федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Казанский национальный исследовательский технический университет  
им. А.Н. Туполева-КАИ»

Кафедра систем информационной безопасности

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе по дисциплине

«Нейросетевые модели и алгоритмы»

на тему «Обработка динамических графов»

Выполнил:

ст. гр. 4167 Иванов И.И.,

Петрова Р.Р.,

Сафин М.Р.

Проверил:

д. т. н, доцент кафедры СИБ

\_\_\_\_\_\_\_\_\_\_\_\_\_ Аникин И.В.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.

Казань 2019



GERMAN-RUSSIAN INSTITUTE OF ADVANCED TECHNOLOGIES

TU-ILMENAU (GERMANY) AND KNRTU-KAI (RUSSIA)

Research in Computer & System Engineering Program

“Neural Networks” Course

Group Project

**Some name of your work**

Submitted by

Ivanov Ivan

Petrova Maria

Supervised by

D.C.S.. Igor Anikin

Kazan, 2019

# Abstract

Include the abstract of your project. This should be no more than 300 words.

##### Table of Contents

[Abstract 2](#_Toc498437051)

[Abbreviations 4](#_Toc498437052)

[List of Figures 5](#_Toc498437053)

[List of Tables 6](#_Toc498437054)

[Algorithms and program codes 6](#_Toc498437055)

[1. Introduction 7](#_Toc498437056)

[1.1. Motivation 7](#_Toc498437057)

[1.2. Problem statement 8](#_Toc498437058)

[1.3. Thesis structure 8](#_Toc498437059)

[2. Background 9](#_Toc498437060)

[2.1. Section name 9](#_Toc498437061)

[2.1.1. Some subsection name 9](#_Toc498437062)

[3. State of the art. Literature overview 10](#_Toc498437063)

[3.1. Section name 10](#_Toc498437064)

[3.2. Summary 10](#_Toc498437065)

[4. Development. System design. 10](#_Toc498437066)

[4.1. Software description 10](#_Toc498437067)

[4.2. Summary 10](#_Toc498437068)

[5. Evaluation & Experiments 11](#_Toc498437069)

[5.1. Experimental results of/Evaluation of… 11](#_Toc498437070)

[6. Conclusion and Perspectives 12](#_Toc498437071)

[Bibliography 13](#_Toc498437072)

# Abbreviations

|  |  |
| --- | --- |
| **API** | Application Programming Interface |
| **HDFS** | Hadoop Distributed File System |
| **RDD** | Resilient Distributed Dataset |
| **NN** | Neural Networks |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# List of Figures

Figure 2.1 Machine learning tasks and algorithms [1] 10

# List of Tables

[Table 1.1: Libraries features 8](#_Toc498437033)

# Algorithms and program codes

[**Algorithm 4‑1** Example of reading CSV to Spark DataFrame 11](#_Toc498437036)

# Introduction

## Motivation

Neural networks and deep learning currently provide the best solutions to many problems in image recognition, speech recognition, and natural language processing. But more relevant to work with all datasets and learning processes using distributed systems because of the huge amount of computations. It helps to parallelize program, to spend less time for learning and validation and eventually to choose the most appropriate model faster. This area of science is extremely important and fast-developing nowadays. It solves a lot of tasks and it is well-presented in the commercial sphere as well.

BigDL is a distributed deep learning library for Apache Spark; with BigDL, users can write their

deep learning applications as standard Spark programs, which can directly run on top of existing

Spark or Hadoop clusters.

BigDL implements possibility to analyze big amount of data (Big Data), and add deep learning functionalities to Big Data programs. Also provides other features such as dynamical share with other workloads.

Examples of BigDL features are presented in Table 1.1.

Table 1.1: Libraries features

|  |  |
| --- | --- |
| Programming Languages | Supports Scala and provides Python API |
| Chips supported | Supports distributed deep learning on  commodity CPUs and not GPUs. BigDL  works on any x86\_64 hardware that uses MK, which is specially designed for Intel hardware |
| Pre-trained model support | Supports models created in TensorFlow, Caffe or Torch framework |
| Linear algebra library | Uses Intel’s MKL |
| Distributed deep neural network training  on Spark | Supports training with CPUs Employs  synchronous parameter averaging to train a  network where Spark’s Block Manager serves  as a parameter server, thus avoiding the driver  being a bottleneck in the communication,  details here. The parameter averaging occurs  once a mini-batch is processed by all the  workers. The mini-batch size is expected to be  a multiple of total cores used in the job |
| ETL & Traditional Machine Learning | Can employ Apache Spark and its ML/ MLlib  algorithms |
| Visualization | Supports visualization through TensorBoard |

## Problem statement

As it was mentioned above…,

Summarizing previously mentioned facts, the main objectives of this work could be concluded. The main goal of this work is to analyze scalability property of different types of NN based approach for …problem, described above. To achieve this goal the following subtasks have to be done:

* Analyze…;
* Analyze…;
* Implement…;
* Preform evaluation of ….

## Thesis structure

The work has the following structure. Chapter 2 gives the background information about..., definition and examples of.... Chapter 3 gives overview on existing.... Chapter 4 describes implemented framework and it’s features. Chapter 5 contain evaluation part. Finally, conclusion and further perspectives of the project are presented in chapter 6.

# Background

This chapter is intended to give an overview on existing approaches in ML algorithms. The first section of this chapter gives a definition and examples of … methods, and also describe differences in comparing with data mining approaches. Second section will introduce concepts for .... Third and fourth sections will describe main approaches ..... Last section will make a conclusion about algorithms for ..., describe their advantages and disadvantages.

## Section name

Here example of the references [1].

### Some subsection name

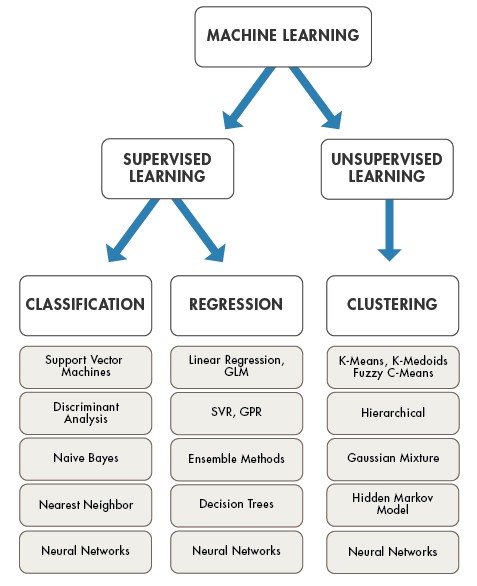


Figure 2.1 Machine learning tasks and algorithms [1]

# State of the art. Literature overview

In previous chapters the concept of … was introduced. The main disadvantages of … is slow processing time. Nowadays there are many systems, which can partially solve this problem. First section of this chapter will provide ...... will be presented in second and third sections respectively.

## Section name

## Summary

The main purpose of this chapter was in giving introduction...

# Development. System design.

This chapter is intended to give information about implementation details of …. The first section describes the general conceptual model of the framework, The second section gives information about actual implementation of the framework.

## Software description

|  |
| --- |
| **Algorithm 4‑1** Example of reading CSV to Spark DataFrame |
| **import** org.apache.spark.sql.SQLContext **import** org.apache.spark.sql.types.{StructType, StructField, StringType, IntegerType};  **val** sqlContext = **new** SQLContext(sc) |

## Summary

Therefore, implementation details of proposed framework were described in this chapter….

# Evaluation & Experiments

This chapter is intended to give information about evaluation of .... The evaluation of work was made by two criteria... First is responsible for.... For this purpose, different experiments was performed. Second criteria, has aim to evaluate...It will be described in second section of this chapter...

## Experimental results of/Evaluation of…

# Conclusion and Perspectives

As a result of this work/During this work/In this project

# Bibliography

|  |  |
| --- | --- |
| [1] | Y. Bengio, Learning Deep Architectures for AI. Foundations and Trends in Machine Learning, Hanover: Now Publishers, 2009. |
| [2] | "Apache Spark," [Online]. Available: http://spark.apache.org. |
| [3] | "Apache Hadoop," [Online]. Available: http://hadoop.apache.org. |
| [4] | "Apache Hive," [Online]. Available: https://hive.apache.org. |