Правительство Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет

«Высшая школа экономики»

нижегородский филиал

Факультет Информатики, математики и компьютерных наук

Кафедра Прикладной математики и информатики

**Отчет по научно-исследовательской практике**

**Выполнил:** Студент группы МАГ 16 ИАД

Боровкова Д.А.

**Руководитель:** Шадрина Е.В.,ст. преподаватель

Нижний Новгород 2018

**Содержание**

1. Введение
2. Прохождение практики
3. Заключение
4. **Введение**

В этом году я проходила практику на кафедре. Задание выполнялось в рамках совместного проекта НИУ ВШЭ - Нижний Новгород и Netcracker Technology по направлению Machine learning and Artificial Intelligence.

**О компании:**

Netcracker Technology - это мировой лидер в разработке и внедрении программного обеспечения для операторов связи. У нас более 20 лет богатейшего отраслевого опыта, 8000 специалистов, тысячи успешных внедрений и высочайший уровень стандартизации в индустрии. Наши успехи неоднократно отмечены самыми престижными отраслевыми наградами и признаны ведущими аналитиками телеком-отрасли, включая Gartner, Analysys Mason и Stratecast. В сотрудничестве со своей материнской компанией - японской корпорацией NEC - Netcracker уже несколько лет занимает ведущие позиции на рынке инновационных технологий SDN/NFV. Центры разработки находятся в России, Украине, Беларуси, Великобритании, Индии, США и Израиле.

Все участники были разделены на группы по 2-3 человека и были сформулированы следующие направления работы:

1. Обработка естественного языка (POS, NER, и т.д.)

2. Понимание темы разговора и/или цели разговора

3. Принятие решение о действии по результатам понимания

4. Свободная тема  
Мной была выбрана тема Обработка естественного языка (POS, NER, и т.д.).

**Цель практики:** придумать решение поставленной проблеме.

**Задачи:**

1. Изучить существующие подходы.
2. Выбрать лучший.
3. Привести пример работы прототипа.
4. Проанализировать результаты.
5. **Прохождение практики**

Задача обработки естественного языка состоит в том, чтобы разобрать предложение, фразу или текст на части речи, а также построить зависимости между словами.

Для решения поставленной задачи было решено использовать SyntaxNet. Это основанная на TensorFlow библиотека определения синтаксических связей, использующая нейронную сеть. В настоящий момент поддерживается 40 языков, в том числе и Русский. Плюсы SyntaxNet это конечно же базирование на гугловском TensorFlow. SyntaxNet уже содержит натренированную модель нейронной сети "Parsey McParseface", пригодная для разбора текста. Код открыт под лицензией Apache 2.0. Точность работы модели "Parsey McParseface" оценивается Google в 94%. Производительность SyntaxNet позволяет обрабатывать приблизительно 600 слов в секунду на обычном настольном компьютере. В качестве сопутствующих инструментов предоставлены средства для анализа лингвистической структуры предложений или высказываний, показывающие функциональную роль каждого слова.

SyntaxNet устанавливается под Unix системы, для Windows можно использовать Docker. Для этого необходимо установить его и скачать образ SyntaxNet. Удобство подхода заключается в том, что в образе SyntaxNet уже настроенный, так же установлены все необходимые зависимости. Далее приведён пример Dockerfile, с помощью которого можно запустить контейнер.

FROM tensorflow/syntaxnet

COPY UD\_Russian /opt/tensorflow/syntaxnet/UD\_Russian

COPY scripts /opt/tensorflow/syntaxnet/my\_script

Для анализа использовался текст на русском языке, взятый с Universal Dependencies. UD - это кросс-лингвистический проект. Он содержит размеченные предложения и тесты на разных языках, чтобы их потом можно было использовать для парсеров, обучений, исследований и так далее. Чтобы проанализировать текст был написан скрипт, который основан на demo.sh скрипте, включённым в образ SyntaxNet. Пример приведён ниже.

$PARSER\_EVAL \

--input=$INPUT\_FORMAT \

--output=stdout-conll \

--hidden\_layer\_sizes=64 \

--arg\_prefix=brain\_morpher \

--graph\_builder=structured \

--task\_context=$CONTEXT \

--resource\_dir=$MODEL\_DIR \

--model\_path=$MODEL\_DIR/morpher-params \

--slim\_model \

--batch\_size=1024 \

--alsologtostderr

Скрипты можно запускать не только для файлов но и для отдельных предложений, передавая их в качестве аргумента.

В результате работы скрипта получаем размеченные слова. Формат вывода conll. Пример этого формата приведён ниже.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | They | they | PRON | PRP | Case=Nom|Number=Plur | 2 | nsubj | 2:nsubj|4:nsubj |
| 2 | buy | buy | VERB | VBP | Number=Plur|Person=3|Tense=Pres | 0 | root | 0:root |
| 3 | and | and | CONJ | CC | \_ | 4 | cc | 4:cc |
| 4 | sell | sell | VERB | VBP | Number=Plur|Person=3|Tense=Pres | 2 | conj | 0:root|2:conj |
| 5 | books | book | NOUN | NNS | Number=Plur | 2 | obj | 2:obj|4:obj |
| 6 | . | . | PUNCT | . | \_ | 2 | punct | 2:punct |

Наибольший интерес представляют следующие колонки:

1. порядковый номер слова в предложении,

2. слово (или символ пунктуации),

4. часть речи,

7. номер родительского слова (или 0 для корня).

Используя номер родительского слова можно построить граф зависимостей в предложении.

Также с помощью Python были проанализированы полученные данные. Для анализа использовались три основных лингвистических параметра POS, LAS и UAS. Для русского языка эти параметры были следующими:

1. **Заключение**

Данный проект был успешно разработан и прошёл защиту перед комиссией компании Netcracker, а также позволил продемонстрировать полученные знания за время учёбы.