографий, сделанных во время события. Однако некоторые фотографии могут содержать не главные объекты, но окружение, которое, вообще говоря, не является специфичным именно для конкретного события (типа события).

В статье [1] приведены данные, что качество распознавания конкретных событий по размеченным коллекциям в настоящее время имеет следующие характерные оценки (на коллекции WIDER [21], 60 классов, 60 000 изображений): 42% корректных ответов среди первых, 60% – среди первых пяти.

Таким образом, пока нет возможности с высокой степенью уверенности опереться на методы определения типа события по изображению, использовать методы порождения описания изображений. При этом в [19] определена характеристика «важности» того или иного типа объекта для описания того или иного типа события, например, изображение местных достопримечательностей для альбома о путешествии или изображения невесты и жениха для фотоальбома о свадьбе. Авторы [19] предлагают выделять наиболее важные объекты путем выявления среднего по большому количеству изображений о событиях одного типа.

4 Модели

Для построения детектора объектов на изображении используется комбинация из двух моделей. Первая из них – обученная на большом объеме изображений сверточная нейросеть, используемая для извлечения вектор-признаков с изображения. Вторая модель – это основной классификатор, который преобразует полученные вектор-признаки первой модели в нужные нам «вероятностные» признаки.

Нейронная сеть – широко используемый метод машинного обучения, показывающий отличные результаты в анализе изображений, текстов, распознавании речи и других областях. В последнее время сверточные нейронные сети получили большое распространение, и эта область машинного обучения сейчас активно развивается.

На вход первой нейронной сети подается изображение, на выходе получается некоторый вектор-признак, который далее подается на вход основному классификатору.

В качестве основного классификатора рассмотрим следующие модели:

- логистическая регрессия;
- градиентный бустинг;
- нейронная сеть.

4.1 Логистическая регрессия

Логистическая регрессия – это линейный алгоритм классификации с логистической функцией потерь. Часто эта модель используется в качестве отправной точки (baseline).

$$a(x, w) = \text{sign}(\sum_{i=1}^{n} w_j f_j(x) - w_0) = \text{sign}\langle w, x\rangle,$$

где w_j – вес j-го признака, w_0 – порог принятия

решения, $w = (w_0, w_1, \dots, w_n)$ – вектор весов, $\langle w, x \rangle$ – скалярное произведение признакового описания объекта на вектор весов. Считается, что $f_0(x) = -1$,

$$L(w) = \sum_{i=1}^{m} \ln(1 + e\exp^{-y_i * \langle x_i, w \rangle}) \to \min_w.$$

Логистическая регрессия – статистическая модель, которая используется для предсказания вероятности возникновения некоторого события по значениям множества признаков:

$P\{y|x\} = \sigma(y\langle x, w\rangle), \quad \sigma(z) = 1/(1 + \exp^{-z})$.

Модели обучаются на вектор-признаках – выходе первой нейронной сети.

Как основной классификатор рассматривается набор моделей логистических регрессий – для каждого выделенного класса используется своя модель. Реализация логистической регрессии берется из библиотеки sklearn с параметрами по умолчанию [11, 12]. Результаты этих моделей затем объединяются в один вектор.

4.2 Градиентный бустинг

Градиентный бустинг – метод машинного обучения, основанный на ансамбле деревьев решений, считающийся одним из наиболее эффективных методов (с точки зрения качества классификации) и обладающий хорошей обобщающей способностью.

Градиентный бустинг, как и любой бустинг-алгоритм, последовательно строит базовые модели так, что каждая следующая улучшает качество всего ансамбля. Градиентный бустинг деревьев решений строит модель в виде суммы деревьев:

$$f(x) = h_0 + \sum_{j=1}^{M} b_j h(x; a_m),$$

где h_0 – некоторое начальное приближение, $b_j \in R$ – параметр, регулирующий скорость обучения и влияние отдельных деревьев на всю модель, $h_j(x; a_n)$ – базовый алгоритм с вектором параметров a_n.

$L = \sum_{i=1}^{N} L(y_i, f_j(x_i)) \to \min_{a_i, b_i}$ – некоторая функция потерь.

Модели обучаются на вектор-признаках – выходе первой нейронной сети.

Аналогично классификатору, использующему логистическую регрессию, рассматривается набор моделей градиентного бустинга – по одной на каждый класс. Реализация градиентного бустинга берется из библиотеки sklearn с параметрами по умолчанию [10, 12]. Результаты этих моделей так же объединяются в один вектор.

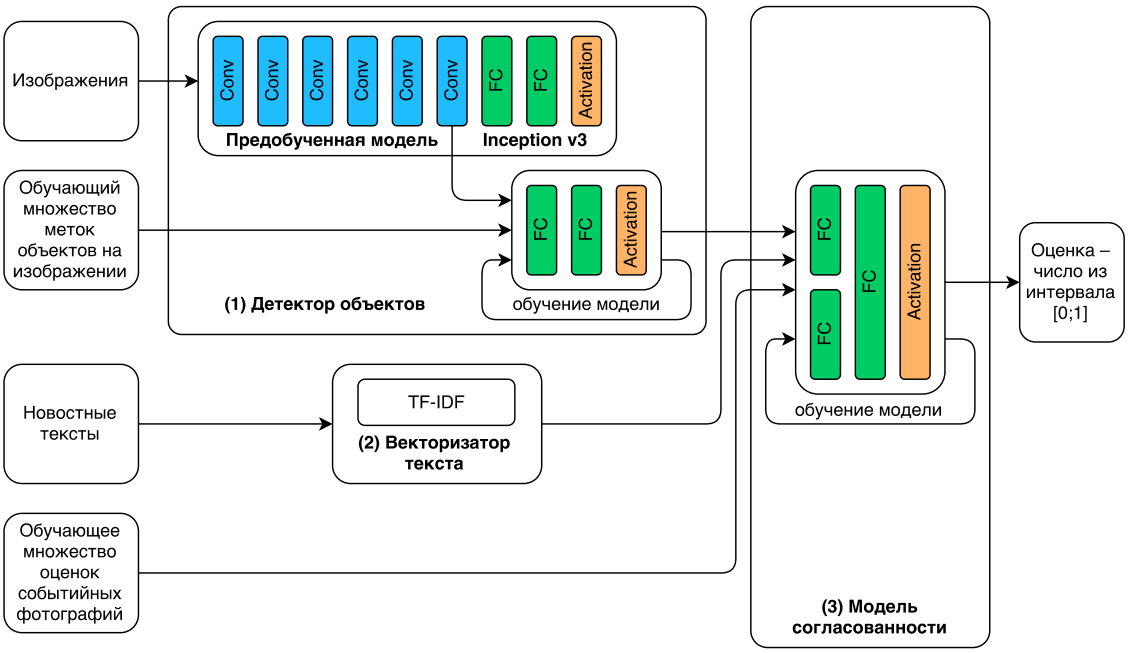

Рисунок 3 Схема потока данных обучения моделей

4.3 Нейронная сеть

Рассмотрим нейросеть, на вход которой подается вектор-признак с первой нейросети, а на выходе получается вектор, описывающий вероятность присутствия классов на изображении.

В проведенном исследовании использована следующая архитектура нейронной сети (см. Рис. 3, детектор объектов). Такая архитектура используется в финальных слоях модели VGG-16 [9], которые идут сразу же за сверточными. В оригинале последний слой – Softmax, который не очень подходит для нашей задачи (при повышении оценки для одного класса все остальные занижаются). Нами последний слой был заменен на Sigmoid, так как решается задача многозначной классификации.

Для обучения нейронной сети использована следующая функция потерь:

$$L(\tilde{y}, y) = -(y \log y + (1 - y)\log(1 - \tilde{y}))$$

– бинарная кросс-энтропия.

5 Методы

5.1 Обработка изображения

Для представления изображения в виде вектора используются модели, описанные в п. 4.1. Для данного преобразования применяется метод, называемый переносом обучения (transfer learning).

Перенос обучения – метод, позволяющий применить знания, полученные в процессе решения одной задачи, для решения другой схожей задачи. Например, можно взять уже предобученную на большом объеме данных нейронную сеть и дообучить ее на своих данных. Применение данного метода обычно позволяет сэкономить большое количество ресурсов (как времени, так и вычислительных ресурсов). В данной работе в качестве предобученной модели использована Inception v3, обученная для ImageNet Large Visual Recognition Challenge на данных 2012 года [14].

В случае обработки изображений берется первая модель (предобученная нейросеть), на вход которой подается изображение. Затем из этой сети извлекается некоторый слой, который и будет являться промежуточным векторным представлением нашего изображения. Данный слой обычно содержит большое количество признаков, помогающих решать задачу классификации. Далее этот слой подается на вход уже второй модели (линейной регрессии, градиентному бустингу или другой нейронной сети). На выходе мы получаем вектор вероятностей присутствия объектов для каждого из классов.

5.2 Обработка текста

Для представления текста в векторном виде используется TF-IDF. Для каждого документа из коллекции его исходный текст токенизируется, а токены приводятся в начальную форму. Затем считается подокументная частотность преобразованных токенов и вычисляется TF-IDF вектор для каждого документа.

5.3 Объединение текста и изображений

Результаты обработки коллекции новостей с изображениями по пп. 5.1 и 5.2 подаются на вход нейросети, которая комбинирует полученную информацию с двух входов и возвращает число в диапазоне от 0 до 1 – вероятность того, что изображение подходит для иллюстрации текста. Нами проверено, влияет ли использование текстовых признаков на качество определения фотографии как событийной или же достаточно использования только признаков, выделенных на изображении.

6 Исходные данные

В качестве данных для обучения и отладки моделей были использованы как готовые наборы данных, так и специально собранные для решения поставленной задачи.

6.1 CIFAR-10

CIFAR-10 – это коллекция размеченных изображений, взятых из другого набора данных подназванием «80 million tiny images» [16].

Описание коллекции:

- 60000 размеченных изображений;
- 10 классов, 6000 изображений на класс (*самолет, автомобиль, птица, кошка, олень, собака, лягушка, лошадь, корабль, грузовик*);
- размер изображений фиксированный, 32x32;
- один класс на одном изображении.

6.2 Pascal VOC2012

Pascal VOC2012 – это коллекция размеченных изображений, которые были собраны для соревнования по распознаванию и классификации объектов [5].

Описание коллекции:

- 11540 размеченных цветных изображений;
- 20 классов, в среднем по 577 изображений на класс (*мотоцикл, автомобиль, человек, домашнее растение, велосипед, автобус, поезд, птица, лодка, лошадь, самолет, бутылка, телевизор, кресло, собака, кот, стол, кровать, корова, овца*);
- размер изображений не фиксирован, но максимальная длина сторон – 500 пикселей;
- не менее одного класса на изображении.

6.3 Коллекция изображений на базе ImageNet

Для обучения детектора объектов нужно просмотреть новостные иллюстрации, понять, какие объекты там чаще всего встречаются, и собрать собственную коллекцию для обучения. Нами выделено 38 классов объектов, которые чаще всего встречаются в новостных иллюстрациях, и для них была собрана коллекция изображений на базе ImageNet [6].

Описание коллекции:

- 62357 размеченных цветных изображений;
- 38 классов, в среднем по 1640 изображений на класс (*воздушная техника, животное, баннер, лодка, здание, церковь, концерт, конструкция, толпа, документ, электронное устройство, огонь/дым, флаг, еда, в помещении, военная воздушная техника, военный транспорт, гора, нефтегазовые строения, картина, человек, растение, общественный транспорт, дорога, корабль, снаружи, солдат, космический корабль, спикер, транспорт специальных служб, спорт, деловой костюм, телекамера, служебная форма, транспорт, военный корабль, вода, оружие);*
- размер изображений не фиксирован;
- не менее одного класса на изображении.

6.4 Коллекция новостей

В качестве обучающей коллекции для определения событийной фотографии был собран набор новостей, содержащий 4114 примеров. В результате разметки получилось 3100 позитивных и 1014 негативных примеров событийных фотографий.

Таблица 1 Значение AP для 4 моделей на наборе данных Pascal VOC2012

	Самолет	Велосипед	Птица	Лодка	Бутылка	Автобус	Автомобиль	Кот	Кресло	Корова	mAP
GBC	0,9664	0,7670	0,8934	0,8090	0,5034	0,8747	0,8079	0,9050	0,7331	0,7241	
LR	**0,9938**	**0,8769**	0,9117	0,8795	0,5294	**0,9105**	0,8197	0,9140	0,7115	0,7402	
NN	0,9628	0,8738	0,9140	0,8885	**0,6142**	0,9089	**0,8229**	**0,9148**	**0,7796**	**0,7466**	
*HCP**	*0,9750*	*0,8430*	*0,9300*	*0,8940*	*0,6250*	*0,9020*	*0,8460*	*0,9480*	*0,6970*	*0,9020*	
	Стол	Собака	Лошадь	Мотоцикл	Человек	Растение	Овца	Кровать	Поезд	Телевизор	
GBC	0,7467	0,8729	0,8832	0,8748	0,8905	0,5145	0,7992	0,6076	0,8888	0,7276	0,7895
LR	0,6350	0,8995	**0,9458**	0,9010	0,9010	0,5017	0,8122	0,6900	0,9034	0,7464	0,8112
NN	**0,7480**	**0,9188**	0,9099	**0,9212**	**0,9062**	**0,5559**	**0,8272**	**0,7716**	**0,9050**	**0,7847**	**0,8337**
*HCP**	*0,7410*	*0,9340*	*0,9370*	*0,8880*	*0,9330*	*0,5970*	*0,9030*	*0,6180*	*0,9440*	*0,7800*	*0,8420*

7 Эксперименты

7.1 Выбор модели для обработки изображений

В качестве основной метрики была использована AP (average precision), определенная в статье [4] и вычисляемая по следующей формуле:

$$AP = \frac{1}{11} \sum_{r \in \{0, 0.1, \ldots, 1\}} p_{interp}(r),$$

$p_{interp}(r) = \max_{\tilde{r}:\tilde{r} \geq r} p(\tilde{r})$ – интерполяция точности, где $p(\tilde{r})$ – это измеренное значение точности для значения полноты $\tilde{r}$, p(x) – кривая «точность–полнота».

Сравнения качества моделей на наборе данных Pascal VOC2012 отображено в таблице1. Здесь также приведены значения AP для модели HCP-2000C [20] на конкурсном тестовом множестве (не на том, который был использован для сравнения моделей 1–3).

Из полученных оценок можно сделать вывод, что нейронная сеть с текущими параметрами лучше подходит для решения поставленной задачи, в дальнейшем будем рассматривать ее как основную модель.

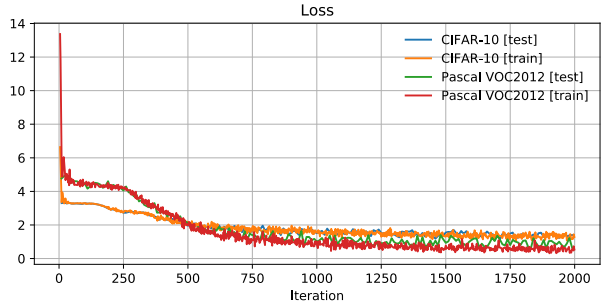

Рисунок 4 Зависимость значения функции потерь от итерации

На графике (см. Рис. 4) можно наблюдать, как сходится функция потерь нейросетевой модели на различных наборах данных. Поведение функций довольно похожее, но на Pascal VOC2012 при более медленной сходимости достигается лучшее качество. Графики MAE (средней абсолютной ошибки) ведут себя одинаковым образом (см. Рис. 5).

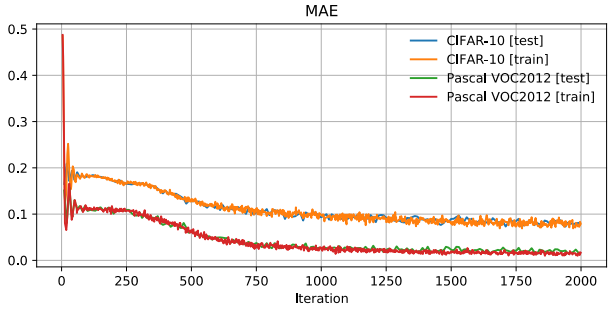

Рисунок 5 Зависимость значения MAE от итерации

7.2 Обучение модели на собственном наборе данных

Убедившись, что модель работает и показывает хорошие результаты на готовых наборах данных, нужно перейти к следующему этапу – обучению модели на собственной коллекции.

7.3 Применение моделей к новостям

Для этого обучается модель согласованности, которая по входным данным определяет, является изображение событийным или нет.

Обучим две модели, одна из которых принимает на вход одно лишь векторное представление изображения, а другая принимает на вход, помимо прочего, векторное представление соответствующего новостного текста. Во второй сети каждый из двух векторов входа преобразуется в вектор общей длины, затем конструируется новый вектор, получающийся конкатенацией поэлементного умножения и поэлементного сложения предыдущих слоев. Финальный слой

каждой сети – Softmax. На выходе нейросети получаем два значения p_1, p_2 в интервале [0;1], первое из которых – вероятность того, что изображение не является событийным для этой фотографии, а второе – что является ($p_1 + p_2 = 1$). Для обучения нейронной сети использована следующая функция потерь: $L(\tilde{y}, y) = -(y \log \tilde{y} + (1 - y)\log(1 - \tilde{y}))$ – софтмакс кросс-энтропия.

На Рис. 6 показаны графики значения функции потерь для каждой из двух моделей. Отметим, что модель, использующая текст, имеет склонность к переобучению после ~1500 итераций.

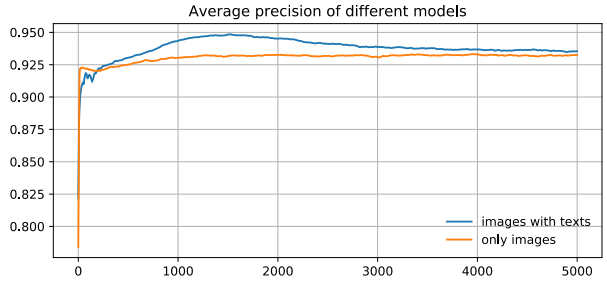

Рисунок 6 Зависимость значений средней точности для различных моделей на тестовых данных

7.4 Результаты

Следующие шаги после обучения детектора – его применение к новостным изображениям, получение векторного представления изображений и перевод текстов в векторную форму. Примеры работы программы изображены на Рис. 7 и 8.

Рисунок 7 Пример работы программы

Рисунок 8 Пример работы программы

8 Интерпретация результатов

В работе исследован метод ранжирования изображений для иллюстрации новостного сюжета, а именно, выявления изображений, которые с большей вероятностью содержат информацию, дополняющую текстовое сообщение. Представлен метод с использованием переноса обучения результатов Inception v3, когда несколько последних слоев обученной нейронной сети заменяются специфическим классификатором для исследуемой коллекции изображений.

В проведенных экспериментах специфический классификатор на основе нейронных сетей несколько превзошел логистическую регрессию и градиентный бустинг (однако для практических целей данные методы также можно использовать).

На коллекции из 4114 изображений (из них 3100 событийных), размеченной одним из авторов, достигнут результат 93,2% средней точности при обучении только по изображениям и 94,7% при использовании текстовой информации.

Целью дальнейших исследований являются применение более сложных и современных моделей классификации, введение дополнительных признаков, выделенных на изображениях, оценка применимости данной работы на таких источниках новостей, как социальные сети, улучшение и расширение собранных коллекций.

Литература

[1] Ahsan, U., Sun, C., Hays, J., Essa, I.: Complex Event Recognition from Images with Few Training Examples. Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on, pp. 669-678 (2017)

[2] Chollet, F. and others: Keras. https://github.com/fchollet/keras

[3] Cui, Y., Liu, D., Chen, J., Chang, S.F.: Building a Large Concept Bank for Representing Events in Video. arXiv preprint arXiv:1403.7591 (2014)

[4] Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes (VOC) Challenge: A Retrospective. Int. J. of Computer Vision, 111 (1), pp. 98-136 (2015)

[5] Everingham, M., Winn, J.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Development Kit (2012)

[6] ImageNet. http://image-net.org/index

[7] Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems, pp. 1097-1105 (2012)

[8] Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Learning and Transferring Mid-Level Image Representations using Convolutional Neural. Networks. M.: CVF (2014)

[9] Simonyan, K., Zisserman, A.: Very Deep 5Convolutional Networks for Large-Scale Image Recognition. arXiv preprint arXiv:1409.1556 (2014)

[10] Sklearn, GradientBoostingClassifier. http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html

[11] Sklearn, LogisticRegression. http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

[12] Sklearn. OneVsRestClassifier. http://scikit-learn.org/stable/modules/generated/sklearn.multiclass.OneVsRestClassifier.html

[13] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A Simple Way to Prevent Neural Networks from Overfitting. J. of Machine Learning Research, 15 (1), pp. 1929-1958 (2014)

[14] Szegedy, C., Vanhoucke, V., Ioffe, S., Wojna, Z., Shlens, J.: Rethinking the Inception Architecture for Computer Vision. Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818-2826 (2016)

[15] TensorFlow: Large-scale machine learning on heterogeneous systems. http://tensorflow.org (2015)

[16] The CIFAR-10 dataset. https://www.cs.toronto.edu/~kriz/cifar.html

[17] Theano Development Team. Theano: A Python Framework for Fast Computation of Mathematical Expressions. arXiv preprint arXiv:1605.02688 (2016)

[18] van Mitenburg, E., Elliot, D.: Room for Improvement in Automatic Image Description: an Error Analysis. arXiv preprint arXiv:1704.04198 (2017)

[19] Wang, Y., Lin, Z., Shen, X., Mech, R., Miller, G., Cottrell, G.W.: Event-specific Image Importance. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, pp. 4810-4819 (2016)

[20] Wei, Y., Xia, W., Huang, J., Ni, B., Dong, J., Zhao, Y., Yan, S.: CNN: Single-label to Multi-label. arXiv preprint arXiv:1406.5726 (2014)

[21] Yang, S., Luo, P., Loy, C.C., Tang, X.: Wider Face: A Face Detection Benchmark. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, pp. 5525-5533 (2016)

[22] Zeiler, M.D., Fergus, R.: Visualizing and Understanding Convolutional Networks. European Conf. on Computer Vision. Springer, Cham, pp. 818-833 (2014)