**Лабораторная работа №1. Логистическая регрессия в качестве нейронной сети**

**Данные:** В работе предлагается использовать набор данных notMNIST, который состоит из изображений размерностью 28×28 первых 10 букв латинского алфавита (A … J, соответственно). Обучающая выборка содержит порядка 500 тыс. изображений, а тестовая – около 19 тыс.

Данные можно скачать по ссылке:

* <https://commondatastorage.googleapis.com/books1000/notMNIST_large.tar.gz> (большой набор данных);
* [https://commondatastorage.googleapis.com/books1000/notMNIST\_small.tar.gz](https://commondatastorage.googleapis.com/books1000/notMNIST_large.tar.gz) (маленький набор данных);

Описание данных на английском языке доступно по ссылке:

<http://yaroslavvb.blogspot.sg/2011/09/notmnist-dataset.html>

**Задание 1.**

Загрузите данные и отобразите на экране несколько из изображений с помощью языка Python;

**Задание 2.**

Проверьте, что классы являются сбалансированными, т.е. количество изображений, принадлежащих каждому из классов, примерно одинаково (В данной задаче 10 классов).

**Задание 3.**

Разделите данные на три подвыборки: обучающую (200 тыс. изображений), валидационную (10 тыс. изображений) и контрольную (тестовую) (19 тыс. изображений);

**Задание 4.**

Проверьте, что данные из обучающей выборки не пересекаются с данными из валидационной и контрольной выборок. Другими словами, избавьтесь от дубликатов в обучающей выборке.

**Задание 5.**

Постройте простейший классификатор (например, с помощью логистической регрессии). Постройте график зависимости точности классификатора от размера обучающей выборки (50, 100, 1000, 50000). Для построения классификатора можете использовать библиотеку SkLearn ([http://scikit-learn.org](http://scikit-learn.org/)).

Результат выполнения заданий опишите в отчете.

**Лабораторная работа №2. Реализация глубокой нейронной сети**

**Данные:** В работе предлагается использовать набор данных notMNIST, который состоит из изображений размерностью 28×28 первых 10 букв латинского алфавита (A … J, соответственно). Обучающая выборка содержит порядка 500 тыс. изображений, а тестовая – около 19 тыс.

Данные можно скачать по ссылке:

* <https://commondatastorage.googleapis.com/books1000/notMNIST_large.tar.gz> (большой набор данных);
* [https://commondatastorage.googleapis.com/books1000/notMNIST\_small.tar.gz](https://commondatastorage.googleapis.com/books1000/notMNIST_large.tar.gz) (маленький набор данных);

Описание данных на английском языке доступно по ссылке:

<http://yaroslavvb.blogspot.sg/2011/09/notmnist-dataset.html>

**Задание 1.**

Реализуйте полносвязную нейронную сеть с помощью библиотеки Tensor Flow. В качестве алгоритма оптимизации можно использовать, например, стохастический градиент (Stochastic Gradient Descent, SGD). Определите количество скрытых слоев от 1 до 5, количество нейронов в каждом из слоев до нескольких сотен, а также их функции активации (кусочно-линейная, сигмоидная, гиперболический тангенс и т.д.).

**Задание 2.**

Как улучшилась точность классификатора по сравнению с логистической регрессией?

**Задание 3.**

Используйте регуляризацию и метод сброса нейронов (dropout) для борьбы с переобучением. Как улучшилось качество классификации?

**Задание 4.**

Воспользуйтесь динамически изменяемой скоростью обучения (learning rate). Наилучшая точность, достигнутая с помощью данной модели составляет 97.1%. Какую точность демонстрирует Ваша реализованная модель?

Результат выполнения заданий опишите в отчете.

**Лабораторная работа №3. Реализация сверточной нейронной сети**

**Данные:** В работе предлагается использовать набор данных notMNIST, который состоит из изображений размерностью 28×28 первых 10 букв латинского алфавита (A … J, соответственно). Обучающая выборка содержит порядка 500 тыс. изображений, а тестовая – около 19 тыс.

Данные можно скачать по ссылке:

* <https://commondatastorage.googleapis.com/books1000/notMNIST_large.tar.gz> (большой набор данных);
* [https://commondatastorage.googleapis.com/books1000/notMNIST\_small.tar.gz](https://commondatastorage.googleapis.com/books1000/notMNIST_large.tar.gz) (маленький набор данных);

Описание данных на английском языке доступно по ссылке:

<http://yaroslavvb.blogspot.sg/2011/09/notmnist-dataset.html>

**Задание 1.**

Реализуйте нейронную сеть с двумя сверточными слоями, и одним полносвязным с нейронами с кусочно-линейной функцией активации. Какова точность построенное модели?

**Задание 2.**

Замените один из сверточных слоев на слой, реализующий операцию пулинга (Pooling) с функцией максимума или среднего. Как это повлияло на точность классификатора?

**Задание 3.**

Реализуйте классическую архитектуру сверточных сетей LeNet-5 (http://yann.lecun.com/exdb/lenet/).

**Задание 4.**

Сравните максимальные точности моделей, построенных в лабораторных работах 1-3. Как можно объяснить полученные различия?

Результат выполнения заданий опишите в отчете.

**Лабораторная работа №4. Реализация приложения по распознаванию номеров домов.**

**Данные:** Набор изображений из Google Street View с изображениями номеров домов, содержащий 10 классов, соответствующих цифрам от 0 до 9.

* 73257 изображений цифр в обучающей выборке;
* 26032 изображения цифр в тестовой выборке;
* 531131 изображения, которые можно использовать как дополнение к обучающей выборке;
* В двух форматах:

o Оригинальные изображения с выделенными цифрами;

o Изображения размером 32 × 32, содержащих одну цифру;

* Данные первого формата можно скачать по ссылкам:

o <http://ufldl.stanford.edu/housenumbers/train.tar.gz> (обучающая выборка);

o <http://ufldl.stanford.edu/housenumbers/test.tar.gz> (тестовая выборка);

o <http://ufldl.stanford.edu/housenumbers/extra.tar.gz> (дополнительные данные);

* Данные второго формата можно скачать по ссылкам:

o <http://ufldl.stanford.edu/housenumbers/train_32x32.mat> (обучающая выборка);

o <http://ufldl.stanford.edu/housenumbers/test_32x32.mat> (тестовая выборка);

o <http://ufldl.stanford.edu/housenumbers/extra_32x32.mat> (дополнительные данные);

* Описание данных на английском языке доступно по ссылке:

o <http://ufldl.stanford.edu/housenumbers/>

**Задание 1.**

Реализуйте глубокую нейронную сеть (полносвязную или сверточную) и обучите ее на синтетических данных (например, наборы MNIST (http://yann.lecun.com/exdb/mnist/) или notMNIST).

Ознакомьтесь с имеющимися работами по данной тематике: англоязычная статья ([http://static.googleusercontent.com/media/research.google.com/en//pubs/archive/42241.pdf](http://static.googleusercontent.com/media/research.google.com/en/pubs/archive/42241.pdf)), видео на YouTube (https://www.youtube.com/watch?v=vGPI\_JvLoN0).

**Задание 2.**

После уточнения модели на синтетических данных попробуйте обучить ее на реальных данных (набор Google Street View). Что изменилось в модели?

**Задание 3.**

Сделайте множество снимков изображений номеров домов с помощью смартфона на ОС Android. Также можно использовать библиотеки OpenCV, Simple CV или Pygame для обработки изображений с общедоступных камер видеонаблюдения (например,<https://www.earthcam.com/>).

Пример использования библиотеки TensorFlow на смартфоне можете воспользоваться демонстрационным приложением от Google (https://github.com/tensorflow/tensorflow/tree/master/tensorflow/examples/android).

**Задание 4.**

Реализуйте приложение для ОС Android, которое может распознавать цифры в номерах домов, используя разработанный ранее классификатор. Какова доля правильных классификаций?

Результат выполнения заданий опишите в отчете.

**Лабораторная работа №5. Применение сверточных нейронных сетей (бинарная классификация)**

**Данные:** Набор данных DogsVsCats, который состоит из изображений различной размерности, содержащих фотографии собак и кошек. Обучающая выборка включает в себя 25 тыс. изображений (12,5 тыс. кошек: cat.0.jpg, …, cat.12499.jpg и 12,5 тыс. собак: dog.0.jpg, …, dog.12499.jpg), а контрольная выборка содержит 12,5 тыс. неразмеченных изображений. Скачать данные, а также проверить качество классификатора на тестовой выборке можно на сайте Kaggle -> <https://www.kaggle.com/c/dogs-vs-cats/data>

**Задание 1.**

Загрузите данные. Разделите исходный набор данных на обучающую, валидационную и контрольную выборки.

**Задание 2.**

Реализуйте глубокую нейронную сеть с как минимум тремя сверточными слоями. Какое качество классификации получено?

**Задание 3.**

Примените дополнение данных (data augmentation). Как это повлияло на качество классификатора?

**Задание 4.**

Поэкспериментируйте с готовыми нейронными сетями (например, AlexNet, VGG16, Inception и т.п.), применив передаточное обучение. Как это повлияло на качество классификатора?

Какой максимальный результат удалось получить на сайте Kaggle? Почему?

Результат выполнения заданий опишите в отчете.

**Лабораторная работа №6. Применение сверточных нейронных сетей (многоклассовая классификация)**

**Данные:** Набор данных для распознавания языка жестов, который состоит из изображений размерности 28x28 в оттенках серого (значение пикселя от 0 до 255). Каждое из изображений обозначает букву латинского алфавита, обозначенную с помощью жеста, как показано на рисунке ниже (рисунок цветной, а изображения в наборе данных в оттенках серого). Обучающая выборка включает в себя 27,455 изображений, а контрольная выборка содержит 7172 изображения. Данные в виде csv-файлов можно скачать на сайте Kaggle -> <https://www.kaggle.com/datamunge/sign-language-mnist>



**Задание 1.**

Загрузите данные. Разделите исходный набор данных на обучающую и валидационную выборки.

**Задание 2.**

Реализуйте глубокую нейронную сеть со сверточными слоями. Какое качество классификации получено? Какая архитектура сети была использована?

**Задание 3.**

Примените дополнение данных (data augmentation). Как это повлияло на качество классификатора?

**Задание 4.**

Поэкспериментируйте с готовыми нейронными сетями (например, AlexNet, VGG16, Inception и т.п.), применив передаточное обучение. Как это повлияло на качество классификатора? Можно ли было обойтись без него?

Какой максимальный результат удалось получить на контрольной выборке?

Результат выполнения заданий опишите в отчете.

**Лабораторная работа №7. Рекуррентные нейронные сети для анализа текста**

**Данные:** Набор данных для предсказания оценок для отзывов, собранных с сайта imdb.com, который состоит из 50,000 отзывов в виде текстовых файлов. Отзывы разделены на положительные (25,000) и отрицательные (25,000). Данные предварительно токенизированы по принципу “мешка слов”, индексы слов можно взять из словаря (imdb.vocab). Обучающая выборка включает в себя 12,500 положительных и 12,500 отрицательных отзывов, контрольная выборка также содержит 12,500 положительных и 12,500 отрицательных отзывов, а также. Данные можно скачать на сайте Kaggle -><https://www.kaggle.com/iarunava/imdb-movie-reviews-dataset>

**Задание 1.**

Загрузите данные. Преобразуйте текстовые файлы во внутренние структуры данных, которые используют индексы вместо слов.

**Задание 2.**

Реализуйте и обучите двунаправленную рекуррентную сеть (LSTM или GRU). Какого качества классификации удалось достичь?

**Задание 3.**

Используйте индексы слов и их различное внутреннее представление (word2vec, glove). Как влияет данное преобразование на качество классификации?

**Задание 4.**

Поэкспериментируйте со структурой сети (добавьте больше рекуррентных, полносвязных или сверточных слоев). Как это повлияло на качество классификации?

**Задание 5.**

Используйте предобученную рекуррентную нейронную сеть (например, DeepMoji или что-то подобное).

Какой максимальный результат удалось получить на контрольной выборке?

Результат выполнения заданий опишите в отчете.

**Лабораторная работа №8. Рекуррентные нейронные сети для анализа временных рядов**

**Данные:** Набор данных для прогнозирования временных рядов, который состоит из среднемесячного числа пятен на солнце, наблюдаемых с января 1749 по август 2017. Данные в виде csv-файла можно скачать на сайте Kaggle -><https://www.kaggle.com/robervalt/sunspots/>

**Задание 1.**

Загрузите данные. Изобразите ряд в виде графика. Вычислите основные характеристики временного ряда (сезонность, тренд, автокорреляцию).

**Задание 2.**

Для прогнозирования разделите временной ряд на обучающую, валидационную и контрольную выборки.

**Задание 3.**

Примените модель ARIMA для прогнозирования значений данного временного ряда.

**Задание 4.**

Повторите эксперимент по прогнозированию, реализовав рекуррентную нейронную сеть (с как минимум 2 рекуррентными слоями).

**Задание 5.**

Сравните качество прогноза моделей.

Какой максимальный результат удалось получить на контрольной выборке?

Результат выполнения заданий опишите в отчете.