

Московский Государственный Технический Университет имени Н. Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Автоматизированные системы обработки информации и управления»



## **Отчёт по лабораторной работе № 1 по дисциплине «Проектирование интеллектуальных систем»**

Исполнитель:

Саврасов П.А.

Группа ИУ5-24М

«\_\_» \_\_\_\_\_ 2021 г.

Преподаватель:

Терехов В.И.

\_\_\_\_\_  
«\_\_» \_\_\_\_\_ 2021 г.

Москва, 2020 г.

## 1. Цель работы

Научить настраивать рабочее пространство для работы с tensorflow framework.

## 2. Установка интерпретатора Python

Установка интерпретатора языка Python не требуется в операционных системах Linux, так как во многих из них он установлен по умолчанию. В случае отсутствия такового, потребуется ввести в терминале команду установки через пакетный менеджер (в данном случае пакетный менеджер apt (Advanced package tool) дистрибутивов, основанных на Debian):

```
>>> sudo apt-get install python3
```

Проверить корректность установки или наличие интерпретатора можно через проверку версии:

```
>>> python3 --version
```

```
<<< Python 3.8.2
```

или вызовом консольного интерпретатора

```
>>> python3
```

```
<<< Python 3.8.2 (default, Mar 13 2020, 10:14:16)
```

```
[GCC 9.3.0] on linux
```

```
Type "help", "copyright", "credits" or "license" for more
```

information.

## 3. Установка virtualenv

Установка virtualenv производится через пакетный менеджер Python pip. Если такового нет, нужно установить его через пакетный менеджер операционной системы:

```
>>> sudo apt-get install python3-pip
```

После установки желательно проверить корректность установки выводом версии пакетного менеджера pip

```
>>> pip3 --version
```

```
<<< pip 20.0.2 from /usr/lib/python3/dist-packages/pip (python
```

```
3.8)
```

Далее установим virtualenv:

```
>>> pip3 install virtualenv
```

После установки может возникнуть ошибка отсутствия пакета virtualenv:

```
>>> virtualenv --version
```

```
<<< bash: virtualenv: команда не найдена
```

Причиной этой ошибки является отсутствие имени virtualenv в переменной PATH. Сам пакет virtualenv по умолчанию устанавливается в папку /home/<username>/.local/bin/. Для добавления этой директории в переменную PATH требуется изменить файл настроек терминала bash путём добавления в файл /home/<username>/.bashrc строк:

```
export PATH=$PATH:/home/<username>/.local/bin/
```

После этого проверить версию пакета virtualenv:

```
>>> virtualenv --version
```

```
<<< virtualenv 20.4.2 from
```

```
/home/<username>/.local/lib/python3.8/site-packages/virtualenv/___init__
```

```
___.py
```

Создание среды производится так же как и на Windows:

```
>>> virtualenv path/to/folder
```

Активация и деактивация среды производится скриптами, хранящимися в директории path/to/folder/bin

```
>>> cd path/to/folder/bin
```

```
>>> ls
```

```
<<< activate ...
```

Иногда файл активации и деактивации среды не является исполняемым скриптом, в таком случае его нужно отметить флагом x (eXecutable). Выполняется это командой

```
>>> sudo chmod +x activate
```

После чего его можно выполнить:

```
>>> echo $VIRTUAL_ENV – проверка запущенной виртуальной среды
<<<
>>> . ./activate
>>> echo $VIRTUAL_ENV
<<< /home/<username>/path/to/folder
```

Для деактивации виртуальной среды используется комбинация клавиш Ctrl+D или вводом команды exit

## **4. Установка Jupyter notebook**

Установка Jupyter Notebook производится так же через пакетный менеджер Python pip.

```
>>> pip3 install jupyter
```

Запуск выполняется командой

```
>>> jupyter notebook
```

Далее будет запущен локальный сервер и откроется страница клиент в браузере.

## **5. Установка Tensorflow**

Установка Tensorflow производится так же через пакетный менеджер Python pip.

```
>>> pip3 install tensorflow
```

## **6. Проверка работы установленного ПО на примере построения вычислительного графа**

Вывод из Jupyter Notebook:

```
In [1]: import tensorflow as tf
        print(tf.__version__)

INFO:tensorflow:Enabling eager execution
INFO:tensorflow:Enabling v2 tensorshape
INFO:tensorflow:Enabling resource variables
INFO:tensorflow:Enabling tensor equality
INFO:tensorflow:Enabling control flow v2
2.5.0-rc1
```

```
In [3]: a = tf.constant(1)
        b = tf.constant(2)

        c = tf.add(a, b)
        print(c)

tf.Tensor(3, shape=(), dtype=int32)
```

```
In [5]: a = tf.constant(2)
        b = tf.constant(3)
        c = tf.multiply(a, b)
        d = tf.pow(c, 2)
        print(d)

tf.Tensor(36, shape=(), dtype=int32)
```

## **7. Выводы по работе:**

Научился настраивать рабочее пространство для работы с tensorflow framework.

## **8. Контрольные вопросы**

### **8.1. Что такое virtualenv?**

virtualenv — это инструмент, позволяющий создавать виртуальные окружения с пакетами. Работая над различными python проектами virtualenv помогает сохранить для каждого проекта набор пакетов с определенными версиями. Без его использования возможен случайный апгрейд пакета, который использует два разных проекта, что может привести к нежелательным ошибкам.

## **8.2. Как запустить Jupyter notebook?**

Запуск выполняется командой `jupyter notebook`

## **8.3. Что такое Tensorflow?**

Tensorflow - открытая программная библиотека для машинного обучения, разработанная компанией Google для решения задач построения и тренировки нейронной сети

## **8.4. Что такое переменная в Tensorflow?**

Тензоры в TensorFlow

TensorFlow хранит данные в тензорах — многомерных массивах очень похожих на массивы `numpy`.