

CLOUD COMPUTING PAPERS FA18

Gregor von Laszewski
Geoffrey C. Fox

laszewski@gmail.com

CLOUD COMPUTING PAPERS

Gregor von Laszewski

(c) Gregor von Laszewski, 2018

CLOUD COMPUTING PAPERS

1 Chapters and Sections Located Elsewhere

1.1 523

1.2 423

1.3 516

2 Sample Paper fa18-523-000, fa18-523-001

2.1 Example

2.2 Workbreakdown

3 Azure Data Services fa18-516-06

3.1 Abstract

3.2 Database Products

3.2.1 Azure SQL Database

3.2.2 Azure MySQL, PostgreSQL, and MariaDB Databases

3.2.3 Azure Cosmos DB

3.2.4 Azure SQL Data Warehouse

3.3 Analytics

3.3.1 Azure HDInsight

3.3.2 Azure Stream Analytics

3.3.3 Azure Data Lake Store and Data Lake Analytics

3.3.4 Azure Data Factory

4 Cloud and Privacy fa18-516-08

4.1 GDPR Compliance and Cloud Computing

4.1.1 GDPR Compliance

4.1.2 Data Processor vs Data Controller

4.1.3 Impact On Cloud Computing

4.1.4 Public or Private Cloud

4.1.5 Common Vendors GDPR Readiness

5 Cloud Foundry - Open Source PaaS fa18-516-11

5.1 Cloud Foundry Overview

5.1.1 Category: Clouds

5.1.2 Category: Frameworks

5.1.3 Category: Services

5.2 Key Components

5.2.1 Router

5.2.2 UAA and Login Server

[5.2.3 Cloud Controller](#)

[5.2.4 Execution and Storage](#)

[5.2.5 Service Brokers](#)

[5.2.6 Monitoring and Logging](#)

[5.3 Conclusion](#)

[6 Cloud Security Alliance \(CSA\) fa18-516-17](#)

[6.1 Introduction to CSA](#)

[6.2 About the CSA](#)

[6.3 Guiding Principles](#)

[6.4 History](#)

[6.5 Research Areas](#)

[6.6 Membership](#)

[6.7 Best ways to leverage CSA](#)

[6.7.1 Security Guidance Publication](#)

[6.7.2 Training and Certifications](#)

[6.7.3 Working Groups](#)

[7 Guided Analytics Using Knime fa18-523-52](#)

[7.1 Introduction](#)

[7.2 KNIME Used in Big Data](#)

[8 Microservices and Kafka fa18-523-53](#)

[8.1 Introduction](#)

[8.2 Architecture](#)

[8.3 Installation and Starting Kafka](#)

[8.4 Use Cases](#)

[8.5 Acknowledgement](#)

[9 Apache NiFi fa18-523-56 fa18-523-83](#)

[9.1 Apache NiFi Introduction](#)

[9.2 Big Data Challenges and NiFi](#)

[9.3 NiFi History](#)

[9.4 NiFi Features](#)

[9.5 NiFi Architecture](#)

[9.5.1 Web Server](#)

[9.5.2 Flow Controller](#)

[9.5.3 FlowFile Repository](#)

[9.5.4 Content Repository](#)

[9.5.5 Provenance Repository](#)

[9.5.6 Processors](#)

9.5.7 NiFi Clusters

9.6 NiFi Download, Installing and Getting Started

9.7 Use Case

9.7.1 File Transfer and Routing at MasterCard

9.7.2 Streaming Analytics Solutions at OpenText Magellan

9.7.3 Social Competitive Intelligence Application at Compose

9.7.4 Real Time Streaming Architecture at Ford

9.8 Work Breakdown

10 PyTorch fa18-523-57

10.1 Background

10.1.1 Deep Learning

10.1.2 Neural Networks

10.1.3 Tensors

10.1.4 Computational Graph

10.1.5 Auto Differentiation

10.1.6 Backpropagation

10.1.7 Autograd

10.2 Getting Started

10.3 Implementation

10.3.1 Define a Neural Network

10.3.2 Constructing an optimizer

10.4 Advantages of PyTorch

10.5 Drawbacks of PyTorch

11 Caffe - A Deep Learning Framework fa18-523-58

11.1 Introduction

11.1.1 Artificial Neural Network

11.1.2 Computer Vision

11.1.3 Deep Learning

11.2 Installation

11.3 Architecture

11.4 Applications

11.5 Limitations and Comparisons

11.6 Conclusion

12 MongoDB in Python fa18-523-60, fa18-523-64, fa18-523-72

12.1 Introduction

12.2 Learning Outcome

12.3 MongoDB

- [12.3.1 Installation](#)
 - [12.3.2 Collections and Documents](#)
 - [12.3.3 MongoDB Querying](#)
 - [12.3.4 MongoDB Basic Functions](#)
 - [12.3.5 Security Features](#)
 - [12.3.6 MongoDB Cloud Service](#)
 - [12.4 PyMongo](#)
 - [12.4.1 Installation](#)
 - [12.4.2 Dependencies](#)
 - [12.4.3 Running PyMongo with Mongo Deamon](#)
 - [12.4.4 Connecting to a database using MongoClient](#)
 - [12.4.5 Accessing Databases](#)
 - [12.4.6 Creating a Database](#)
 - [12.4.7 Inserting and Retrieving Documents \(Querying\)](#)
 - [12.4.8 Limiting Results](#)
 - [12.4.9 Updating Collection](#)
 - [12.4.10 Counting Documents](#)
 - [12.4.11 Indexing](#)
 - [12.4.12 Sorting](#)
 - [12.4.13 Aggregation](#)
 - [12.4.14 Deleting Documents from a Collection](#)
 - [12.4.15 Copying a Database](#)
 - [12.4.16 PyMongo Strengths](#)
 - [12.5 MongoEngine](#)
 - [12.5.1 Installation](#)
 - [12.5.2 Connecting to a database using MongoEngine](#)
 - [12.5.3 Querying using MongoEngine](#)
 - [12.6 Flask-PyMongo](#)
 - [12.6.1 Installation](#)
 - [12.6.2 Configuration](#)
 - [12.6.3 Connection to multiple databases/servers](#)
 - [12.6.4 Flask-PyMongo Methods](#)
 - [12.6.5 Additional Libraries](#)
 - [12.6.6 Classes and Wrappers](#)
 - [12.7 Workbreakdown](#)
 - [13 Natural Language Applications and Challenges within Big Data](#)
-
- fa18-523-61

13.1 Abstract	
13.2 Introduction	
13.3 Natural Language Challenges	
13.4 Natural Language Processing Solutions	
13.5 Conclusion	
14 Big Data and Streaming	fa18-523-62 fa18-523-69
14.1 Introduction	
14.2 Stream Processing vs Batch Processing	
14.3 Challenges in Stream Processing	
14.4 Big Data Streaming Architecture and Technologies	
14.4.1 Apache Spark	
14.4.2 Apache Storm	
14.4.3 Apache Flink	
14.4.4 Apache Kafka	
14.4.5 Amazon Kinesis	
14.4.6 Hortonworks Dataflow	
14.5 Industrial Use Cases of Big Data Streaming	
15 Scikit-learn	fa18-523-63
15.1 The supervised algorithms (some of them)	
15.2 The unsupervised algorithms (some of them)	
15.3 Other Method groupings within sklearn	
15.4 Further functionalities to Scikit	
15.5 Real world applications for scikit learn	
16 Natural language text processing and Language generation	
	fa18-523-67, fa18-523-65
16.1 Purpose of Natural language processing	
16.2 Levels of NLP	
16.2.1 Phonology	
16.2.2 Morphology	
16.2.3 Lexical	
16.2.4 Syntactic	
16.2.5 Semantic	
16.2.6 Pragmatic	
16.2.7 Discourse	
16.3 Natural Language generation	
16.4 Approaches to NLP:	
16.5 Related Work	

[16.6 Applications of NLP](#)

[16.7 Problems with NLP: linguistic variation and ambiguity](#)

[16.8 NLP in textual information retrieval](#)

[16.9 Statistical processing of natural language](#)

[16.10 Linguistic processing of natural language](#)

[16.11 NLP for Big data](#)

[16.12 NLP in Yelp data review](#)

[16.13 Building a NLP pipeline](#)

[16.13.1 Sentence Segmentation](#)

[16.13.2 Word Tokenization](#)

[16.13.3 Predicting parts of speech](#)

[16.13.4 Text Lemmatization](#)

[16.13.5 Identifying Stop Words](#)

[16.13.6 Dependency Parsing](#)

[16.13.7 Finding noun phrases](#)

[16.13.8 Named Entity Recognition](#)

[16.14 Installation](#)

[16.15 Example](#)

[16.16 Conclusion](#)

[16.17 Team Members and work breakdown](#)

[17 SAS Viya fa18-523-66](#)

[17.1 Introduction](#)

[17.2 SAS Viya Components](#)

[17.2.1 SAS Cloud Analytic Services](#)

[17.2.2 SAS Studio](#)

[17.2.3 SAS Visual Analytics](#)

[17.2.4 SAS Visual Statistics](#)

[17.2.5 SAS Visual Data Mining and Machine Learning](#)

[17.2.6 SAS Econometrics](#)

[17.2.7 SAS Visual Forecasting](#)

[17.2.8 SAS Visual Text Analytics](#)

[17.2.9 SAS Optimization](#)

[17.3 Deployment](#)

[17.3.1 System Requirements](#)

[17.3.2 Installation](#)

[17.4 Sample Illustration](#)

[17.5 Conclusion](#)

18 Distributed TensorFlow fa18-523-68

- 18.1 Abstract
- 18.2 Introduction
- 18.3 Parameter Server
- 18.4 TensorFlow Cluster
- 18.5 Parameter Server
- 18.6 Shared Variables
- 18.7 Synchronous Data Parallelism
- 18.8 Asynchronous Data Parallelism
- 18.9 In-graph replication
- 18.10 Between-graph replication
- 18.11 Asynchronous training
- 18.12 Synchronous training

19 Big Data Application in recommender systems fa18-523-70

- 19.1 Introduction
- 19.2 What is a recommender system?
- 19.3 How does the recommender system work?
 - 19.3.1 Collection of Data
 - 19.3.2 Storing the data
 - 19.3.3 Analyzing the data
 - 19.3.4 Filtering the data
- 19.4 Types of recommender systems
- 19.5 Algorithms
 - 19.5.1 K-Nearest Neighbors
 - 19.5.2 Association Rules
 - 19.5.3 Matrix Factorization
 - 19.5.4 Deep Neural Networks
- 19.6 Evaluation of recommender systems
 - 19.6.1 Validation of Recommender System
 - 19.6.2 Root mean squared error
 - 19.6.3 Top N Recommendations
- 19.7 Acknowledgement

20 IOT and Big Data: Applications and Future Trends fa18-523-71 fa18-523-59

- 20.0.1 Introduction
- 20.0.2 Architecture
- 20.0.3 Big Data and IoT Together

[20.0.4 Impacts of IOT on Big data](#)

[20.0.5 Challenges](#)

[20.0.6 Applications](#)

[20.0.7 Use cases](#)

[20.0.8 Future Trends](#)

[20.0.9 Conclusion](#)

[20.0.10 Acknowledgments](#)

[21 Big Data in Healthcare fa18-523-73](#)

[21.1 Introduction](#)

[21.2 Requirements of the Electronic Health Records](#)

[21.3 The Architecture of the Electronic Health Record](#)

[21.4 Implementation of Big Data in Electronic Health Record](#)

[21.5 Benchmark of Big Data in EHR Systems](#)

[21.5.1 Benefits of Big Data in EHR Systems](#)

[21.5.2 Challenges of Big Data in EHR Systems](#)

[21.6 Conclusion](#)

[22 Big Data and Privacy fa18-523-74](#)

[22.1 Introduction](#)

[22.2 Academic Theory](#)

[22.2.1 Requirements](#)

[22.2.2 Architecture](#)

[22.2.3 Implementation](#)

[22.2.4 Big Data and the Web](#)

[22.3 Benchmark and Privacy](#)

[22.4 Conclusions](#)

[23 QlikView fa18-523-79](#)

[23.1 Keywords](#)

[23.2 Introduction](#)

[23.3 Architecture](#)

[23.4 The Front End](#)

[23.5 The Back End](#)

[23.6 Associative In-Memory Technology](#)

[23.7 QlikView Server \(QVS\)](#)

[23.7.1 CPU](#)

[23.7.2 Memory](#)

[23.7.3 Data Compression](#)

[23.8 QlikView Publisher](#)

[23.8.1 CPU](#)

[23.8.2 Hard Drive](#)

[23.8.3 Memory](#)

[23.9 Uses and advantages](#)

[23.10 Conclusion](#)

[24 Utilizing Python Matplotlib Package for Data Visualization of In Cancer Clinical Trials](#) fa18-523-80

[24.1 Introduction](#)

[24.2 Architecture of Electronic Data Capture Systems](#)

[24.3 Matplotlib Use Case for Clinical Trials Visualizations](#)

[24.4 Matplotlib Architecture](#)

[24.5 Matplotlib Features for Clinical Trial Research](#)

[24.6 Conclusion](#)

[25 IBM Cognos Business Intelligence](#) fa18-523-81

[25.1 Introduction](#)

[25.2 History](#)

[25.3 Architecture](#)

[25.4 Components](#)

[25.4.1 IBM Cognos Connection](#)

[25.4.2 IBM Cognos Insight](#)

[25.4.3 IBM Cognos Workspace](#)

[25.4.4 IBM Cognos Workspace Advanced](#)

[25.4.5 IBM Cognos Report Studio](#)

[25.4.6 IBM Cognos Event Studio](#)

[25.4.7 IBM Cognos Metric Studio](#)

[25.4.8 IBM Cognos Query Studio](#)

[25.4.9 IBM Cognos Analysis Studio](#)

[25.5 Cognos Analytics](#)

[25.5.1 Big data and Cognos](#)

[25.6 Conclusions](#)

[26 IBM Watson Construction and its Services](#) fa18-523-82

[26.1 Introduction](#)

[26.1.1 Watson's Methodology of Working](#)

[26.1.2 Role of Artificial Intelligence in Building Watson](#)

[26.1.3 Role of Machine Learning in Building Watson](#)

[26.1.4 Role of Transfer Learning in Building Watson](#)

[26.2 IBM Watson and its Services](#)

- 26.2.1 IBM Watson Analytics
- 26.2.2 IBM Watson Machine Learning
- 26.3 Conclusion
- 27 Smart Home IoT Sensors for Raspberry Pi fa18-523-84
 - 27.1 Project Location
 - 27.2 Outline
 - 27.2.1 Contribute to sensor specific sections
 - 27.2.2 Sample Project
- 28 Big Data Analytics in E-commerce fa18-523-85
 - 28.1 Abstract
 - 28.2 Introduction
 - 28.3 Technology Background
 - 28.4 Big Data Applications in E-commerce
 - 28.5 Features of User Behavior in the E-commerce Platform
 - 28.6 Applications
 - 28.7 Conclusion
- 29 SAP fa18-523-86
 - 29.1 Introduction
 - 29.2 Implementation
 - 29.3 Conclusions
- 30 TBD fa18-523-88
- 31 Big Data Security and Privacy HID MISSING
 - 31.1 Introduction
 - 31.2 What is Big Data
 - 31.3 Big Data Needs Big Security
 - 31.4 Big Data Security Challenges
 - 31.4.1 Access Control
 - 31.4.2 Audit Control
 - 31.4.3 Real Time Compliance Control
 - 31.4.4 Non Relational Databases Privacy
 - 31.4.5 End-Point Input Validation
 - 31.4.6 Securing Transaction Logs and Data
 - 31.4.7 Securing Distributed Framework
 - 31.4.8 Data Provenance
 - 31.5 Big Data Security Stakeholders
 - 31.6 Best Practices for securing Big Data
 - 31.6.1 Authentication

[31.6.2 Cryptography](#)

[31.6.3 Data Masking](#)

[31.6.4 Access Control](#)

[31.6.5 Physical Security](#)

[31.7 Future of Big Data Security](#)

[31.7.1 Virtualization and Cloud Computing](#)

[31.7.2 IOT Security](#)

[31.7.3 External Password Vaults](#)

[31.7.4 Penetration Tests](#)

[31.8 Conclusions](#)

[31.9 Work Breakdown](#)

[Refernces](#)

1 CHAPTERS AND SECTIONS LOCATED ELSEWHERE

Please put here your links in the following format

1.1 523

If your paper or project report is located elsewhere, please modify this file in github or better contact the TAs

1.2 423

fa18-423-05, Yixing Hu fa18-423-02, Kelvin Liuwie fa18-423-03, Omkar Tamhankar fa18-423-06, Chandler Mick

wrong directory name

- Project: <https://github.com/cloudmesh-community/fa18-423-03/blob/master/project/project.md>
- Paper:

fa18-423-08, Yuli Zhao

- Project: <https://github.com/cloudmesh-community/fa18-423-08/blob/master/project-report/report.md>
- Paper:

fa18-423-07, Michael Gillum

- Project:
- Paper:

1.3 516

fa18-516-02, Vineet Barshikar

- Paper: <https://github.com/cloudmesh->

[community/book/blob/master/chapters/msg/graphql.md](https://github.com/cloudmesh-community/book/blob/master/chapters/msg/graphql.md)

- Other: <https://github.com/cloudmesh-community/fa18-516-21/tree/master/graphql-examples/cloudmeshrepo>

fa18-516-03, Jonathan Branam

- Project: <https://github.com/cloudmesh-community/cm-burn>
- Paper: <https://github.com/cloudmesh-community/book/tree/master/chapters/pi/kubernetes>
- Other (grammar and spell check Virtualization chapter) : <https://github.com/cloudmesh-community/book/tree/master/chapters/cloud/virtualization.md>
- Other (grammar and spell check E516 summary) : <https://github.com/cloudmesh-community/book/tree/master/chapters/class/e516-summary.md>

fa18-516-04, David Demeulenaere

- Section: <https://github.com/cloudmesh-community/fa18-516-04/blob/master/section/vs-code.md>
- Project: <https://github.com/cloudmesh-community/fa18-516-04/blob/master/project-paper/report.md>
- Paper:
- Other (Add data service code to cm) : <https://github.com/cloudmesh-community/cm/tree/master/cm4/data>

fa18-516-17, Brad Pope

- Section: <https://github.com/cloudmesh-community/book/edit/master/chapters/arch.md>
- Section: <https://github.com/cloudmesh-community/book/edit/master/chapters/iaas/watson/watson.r>
- Section: <https://github.com/cloudmesh-community/book/edit/master/chapters/mapreduce/hadoop-installation.md>
- Project: <https://github.com/cloudmesh-community/fa18-516->

[17/blob/master/project-paper/report.md](https://github.com/cloudmesh-community/fa18-516-17/blob/master/project-paper/report.md)

- Paper: <https://github.com/cloudmesh-community/fa18-516-17/blob/master/paper/paper.md>

fa18-516-10, Rui Li

- Project:
- Paper:

fa18-516-19, De'Angelo Rutledge

- Project: <https://github.com/cloudmesh-community/fa18-516-19/blob/master/project-report/report.md>
- Paper:
- Other (python setup script): <https://github.com/cloudmesh-community/fa18-516-19/blob/master/project-code/setproc.py>
- Other (bootstrap): <https://github.com/cloudmesh-community/fa18-516-19/blob/master/project-code/bootstrapbackup.bash>

fa18-516-26, Vafa Andalibi

- Section: https://github.com/cloudmesh-community/book/blob/master/chapters/prg/python/python.m_expressions-new
- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/prg/python/python.m>
- Project: <https://github.com/cloudmesh-community/cm/tree/master/cm4/vcluster>
- Paper: https://github.com/cloudmesh-community/book/blob/master/chapters/prg/python/python_parallel.md
- Paper: <https://github.com/cloudmesh-community/book/blob/master/chapters/faas/openwhisk.md>
- Other (Fixing typos in md files): <https://github.com/cloudmesh-community/book/commit/e0df3d097935f09fc8abffd7801f0c3>
- Other (Fixing typos in bib and tex files):

<https://github.com/cloudmesh-community/book/commit/69f9e1e6e9c6cf2c4c2f044b58c2364>

- Other (Fixing typos in makefiles, dot, java, sh and conf files):
<https://github.com/cloudmesh-community/book/commit/e8c71365b350ba31843b3d876bdd3>

fa18-516-21, Mihir Shanishchara

- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/cloud/datacenter.md>
- Paper: <https://github.com/cloudmesh-community/book/blob/master/chapters/msg/graphql.md>
- Project: <https://github.com/cloudmesh-community/fa18-516-21/blob/master/project-paper/report.md>
- Other (Validate YAML script in Python):
https://github.com/cloudmesh-community/book/blob/master/examples/yaml-validation/validate_yml.py
- Other (Create example of using graphql with python):
<https://github.com/cloudmesh-community/book/tree/master/examples/graphql/cloudmeshr>

fa18-516-18, Richa Rastogi

- Section: https://github.com/cloudmesh-community/book/blob/master/chapters/prg/python/python-install.md*install-pyenv-on-ubuntu
- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/prg/python/python-editor.md>
- Section: https://github.com/cloudmesh-community/book/blob/master/chapters/cloud/datacenter.md*center-carbon-footprint
- Section: https://github.com/cloudmesh-community/book/blob/master/chapters/prg/python/python.m_expressions-new
- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/prg/python/python.m>

new

- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/prg/python/python.md>
- Paper: <https://github.com/cloudmesh-community/book/blob/master/chapters/iaas/aws/aws-lambda.md>
- Paper: <https://github.com/cloudmesh-community/book/blob/master/chapters/mapreduce/mapreduce.md>
- Project:
- Other (Spell fix): <https://github.com/cloudmesh-community/book/commit/abb2464a00e5658531d5ebeafdf260f0457bbb05d4eafc8363cccf9b4168a1>

fa18-516-11, Murali Cheruvu

- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/devops/devop-ci.md>
- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/devops/devop-azure-monitor.md>
- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/devops/devop-aws.md>
- Chapter: https://github.com/cloudmesh-community/book/blob/master/chapters/cloud/cloud_foundry.md
- Project: <https://github.com/cloudmesh-community/fa18-516-11/blob/master/project-paper/paper.md>

fa18-516-29, Shilpa Singh

- Chapter:
- Section:

fa18-516-08, Varun Joshi

- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/faas/google-cloud-functions.md>

- Section: <https://github.com/cloudmesh-community/book/edit/master/chapters/faas/microsoft-azure-functions.md>
- Paper: <https://github.com/cloudmesh-community/fa18-516-08/blob/master/paper/paper.md>

fa18-516-06, Paul Filliman

- Section: <https://github.com/cloudmesh-community/fa18-516-06/blob/master/sections/AzureDataFactory.md>
- Section: <https://github.com/cloudmesh-community/fa18-516-06/blob/master/sections/AzureIoT.md>
- Section: <https://github.com/cloudmesh-community/fa18-516-06/blob/master/sections/VisualStudioCloudComputing.md>
- Paper: <https://github.com/cloudmesh-community/fa18-516-06/blob/master/chapter/whatever.md>
- Project: <https://github.com/cloudmesh-community/fa18-516-06/blob/master/chapter/whatever.md>

fa18-516-14, Gerald Manipon

- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/cloud/packer.md>
- Paper: <https://github.com/cloudmesh-community/book/blob/master/chapters/cloud/use-cases.md>
- Project: <https://github.com/cloudmesh-community/fa18-516-14/blob/master/project-report/report.md>

fa18-516-22, Ian Sims

- Section: <https://github.com/cloudmesh-community/fa18-516-22/blob/master/section/AWS-Admin-Access.md>
- Section: <https://github.com/cloudmesh-community/fa18-516-22/blob/master/section/AWS-CLI.md>
- Paper: <https://github.com/cloudmesh-community/fa18-516-22/blob/master/chapter/AWS-EMR.md>
- Project:

fa18-516-12, Yu Luo

- Project: <https://github.com/cloudmesh-community/cm>
- Other (contribute to the AWS in cm4 project):
<https://github.com/cloudmesh-community/cm/tree/master/cm4/aws>
- Other (contribute to the vm in cm4 project):
<https://github.com/cloudmesh-community/cm/tree/master/cm4/vm>
- Other (contribute to the MongoDB in cm4 project):
<https://github.com/cloudmesh-community/cm/tree/master/cm4/cmmongo>
- Project: cm4

fa18-516-31, Jordan Spell

fa18-516-25, Chun Sheng Wu

- Section: <https://github.com/cloudmesh-community/cm/tree/master/cm4/vagrant/vagrant.py>
- Section: <https://github.com/cloudmesh-community/cm/tree/master/cm4/openstack/OpenstackCM.py>
- Paper: <https://github.com/cloudmesh-community/cm/tree/master/cm4/vagrant/README.md>
- Other (contribute to cm4 project):
<https://github.com/cloudmesh-community/cm/tree/master/cm4/configrue/name.py>
- Project: cm4

fa18-516-24, Sachith Withana

- Section: <https://github.com/cloudmesh-community/cm/tree/master/cm4/configuration>
- Chapter: <https://github.com/cloudmesh-community/book/blob/master/chapters/pi/clusters/pi-spark.md>
- Project: cm4 <https://github.com/cloudmesh-community/cm>
- Other (Contributed to the CM Vagrant Script):

<https://github.com/cloudmesh-community/cm/tree/master/cm4/vagrant>

- Other (Ansible Spark Cluster):
<https://github.com/swsachith/pi-spark-cluster>
- Other (Config file): <https://github.com/cloudmesh-community/cm/tree/master/cm4/configuration>
- Other (Counter): <https://github.com/cloudmesh-community/cm/tree/master/cm4/configuration>
- Other (CM4 REST API): https://github.com/cloudmesh-community/cm/tree/master/cm4/flask_rest_api

fa18-516-23, Anand Sriramulu

- Section: <https://github.com/cloudmesh-community/book/blob/master/chapters/faas/openfaas.md>
- Chapter: <https://github.com/cloudmesh-community/book/blob/master/chapters/pi/setup-multiple.md>
- Paper: <https://github.com/cloudmesh-community/cmburn/blob/master/README.md>
- Project: <https://github.com/cloudmesh-community/fa18-516-23/blob/master/project-paper/paper.md>

Gregor von Laszewski, Albert Zweistein
laszewski@gmail.com, zwei@example.edu
Indiana University, Example University
hid: fa18-523-000, fa18-523-001
github: [cloud](#)

Keywords: Cloud, Example

Please introduce the topic here in contrast to a report, you do not need a section introduction.

One possible way of structuring the document. We may have to tweak this example as we progress.

Make sure paragraphs are 80 chars wide

Place images in an images directory

Use empty lines before and after headings

In [1] we can find a sample report.

Naturally the headings are just suggestions and you may change them as appropriate for your project.

2.1 EXAMPLE

All images must be referred to in the text. The words below and above must not be used in your paper for images, tables, and code.

Figure 2 shows a nice figure exported from Powerpoint to png. If you like you can use this as a basis for your drawings.



Figure 1: A simple flow chart

Figures must not be cited with an explicit number, but automated numbering must be used. Here is how we did it for this paper:

```
+@fig:fromonetothether shows a nice
figure exported from Powerpoint to png.
If you like you can use this as a basis
for your drawings.
```

```
![A simple flow chart](images/from-one-to-the-other.png){#fig:fromonetothether}
```

If the paper is copied from another source you MUST use a citation in the caption.



Figure 2: A simple flow chart [1]

This is done as follows, where all of this is in one line without spaces between the various brackets

```
![A simple flow chart [@vonLaszwska-fa18-sample-report]](images/from-one-to-the-other.png){#fig:fromonetothether}
```

Introduce sections as needed. If you write a very long text over multiple pages consider introducing sections and subsections.

2.2 WORKBREAKDOWN

Only needed if you work in a group.

Paul Filliman
pfillima@iu.edu
Indiana University
hid: fa18-516-06
github: [pfillima](#)

Keywords: Azure

3.1 ABSTRACT

This chapter focuses on an overview of the many data services highlights within the Microsoft Azure cloud. We detail the different relational and non-relational NoSQL databases as well as the many data analytics services.

3.2 DATABASE PRODUCTS

3.2.1 Azure SQL Database

3.2.1.1 Overview

The Azure SQL database is the cloud-based, SQL Server database as a service, relational database engine using the latest version of the SQL Server. There are many advantages to using an Azure SQL database as opposed to an on-premises SQL Server database platform. There are also many pricing level choices based on a function of hardware resources used.

3.2.1.2 Advantages

The biggest advantage to using an Azure SQL database rather than an

on-premises SQL database is scalability. Users can choose from many options of pricing model depending upon the utilization of their needs. Companies can start out with a low cost Azure SQL database having a fully managed database platform without the expense of an on-premises server and administrative costs and quickly scale up to a higher-cost pricing model with expanded system resources [2]. Other major benefits are database high availability and low administrative duties with operational database administrator or Windows server administration duties.

3.2.1.3 Pricing Models

There are two purchasing models. The Database Transaction Unit (DTU) based model and the vCore based model. The DTU model uses a measure using a combination of compute, memory, and storage [3]. In the DTU purchasing model, users can choose from three different configurations, Basic, Standard, and Premium, corresponding to the extent of the needed resources. Within the vCore model, users can individually choose the compute, memory, and storage values. The Gen4 generation allows for up to 24 virtual CPU cores and 168 GB memory and the Gen5 generation allows for up to 80 virtual cores and 408 GB memory. The maximum data size within the Gen4 is 1 TB and with Gen5 it is 4 TB.

3.2.1.4 Creating an Azure SQL Database

The creation of an Azure SQL database is very easy:

1. Log in to the Azure portal
2. From the Azure portal, select Create a Resource, then choose SQL Database within Databases
3. Enter the name of the database to create
4. Enter the container for the resource group, create a new resource group, if desired
5. Choose if this created database will use an elastic pool—example
6. Select the pricing model

7. Click the Create button

Microsoft Azure

Home > New > SQL Database

SQL Database

Create a resource

All services

FAVORITES

- Dashboard
- All resources
- Resource groups
- App Services
- SQL databases
- Azure Cosmos DB
- Virtual machines
- Load balancers
- Storage accounts
- Virtual networks
- Security Center
- Cost Management + Bill...
- Help + support
- Data factories
- Data Lake Analytics
- Data Lake Storage Gen1
- HDInsight clusters
- IoT Hub
- SQL servers
- Data Catalog
- Machine Learning Studi...
- Machine Learning Studi...
- Marketplace

Database name: test1

Subscription: Visual Studio Enterprise (8b278fc5-e24f-4641-9...)

Resource group: SQLRG

Select source: Blank database

Server: pfillimansql (East US)

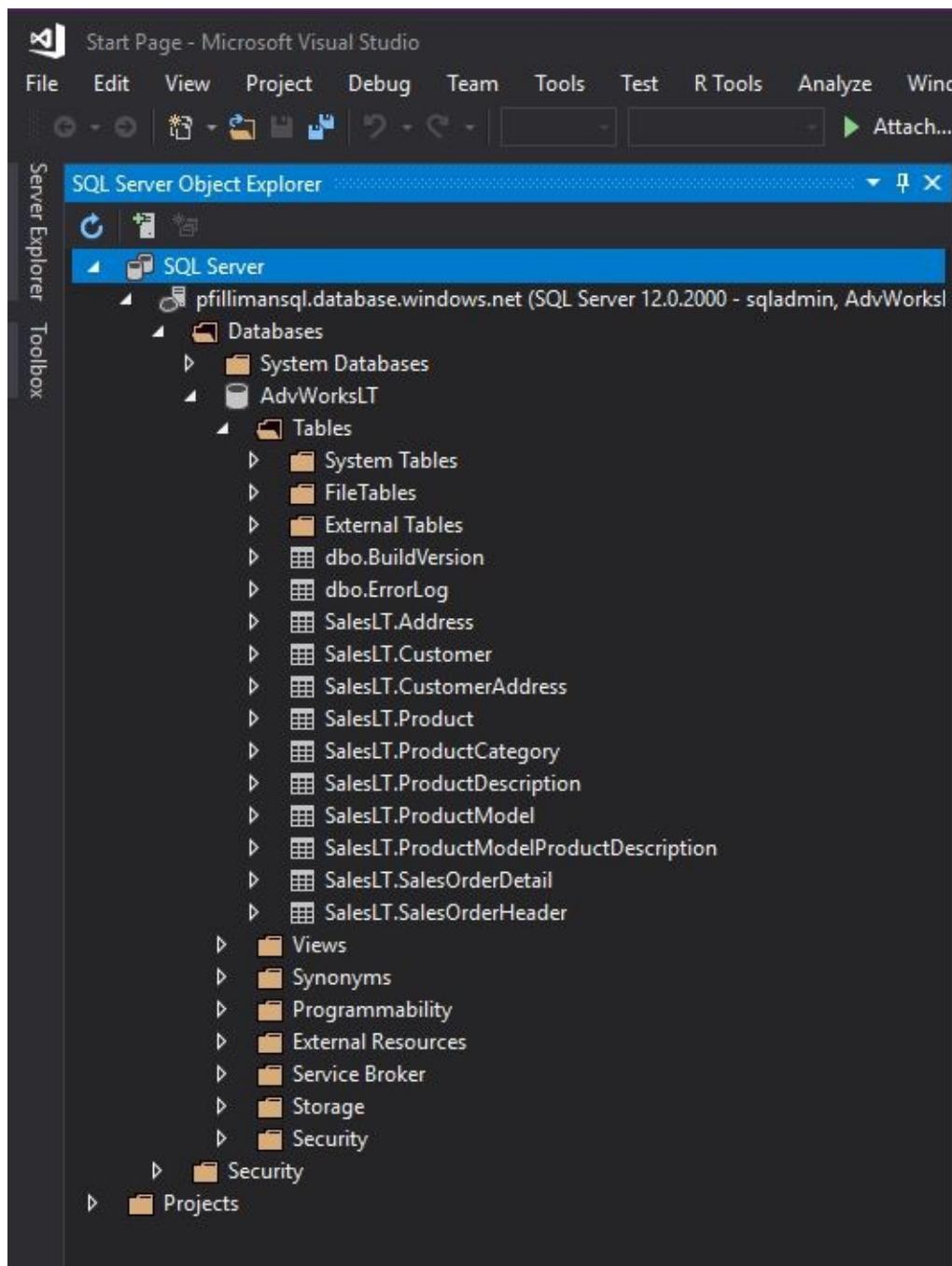
Want to use SQL elastic pool? Yes Not now

Pricing tier: Standard S0: 10 DTUs, 250 GB

Collation: SQL_Latin1_General_CI_AS

Create Automation options

Once the database has been created, we can use Microsoft Visual Studio as the development tool to the new Azure SQL database, much like an on-premises database using SQL Server Management Studio, as shown in +???.



3.2.2 Azure MySQL, PostgreSQL, and MariaDB Databases

Within the Azure ecosystem, it is possible to use three different open-source databases, MySQL, PostgreSQL, and MariaDB. Each of these are the cloud-based community versions of the databases. Much like Azure SQL, these have the benefits of using a cloud-based database service, for example scalability and uptime [fa18-516-06-AzureOpenSourceDB]. These Azure relational database allow users to keep using their desired open-source database platforms in the Azure cloud environment.

3.2.3 Azure Cosmos DB

The Azure Cosmos DB offers various multimodel, highly available databases for world-wide use. Cosmos DB supports many NoSQL data models including document, graph, key-value, and column-family models and is built on the atom-record-sequence data model which supports many APIs including MongoDB, Cassandra, Gremlin, and SQL [4].

Cosmos DB uses turnkey global distribution by distributing data near to where the current users are located to enable low network latency [4]. This is done through the multi-homing APIs where an application is aware of the location of the application user and can move data to the closest Azure region.

Cosmos DB service has high availability and throughput service level agreements, including a 99.999% availability and IO reads of less than 10 ms and IO writes of under 15 ms [5]. Users needing a highly available NoSQL database at a global scale, such as global web application databases, could gain from using Cosmos DB.

3.2.4 Azure SQL Data Warehouse

The Azure SQL Data Warehouse is a cloud-based, data warehouse that uses massive parallel processing for use with querying large amounts of data. The Azure SQL Data Warehouse uses Azure virtual machines for the compute nodes and Azure page blobs for storage. This separation allows for scalability for compute and storage

independently [6].

One of strengths of Azure SQL Data Warehouse is its ability to ingest modern data sources, for exampledatalakes and Hadoop as shown in the figure below. With the ability of using Polybase, a user can query non-relation as well as relation data sources that are stored in Azure SQL Data Warehouse [7]. Various Azure services can be used having the Azure SQL Data Warehouse as a source, including Azure Analysis Services, other Azure SQL Data Warehouses, and Azure SQL Databases.



Figure 3: Azure Data Warehouse [8]

Another strength is the ability to only use this service during a particular time of day or week. If the data warehouse user only need access during a regular work week, this could save cost rather than running this service all of the time. Much like Azure SQL Database described above, this has high-availability and backup and recoverability features as well [6].

3.3 ANALYTICS

3.3.1 Azure HDInsight

HDInsight is the Azure services for clustering Apache Hadoop, Apache Spark, Kafka, Apache HBase, Hive, and Storm and are built around the Hortonworks Data Platform. The concepts of the Hadoop ecosystem go beyond the scope of this chapter, but this section is an overview of the different HDInsight services available and how they can be used within Azure.

Azure HDInsight services are typically used when working with massive amounts of data in the internet of things and streaming real-time analytics scenarios. There are many ways under the HDInsight umbrella to setup clusters according to business needs. The following show for example configuring clusters using Apache Spark for parallel processing or Apache Storm for use with real-time streaming analytics. Apache HBase can be clustered in Azure for businesses needing a NoSQL database to store unstructured or semi-structured data. HBase brings very large tables having billions of rows and millions of columns. Apache Kafka can also be clustered under Azure HDInsight. Apache Kafka is a popular platform for streaming pipelines ???.

The following figure shows HDInsight within a modern data warehouse. There are multiple data sources from log files, and structured and unstructured data as batch processes for the HDInsight data sources. These data are into Azure Storage or Azure Data Lake Stores. Spark and HiveQL can then be used to query the Azure storage and these can be used to build business intelligence data models, for example Azure Analysis Services models. Finally, these data can be visualized using PowerBI.

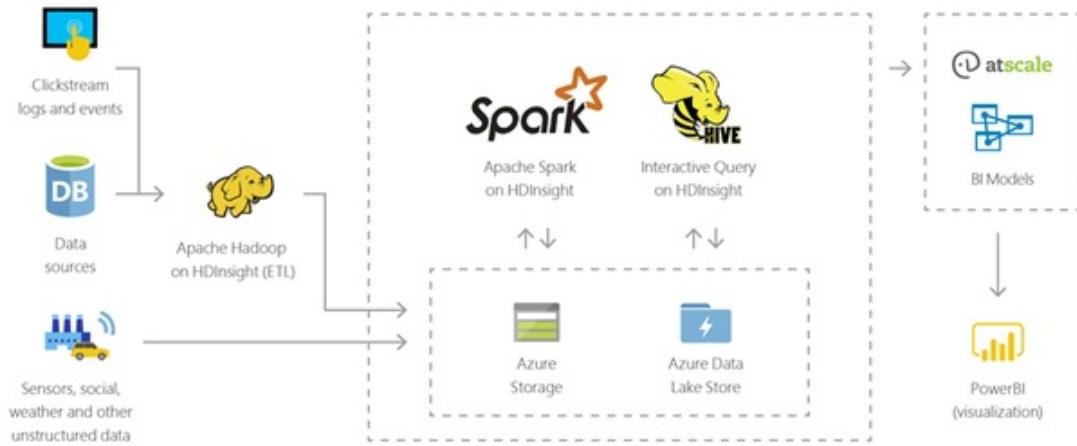


Figure 4: Modern Data Warehouse using HDInsight [8]

The next figure shows Azure HDInsight in an Internet of Things scenario. Various IoT streams can be fed into IoT hubs then read into HDInsight using the Storm, Kafka, or Spark services, then real-time visualizations or applications can be fed data from HDInsight.

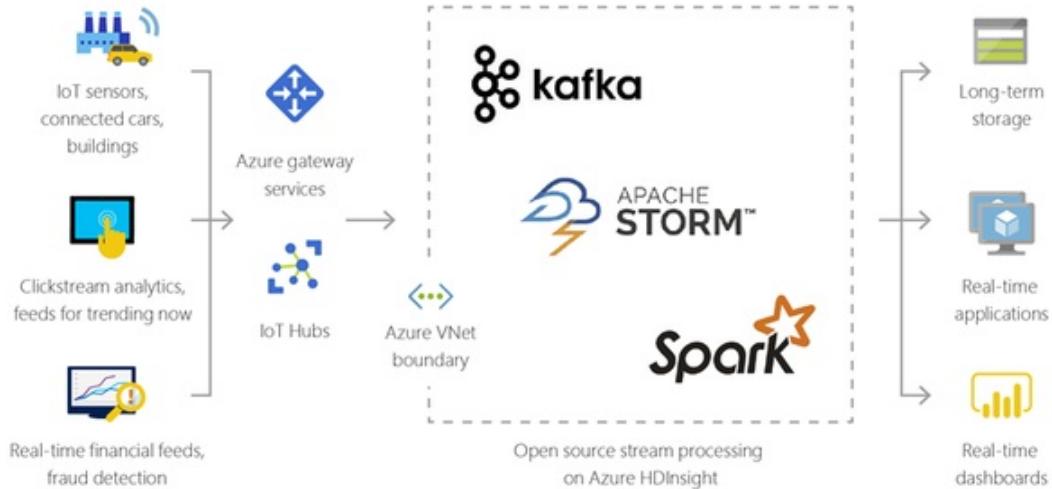


Figure 5: HDInsight in an IoT scenario [9]

One of the strengths of HDInsight is that these services are available in Azure without the work of implementing these clusters in on-premises servers and also having seamless integration with other Azure services. These services have high performance, five nines (99.999%) SLA and can be used on a per-use basis therefore cutting

costs of permanent uptime.

3.3.2 Azure Stream Analytics

The Azure Stream Analytics service processes output from various IoT sources and can be used to analyze real-time data. Real-time data analytics is needed when data is in movement, for example, in cases such as detecting fraudulent bank transactions before the account is deducted. In past analytic systems, where an ETL load happened once per day, this system could not detect this transaction in real-time. Azure Stream Analytics is the service that manages these continuous real-time output.

Azure Stream Analytics is a part of the Azure IoT suite and ingest data from the Azure IoT Hub as well as Azure Event Hubs, Blob storage, and other relational or non-relational data sources. Once ingested into Azure Stream Analytics, real-time analytics can be gained using machine learning algorithms, for example detecting a fraudulent bank transaction. The data output from Azure Stream Analytics can also be loaded into other and uses is a part of the IoT.

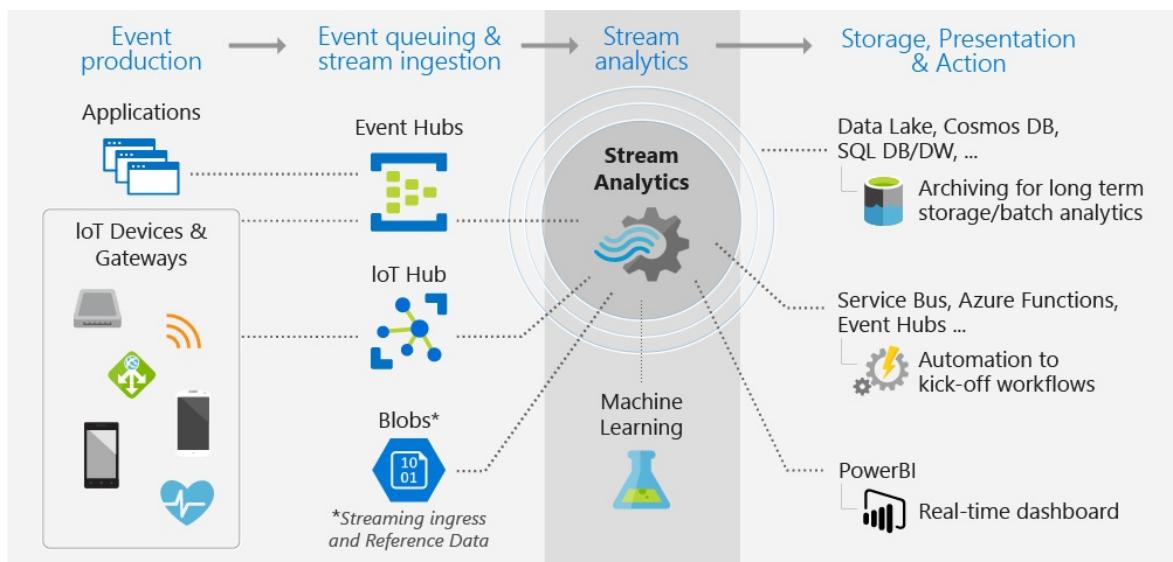


Figure 6: Azure Stream Analytics [10]

There are three basic parts to using Azure Stream Analytics. The first part is creating a stream job which designates the data source and

uses a query language similar to SQL to make any transformations on the incoming data. The third step is specifying where to output the data.

3.3.3 Azure Data Lake Store and Data Lake Analytics

Data lakes are scalable repositories of data stored in its original format. The Azure Data Lake Store allows users to store data within a Hadoop Distributed File System (HDFS) -compliant file system for use with big data analytics. Azure Data Lake is a cost-effective way to store scalable unstructured data in secure, active-directory environment [fa18-516-06-ADLS1].

The latest release of Azure Data Lake in June, 2018, named Gen2, is multimodal in that there is both BLOB object storage and now file system storage. This version has a Hadoop file system with hierarchical directories which allows for higher performance than a flat object namespace. This new feature in Gen2 can eliminate unneeded REST service calls, for example in moving files. Instead of separate REST service calls for copying a file to a new location and another for deleting the file from its original location, with Gen2 this process can be done in a single operation using file system storage [11].

Together with Azure Data Lake is Azure Data Lake Analytics. This service provides methods for running analytics job at a pay per use cost. The creation of data lake analytics jobs can be done using Visual Studio and U-SQL to load and transform data. Azure data lake analytics can also be used with data sources from Azure SQL Database, Azure Storage, and Azure SQL Data Warehouse, as well as the Azure Data Lake Store [12].

3.3.4 Azure Data Factory

Azure Data Factory is the integration engine within Microsoft Azure. This data service is responsible for automated movement of both structured and unstructured data within Azure and on-premisis data

repositories. This work is accomplished by source and target connections together with pipelines between those connections and activities. Azure Data Factory can run in typical data warehouse environments as an extract transform and load workflow using the Azure-SSIS runtime as well as with big data workflows using unstructured data Azure HDInsight or Azure Data Lake [13].

A pipeline is a task within a data factory that comprises activities. For example, a pipeline can be used as a copy task or a data transformation task. Pipelines can be scheduled as a one-time event, hourly, daily, etc.

An activity within Data Factory is either a copy utility or a data transformation utility. A data copy utility has numerous sources and targets which can move data between cloud and on-premises relational and NoSQL databases. A data transformation utility can manipulate the data from the previously mentioned data stores using Data Lake U-SQL queries, an HDInsight Hive or Pig activities [14].

Varun Joshi
vajoshi@iu.edu
Indiana University
hid: fa18-516-08
github: [blue user icon](#)

Add Learning outcome

introduction focusses to much about what we will learn, focus on content, use the learning outcome bullet section we use in the handbook instead

introduce the problem instead

bibtex issues

urls done wrong

The general question should be if this chapter is actually Cloud Computing and privacy and GDPR is a subsection to it.

4.1 GDPR COMPLIANCE AND CLOUD COMPUTING

European Union's General Data Protection Regulation (GDPR) came into effect on May 25, 2018.

this is not an introduction, but an explanation what is covered, which is fine. But so far there is no information covered, and we may need to discuss if this should be changed or deleted. we also need a learning outcome section as we use in other epubs.

We will cover the guidance of GDPR to cloud computing and how it applies to the businesses inside and outside European Union. We will touch upon the terms data processor and data controllers for cloud computing as related to GDPR compliance. We will also cover GDPR

readiness and privacy statements of some common cloud vendors like Amazon, Microsoft and Google. We will also present general market outlook in the wake of GDPR for preference of public vs private cloud.

4.1.1 GDPR Compliance

The core of the GDPR compliance is to protect EU citizens from privacy and data breaches **???** Key Changes. It aims to give back the control of personal data to citizens and residents.

We may wonder that GDPR is applicable only for protecting EU citizens and the organizations based outside of EU need not be GDPR compliant. However, GDPR applies to any organization with business in EU and collect, store and process data of EU citizens. With the digital age and the organizations moving towards cloud computing, the GDPR brings new challenges both for cloud computing vendors who have data centers in EU as well as for organizations like Uber, Visa, Apple and many more who are ubiquitous in their business models and deal with EU citizens personal data.

??? Definitions lists personal data as defined in Article 4 of GDPR:

“personal data means any information relating to an identified or identifiable natural person (data subject); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person;”

In simpler language it could be any information like identification number (US equivalent for SSN), phone number, address, birth date, IP address etc. which can uniquely identify a person.

GDPR compliance poses strict fines for any personal data breach.

Fines are either up to EUR 20 million or 4% of annual revenue, which ever is higher.

To summarize, GDPR compliance will protect EU citizens personal data and will enforce organizations worldwide to be GDPR compliant. Under GDPR, it is mandatory for organizations to disclose how and when personal data is collected and where it is stored, what it is used for and how the data will be erased when the data subject is no longer needed or chooses to opt out. Organizations need to explain in simplest terms to the data subject about usage of their personal information and give them option to opt out if they choose to do so. In case of data hack, the data subjects should be notified immediately when the hack happens.

In technical terms to be GDPR compliant, specifically for cloud computing use case, cloud users and cloud providers will collectively be responsible for encryption of data (both at rest and in transit), should have ability to monitor the usage, authorized access control of data, choose data storage - geolocation/type of storage/data access, dedicated data protection officers and industry certifications for data security. Privacy by design is now a legal obligation.

Now that we are familiar with the GDPR compliance, in next sections we will look into its impact specifically for cloud computing platform and data privacy in cloud data centers. Before that, let's define Data Processor and Data controller with respect to cloud and as related to GDPR.

4.1.2 Data Processor vs Data Controller

Cloud solutions like AWS, Azure, GCP are all considered data processors because they offer resources and infrastructure to process the data.

Organizations, authority or agency which collect and direct the personal data and define mandate on how the collected personal data is processed are known as data controllers.

For example an organization like Airbnb collects personal data of its users and customers and controls the usage of the personal data like how and where it is stored, how it is used - such as providing rental suggestions or generating analytics on usage statistics. Airbnb uses AWS as its operation infrastructure. In this example AWS is data controller whereas AWS acts as data processor.

AWS also acts as data controller for the data it collects like user account registration information of its customers, administration, service access etc.

Collection, storage, recording, organizing, structuring, alteration, consultation, retrieval, sharing, restriction and erasing and destruction all come under personal data processing.

GDPR defines collective responsibility for both data processors and data controllers for safeguarding personal data.

The defining of roles extends further if there is a third-party involved between cloud solution vendors and cloud users. For example if a company is using services of a third-party and the third-party is using cloud solution vendors directly then the roles should be understood clearly for being GDPR compliant and avoiding audit and fines.

Now we have understood the difference between data processors and data controllers, let's look in to its impact on cloud computing by relating it to GDPR.

4.1.3 Impact On Cloud Computing

GDPR imposes collective responsibility on data controllers and data processors for personal data protection. Organizations or cloud users who deal with the personal data of their customers or consumers of their applications should be careful in choosing a cloud solution which is GDPR compliant and provides infrastructure and services options which are GDPR compliant. Data controllers should have options to define data privacy and security operations within the cloud infrastructure. Taking example of AWS as data processor, the

resources like EC2, EBS, Amazon VPC all offer operations mechanism for a data controller to configure for robust data privacy and security. At the same time, AWS as a data processor needs to disclose in its contract with the data controller the options it provides for data storage and region and site for each chosen services.

Since data protection is a collective responsibility and design by principle, the data controller will have to keep the following check list [15] when choosing a cloud solution provider:

- Options to configure resources and desired settings as related to the data Privacy
- Ability to get snapshot of the current configurations in cloud
- On demand retrieval of configurations
- Historical logging
- Ability to get automatically notified of any changes in configuration
- View how resources communicate in cloud infrastructure
- Ability of cloud provider to encrypt the data either in transit or rest
- Options to set data access controls - granular access to data, multi factor authentication, geo-restrictions on data access

In summary, data controllers should define and are responsible for defining all data privacy and security rules when using a cloud infrastructure and resources. Data processors are responsible for providing resources and services to be GDPR compliant. If the processing happens with a third-party, the information needs to be disclosed and again the responsibility is collective.

Complexity the compliance may cause changes to how the cloud computing infrastructure is used by the organizations. The debate would be between private or public cloud for services which deal with the personal data. Let's look more in detail in next section.

4.1.4 Public or Private Cloud

Let us define few terms before we dig deep into the choice of public

or private cloud given GDPR compliance.

this has been defined elsewhere in the book and either should be moved there if not and we simply reference here and how it is relevant, we can use the terms here with a back reference to that section.

- Private cloud is an internal or enterprise dedicated cloud where data center resides and belongs to the enterprise and is behind enterprise firewall [16]. The enterprise is solely responsible for managing and scaling the infrastructure as needed. Examples are AT&T, Cisco, T-Mobile have their own private cloud hosted in their own data centers. the nist definition must be used first. Expedient did not define these terms but nist did
- Public cloud is a multi-tenant architecture where the cloud infrastructure provider hosts many enterprises in its data centers [16]. The enterprises and their data are virtually separated in the data center and enterprises have their own virtual private cloud network (example Amazon VPC). The cloud provider is responsible for managing and scaling the infrastructure at the data center. The enterprises can choose to individually upgrade and update the resources for patching, security etc. Examples are AWS, GCP, Azure.

Based on the above concepts, there could be multiple iterations of the cloud solution.

- Hybrid cloud is one such solution where there is a mix usage of on premise data center and public cloud data center. The use case could be based on mission critical applications, data privacy, need for on demand scalability, high availability etc.
- Managed cloud services provides enterprise an option to outsource the management of the cloud infrastructure and services to a third-party company like Rackspace or Expedient. Again the managed cloud service option depends on the need of usage. Managed services can be leveraged on private cloud, public cloud or companies like Rackspace,

Expedient provide their own data centers which can be dedicated completely to an enterprise for their infrastructure needs and managed by the providers. Rackspace, for example, uses Openstack software for their managed cloud services solution.

Now with the knowledge of above concepts, it's clear to define the cloud strategy. Keeping in mind the requirements of GDPR compliance and the responsibility of data controllers and data processors, a cloud solution can be chosen which is secured, robust, optimized and cost-effective.

Highly secured and sensitive data, for example HIPAA, can be managed in a private cloud or hybrid cloud. Other sensitive personal data which requires services of third-party for analytics generation like movie recommendation apps, shopping recommendation, election surveys, likes, social mining etc. can leverage public cloud scaling in a virtual private network utilizing GDPR compliant cloud data processor and rules and security defined by data controllers.

One important consideration is while using opensource solution like Openstack. Openstack can be used in a managed cloud service setting or independently for private cloud solution. The key is to use open source resources and their configurations which provide robust data security for compute, storage, network etc. which are integrated in Openstack software[17].

How GDPR and other data privacy compliances will shift the revenue model of major cloud vendors will be an interesting trend to observe. The trend will also reflect choice of enterprises for cloud solution provider in their journey to achieve less overhead of maintaining data centers, achieving scalability and at the same time protecting the interests of data subjects.

4.1.5 Common Vendors GDPR Readiness

Major cloud solution vendors like AWS, GCP and Azure are GDPR compliant and offer resources, services and configurations which are

GDPR ready. Other vendors offering specifically SaaS and PaaS are also GDPR compliant. Privacy statements of vendors has also been updated to reflect their GDPR readiness. Refer the following for major vendors GDPR readiness:

wrong

- AWS

[<https://aws.amazon.com/blogs/security/all-aws-services-gdpr-ready/>]

- GCP

[<https://cloud.google.com/blog/topics/inside-google-cloud/google-cloud-ready-for-gdpr>]

- Azure

[<https://techcommunity.microsoft.com/t5/Enterprise-Mobility-Security/Assess-GDPR-readiness-with-Microsoft-Cloud-App-Security/ba-p/250572>]

Example of updated privacy policy in the wake of GDPR:

- Redhat

[<https://www.redhat.com/en/about/privacy-policy>]

Important takeaways from the RedHat privacy statement [18] is that Redhat claims the following rights:

- The right to access your personal data;
- The right to rectify the personal data we hold about you;
- The right to erase your personal data;
- The right to restrict our use of your personal data;
- The right to object to our use of your personal data;
- The right to receive your personal data in a usable electronic format and transmit it to a third party (also known as the right of data portability); and

- The right to lodge a complaint with your local data protection authority;

it is unclear in what relationship this is used and on what this is applied Is is the OS is it a service is it something else ...

Murali Cheruvu
mcheruvu@iu.edu
Indiana University
hid: fa18-516-11
github: [blue icon](#)

citations in captions missing (as is now = plagiarizm if citation is not added)

conlusion reads like advertisement. I am wondering if the conclusion needs to focus on technical issue. Maybe it actually needs to go in the introduction and the conclusion may need to be deleted. However, if used, you may want to rethink the advertisement aspect that this conclusion seems to reflect.

12-factor application development is not explained, thus the conclusion can not be understood.

this could actually contributed to the book, if it woudlnot be for the small issues pointed out here.

Keywords: Cloud Foundry, Open Source, PaaS

5.1 CLOUD FOUNDRY OVERVIEW

Cloud Foundry (CF) is, an open source, Platform as a Service (PaaS) available through a variety of private and public cloud distributions, which is developed and operated by VMware and then transferred to Pivotal Software, a joint venture by EMC, VMware and General Electric. Pivotal Cloud Foundry is an alternative paid version with additional commercial features, support and documentation.

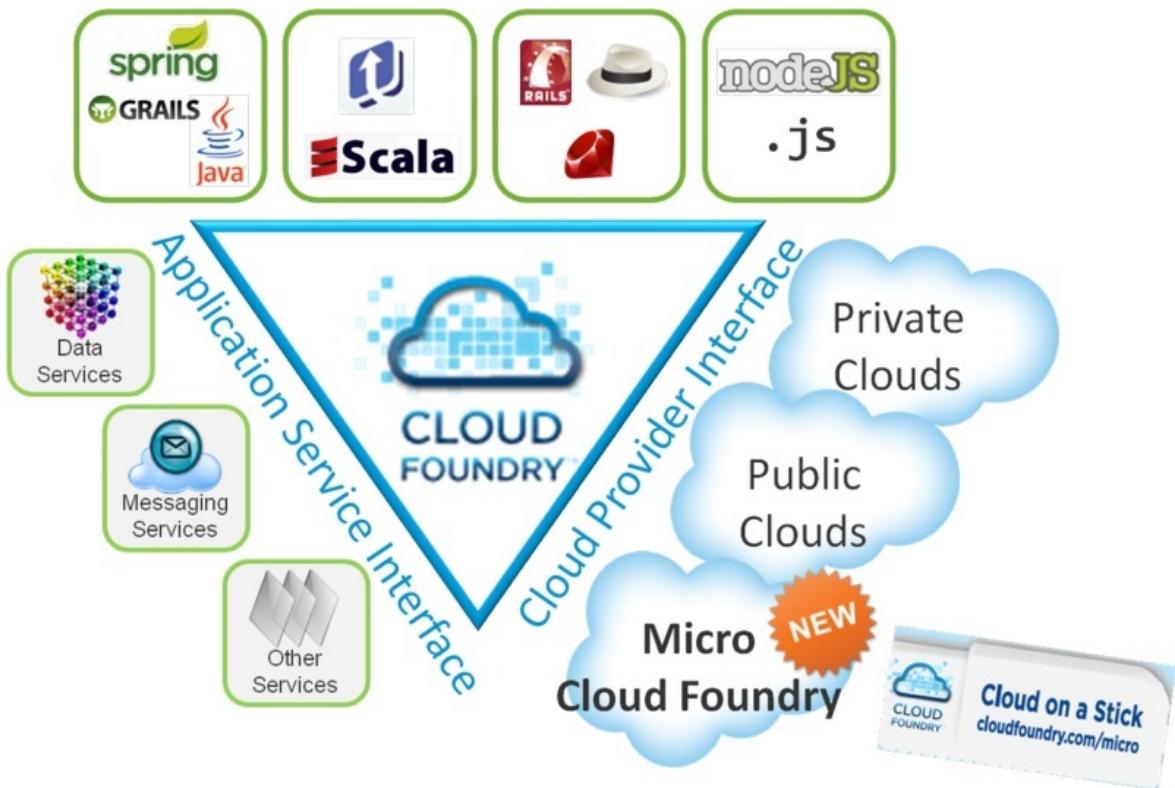


Figure 7: Cloud Foundry - Open PaaS

As opposed to most of the PaaS offerings in the market, which are tied to proprietary implementations, Cloud Foundry is an open source software with flexibility to allow integrations with external systems. Cloud Foundry provides all the PaaS capabilities like the popular public and private PaaS Providers - built-in scalable infrastructure, middleware, and various tools for development, deployment and support. To enable infrastructure-agnostic architecture, Cloud Foundry, as shown in Figure 7 focused on three main categories: Clouds, Frameworks and Services [19].

5.1.1 Category: Clouds

Public and private clouds have their advantages and disadvantages. While public clouds provide flexibility and faster deployments, private clouds offer operational efficiency and total control. Hybrid cloud approach gives best of both public and private cloud offerings: infrastructure scalability, deployment and monitoring tools, data locality, industry regulations, zero-changes to the existing

applications or develop with cloud-native mindset - API Gateways in combination of Micro-Services. Cloud Foundry is an open PaaS with ability to extend and collaborate with other private and public cloud systems. It can co-exist with other PaaS and IaaS platforms, in a way, it can be hosted on top of popular cloud environments like AWS, Azure, Google Cloud and OpenStack.

5.1.2 Category: Frameworks

Most of the cloud environments are restricted to fewer frameworks and programming languages. Though there is a good coverage of runtime environments and programming languages offered by most of the public clouds including AWS and Azure, there are still some restrictions either in the way we will need to use them or in the pricing and licensing models. Cloud Foundry, at architecture level, is generic and intended to host any programming language. It is written in Ruby and currently supports popular programming languages like Spring, Java, Ruby and NodeJS.

5.1.3 Category: Services

Similarly most of the clouds have limited set of support for various types of data, messaging and other services, restricting companies and development teams by forcing them to use specific technologies.

Cloud Foundry, out of the box, shipped with support to various services like relational (MySQL, PostgreSQL), NoSQL (MongoDB), Key-Value pair (Redis) databases and RabbitMQ message queuing system along with extensibility to allow third-party systems to be added later.

5.2 KEY COMPONENTS

Cloud Foundry comes with lots of ready-made components to support all the key aspects of PaaS cloud computing in a scalable fashion [20]. Figure 8 is a good representation of the major components of Cloud Foundry:

Cloud Foundry Architecture

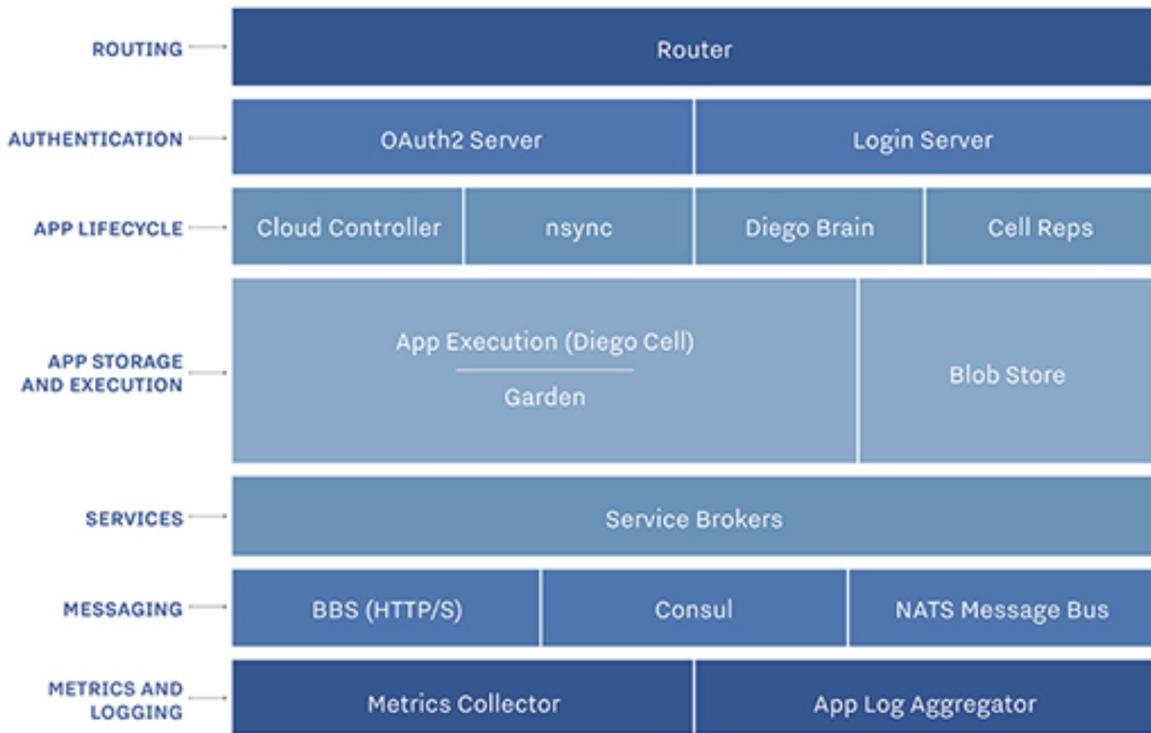


Figure 8: Cloud Foundry Architecture

5.2.1 Router

Router is responsible for controlling all the external and application level traffic, and also directing the incoming traffic to appropriate components. Router configuration allows to have number of routers to enable proper load balancing and high availability of the cloud foundry environment. Each router maintains a dynamic route table with all details of the deployed applications. Router interacts with Cloud Controller and Droplet Execution Agent (DEA) to facilitate the updated routing information across the Cloud Foundry Environment. Router is implemented in Go programming language that can offer optimal performance.

5.2.2 UAA and Login Server

User Account and Authentication (UAA) and Login Server components

are the identity management system in CloudFoundry. Cloud Foundry uses OAuth2 (Open Authorization) standards driven token- based authentication and authorization to manage user security tokens.

5.2.3 Cloud Controller

We can think of Cloud Controller (CC) as the brain of Cloud Foundry Environment and the main responsibility of Cloud Controller is to manage the applications life cycle - deployment, application metadata, staging and running the applications. CC uses Diego Brain, CC-Bridge and Diego Cells to stage and run the applications. CC redirects the first requests to the appropriate Droplet Execution Engine (DEA) available in the load balancing pool. CC user permissions are maintained at various levels - Orgs, Spaces and Roles for greater scalability to the role-based access control. Application deployment artifacts - code packages, build-packs and droplets are maintained in Blob Store.

5.2.4 Execution and Storage

Droplet Execution Engine (DEA) is responsible for application deployment and runtime management - selecting appropriate build-pack, stage the application and ensuring end-to-end life cycle management of the application instance. Build packs are the scripts to identify the required framework for applications to run properly. Droplet is the unit of execution - deployed build pack of the application along with the application metadata. Wardens are containers to host the droplets and isolate them in resource-controlled environments. BOSH is used as tool for release management of complex distributed systems.

5.2.5 Service Brokers

Most of the applications have aspects like interacting with database, sending messages using Service-Oriented Architecture (SOA) and interfacing third-party components. Services cannot directly interact with the applications given they run in the containers which are not

persistent. Cloud Foundry uses service brokers in a decoupled fashion through which application developers can facilitate and use the services in applications. Cloud Controller uses NATS based messaging system. It uses a light-weight publish-subscribe mechanism to distribute the queued messages among applications.

5.2.6 Monitoring and Logging

CC provides various tools for continuous monitoring - (a) Health Manager: Responsible for monitoring health of the application instances by interacting with DEA and CC, (b) Metric Collector: Gathers various application metrics from the running instances, and (c) Log Aggregator: Streams application logs. Developers and support teams can access these logs and metrics to monitor, support and take necessary actions to keep systems up and running.

5.3 CONCLUSION

Lots of Fortune 500 companies from all over the world - various industries and government organizations, rely on Cloud Foundry for all the benefits of flexible open source PaaS offerings [21]. Cloud Foundry is setting the new standards for cloud computing with emphasis on scalability and industry best practices by promoting 12-factor application development [22].

6 CLOUD SECURITY ALLIANCE (CSA) FA18-516-17

lots of wrong punctuations YOU CAN NOT HAVE THE CITATION AFTER THE DOT AS IT OTHERWISE APPLIES TO THE NEXT SENTENCE

bulleted list use phrase such as according to - Fixed below! it was not fixed, so I fixed it

this is incomplete as entire sections in the introductions are missing according to you, therefore this will not be further reviewed till not fixed - BP - Fixed below!

no superscript as char use markdown for that - BP - took out the incorrect superscript character

github: [!\[\]\(e83b2296dc4c9009963f494078a8a780_img.jpg\)](#)

6.1 INTRODUCTION TO CSA

The Cloud Security Alliance (CSA) is a nonprofit organization that provides a variety of security resources to institutions including guidelines, education and best practices for adoption. There are many different audiences and variety of cloud and compute configurations. As this is the case, the CSA has a diverse membership, a variety of research areas and a breadth of recommendations and guidelines. In this chapter CSA topics include:

- About the CSA
- Guiding Principles
- History
- Research areas
- Membership
- Best ways to leverage CSA

6.2 ABOUT THE CSA

The Cloud Security Alliance (CSA) is dedicated to defining and raising awareness of best practices to help ensure a secure cloud computing environment. The CSA leverages the expertise of industry practitioners, associations and governments, corporations and individual members. They offer research, education, certification, events, and products specific to cloud security that benefit the entire community. They essentially provide a forum through which these different parties can work together to create and maintain these recommendations for a trusted cloud ecosystem. [23] The industry group also provides security education and offers guidance to companies in different stages of cloud adoption. They also offer certification programs for cloud security providers and manage a global consulting program that allows participants to work with a network of qualified cloud security professionals. [23]

6.3 GUIDING PRINCIPLES

The Cloud Security Alliance members are united by the following objectives according to their LinkedIn page:

- "Promote a common level of understanding between the consumers and providers of cloud computing regarding the necessary security requirements and attestation of assurance.
- Promote independent research into best practices for cloud computing security.
- Launch awareness campaigns and educational programs on the appropriate uses of cloud computing and cloud security solutions.
- Create consensus lists of issues and guidance for cloud security assurance." [24]

6.4 HISTORY

The Cloud Security Alliance was forged due to the need to have to have security best practices in a cloud computing environment [25].

As the concept of cloud computing increased in popularity in 2008, so did the surrounding issues and opportunities. The information security community took note and in November at an ISSA CISO Forum in Las Vegas the concept of the Cloud Security Alliance was born. After acknowledging emerging trends, proactive participants outlined the initial mission and strategy of the CSA. What followed was a series of meetings with industry leaders later that year which codified the foundation of CSA as we know it today.

6.5 RESEARCH AREAS

The CSA currently has working groups that cover 38 domains of Cloud Security. These working groups publish a variety of white papers, reports, tools, trainings, and services that benefit the cloud security community.

For instance, the Blockchain working group meets every other week to discuss current events that surround Blockchain and any potential security implication. In addition, they published multiple documents outlining successful Blockchain projects, how to use the technology to secure the Internet of Things and a reference glossary of terms for the industry's benefit.

6.6 MEMBERSHIP

The Cloud Security Alliance employs roughly sixteen full-time and contract staff worldwide. It has over 400 active volunteers participating in research at any time. The CSA is a member-driven organization and individuals who are interested in cloud computing and have the experience to assist in making it more secure receive a complimentary individual membership based on a minimum level of participation.[26]

The Cloud Security Alliance has a network of chapters worldwide. Chapters are separate legal entities from the Cloud Security Alliance, but operate within guidelines set down by the Cloud Security Alliance

In the United States. Chapters are encouraged to hold local meetings and participate in areas of research. Chapter activities are coordinated by the Cloud Security Alliance worldwide. [26]

6.7 BEST WAYS TO LEVERAGE CSA

6.7.1 Security Guidance Publication

The Security Guidance for Critical Areas of Focus in Cloud Computing documentation provides guidance to support business goals while mitigating the risks associated with the adoption of cloud computing technology. It is derived from focused research, participation from the Cloud Security Alliance members, working groups, and industry experts. It incorporates new cloud technology, reflects on real-world cloud security practices, integrates the latest Cloud Security Alliance research projects, and offers guidance for related technologies. [27]

These security guidelines use NIST industry standards and are easy to read with models and illustrations that make key concepts more digestible for most readers.

6.7.2 Training and Certifications

6.7.2.1 CSA STAR

This is an industry-recognized security assurance in the cloud. STAR encompasses key principles of transparency, rigorous auditing, and harmonization of standards. According to the CSA website,

"STAR consists of three levels of assurance (Self Assessment, 3rd party certification and continuous auditing), based upon:

- The CSA Cloud Controls Matrix (CCM)
- The Consensus Assessments Initiative Questionnaire (CAIQ)
- The CSA Code of Conduct for GDPR

Compliance" [28]

6.7.2.2 Certificate of Cloud Security Knowledge (CCSK)

Since Cloud Security Alliance first released the Certificate of Cloud Security Knowledge (CCSK) in 2010, thousands of IT and security professionals have used it to upgrade their skills. Certification Magazine listed CCSK at number one on the Average Salary Survey 2016. [29]

Accoring to thier CCSP page,

"Certified Cloud Security Professional (CCSP) The CCSP is a global credential that represents the highest standard for cloud security expertise. It was co-created by Cloud Security Alliance and (ISC)² — leading stewards for information security and cloud computing security." [30]

wrong punctuation

6.7.2.3 CSA Global Consulting Program

"The Cloud Security Alliance Global Consulting Program (CSA GCP) allows cloud users to work with a network of trusted security professionals and organizations that offer qualified professional services based on CSA best practices. These providers bring with them a broad understanding of the challenges organizations face when moving to the cloud." [31]

wrong punctuation

6.7.3 Working Groups

As mentioned above working groups are a great way to learn and share with industry experts on topics you or your institution are

focused on. [32]

A list of upcoming meetings is published on the CSA site and meetings are open to the public to join!

Anna Heine
avheine@iu.edu
Indiana University
hid: fa18-523-52
github: [blue user icon](#)

Keywords: KNIME, workflow, workbench

7.1 INTRODUCTION

KNIME [33] stands for KoNstanz Information MinEr and is an open source data analytics software that creates services and applications for data science projects. KNIME allows its users to create visual workflows with a user-friendly drag and drop graphical interface that depletes the need for any programming. However, KNIME does allow implementation of other scripting languages such as Python [33] or R [33] that creates connections to abilities within Apache Spark or other machine learning tools. KNIME allows imports of datasets from a variety of formats, some of which include CSV [34], PDF [34], JSON [35] and more. The workflows and visualizations that KNIME produces allows export in many of these formats as well. It also supports several unstructured data types from images, documents, and certain networks. KNIME operates by a node system that includes embedded modules that help its users build their workflow. With this node system, users can make changes at every step of their analysis to ensure the most current version. KNIME also provides detailed visualizations from a set of defined graphs and charts which can lead to predictive analyses and machine learning implementations. Users can shape their data by a variety of mathematical models such as statistical tests, standard deviations, and means. Users can even select specific features for use in possible machine learning datasets and apply filters to mark out some of the

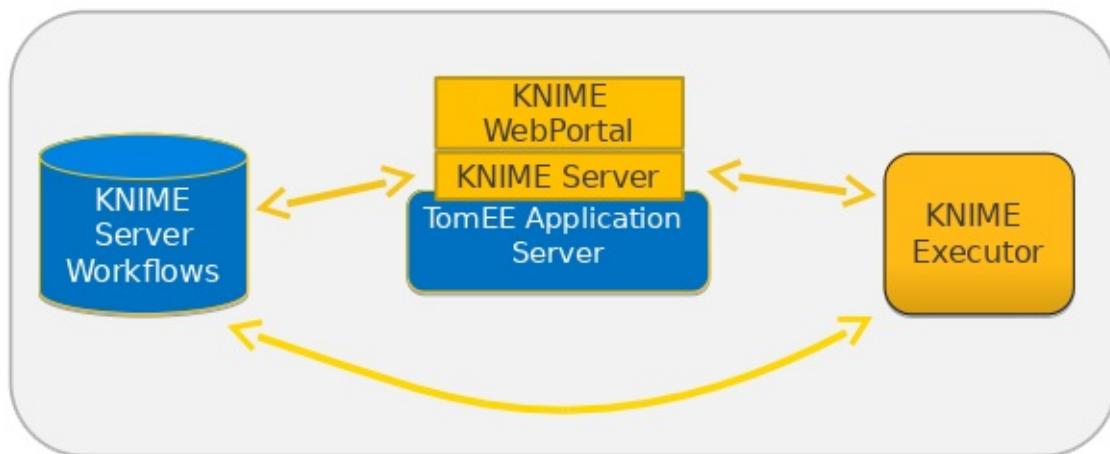
data if needed.

KNIME is a platform that can perform intense data analytics on a graphical user interface and incorporate a user-friendly workflow. It incorporates large or small data sets and even projects as broad as deep learning. KNIME is diverse in that its users do not necessarily need to know any coding languages to use it. KNIME is a process-oriented, single base workflow with basic input/output manipulations. KNIME is an open source platform that uses thousands of its documented nodes within the node repository for use in the KNIME workbench. A node is a single processing point of data manipulations within your workflow. A workflow is described as a sequence of steps a user follows in their platform that is used to complete their final product. The collection of nodes that creates a KNIME workbench is able to be executed locally or within the KNIME web portal on its own server. The workflow that KNIME follows first begins with data collection, data cleaning, data integration, and finally, feature extraction. This workflow allows for large files such as a CSV to be accessed through the web portal and it can therefore be manipulated through several wizards [36].

7.2 KNIME USED IN BIG DATA

KNIME is useful in Big Data applications because it aids in the process of guided analytics. Guided analytics is a process that functions by providing automation to data science projects. This usage brings to light some of the features of big data that are sometimes hidden when visualization tools are unused. Depending on the wizard the user selects, the data can then be viewed in a formatted table using forward feature selection methods. This means that the user can then select their data based on the most accurate correlating information. This will, of course, be varied based on the type of information that is entered into the table. The next step in the data-selection process is a second screen with a wizard that asks the user to choose a target variable. The next screen will then show to offer a selection of algorithms which the user can use to entrain the dataset. After choosing an algorithm, the user may browse from a list of

displayed visualizations that include their dataset such as bar graphs, an ROC curve, and more. The user can then download their data model in a PMML format, which is universally configurable in any enterprise application. The steps to download and set up your KNIME platform is quite simple. First, go to the KNIME website and obtain the download. Then, install KNIME and set its working directory for KNIME to store its files in. To set up a KNIME workflow, go to the File menu on the platform and choose New. Give your workflow a name and then click Finish. This process establishes a basic, empty workflow in which users can drag nodes from the repositories on the left-hand side into the workflow space. To increase user collaboration and support, KNIME also includes a Workflow Hub [37]. This hub allows users to share their workflows and make comments or suggest improvements to their designs. The components of a KNIME server after installation are hosted on the same machine. The components include: a workflow repository, the executor, the server. Figure 9 shows the simple architecture of KNIME's server for a single user. Figure 10 shows the basic KNIME platform setup with graphs, repositories, and the workbench.



Communication via RMI, HTTP, and local file system

Figure 9: KNIME Diagram [38]

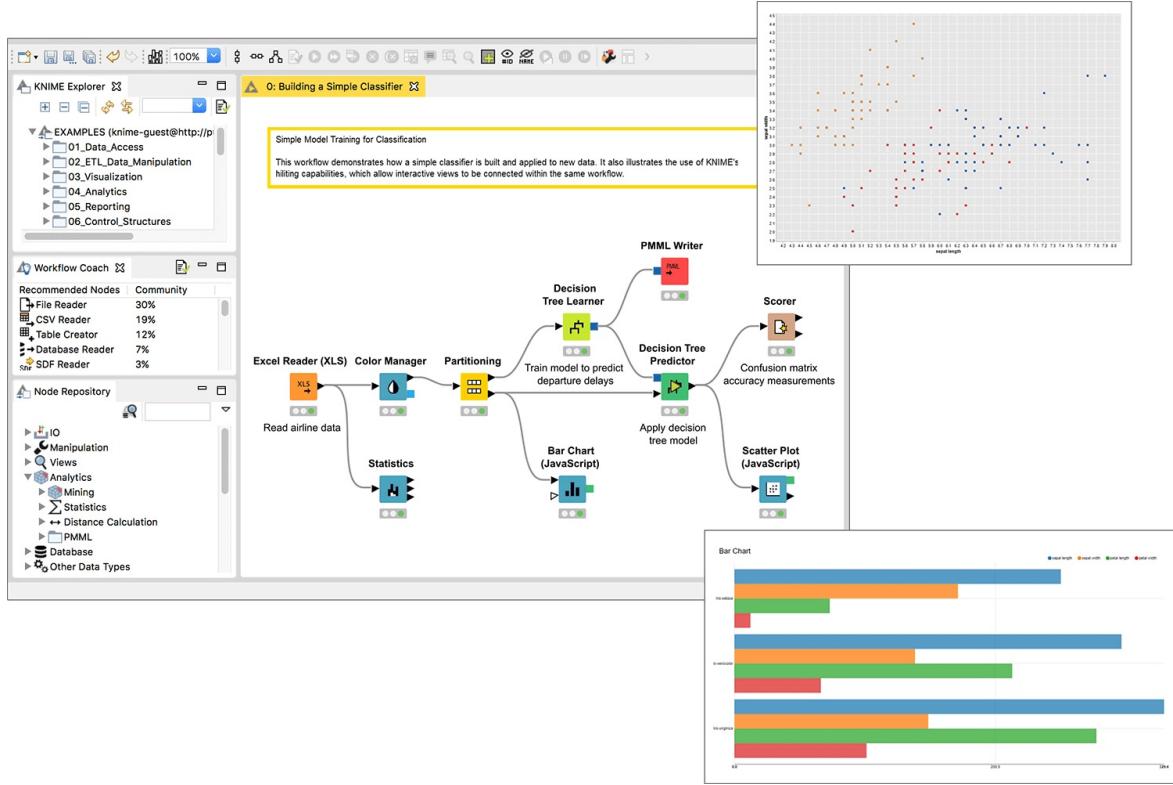


Figure 10: KNIME Architecture [39]

An example of big data analysis hosted by KNIME is a store trying to compare its products sold over multiple store locations. The first step in the visualization process would be to import your data. Within the workflow, users can view differences or find a possible correlation in their dataset by searching for linear correlation in the repository. After choosing this feature, the user must connect the data set to the linear correlation node via a line on the workflow grid. The execute option can then be chosen to view the correlation matrix. The user can hover over a specific cell to select the feature you want to use for further prediction. The next few steps are used in visualization and analysis of user data. Under the Views tab, the user can search for different graphs or plots. A scatter plot, in this case, would be a great way to visualize data from multiple items within a store. You then must drag and drop it and connect it within the workflow like before. You can then configure just how many rows that you would like to look at. Figure 11 shows an example of user analysis by the drag and drop method.

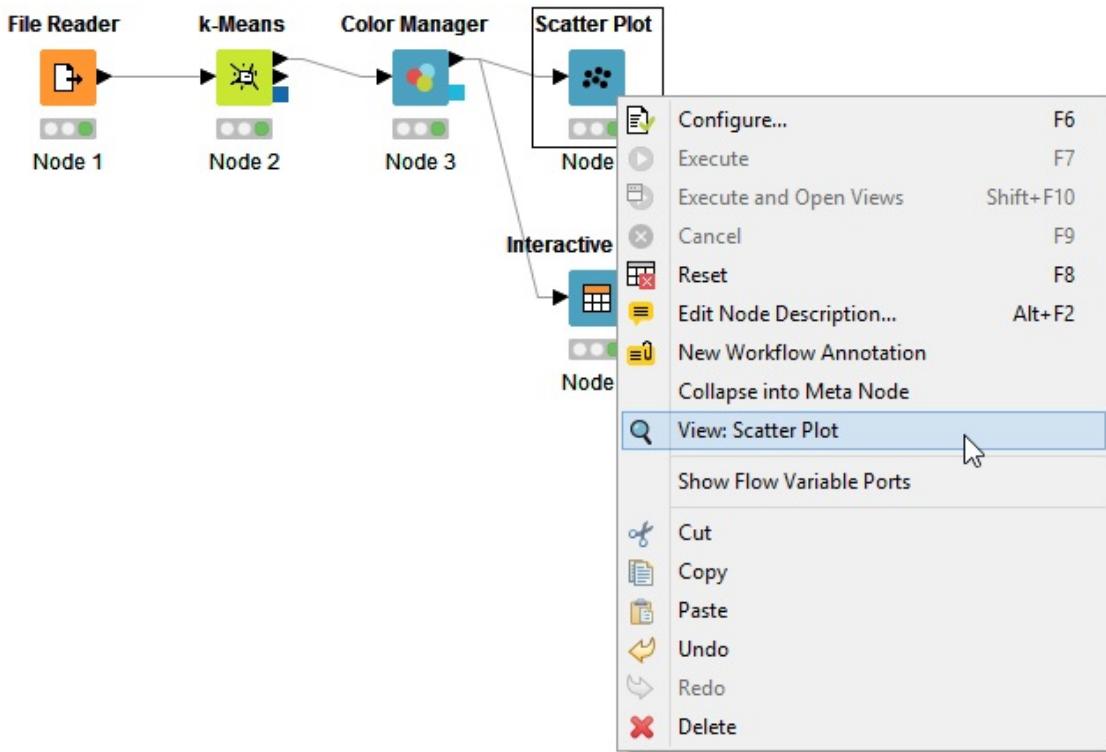


Figure 11: KNIME K-Means [40]

KNIME also includes nodes that can show missing values from certain datasets. KNIME includes a special node in which users may find imputations in their dataset. This is displayed as Missing Values in the output portal. From here, you can choose from a variety of options that allow handling of these imputations. For example, with strings you can move forward and backwards in between rows, create a custom row, or remove a row. You can also manipulate numerical data values by performing several mathematical functions. The basic limitations for KNIME include visualizations that are not extremely neat or detailed as other software. The software's updates sometimes cause user issue and result in necessary re-installation. As this is not as popular of a program as Python or other editing platforms, the community does not have as rich of a support system, therefore, users sometimes struggle with researching issues.

KNIME has the ability to be integrated with other technologies for larger open-source projects. These cloud services allow for user's

projects to be analyzed even further. For example, KNIME can be used with Amazon AWS [41] and Azure [42]. KNIME's platform can be hosted on Microsoft Azure Cloud Services. Azure allows KNIME to perform its analytical, machine learning, and deep learning tasks on its integrated server. This application can be downloaded from Azure's Marketplace. KNIME can also be incorporated with Amazon AWS. When KNIME is connected to AWS resources, users can leverage the memory available while connected to the relational database service to construct SQL queries visually. KNIME's Analytic Platform is a free service for all who use it. However, if you are using KNIME on a cloud service such as Azure or AWS, there are often subscription fees associated. Students and other organizations can receive discounts or allocated amounts for a specified time of use. KNIME is a data analysis software platform that allows for easy read and manipulation of large datasets that can ultimately be used to make inferences and predictions. Its user-friendly interface allows for a broad integration of users and sometimes more efficient workflows. KNIME has several applications for its users such as data modeling, machine learning, predictive analysis, and more. After visualization, users can extract specific features from their data and implement it into a model of their choice, which can then be exported as a CSV file.

Chaitanya Kakarala
ckakara@iu.edu
Indiana University
hid: fa18-523-53
github: [ckakara](#)

Keywords: fa18-523-53, Microservices, Kafka, Python

8.1 INTRODUCTION

Apache Kafka [43] is a distributed streaming platform which works on a subscribe-publish model. Data flows through a streaming channel also known as a Kafka cluster by either subscribing or publishing the topics. The primary objective of Kafka is to persist the message data so multiple consumers can access the same and provide horizontal scaling. With the increase in demand for Agile methodology in software development where the user stories should be completed in a given sprint, there is great need of shrinking down the applications to smaller units. These smaller units are otherwise known as Microservices. These Microservices are loosely coupled and the instructions inside them are light weight. Microservices are autonomous by nature and they can be plugged in any host and bring them up provided the software and hardware requirements are met. As a result, small independent teams can work on these Microservices in parallel and deploy them independently. Depending on the complexity of an application there could be hundreds of Microservices defined and each of them might interact with each other. In other words, an event occurred on one Microservice could start another Microservice. With the interactions between these Microservices increasing, it's hard to trace the connection between them and hence creating a technical debt to mitigate the issue. Apache Kafka is a solution to mitigate this issue.

8.2 ARCHITECTURE

The unit of data within Kafka is message. These messages are nothing but an array of bytes and Kafka is least worried about the content of these messages. Optionally a message can have a key which is again an array of text whose hash value determines the partition the message will be written to. Doing so will guarantee that the messages with same hash value will be stored into the same partition. Messages can also be sent in batches which in other words, a bunch of messages sent all at once. That leaves us with questions like, what are these messages? Where are they stored? who uses these messages? Messages in Kafka are classified into Topics. Topics are nothing but a group of partitions (Can also be described as disk space) where a collection of similar messages are stored. Messages will be appended to these partitions and will be read from beginning to end fashion. The Partitions can be hosted by different servers which makes the topic scale horizontally. All the partitions for a topic is often termed as Stream. Figure [12](#) describes four partitions of a single topic.

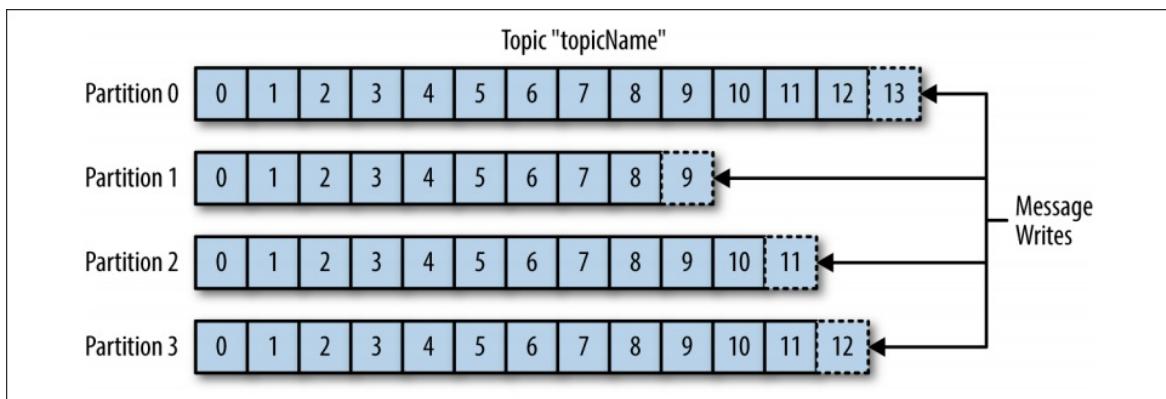


Figure 12: Representation of topic with multiple partitions[44].

There are basically two users of Kafka system. They are Producers and Consumers. Producers create messages to a specific topic. Producers are also termed as publishers. Producers by default does not care which partition they are writing the message to. However, in some cases the hash value of the key decided the partition and

ensures all the messages for the same key reside in the same partition. Consumers read messages from the partitions in the order they were published by the producers. Consumers are also termed as subscribers. While reading the messages from partitions, consumers store the offset to keep track of the read messages. By storing the offset, the system can be re-started from the point of failure without starting all over again. Consumers are bundled together as a consumer group that restricts a given partition to be read by a unique consumer. Consumer groups help in scaling the consumers horizontally. Figure 13 illustrates how consumer group works.

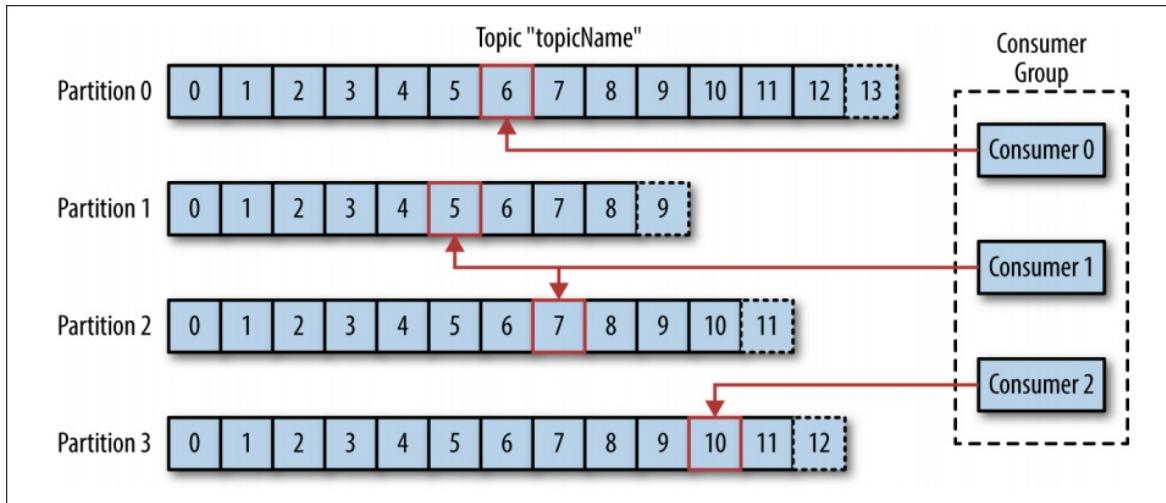


Figure 13: A consumer group reading from a topic[44].

A single kafka server is called as Broker. Each broker receives messages from producers and writes them to the partitions on the disk. They will then save the offset for each message in a partition. They also respond to the consumer programs for data requests from partitions and commit the same. Kafka is designed to have multiple brokers and a collection of all of them is termed as a Kafka cluster. Each cluster can have multiple brokers where the leader broker replicates the data to others. Replication of data helps in durability of data even when one of the brokers fail. Figure 14 explains how multiple brokers are replicated in a kafka cluster.

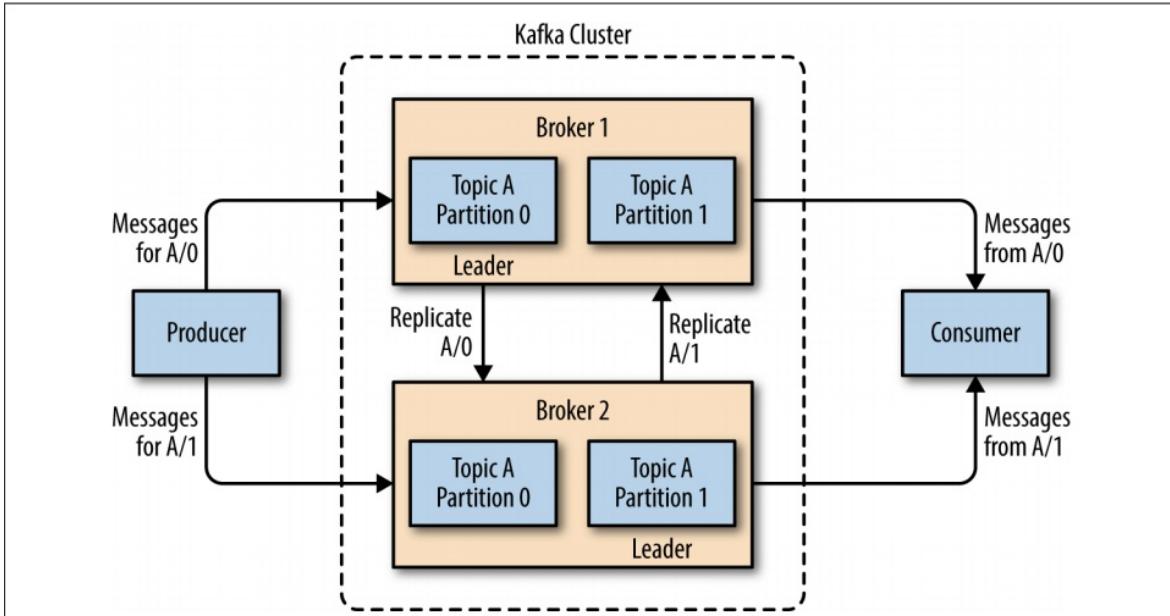


Figure 14: Representation of partitions in a cluster[44].

The major aspect of kafka is the data retention in the partitions. By default the messages in the partitions will be retained for a period of time or Size. For example, the messages in a topic can be retained for one week or until the partition reaches 1 GB. The default behavior can be overridden for topics by changing their settings. Kafka also supports multiple clusters communicating across multiple data centers.

8.3 INSTALLATION AND STARTING KAFKA

- Kafka Installation:
 - Kafka tar file can be obtained from [45]. Please download and save it on the server. Please be aware that kafka requires Java to be installed on the server.
 - Untar the downloaded file using below commands

```
tar -xzf kafka_2.11-1.1.0.tgz
```

- If the java version in your server is having a LTS (Long Time Support) then below fix is needed in kafka-run-class.sh located in bin folder under the kafka home directory. This is a known fix and kafka is working to address this issue for future releases [46].

Change below line

```
JAVA_MAJOR_VERSION=$(($JAVA -version 2>&1 | sed -E -n 's/.*/ version "([^. -]*).*/\1/p')
```

to

```
JAVA_MAJOR_VERSION=$(($JAVA -version 2>&1 | sed -E -n 's/.*/ version "([^. -]*).*/\1/p')
```

- Start the zookeeper server using the below command. You need to be in kafka home directory to be able to successfully execute the below command.

```
bin/zookeeper-server-start.sh config/zookeeper.properties
```

- Start the Kafka server using the below command. You need to be in kafka home directory to be able to successfully execute the below command.

```
bin/kafka-server-start.sh config/server.properties
```

8.4 USE CASES

Kafka was primarily built in LinkedIn to optimize their activity logging system that has multiple Microservices. When the user does an action from the frontend, messages streams through the kafka cluster which are then subscribed by the backend consumer process. Kafka is also used in organizations for collecting the logs and metrics from their microservices. Since Kafka is analogous to the commit log in databases, it can be used to monitor the databases as an event occurs. Kafka is also used in the applications which are developed to process data in streams.

8.5 ACKNOWLEDGEMENT

The author would like to thank Professor Gregor von Laszewski and associate instructors for their help and guidance.

Daniel Hinders , Nhi Tran
dhinders@iu.edu, nytran@iu.edu
Indiana University
hid: fa18-523-56 , fa18-523-83
github: [github](#)

Keywords: NiFi, NSA, Data Stream, ETL

9.1 APACHE NiFi INTRODUCTION

NiFi is a customizable tool for building flexible data flows while preserving data provenance and security [47]. NiFi provides the ability to build or alter an ETL flow with a few clicks. NiFi builds Gets, Converts, and Pulls in a GUI and allows the user to build and customize the flow [48]. This flexibility and usability is key to NiFi's value in a big data world where stovepipes and inflexibility are frequently challenges.

NiFi is a tool for:

- Moving data between systems, including modern systems such as social media sources, AWS cloud server, Hadoop, MongoDB, and so on [49]
- Delivering data to analytics platforms[49]
- Format Conversion, extracting/parsing data[49]
- Data or files routing decisions[49]
- Real-time data streaming[49]

NiFi is not recommended for:

- Distributed Computation [49]
- Complex Event Processing [49]
- Join/ Aggregated Functions [49]

9.2 BIG DATA CHALLENGES AND NiFi

Big data can be a fantastic source of information for decision making and business process definition and actualization. However, the complexity of individual datasets, the variability of dataset structure and composition, and the sheer volume of data are challenges to truly leveraging big data in the real world. This is a multifaceted problem with many inherently overlapping challenges. ETL or Extract, Transform, and Load encompass a number of potential tasks such as harvesting and moving data into a database from some other location and/or and cleaning, normalizing or even structuring data. In a case where a single dataset emerges from an ETL process and the data is somewhat structured and located somewhere predictably accessible, then we can start to leverage analytics or visualization tools to understand the data and use it to make decisions and learn things. Furthermore, productization and dissemination of that data is fairly straightforward.

But this is rarely where the real use case for big data solutions ends. The bigger challenge is dealing with disparate datasets and connecting points of information in a multi-sourced dataset environment. Consolidation of disparate data is therefore extremely important. Furthermore leveraging the correctly sourced data out of consolidated data store environment and then loaded this data into the correct product is challenging.

Apache NiFi is an application that seeks to address this big data problem. NiFi is a tool that has emerged from a unique background as a tool created by the National Security Agency then curated and improved by the open source community.

9.3 NiFi HISTORY

NiFi was first developed at the National Security Agency but was released as an open source project to the public.

“NiFi was submitted to The Apache Software

Foundation (ASF) in November 2014 as part of the NSA Technology Transfer Program" [50].

Since then, Apache Foundation has used its volunteer organization to grow and mature the project [48].

9.4 NiFi FEATURES

NiFi incorporates a straightforward User Interface (UI) to engineer traceable data provenance with configurable components. NiFi offers up the ability to custom build processors and incorporate them into a highly customizable flow. Through

"data routing, transformation, and system mediation logic" [47],

NiFi seeks to automate data flow in a big data environment and gives architects the ability to keep data flowing between evolving systems quickly. Amongst a host of features, NiFi offers, one sticks out as particularly important because of the challenges associated with what the feature addresses: data errors, data inconsistency, and data irregularity handling. NiFi provides users with the ability to incorporate in the flow, processes to catch these non-happy path realities in big data. As new situations are discovered, a user can quickly build if-then forks in the process to catch, store, or resolve the data issues.

NiFi's main features are:

- Guaranteed delivery: use purpose-built persistent write-ahead log and content repository to ensure guaranteed delivery in an effective way [47] [51]
- Web-based user interface: easy to use web-based GUI with drag and drop features that allows users to build, schedule, control, and monitor data flow[47] [51]
- Provenance: provide the ability to track data flows through the systems with audit trail and traceability functionalities [47]

[51]

- Queue Prioritization: provide the ability to configure and prioritize job flow and determine the order of events [47] [51]
- Secure: provide and support multiple security protocols and encryptions, as well as authorization management [47] [51]
- Extensibility: provide flexibility by allowing pre-built and built-your-own extension to be integrated [47] [51]
- Scalability: supports scale-out by clustering architecture as well as scale-up and scale-down [47] [51]

9.5 NiFi ARCHITECTURE

Figure 15 shows the main components in NiFi architecture [51].

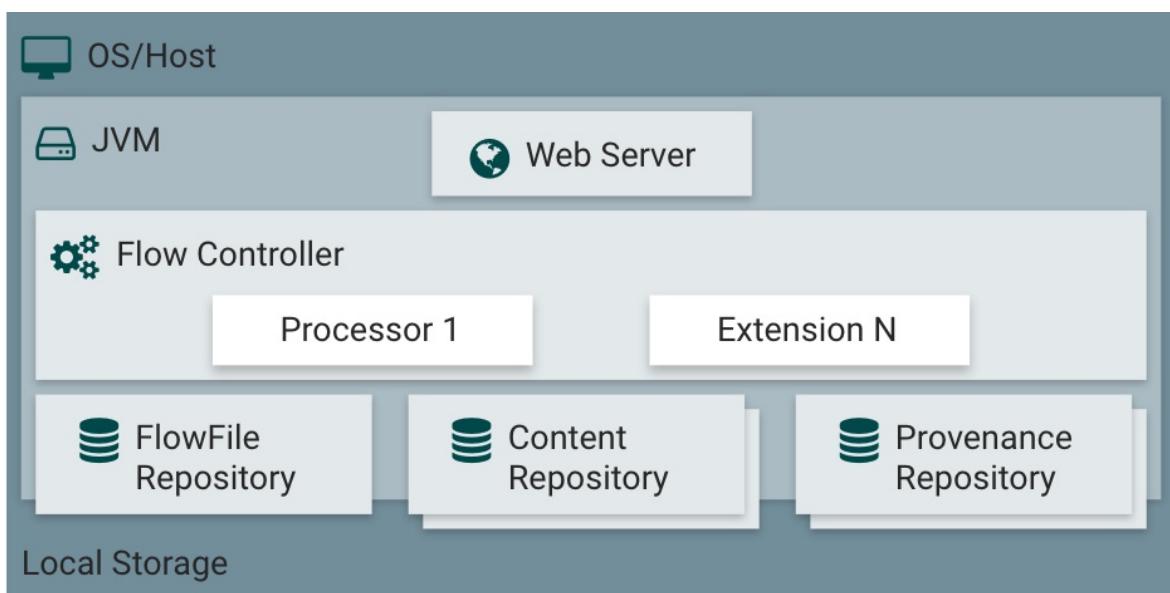


Figure 15: NiFi Architecture [51]

From the top down, NiFi is web browser accessible by a NiFi hosted Web Server. NiFi processor operations are managed through the Flow Controller and the three repositories; FlowFile, Content, and Provenance work to process data on and off disk and in a NiFi flow. NiFi is hosted in the Java Virtual Machine environment or JVM [51].

9.5.1 Web Server

NiFi's easy-to-use graphic user interface(GUI) is hosted on the Web Server within the JVM [51].

9.5.2 Flow Controller

NiFi central operations hub is the Flow Controller. Threads are managed and allocated to the processors and the FlowFiles are passed through and managed through the Flow Controller [52].

9.5.3 FlowFile Repository

Files in an active NiFi flow are tracked in a write-ahead log so that as data moved through the flow NiFi can keep track of what is known about files as they pass through [51].

9.5.4 Content Repository

The real data for a flow file is in the NiFi content repository. NiFi uses simple blocks of data in a file system to store this FlowFile data [51]. Multiple file systems can be used in order to increase speed with multiple volumes being utilized.

9.5.5 Provenance Repository

In NiFi, the provenance repository stores historic event data. The provenance data about flows is indexed to enable search of the records [52].

9.5.6 Processors

NiFi provides more than 260 processors and more than 48 controller services for users to integrate into a flow from the graphic user interface(GUI) of Nifi [53]. Processors are base on underlying controller services in the java virtual machine. Controller services can be centered around a security implementation, database CRUD (create, read, updates, and deletes), and many other foundational areas. Users can create custom processors from existing controller

services or create a customer controller service as well [53].

9.5.6.1 Processor Examples

- **Get**

- Examples: GetFTP, GetMongo, GetTCP, etc. [51]
- Similar input type processors: Consume, Extract, Fetch, Listen, etc.

Nifi provides dozens of Get processor options and many other similar input type processors. A Get processor is commonly used to pick up a file or data and launch a FlowFile. The Get file processor setup typically gives configuration options to point to a host, set timing increments for polling and timeouts, set proxy settings, and more [51].

- **Convert**

- Examples: ConvertJSONToSQL, Convert Record, ConvertExceltoCSVProcessor, etc. [51]
- Similar transformation type processors: Evaluate, Merge, Split, etc.

Once data is in the flow, NiFi provides dozens of processors to manipulate or transform data. The Convert processors can be configured to the expected schema or type from the Get processor and transform, edit, thin, enrich, or many other functions on the data in the flow [51].

- **Put**

- Examples: PutFile, PutFTP, PutSQL, PutElasticSearch, PutAzureBlobStorage, etc. [51]
- Similar output type processors: Publish, etc.

A critical part of a flow in NiFi is pushing the right data out of the flow into the right spot. There are dozens of Put processors that can be

configured to set the directory to write files too. Additional configuration options are specific to the destination type to include SSL configuration, cache options, batching options, and many other configuration options based on the destination type [51].

9.5.7 NiFi Clusters

NiFi can also be integrated with ZooKeeper to operate within a cluster. Figure 16 shows how ZooKeeper manages NiFi's nodes by determining the primary node, Zookeeper Coordinator, and failover node [51]. Each of the nodes performs the same tasks but processes different dataset(s) [51].

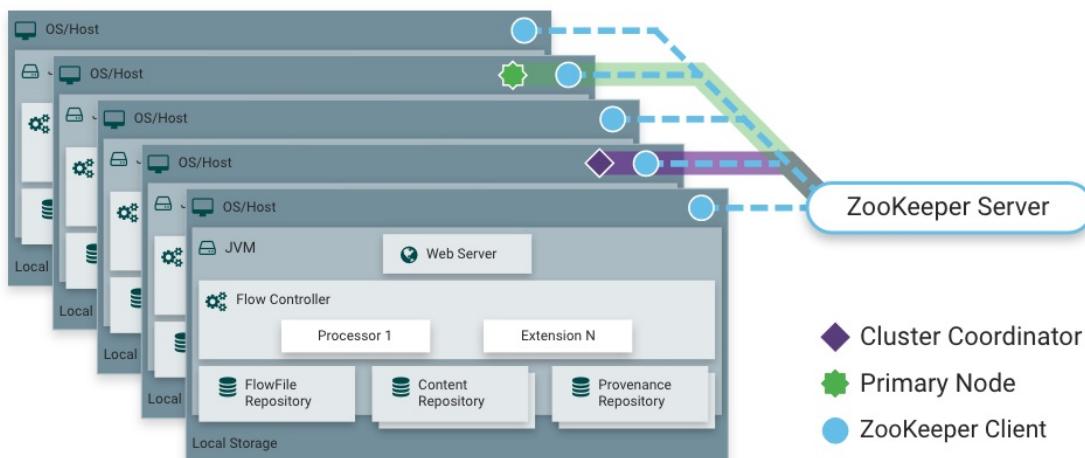


Figure 16: NiFi Cluster Architecture [51]

9.6 NIIFI DOWNLOAD, INSTALLING AND GETTING STARTED

NiFi can be downloaded and installed from its Downloads Page [54] with Linux/Mac tarball option, or zip file option for Windows, or Homebrew option for Mac [55].

For Window

Double-click to run `run-nifi.bat` file from NiFi `bin` subfolder within the installed folder [55].

For Linux/Mac OS X Users

Use Terminal to run `bin/nifi.sh`. An application will run and will be shutdown when the command is terminated [55].

NiFi as a Service in Linux and Max OS X

To install NiFi as a Service, run the command `bin/nifi.sh install <service_name>`. Without specifying specific `<service_name>`, nifi service name will default to nifi [55].

To start NiFi service after installation, run `sudo service <service_name> start`. To stop, run `sudo service <service_name> stop` [55].

Once NiFi has been started, the GUI can be accessed using a web browser via <http://localhost:8080/nifi>. The port and hostname can be configured and changed depending on which server or setting in `conf/nifi.properties` is used [55].

9.7 USE CASE

9.7.1 File Transfer and Routing at MasterCard

MasterCard is a card payment and technology company that connects digital transactions globally. Figure 17 shows one of the use cases for NiFi at MasterCard, which is a file transfer mechanism [56].

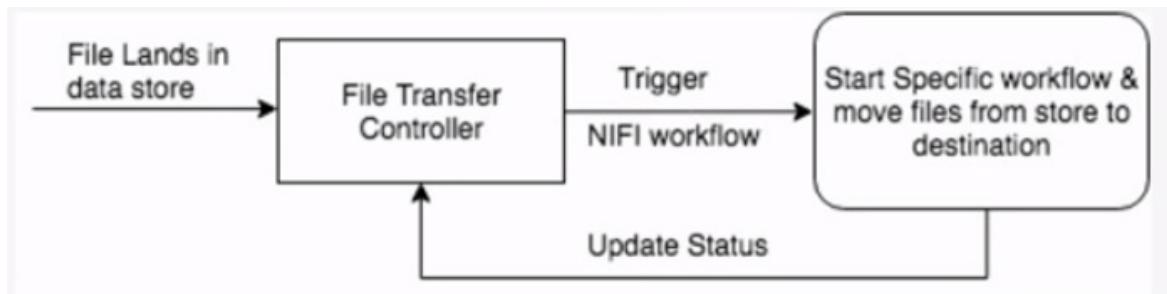


Figure 17: Mastercard NiFi Flow [56]

Batch processing is still a major part of MasterCard's ecosystem which requires multiple formatted flat files being created,

transferred, and picked up by applications [56]. MasterCard uses NiFi's file transfer features to convert files source into data stream(s) and perform specific workflow to direct data into various target systems [56]. Target system could be a messaging systems, Hadoop landing zone, databases. NiFi can also feed data and trigger a map-reduce or spark jobs after transfer [56].

MasterCard provided a demo which demonstrates the use case of them using NiFi to call web services from a file transfer controller, the data flow then has a mechanism to determine which process groups NiFi should distribute data into based on file name/ format logic [56]. The process groups contain workflows that can either feed data to a different system, to Hadoop, or to Postgres database [56]. Once each process flow is completed, the process status will be captured and reported into a Status Handler process [56].

9.7.2 Streaming Analytics Solutions at OpenText Magellan

OpenText Magellan is an artificial intelligence product that supports machine learning and advanced analytics. At OpenText Magellan organization, NiFi was utilized as part of their streaming analytics infrastructure to allow continuous process and real-time analysis [57]. OpenText Magellan's infrastructure involves source applications, NiFi, Apache Spark, Python, R, Scala, and other Magellan BI and Reporting tools [57]. Figure [18](#) shows a typical process of streaming analytics process at OpenText Magellan, which involves six steps: (1) Data Acquisition, (2) Data Routing, (3) Streaming Processing, (4) Machine Learning, (5) Prediction Results, and (6) Actionable Insights [57].

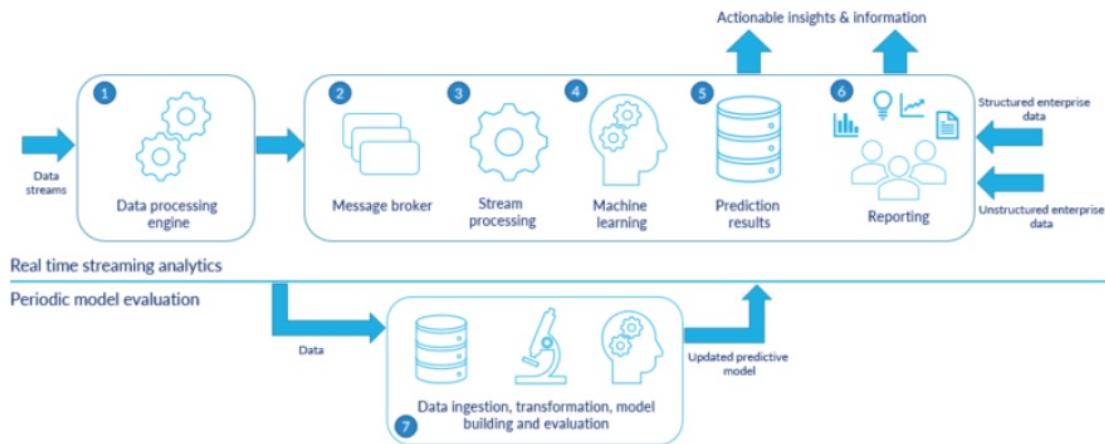


Figure 18: Opentext Magellan NiFi Flow [57]

NiFi is used during the first Data Acquisition steps to collect data from multiple sources such as smart devices, social media, online transactions, and log monitoring [57]. The real-time data can then be combined with other historical data or other data sources before being feed into a downstream system [57]. Data is then being streamed by Kafka in Data Routing step and then being read and applied business rules by Spark Streaming API before it is being stored in a data lake [57]. Spark Streaming API will apply machine learning prediction model in Machine Learning step and then being saved in Prediction Results [57]. One the result is created, organizations can take quick decisions to provide business benefits and insights [57].

As a result, the organization was able to create low-cost solutions that has the flexibility and extensibility of open source software.

9.7.3 Social Competitive Intelligence Application at Compose

Compose is an IBM company launched in 2010 that offer databases as a service on the cloud that is production ready and is easy to manage. NiFi in being used in Compose as part of their Competitive Intelligence infrastructure that involves other software such as

Twitter, IBM Watson, Redis, and MongoDB [58]. NiFi was used to extract filtered Twitter Stream data and attributes and send tweet data to IBM Watson for Sentiment analysis, as well as updating Redis for dashboards and reporting purpose and at the same time store all data in MongoDB [58] .

9.7.4 Real Time Streaming Architecture at Ford

Ford is an automobiles manufacturing company in the United States. Being a large company, data are stored and generated constantly in many applications within the enterprise such as assembly plants data, vehicle sensor data, dealership data, vehicle diagnostic data, and so on [59]. Ford came up with a solution called Real Time Streaming Architecture (RTSA) to allow data being flow between systems in real-time with proper data governance [59].

Ford's data are being sourced from Open XC which contains vehicle and phone application data into a private cloud via Cloud Foundry WebSocket or Event Hub [59]. Data from Websocket are streamed via Kafka into a cloud-based NiFi cluster together with the Event Hub data [59]. From the cloud-based NiFi Cluster, the combined data then flows to a private in-house NiFi cluster in Ford's data center and then publish to Kafka for downstream system distributions or being stored in Hadoop [59].

9.8 WORK BREAKDOWN

- Nhi Tran fa18-523-83

Use Case, NiFi Architecture image, NiFi Cluster Architecture's image, NiFi Download Installing and Getting Started

- Daniel Hinders fa18-523-56

NiFi Introduction, Big Data Challenges and NiFi, NiFi History, NiFi Architecture

- Both

NiFi Features

Divya Rajendran
divrajen@iu.edu
Indiana University
hid: fa18-523-57
github: [blue user icon](#)

- do not remove the this is for the instructors a sign that we reviewed
-

Keywords: Hid fa18-523-57, [Deep Learning](#), gradient descents, Python, [Neural Networks](#), [Tensors](#), [Computational Graph](#), [Autograd](#), [Auto Differentiation](#), [Backpropagation](#)

PyTorch [60] is a python based deep learning framework used for scientific calculations. It is popular among Neural Nets and Deep Learning developers due to its faster implementation of the algorithms and the maximum flexibility of its use. It is also used as a replacement for NumPy [61], an existing package, available in Python on scientific computing. The reason being, PyTorch mimicked most of NumPy's functionality with an addition of increased speed by making use of the Graphical Processing Unit (GPU) [62].

Being written in a commonly used language by Machine Learning and Artificial Intelligence developers, Python, PyTorch has been gaining popularity since its inception in 2016 [63]. It is also less complex and easy to use when compared to existing Deep Learning frameworks like TensorFlow [64], Keras [65], Caffe [66], Chainer [67], MXNet [68], CNTK [69], Deeplearning4j [70].

PyTorch has been developed by the Artificial Intelligence group at Facebook [63] and is a successor framework of Torch [71] and has been built on it. Torch is a computing framework for scientific calculations wrapped in Lua, a programming language written in a

general-purpose programming language C [72]. This framework, Torch, runs even in constrained platforms through LuaJIT [73] a platform specific compiler. It is used extensively to implement machine learning and deep learning algorithms [63]. It has a plethora of packages commonly used for Machine Learning, Signal Processing, Computer Vision among others, these have been inherited into PyTorch as well [71].

PyTorch seamlessly integrates all the packages which Torch offers and builds all of its functionality using Python, making Python its integral part. This makes the implementation of algorithms even faster than Torch. The main package of Torch and PyTorch is torch using which we can train neural networks, define the loss function and calculate the gradients for loss function [63].

10.1 BACKGROUND

Before we start using PyTorch, we need to have a background or working knowledge on the below concepts.

10.1.1 Deep Learning

Deep Learning [74] is a branch of Machine Learning which takes its inspiration from the function and structure of a human brain [75]. It uses a subset of machine learning algorithms which processes input data in multiple layers through feature extraction and transformation and predicts the output labels [74]. It is being vastly used in different fields of computer vision, audio, and video signal processing, natural language and speech recognition and such [74].

10.1.2 Neural Networks

Neural Nets [76] is a collection of various connected nodes called neurons mimicking the neuron structure in the human brain. Each neuron receives an input, processes this input by performing a set of operations and then sends it to a next layer in the neural nets. Each layer has a weight and bias associated with it, on which a

computation is done. This processing is based on some pre-defined function, a gradient descent algorithm, which transforms the input in each layer and this entire process is repeated a huge number of times until the error calculated is diminished [76].

10.1.3 Tensors

Tensor [77] is an inbuilt data structure in PyTorch and can be defined as a matrix of matrices or can be defined as a multi-dimensional array with dimensions greater than 3. So a tensor with 3 dimensions is called a 3-D tensor, a tensor with 4 dimensions is called a 4-D tensor and so on [77]. This tensor is used on a GPU which accelerates the computing process and calculation time on matrix operations when compared to existing NumPy's ndarrays [62].

10.1.4 Computational Graph

A computational graph [78] is an internal representation of the operations performed during the neural nets training. It is also called a data graph and is also an inbuilt data structure in PyTorch. It consists of a set of nodes and edges, with nodes representing each operation and edges representing the values being sent from each operation from one layer to another layer in neural networks [78].

10.1.5 Auto Differentiation

Auto Differentiation [79] is a series of techniques used to numerically calculate the derivative of the transformation and loss functions defined in our neural networks [79].

10.1.6 Backpropagation

Backpropagation [80] is a technique in neural networks which calculates the gradient values for the peaks and troughs of the loss function and send the error values obtained to the previous layer going in the reverse or backward direction [80].

10.1.7 Autograd

Autograd [81] is a function in the torch library of PyTorch which calculates the gradients of the transformation and loss functions used in neural networks. The function Autograd uses a technique called tape-based Auto-Differentiation [82], a kind of the Auto-Differentiation technique. This technique saves the operations performed in each layer of the neural network in a reverse order, mimicking how a tape recorder works. This function also saves the gradients calculated in each layer of neural nets and replays them in a reverse order. Autograd's implementation in PyTorch is faster than the same implementation in existing frameworks like TensorFlow [82]. We also use this function to train weights for neural networks through backpropagation [81].

10.2 GETTING STARTED

If you have never used PyTorch before you need to install PyTorch. Check [83] for more instructions on the system requirements and different ways to install PyTorch [84].

In your system terminal or command prompt, enter the below line to install the PyTorch library.

```
$ pip install torch torchvision
```

You would see a message in your terminal or command prompt that the package has been installed.

To use PyTorch we need to first import a library called torch. A sample initialization of tensors using PyTorch is done as below.

```
from __future__ import print_function
import torch

#### Creating an empty matrix
torch.empty(4, 3)

#### Creating a randomly initialized matrix
torch.rand(4, 3)
```

```
#### Constructing a tensor  
torch.tensor(torch.rand(4, 3))
```

Step by step instructions for using PyTorch can be found at [85].

10.3 IMPLEMENTATION

PyTorch can be initialized using the package torch. It contains the below functions.

1. torch.nn is a library of functions used to train or build the neural networks [86].
2. torch.nn.Linear is a function which applies a transformation, linear in nature, on the incoming values [86].
3. torch.nn.Sequential is a function used to initialize a model with a linear-stack of layers [86].
4. torch.nn.MSELoss() is used to initialize the loss function and also calculates the error between the input and the output layer in the neural nets [86].
5. torch.optim is a function which defines an optimizer algorithm which updates the weights at each layer [86].

Let us see an example of how to use these above functions to train a neural network as below.

10.3.1 Define a Neural Network

In this example, we would learn how to create and initialize a neural net model with two layers and how to apply an optimizer function on this neural network. This example is copied from [87].

```
#####
#   Title: Classifying Text with Neural Networks and Pytorch
#   Author: Mesquita, Déborah
#   Date: Oct 25, 2017
#   Code version: 1
#   Availability: https://github.com/dmesquita/understanding\_pytorch\_nn
#
#####
import torch
import torch.nn as nn
```

```

class OurNet(nn.Module):
    def __init__(self, input_size, hidden_size, num_classes):
        super(Net, self).__init__()
        self.layer_1 = nn.Linear(n_inputs,hidden_size, bias=True)
        self.relu = nn.ReLU()
        self.layer_2 = nn.Linear(hidden_size, hidden_size, bias=True)
        self.output_layer = nn.Linear(hidden_size, num_classes, bias=True)

    def forward(self, x):
        out = self.layer_1(x)
        out = self.relu(out)
        out = self.layer_2(out)
        out = self.relu(out)
        out = self.output_layer(out)
        return out

```

The previous code initiates the neural network with two hidden layers and one output layer. The function `nn.Linear` transforms the input layer data linearly, by multiplying the data with weights and adding a bias value. The transformation of our neural network is taken through the function `forward` which we defined above. In this function, we call the initialized values and functions and transform the output and return this value. [87]

To update the weights for our neural network, we use the optimizer algorithm Adaptive Moment Estimation (Adam) through the package `torch.optim`. This optimizer holds the state of the object and the components based on the gradient computation. To calculate the loss we use a function called `torch.nn.CrossEntropyLoss`. Below is a sample code on constructing the optimizer [87].

10.3.2 Constructing an optimizer

The step `OurNet()` is our initialization step for the neural nets we created. We get the initialized neural nets parameters and use it to initialize our optimizer function.

```

net = OurNet(input_size, hidden_size, num_classes)
optimizer = torch.optim.Adam(net.parameters(), lr=learning_rate)
criterion = nn.CrossEntropyLoss()

```

Our next step would be to load a dataset from `sklearn datasets`, or any other set of datasets and use it to test our functions written. The entire code for initializing neural networks, loss, and optimizer

functions, including training and testing our models while applying them to a dataset of our choice is available can be found at [88].

10.4 ADVANTAGES OF PyTorch

When we compare PyTorch over the existing frameworks like TensorFlow, Caffe, Keras, Chainer and such, the below are the most promising advantages.

1. Ramp Up Time is the time taken to execute all the threads or layers and their iterations. This time for code execution using PyTorch is much faster than its competitor TensorFlow [89], in that it uses dynamic creation of graphs rather than the static ones in TensorFlow [89]. Here, the compilation time for the code is much smaller for PyTorch, it uses GPU to increase the speed of execution and the graph is built during run-time, making it significantly faster than TensorFlow [89].
2. Debugging in PyTorch is easy as the underlying language is Python, which is a common language used by developers and it is quite easier when compared to TensorFlow. We can use print statements to keep track of what values our variables take and to identify where our code fails [89].
3. Data Loading is much faster in PyTorch as the APIs for loading the data are designed in a manner to best utilize its parallelizing data loading capability. One can use either NumPy, Pandas or any other library of choice and can load data quite faster as PyTorch utilizes GPU [89].
4. PyTorch is highly extensible in that it has many custom extensions and it is easier to implement them for both GPU and CPU versions of its code [89]. It can be extended using NumPy, Scipy [90] and many other libraries [91]. Examples of extending PyTorch through Scipy and NumPy can be found at [91].

10.5 DRAWBACKS OF PyTorch

When we compare PyTorch over its competitor like TensorFlow [92] we identify the below areas where TensorFlow outperforms PyTorch [89].

1. Coverage of functionality is less in PyTorch when compared to TensorFlow. The functions like NumPy's flip along a dimension, checking NaN values, fast fourier transformation are not readily available in PyTorch whereas these functions and many higher functions are available in TensorFlow [89].
2. Serialization [93] can be defined as the process of translating the input data into a format which can be easily transferred across different platforms [93]. This capability in TensorFlow is better than PyTorch that it even is capable of saving the graphs can be saved as well. These graphs can easily be loaded into different languages like C++ and Java. This enables the deployments to not depend on Python alone [89].
3. Deployment is an activity which makes a code available to be used on a system where the code is placed. Since the code developed in TensorFlow can be easily saved in a format which can be used in different languages, its code can be easily deployed even in mobile applications [89].

Pramod Duvvuri
vduvvuri@iu.edu
Indiana University
hid: fa18-523-58
github: [blob](#)

does not provide a full example

11.1 INTRODUCTION

The amount of data generated has increased exponentially and so did the advancement of computing power, both these have led us to the era of deep learning. This paper aims to summarize a deep learning framework known as Caffe [94] which was developed by a post-doctorate student Yangqing Jia at the University of California, Berkeley in 2013. It is written in C++ and is known for its fast execution and its Python interface allows it to be used by the vast majority of Python users. The framework has then been open-sourced, allowing many users to use, develop and contribute to improve the framework. The deep learning [95] revolution has led to the need for state-of-the-art implementations of Artificial Neural Network (ANN) architectures. These architectures are too hard to code from scratch for most people even with a conceptual understanding. The first deep learning framework to gain popularity was Theano [96]. Theano was developed at the University of Montreal in 2007. It was primarily used by academic researchers at the university. Theano was built using Python which essentially made it slower for larger models, for production-grade models speed is imperative. This meant there was a need for a new popular and fast deep learning framework, especially in computer vision. Caffe was built using C++ and this made it very fast and ideally suitable for deployment in production. Caffe [97] at the time of public release or

open sourcing had the best implementation of a Convolutional Neural Network [98], which is primarily used in solving computer vision problems. This public release made all the computer vision researchers and other people in the computer vision community adopt Caffe.

Keywords: hid fa18-523-58, [Artifical Neural Network](#), C++, [Computer Vision](#), [Deep Learning](#), Machine Learning, Python

11.1.1 Artifical Neural Network

Neural Network is a machine learning algorithm which mimics the human nervous system. It consists of various nodes or artificial neurons that are interconnected and perform machine learning tasks. The advancements in computation and the introduction of Graphical Processing Units [99] (GPUs) have made it feasible for us to run such sophisticated algorithms. Neural networks have performed exceptionally well on data in comparison to other industry standard machine learning algorithms, which is why they have been adopted by both the academia and the industry. They require far more training data or examples than other algorithms and also require a considerable amount of computational resources to run.

11.1.2 Computer Vision

Computer vision [100] primarily consists of computers trying to extract or understand meaningful information from images or videos from the real world. It is a vast field that consists of many domains under it. The main goal of computer vision is to build a system that can mimic the human vision or visualize an image and understand the context and semantics. The input for such a system can take multiple forms such as a single image, sequence of images or a video or multi-dimensional data. Some of the most common are object recognition, object tracking, image segmentation, image processing. The deep learning revolution has essentially revitalized the field of

computer vision. Many problems which were considered impractical have been solved using deep learning. Artificial Intelligence [101] and Computer Vision have a lot of common topics. Quite a few of these problems such as pattern recognition in vision were solved with the help of artificial intelligence and this made computer vision an integral part of artificial intelligence.

11.1.3 Deep Learning

Deep learning is a set of techniques or architectures that use Deep Neural Networks (DNNs) to solve problems in various fields such as computer vision, signal processing [102], natural language processing [103]. These DNNs can be used to solve any type of machine learning problem. Deep Neural Networks are Neural networks with more than two layers. There are usually two main types of in machine learning:

1. Supervised Learning: In supervised learning, the data used to train our machine learning model is categorized into various categories also known as labels. The model is trying to learn how to categorize our data into these categories.
2. Unsupervised Learning: In unsupervised learning, there are no categories. The algorithms are trying to find patterns or similarities in the data we give as input.

Any deep learning architecture at its core consists of a perceptron. A perceptron [104] is a machine learning algorithm which was made to mimic the function of a human neuron. Deep learning architectures use multiple such perceptrons or nodes as layers which is the reason these are referred to as deep learning architectures. In computer vision problems each layer serves a specific purpose and the combination of all these layers aids in solving specific problems.

11.2 INSTALLATION

To use Caffe it is recommended to install a containerized image of it. This can be done using the help of Docker [105]. The official Caffe

image can be found on DockerHub and can be installed using the GUI. It can also be installed using the following command with docker already running on your local machine. In your docker terminal please paste the following command:

```
$ docker run -ti bvlc/caffe:cpu caffe --version
```

```
caffe version 1.0.0
```

As indicated the latest version is 1.0.0, the above command is mostly used if the machine does not contain a GPU [99]. If your machine contains a dedicated GPU then another command can be used to install Caffe using Docker. In your docker terminal please paste the following command:

```
$ nvidia-docker run -ti bvlc/caffe:gpu caffe --version
```

```
caffe version 1.0.0
```

With Caffe now installed it can be used with an Interactive Python notebook. The below command must be used to launch an IPython notebook in the docker terminal and then import caffe in the interactive Python (IPython) notebook [106] before we can write any code in Caffe.

```
$ docker run -ti bvlc/caffe:cpu ipython  
[1] import caffe
```

11.3 ARCHITECTURE

The Caffe architecture mainly consists of layers or it had a layer-wise design all designed and built from scratch using C++ and the CUDA [107] architecture with various interfaces to write code in MATLAB [108] and Python. This architecture at the time of its creation was considered really good but since then newer deep learning framework's such as Tensorflow which was created by Google has a much flexible design. This flexible design is with respect to the various nodes in the Artificial Neural Network [109] (ANN) since ANNs primarily consist of layers and each layer has multiple nodes. The ability to flexibly design these nodes was very important to the

researchers since this helped them achieve higher accuracy rates for their model and this helped with benchmarking and comparison with other similar models aimed to solve similar tasks.

11.4 APPLICATIONS

Some of industry grade production levels applications [110] of Caffe are:

- Facebook used Caffe to generate alternate texts who people who are visually challenged. All the photos uploaded to Facebook were run through a caffe model to generate such text. Facebook also used caffe to detect objectionable content. As the amount of data on social media increases so has the need for a protocol to regulate and report objectionable content risen.
- Pinterest used Caffe for object detection in the images. All the images that were uploaded were run through a model for object detection. The Caffe deep learning model for visual search could search over billions of images in just under 250 milliseconds. As many as 4 million images are uploaded onto Pinterest on a daily basis.
- Yahoo used Caffe for user recommendations in Japan. The news feed on Yahoo had stories curated using a caffe model and also made restaurant suggestion using photos. Yahoo also had models to automatically arrange photos of users into albums.

11.5 LIMITATIONS AND COMPARISONS

Caffe was developed by a post-doctorate student and then open sourced in 2013, the deep learning revolution had just begun when Caffe was launched. After its initial launch, there were more than 150 developers who were actively contributing to the framework. At the same time, large companies such as Amazon, Facebook, Google,

Microsoft had all begun working on deep learning frameworks which suited their needs and fit perfectly in their respective technology stacks. The public launch of Tensorflow [64] by Google made a lot of people adopt it quickly since Google had consistently invested more time and money in the development and maintenance of this framework. Currently, there are more 1500 people who actively contribute to the Tensorflow framework. PyTorch [63] is another popular deep learning framework which was developed in 2017 at Facebook and had dynamic graph computational ability which was lacking in Tensorflow. PyTorch was received very well by the research community since it combines two of the most popular languages used by the artificial intelligence community Torch and Python. Caffe main strength was its implementation of a fast CNN [98] and ready to use GPU [99] support. Although CNNs could be used for Natural language processing [103] (NLP) tasks there were other deep learning architectures which were more suitable for NLP related tasks. Other deep learning frameworks such as Theano and Torch had better implementations of these architectures hence they were preferred to Caffe. This meant that caffe's usability was very limited outside computer vision tasks. Caffe does not offer multi-GPU support. The exponential rise in our data meant we needed to use multiple powerful GPUs to train our models. Hence modern frameworks such as Tensorflow and PyTorch were built to serve as an all-purpose deep learning framework that had the state-of-the-art implementations of the latest deep learning architectures and also had out of the box multi-GPU support. Gradually people adopted these frameworks over Caffe. These frameworks had a much more robust architecture. Caffe was restricted by the format for input and output. It only supports one output format called HDF5 [111]. Caffe had less documentation and fewer hands-on tutorials which made it less developer friendly [112].

11.6 CONCLUSION

Caffe was mainly intended to support vision tasks and was not suitable for other tasks such as speech recognition, language

modeling and time series data. This made its applications and also usability limited. But the lack of proper documentation and examples made it harder to adopt for the community. All modern deep learning frameworks were built to overcome the limitations of Caffe. They also borrowed its excellent CNN implementation. More people have since then moved to Tensorflow for academic research. Caffe has contributed a lot to the computer vision and deep learning communities. Caffe helped these communities make valuable contributions to research with its fast execution times. Caffe2 [113] was a project that was started at Facebook after the success of Caffe. Caffe2 was open sourced by Facebook in April 2017. By the end of March 2018, Caffe2 was merged with PyTorch by Facebook. These days the choice of a deep learning framework, when you have huge amounts of data, is either PyTorch or Tensorflow. Both Google and Facebook constantly keep updating these frameworks from the feedback they receive from the developer community to make it more developer friendly with the ability to visualize the computational graphs [78]. The goal now is to make deep learning more accessible to everybody and reduce the steep learning curves when it comes to these deep learning frameworks and caffe has contributed invaluable to achieve this goal.

Izolda Fetko, Nishad Tupe, Vishal Bhoyar
ifetko@iu.edu, ntupe@iu.edu, vbhoyar@iu.edu
Indiana University
hid: fa18-523-60, fa18-523-64, fa18-523-72
github: [blob](#)

flask pymongo section seemnot ready there is lots if programming detail in it that seems great, but the presentation of that section seems abit dense

lost of ??? erros bibtex errors? you must look at epub

- we already pointed out that first section is too long - broke it down to multiple segments
- proposed title change: MongoDB in Python - changed
- introduction and learning outcome missing - added introduction and learning outcomes
- wrong quotes - fixed quotes
- do not use quotes for non cited text such as in "_id" that is `_id` - fixed
- use bash and python after the 3 quotes, bash has a \$ at the beginning - fixed

12.1 INTRODUCTION

In today's era, NoSQL databases have developed an enormous potential to process the unstructured data efficiently. Modern information is complex, extensive, and may not have pre-existing relationships. With the advent of the advanced search engines, machine learning, and Artificial Intelligence, technology expectations to process, store, and analyze such data have grown tremendously [114]. The NoSQL database engines such as MongoDB, Redis, and

Cassandra have successfully overcome the traditional relational database challenges such as scalability, performance, unstructured data growth, agile sprint cycles, and growing needs of processing data in real-time with minimal hardware processing power [115]. The NoSQL databases are a new generation of engines that do not necessarily require SQL language and are sometimes also called Not Only SQL databases. However, most of them support various third-party open connectivity drivers that can map NoSQL queries to SQL's. It would be safe to say that although NoSQL databases are still far from replacing the relational databases, they are adding an immense value when used in hybrid IT environments in conjunction with relational databases, based on the application specific needs [115]. In this paper, we will be covering the MongoDB technology, its driver PyMongo, its object-document mapper MongoEngine, and the Flask-PyMongo micro-web framework that make MongoDB more attractive and user-friendly.

12.2 LEARNING OUTCOME

The learning outcome of this paper is to equip the readers with a basic MongoDB knowledge, as well as on how to use the PyMongo driver in conjunction with this NoSQL database. Other than the aforementioned, the reader will be introduced to some basic functionalities of the MongoEngine, an Object-Document mapper, and Flask-Mongo, a micro-web framework.

12.3 MONGODB

Today MongoDB is one of leading NoSQL database which is fully capable of handling dynamic changes, processing large volumes of complex and unstructured data, easily using object-oriented programming features; as well as distributed system challenges [116]. At its core, MongoDB is an open source, cross-platform, document database mainly written in C++ language.

12.3.1 Installation

MongoDB can be installed on various Unix Platforms, including Linux, Ubuntu, Amazon Linux, etc [117]. This section focuses on installing MongoDB on Ubuntu 18.04 Bionic Beaver used as a standard OS for a virtual machine used as a part of Big Data Application Class during the 2018 Fall semester.

12.3.1.1 Installation procedure

Before installing, it is recommended to configure the non-root user and provide the administrative privileges to it, in order to be able to perform general MongoDB admin tasks. This can be accomplished by login as the root user in the following manner [118].

```
$ adduser mongoadmin  
$ usermod -aG sudo sammy
```

When logged in as a regular user, one can perform actions with superuser privileges by typing sudo before each command [118].

Once the user set up is completed, one can login as a regular user (mongoadmin) and use the following instructions to install MongoDB.

To update the Ubuntu packages to the most recent versions, use below command:

```
$ sudo apt update
```

To install the MongoDB package:

```
$ sudo apt install -y mongodb
```

To check the service and database status:

```
$ sudo systemctl status mongodb
```

Verifying the status of a successful MongoDB installation can be confirmed with an output similar to this:

```
$ mongodb.service - An object/document-oriented database  
Loaded: loaded (/lib/systemd/system/mongodb.service; enabled; vendor preset: enabled)  
Active: **active** (running) since Sat 2018-11-15 07:48:04 UTC; 2min 17s ago
```

```
Docs: man:mongod(1)
Main PID: 2312 (mongod)
    Tasks: 23 (limit: 1153)
   CGroup: /system.slice/mongodb.service
           └─2312 /usr/bin/mongod --unixSocketPrefix=/run/mongodb --config
/etc/mongodb.conf
```

To verify the configuration, more specifically the installed version, server, and port, use the following command:

```
$ mongo --eval 'db.runCommand({ connectionStatus: 1 })'
```

Similarly, to restart MongoDB, use the following:

```
$ sudo systemctl restart mongodb
```

To allow access to MongoDB from an outside hosted server one can use the following command which opens the fire-wall connections [117].

```
$ sudo ufw allow from your_other_server_ip/32 to any port 27017
```

Status can be verified by using:

```
$ sudo ufw status
```

Other MongoDB configurations can be edited through the /etc/mongodb.conf files such as port and hostnames, file paths.

```
$ sudo nano /etc/mongodb.conf
```

Also, to complete this step, a server's IP address must be added to the bindIP value [117].

```
$ logappend=true
bind_ip = 127.0.0.1,your_server_ip
*port = 27017*
```

MongoDB is now listening for a remote connection that can be accessed by anyone with appropriate credentials [117].

12.3.2 Collections and Documents

Each database within Mongo environment contains collections which

in turn contain documents. Collections and documents are analogous to tables and rows respectively to the relational databases. The document structure is in a key-value form which allows storing of complex data types composed out of field and value pairs. Documents are objects which correspond to native data types in many programming languages, hence a well defined, embedded document can help reduce expensive joins and improve query performance. The *_id* field helps to identify each document uniquely [115].

MongoDB offers flexibility to write records that are not restricted by column types. The data storage approach is flexible as it allows one to store data as it grows and to fulfill varying needs of applications and/or users. It supports JSON like binary points known as BSON where data can be stored without specifying the type of data. Moreover, it can be distributed to multiple machines at high speed. It includes a sharding feature that partitions and spreads the data out across various servers. This makes MongoDB an excellent choice for cloud data processing. Its utilities can load high volumes of data at high speed which ultimately provides greater flexibility and availability in a cloud-based environment [114].

The dynamic schema structure within MongoDB allows easy testing of the small sprints in the Agile project management life cycles and research projects that require frequent changes to the data structure with minimal downtime. Contrary to this flexible process, modifying the data structure of relational databases can be a very tedious process [114].

12.3.2.1 Collection example:

The following collection example for a person named Corey includes additional information such as age, status, and group [119].

```
{  
  name: "Corey"  
  age: "21"  
  status: "Open"  
  group: ["AI" , "Machine Learning"]  
}
```

12.3.2.2 Document structure:

```
{  
    field1: value1,  
    field2: value2,  
    field3: value3,  
    ...  
    fieldN: valueN  
}
```

12.3.2.3 Collection Operations

If collection does not exists, MongoDB database will create a collection by default.

```
> db.myNewCollection1.insertOne( { x: 1 } )  
> db.myNewCollection2.createIndex( { y: 1 } )
```

12.3.3 MongoDB Querying

The data retrieval patterns, the frequency of data manipulation statements such as insert, updates, and deletes may demand for the use of indexes or incorporating the sharding feature to improve query performance and efficiency of MongoDB environment [115]. One of the significant difference between relational databases and NoSQL databases are joins. In the relational database, one can combine results from two or more tables using a common column, often called as key. The native table contains the primary key column while the referenced table contains a foreign key. This mechanism allows one to make changes in a single row instead of changing all rows in the referenced table. This action is referred to as normalization. MongoDB is a document database and mainly contains denormalized data which means the data is repeated instead of indexed over a specific key. If the same data is required in more than one table, it needs to be repeated. This constraint has been eliminated in MongoDB's new version 3.2. The new release introduced a \$lookup feature which more likely works as a left-outer-join. Lookups are restricted to aggregated functions which means that data usually need some type of filtering and grouping operations to be conducted beforehand. For this reason, joins in MongoDB

require more complicated querying compared to the traditional relational database joins. Although at this time, lookups are still very far from replacing joins, this is a prominent feature that can resolve some of the relational data challenges for MongoDB [120]. MongoDB queries support regular expressions as well as range asks for specific fields that eliminate the need of returning entire documents [115]. MongoDB collections do not enforce document structure like SQL databases which is a compelling feature. However, it is essential to keep in mind the needs of the applications[114].

12.3.3.1 Mongo Queries examples:

The queries can be executed from Mongo shell as well as through scripts.

To query the data from a MongoDB collection, one would use MongoDB's find() method.

```
> db.COLLECTION_NAME.find()
```

The output can be formatted by using the pretty() command.

```
> db.mycol.find().pretty()
```

The MongoDB insert statements can be performed in the following manner:

```
> db.COLLECTION_NAME.insert(document)
```

"The \$lookup command performs a left-outer-join to an unsharded collection in the same database to filter in documents from the joined collection for processing" [121].

```
$ {
  $lookup:
  {
    from: <collection to join>,
    localField: <field from the input documents>,
    foreignField: <field from the documents of the "from" collection>,
    as: <output array field>
  }
}
```

This operation is equivalent to the following SQL operation:

```
$ SELECT *, <output array field>
  FROM collection
 WHERE <output array field> IN (SELECT *
                                FROM <collection to join>
                                WHERE <foreignField> = <collection.localField>);`
```

To perform a Like Match (Regex), one would use the following command:

```
> db.products.find( { sku: { $regex: /789$/ } } )
```

12.3.4 MongoDB Basic Functions

When it comes to the technical elements of MongoDB, it posses a rich interface for importing and storage of external data in various formats. By using the Mongo Import/Export tool, one can easily transfer contents from JSON, CSV, or TSV files into a database. MongoDB supports CRUD (create, read, update, delete) operations efficiently and has detailed documentation available on the product website. It can also query the geospatial data, and it is capable of storing geospatial data in GeoJSON objects. The aggregation operation of the MongoDB process data records and returns computed results. MongoDB aggregation framework is modeled on the concept of data pipelines [122].

12.3.4.1 Import/Export functions examples:

To import JSON documents, one would use the following command:

```
$ mongoimport --db users --collection contacts --file contacts.json
```

The CSV import uses the input file name to import a collection, hence, the collection name is optional [122].

```
$ mongoimport --db users --type csv --headerline --file /opt/backups/contacts.csv
```

“Mongoexport is a utility that produces a JSON or CSV export of data stored in a MongoDB instance” [122].

```
$ mongoexport --db test --collection traffic --out traffic.json
```

12.3.5 Security Features

Data security is a crucial aspect of the enterprise infrastructure management and is the reason why MongoDB provides various security features such as role based access control, numerous authentication options, and encryption. It supports mechanisms such as SCRAM, LDAP, and Kerberos authentication. The administrator can create role/collection-based access control; also roles can be predefined or custom. MongoDB can audit activities such as DDL, CRUD statements, authentication and authorization operations [123].

12.3.5.1 Collection based access control example:

A user defined role can contain the following privileges [123].

```
$ privileges: [
  { resource: { db: "products", collection: "inventory" }, actions: [ "find", "update" ] },
  { resource: { db: "products", collection: "orders" }, actions: [ "find" ] }
]
```

12.3.6 MongoDB Cloud Service

In regards to the cloud technologies, MongoDB also offers fully automated cloud service called Atlas with competitive pricing options. Mongo Atlas Cloud interface offers interactive GUI for managing cloud resources and deploying applications quickly. The service is equipped with geographically distributed instances to ensure no single point failure. Also, a well-rounded performance monitoring interface allows users to promptly detect anomalies and generate index suggestions to optimize the performance and reliability of the database. Global technology leaders such as Google, Facebook, eBay, and Nokia are leveraging MongoDB and Atlas cloud services making MongoDB one of the most popular choices among the NoSQL databases [124].

12.4 PyMONGO

PyMongo is the official Python driver or distribution that allows work with a NoSQL type database called MongoDB [125]. The first version of the driver was developed in 2009 [126], only two years after the development of MongoDB was started. This driver allows developers to combine both Python's versatility and MongoDB's flexible schema nature into successful applications. Currently, this driver supports MongoDB versions 2.6, 3.0, 3.2, 3.4, 3.6, and 4.0 [127]. MongoDB and Python represent a compatible fit considering that BSON (binary JSON) used in this NoSQL database is very similar to Python dictionaries, which makes the collaboration between the two even more appealing [128]. For this reason, dictionaries are the recommended tools to be used in PyMongo when representing documents [129].

12.4.1 Installation

Prior to being able to exploit the benefits of Python and MongoDB simultaneously, the PyMongo distribution must be installed using pip. To install it on all platforms, the following command should be used [130]:

```
$ python -m pip install pymongo
```

Specific versions of PyMongo can be installed with command lines such as in our example where the 3.5.1 version is installed [130].

```
$ python -m pip install pymongo==3.5.1
```

A single line of code can be used to upgrade the driver as well [130].

```
$ python -m pip install --upgrade pymongo
```

Furthermore, the installation process can be completed with the help of the easy_install tool, which requires users to use the following command [130].

```
$ python -m easy_install pymongo
```

To do an upgrade of the driver using this tool, the following command is recommended [130]:

```
$ python -m easy_install -U pymongo
```

There are many other ways of installing PyMongo directly from the source, however, they require for C extension dependencies to be installed prior to the driver installation step, as they are the ones that skim through the sources on GitHub and use the most up-to-date links to install the driver [130].

To check if the installation was completed accurately, the following command is used in the Python console [131].

```
import pymongo
```

If the command returns zero exceptions within the Python shell, one can consider for the PyMongo installation to have been completed successfully.

12.4.2 Dependencies

The PyMongo driver has a few dependencies that should be taken into consideration prior to its usage. Currently, it supports CPython 2.7, 3.4+, PyPy, and PyPy 3.5+ interpreters [127]. An optional dependency that requires some additional components to be installed is the GSSAPI authentication [127]. For the Unix based machines, it requires pykerberos, while for the Windows machines WinKerberos is needed to fullfill this requirement [127]. The automatic installation of this dependency can be done simultaneously with the driver installation, in the following manner:

```
$ python -m pip install pymongo[gssapi]
```

Other third-party dependencies such as ipaddress, certifi, or wincerstore are necessary for connections with help of TLS/SSL and can also be simultaneously installed along with the driver installation [127].

12.4.3 Running PyMongo with Mongo Deamon

Once PyMongo is installed, the Mongo deamon can be run with a very

simple command in a new terminal window [131].

```
$ mongod
```

12.4.4 Connecting to a database using MongoClient

In order to be able to establish a connection with a database, a MongoClient class needs to be imported, which sub-sequentially allows the MongoClient object to communicate with the database [131].

```
from pymongo import MongoClient client = MongoClient()
```

This command allows a connection with a default, local host through port 27017, however, depending on the programming requirements, one can also specify those by listing them in the client instance or use the same information via the Mongo URI format [131].

12.4.5 Accessing Databases

Since MongoClient plays a server role, it can be used to access any desired databases in an easy way. To do that, one can use two different approaches. The first approach would be doing this via the attribute method where the name of the desired database is listed as an attribute, and the second approach, which would include a dictionary-style access [131]. For example, to access a database called cloudmesh_community, one would use the following commands for the attribute and for the dictionary method, respectively.

```
db = client.cloudmesh_community  
db = client['cloudmesh_community']
```

12.4.6 Creating a Database

Creating a database is a straight forward process. First, one must create a MongoClient object and specify the connection (IP address) as well as the name of the database they are trying to create [132]. The example of this command is presented in the following section:

```
import pymongo
client = pymongo.MongoClient('mongodb://localhost:27017/')
db = client['cloudmesh']
```

12.4.7 Inserting and Retrieving Documents (Querying)

Creating documents and storing data using PyMongo is equally easy as accessing and creating databases. In order to add new data, a collection must be specified first. In this example, a decision is made to use the cloudmesh group of documents.

```
cloudmesh = db.cloudmesh
```

Once this step is completed, data may be inserted using the `insert_one()` method, which means that only one document is being created. Of course, insertion of multiple documents at the same time is possible as well with use of the `insert_many()` method [131]. An example of this method is as follows:

```
course_info = {
    'course': 'Big Data Applications and Analytics',
    'instructor': ' Gregor von Laszewski',
    'chapter': 'technologies'
}
result = cloudmesh.insert_one(course_info)
```

Another example of this method would be to create a collection. If we wanted to create a collection of students in the `cloudmesh_community`, we would do it in the following manner:

```
student = [ {'name': 'John', 'st_id': 52642},
            {'name': 'Mercedes', 'st_id': 5717},
            {'name': 'Anna', 'st_id': 5654},
            {'name': 'Greg', 'st_id': 5423},
            {'name': 'Amaya', 'st_id': 3540},
            {'name': 'Cameron', 'st_id': 2343},
            {'name': 'Bozer', 'st_id': 4143},
            {'name': 'Cody', 'price': 2165} ]
client = MongoClient('mongodb://localhost:27017/')
with client:
    db = client.cloudmesh
    db.students.insert_many(student)
```

Retrieving documents is equally simple as creating them. The `find_one()` method can be used to retrieve one document [131]. An implementation of this method is given in the following example.

```
gregors_course = cloudmesh.find_one({'instructor':'Gregor von Laszewski'})
```

Similarly, to retrieve multiple documents, one would use the `find()` method instead of the `find_one()`. For example, to find all courses thought by professor von Laszewski, one would use the following command:

```
gregors_course = cloudmesh.find({'instructor':'Gregor von Laszewski'})
```

One thing that users should be cognizant of when using the `find()` method is that it does not return results in an array format but as a cursor object, which is a combination of methods that work together to help with data querying [131]. In order to return individual documents, iteration over the result must be completed [131].

12.4.8 Limiting Results

When it comes to working with large databases it is always useful to limit the number of query results. PyMongo supports this option with its `limit()` method [132]. This method takes in one parameter which specifies the number of documents to be returned [132]. For example, if we had a collection with a large number of cloud technologies as individual documents, one could modify the query results to return only the top 10 technologies. To do this, the following example could be utilized:

```
client = pymongo.MongoClient('mongodb://localhost:27017/')
db = client['cloudmesh']
col = db['technologies']
topten = col.find().limit(10)
```

12.4.9 Updating Collection

Updating documents is very similar to inserting and retrieving the same. Depending on the number of documents to be updated, one would use the `update_one()` or `update_many()` method [132]. Two

parameters need to be passed in the `update_one()` method for it to successfully execute. The first argument is the query object that specifies the document to be changed, and the second argument is the object that specifies the new value in the document. An example of the `update_one()` method in action is the following:

```
myquery = { 'course': 'Big Data Applications and Analytics' }
newvalues = { '$set': { 'course': 'Cloud Computing' } }
```

Updating all documents that fall under the same criteria can be done with the `update_many` method [132]. For example, to update all documents in which course title starts with letter B with a different instructor information, we would do the following:

```
client = pymongo.MongoClient('mongodb://localhost:27017/')
db = client['cloudmesh']
col = db['courses']
query = { 'course': { '$regex': '^B' } }
newvalues = { '$set': { 'instructor': 'Gregor von Laszewski' } }

edited = col.update_many(query, newvalues)
```

12.4.10 Counting Documents

Counting documents can be done with one simple operation called `count_documents()` instead of using a full query [133]. For example, we can count the documents in the `cloudmesh_community` by using the following command:

```
cloudmesh = count_documents({})
```

To create a more specific count, one would use a command similar to this:

```
cloudmesh = count_documents({'author': 'von Laszewski'})
```

This technology supports some more advanced querying options as well. Those advanced queries allow one to add certain constraints and narrow down the results even more. For example, to get the courses thought by professor von Laszewski after a certain date, one would use the following command:

```
d = datetime.datetime(2017, 11, 12, 12)
for course in cloudmesh.find({'date': {'$lt': d}}).sort('author'):
```

```
pprint.pprint(course)
```

12.4.11 Indexing

Indexing is a very important part of querying. It can greatly improve query performance but also add functionality and aide in storing documents [133].

“To create a unique index on a key that rejects documents whose value for that key already exists in the index” [133].

We need to firstly create the index in the following manner:

```
result = db.profiles.create_index([('user_id', pymongo.ASCENDING)],  
unique=True)  
  
sorted(list(db.profiles.index_information()))
```

This command acutally creates two different indexes. The first one is the `*_id*` , created by MongoDB automatically, and the second one is the `user_id`, created by the user.

The purpose of those indexes is to cleverly prevent future additions of invalid `user_ids` into a collection.

12.4.12 Sorting

Sorting on the server-side is also available via MongoDB. The PyMongo `sort()` method is equivalent to the SQL `order by` statement and it can be performed as `pymongoascending` and `pymongodescending` [134]. This method is much more efficient as it is being completed on the server-side, compared to the sorting completed on the client side. For example, to return all users with first name Gregor sorted in descending order by birthdate we would use a command such as this:

```
users = cloudmesh.users.find({'firstname':'Gregor'}).sort(('dateofbirth',  
pymongo.DESCENDING))  
for user in users:  
    print user.get('email')
```

12.4.13 Aggregation

Aggregation operations are used to process given data and produce summarized results. Aggregation operations collect data from a number of documents and provide collective results by grouping data. PyMongo in its documentation offers a separate framework that supports data aggregation. This aggregation framework can be used to

“provide projection capabilities to reshape the returned data” [135].

In the aggregation pipeline, documents pass through multiple pipeline stages which convert documents into result data. The basic pipeline stages include filters. Those filters act like document transformation by helping change the document output form. Other pipelines help group or sort documents with specific fields. By using native operations from MongoDB, the pipeline operators are efficient in aggregating results.

The addFields stage is used to add new fields into documents. It reshapes each document in stream, similarly to the project stage. The output document will contain existing fields from input documents and the newly added fields [136]. The following example shows how to add student details into a document.

```
db.cloudmesh_community.aggregate([
{
    $addFields: {
        "document.StudentDetails": {
            $concat:[ '$document.student.FirstName', ' $document.student.LastName' ]
        }
    }
} ])
```

The bucket stage is used to categorize incoming documents into groups based on specified expressions. Those groups are called buckets [136]. The following example shows the bucket stage in action.

```
db.user.aggregate([
{ "$group": {
```

```

"_id": {
  "city": "$city",
  "age": {
    "$let": {
      "vars": {
        "age": { "$subtract": [{ "$year": new Date() }, { "$year": "$birthDay" }] },
        "in": {
          "$switch": {
            "branches": [
              { "case": { "$lt": [ "$$age", 20 ] }, "then": 0 },
              { "case": { "$lt": [ "$$age", 30 ] }, "then": 20 },
              { "case": { "$lt": [ "$$age", 40 ] }, "then": 30 },
              { "case": { "$lt": [ "$$age", 50 ] }, "then": 40 },
              { "case": { "$lt": [ "$$age", 200 ] }, "then": 50 }
            ] } } } },
        "count": { "$sum": 1 } } })

```

In the `bucketAuto` stage, the boundaries are automatically determined in an attempt to evenly distribute documents into a specified number of buckets. In the following operation, input documents are grouped into four buckets according to the values in the `price` field [136].

```

db.artwork.aggregate( [
  {
    $bucketAuto: {
      groupBy: "$price",
      buckets: 4
    }
  }
] )

```

The `collStats` stage returns statistics regarding a collection or view [136].

```

db.matrices.aggregate( [ { $collStats: { latencyStats: { histograms: true } } }
] ) )

```

The `count` stage passes a document to the next stage that contains the number documents that were input to the stage [136].

```

db.scores.aggregate( [ {
  $match: { score: { $gt: 80 } } },
  { $count: "passing_scores" } ] )

```

The `facet` stage helps process multiple aggregation pipelines in a single stage [136].

```

db.artwork.aggregate( [ {
  $facet: { "categorizedByTags": [ { $unwind: "$tags" },
    { $sortByCount: "$tags" } ],
    "categorizedByPrice": [
      // Filter out documents without a price e.g., _id: 7
      { $match: { price: { $exists: 1 } } },

```

```

    { $bucket: { groupBy: "$price",
      boundaries: [ 0, 150, 200, 300, 400 ],
      default: "Other",
      output: { "count": { $sum: 1 },
        "titles": { $push: "$title" }
      } } } ]], "categorizedByYears(Auto)": [
  { $bucketAuto: { groupBy: "$year", buckets: 4 }
} ]}])

```

The geoNear stage returns an ordered stream of documents based on the proximity to a geospatial point. The output documents include an additional distance field and can include a location identifier field [136].

```

db.places.aggregate([
  { $geoNear: {
    near: { type: "Point", coordinates: [ -73.99279 , 40.719296 ] },
    distanceField: "dist.calculated",
    maxDistance: 2,
    query: { type: "public" },
    includeLocs: "dist.location",
    num: 5,
    spherical: true
  } }])

```

The graphLookup stage performs a recursive search on a collection. To each output document, it adds a new array field that contains the traversal results of the recursive search for that document [136].

```

db.travelers.aggregate( [
  {
    $graphLookup: {
      from: "airports",
      startWith: "$nearestAirport",
      connectFromField: "connects",
      connectToField: "airport",
      maxDepth: 2,
      depthField: "numConnections",
      as: "destinations"
    }
  }
] )

```

The group stage consumes the document data per each distinct group. It has a RAM limit of 100 MB. If the stage exceeds this limit, the group produces an error [136].

```

db.sales.aggregate(
  [
    {
      $group : {
        _id : { month: { $month: "$date" }, day: { $dayOfMonth: "$date" },
        year: { $year: "$date" } },
        totalPrice: { $sum: { $multiply: [ "$price", "$quantity" ] } },
        ...
      }
    }
  ]
)

```

```
        averageQuantity: { $avg: "$quantity" },
        count: { $sum: 1 }
    }
]
)
```

The indexStats stage returns statistics regarding the use of each index for a collection [136].

```
db.orders.aggregate( [ { $indexStats: { } } ] )
```

The limit stage is used for controlling the number of documents passed to the next stage in the pipeline [136].

```
db.article.aggregate(
    { $limit : 5 }
)
```

The listLocalSessions stage gives the session information currently connected to mongos or mongod instance [136].

```
db.aggregate( [ { $listLocalSessions: { allUsers: true } } ] )
```

The listSessions stage lists out all session that have been active long enough to propagate to the system.sessions collection [136].

```
use config

db.system.sessions.aggregate( [ { $listSessions: { allUsers: true } } ] )
```

The lookup stage is useful for performing outer joins to other collections in the same database [136].

```
{
  $lookup:
  {
    from: <collection to join>,
    localField: <field from the input documents>,
    foreignField: <field from the documents of the "from" collection>,
    as: <output array field>
  }
}
```

The match stage is used to filter the document stream. Only matching documents pass to next stage [136].

```
db.articles.aggregate(
    [ { $match : { author : "dave" } } ]
)
```

The project stage is used to reshape the documents by adding or deleting the fields.

```
db.books.aggregate( [ { $project : { title : 1 , author : 1 } } ] )
```

The redact stage reshapes stream documents by restricting information using information stored in documents themselves [136].

```
db.accounts.aggregate(  
[  
  { $match: { status: "A" } },  
  {  
    $redact: {  
      $cond: {  
        if: { $eq: [ "$level", 5 ] },  
        then: "$$PRUNE",  
        else: "$$DESCEND"  
      }     }   }]);
```

The replaceRoot stage is used to replace a document with a specified embedded document [136].

```
db.produce.aggregate(  
[  
  {  
    $replaceRoot: { newRoot: "$in_stock" }  
  }  
])
```

The sample stage is used to sample out data by randomly selecting number of documents from input [136].

```
db.users.aggregate(  
[ { $sample: { size: 3 } } ]  
)
```

The skip stage skips specified initial number of documents and passes remaining documents to the pipeline [136].

```
db.article.aggregate(  
  { $skip : 5 }  
)
```

The sort stage is useful while reordering document stream by a specified sort key [136].

```
db.users.aggregate(  
[  
  { $sort : { age : -1, posts: 1 } }  
])
```

The sortByCounts stage groups the incoming documents based on a specified expression value and counts documents in each distinct group [136].

```
db.exhibits.aggregate( [ { $unwind: "$tags" }, { $sortByCount: "$tags" } ] )
```

The unwind stage deconstructs an array field from the input documents to output a document for each element [136].

```
db.inventory.aggregate( [ { $unwind: "$sizes" } ] )
db.inventory.aggregate( [ { $unwind: { path: "$sizes" } } ] )
```

The out stage is used to write aggregation pipeline results into a collection. This stage should be the last stage of a pipeline [136].

```
db.books.aggregate( [
    { $group : { _id : "$author", books: { $push: "$title" } } },
    { $out : "authors" }
] )
```

Another option from the aggregation operations is the Map/Reduce framework, which essentially includes two different functions, map and reduce. The first one provides the key value pair for each tag in the array, while the latter one

"sums over all of the emitted values for a given key"
[135].

The last step in the Map/Reduce process it to call the map_reduce() function and iterate over the results [135]. The Map/Reduce operation provides result data in a collection or returns results in-line. One can perform subsequent operations with the same input collection if the output of the same is written to a collection [137]. An operation that produces results in a in-line form must provide results with in the BSON document size limit. The current limit for a BSON document is 16 MB. These types of operations are not supported by views [137]. The PyMongo's API supports all features of the MongoDB's Map/Reduce engine [138]. Moreover, Map/Reduce has the ability to get more detailed results by passing full_response=True argument to the map_reduce() function [138].

12.4.14 Deleting Documents from a Collection

The deletion of documents with PyMongo is fairly straight forward. To do so, one would use the remove() method of the PyMongo Collection object [134]. Similarly to the reads and updates, specification of documents to be removed is a must. For example, removal of the entire document collection with a score of 1, would required one to use the following command:

```
cloudmesh.users.remove({"score":1, safe=True})
```

The safe parameter set to True ensures the operation was completed [134].

12.4.15 Copying a Database

Copying databases within the same mongod instance or between different mongod servers is made possible with the command() method after connecting to the desired mongod instance [139]. For example, to copy the cloudmesh database and name the new database cloudmesh_copy, one would use the command() method in the following manner:

```
client.admin.command('copydb',
                      fromdb='cloudmesh',
                      todb='cloudmesh_copy')
```

There are two ways to copy a database between servers. If a server is not password-protected, one would not need to pass in the credentials nor to authenticate to the admin database [139]. In that case, to copy a database one would use the following command:

```
client.admin.command('copydb',
                      fromdb='cloudmesh',
                      todb='cloudmesh_copy',
                      fromhost='source.example.com')
```

On the other hand, if the server where we are copying the database to is protected, one would use this command instead:

```
client = MongoClient('target.example.com',
                      username='administrator',
                      password='pwd')
```

```
client.admin.command('copydb',
    fromdb='cloudmesh',
    todb='cloudmesh_copy',
    fromhost='source.example.com')
```

12.4.16 PyMongo Strengths

One of PyMongo strengths is that allows document creation and querying natively

"through the use of existing language features such as nested dictionaries and lists" [134].

For moderately experienced Python developers, it is very easy to learn it and quickly feel comfortable with it.

"For these reasons, MongoDB and Python make a powerful combination for rapid, iterative development of horizontally scalable backend applications" [134].

According to [134], MongoDB is very applicable to modern applications, which makes PyMongo equally valuable [134].

12.5 MONGOENGINE

"MongoEngine is an Object-Document Mapper, written in Python for working with MongoDB" [140].

It is actually a library that allows a more advanced communication with MongoDB compared to PyMongo. As MongoEngine is technically considered to be an object-document mapper(ODM), it can also be considered to be

"equivalent to a SQL-based object relational mapper(ORM)" [131].

The primary technique why one would use an ODM includes data conversion between computer systems that are not compatible with each other [141]. For the purpose of converting data to the

appropriate form, a virtual object database must be created within the utilized programming language [141]. This library is also used to define schemata for documents within MongoDB, which ultimately helps with minimizing coding errors as well defining methods on existing fields [142]. It is also very beneficial to the overall workflow as it tracks changes made to the documents and aids in the document saving process [143].

12.5.1 Installation

The installation process for this technology is fairly simple as it is considered to be a library. To install it, one would use the following command [144]:

```
$ pip install mongoengine
```

A bleeding-edge version of MongoEngine can be installed directly from GitHub by first cloning the repository on the local machine, virtual machine, or cloud.

12.5.2 Connecting to a database using MongoEngine

Once installed, MongoEngine needs to be connected to an instance of the mongod, similarly to PyMongo [145]. The connect() function must be used to successfully complete this step and the argument that must be used in this function is the name of the desired database [145]. Prior to using this function, the function name needs to be imported from the MongoEngine library.

```
from mongoengine import connect  
connect('cloudmesh_community')
```

Similarly to the MongoClient, MongoEngine uses the local host and port 27017 by default, however, the connect() function also allows specifying other hosts and port arguments as well [145].

```
connect('cloudmesh_community', host='196.185.1.62', port=16758)
```

Other types of connections are also supported (i.e. URI) and they can be completed by providing the URI in the connect() function [145].

12.5.3 Querying using MongoEngine

To query MongoDB using MongoEngine an objects attribute is used, which is, technically, a part of the document class [146]. This attribute is called the QuerySetManager which in return

“creates a new QuerySet object on access” [146].

To be able to access individual documents from a database, this object needs to be iterated over. For example, to return/print all students in the cloudmesh_community object (database), the following command would be used.

```
for user in cloudmesh_community.objects:  
    print cloudmesh_community.student
```

MongoEngine also has a capability of query filtering which means that a keyword can be used within the called QuerySet object to retrieve specific information [146]. Let’s say one would like to iterate over cloudmesh_community students that are natives of Indiana. To achieve this, one would use the following command:

```
indy_students = cloudmesh_community.objects(state='IN')
```

This library also allows the use of all operators except for the equality operator in its queries, and moreover, has the capability of handling string queries, geo queries, list querying, and querying of the raw PyMongo queries [146].

The string queries are useful in performing text operations in the conditional queries. A query to find a document exactly matching and with state ACTIVE can be performed in the following manner:

```
db.cloudmesh_community.find( State.exact("ACTIVE") )
```

The query to retrieve document data for names that start with a case sensitive AL can be written as:

```
db.cloudmesh_community.find( Name.startswith("AL") )
```

To perform an exact same query for the non-key-sensitive AL one

would use the following command:

```
db.cloudmesh_community.find( Name.startswith("AL") )
```

The MongoEngine allows data extraction of geographical locations by using Geo queries. The geo_within operator checks if a geometry is within a polygon.

```
cloudmesh_community.objects(  
    point_geo_within=[[40, 5], [40, 6], [41, 6], [40, 5]])  
cloudmesh_community.objects(  
    point_geo_within={"type": "Polygon",  
        "coordinates": [[[40, 5], [40, 6], [41, 6], [40, 5]]]})
```

The list query looks up the documents where the specified fields matches exactly to the given value. To match all pages that have the word coding as an item in the tags list one would use the following query:

```
class Page(Document):  
    tags = ListField(StringField())  
  
Page.objects(tags='coding')
```

Overall, it would be safe to say that MongoEngine has good compatibility with Python. It provides different functions to utilize Python easily with MongoDB which makes this pair even more attractive to application developers.

12.6 FLASK-PyMONGO

"Flask is a micro-web framework written in Python"
[147].

It was developed after Django, and it is very pythonic in nature which implies that it is explicitly targeting the Python user community. It is lightweight as it does not require additional tools or libraries and hence is classified as a Micro-Web framework. It is often used with MongoDB using PyMongo connector, and it treats data within MongoDB as searchable Python dictionaries. The applications such as Pinterest, LinkedIn, and the community web page for Flask are using the Flask framework. Moreover, it supports various features such as

the RESTful request dispatching, secure cookies, Google app engine compatibility, and integrated support for unit testing, etc [147]. When it comes to connecting to a database, the connection details for MongoDB can be passed as a variable or configured in PyMongo constructor with additional arguments such as username and password, if required. It is important that versions of both Flask and MongoDB are compatible with each other to avoid functionality breaks [148].

12.6.1 Installation

Flask-PyMongo can be installed with an easy command such as this:

```
$ pip install Flask-PyMongo
```

PyMongo can be added in the following manner:

```
from flask import Flask
from flask_pymongo import PyMongo
app = Flask(__name__)
app.config["MONGO_URI"] = "mongodb://localhost:27017/cloudmesh_community"
mongo = PyMongo(app)
```

12.6.2 Configuration

There are two ways to configure Flask-PyMongo. The first way would be to pass a MongoDB URI to the PyMongo constructor, while the second way would be to

“assign it to the MONGO_URI Flask configuration variable” [148].

12.6.3 Connection to multiple databases/servers

Multiple PyMongo instances can be used to connect to multiple databases or database servers. To achieve this, one would use a command similar to the following:

```
app = Flask(__name__)
mongo1 = PyMongo(app, uri="mongodb://localhost:27017/cloudmesh_community_one")
mongo2 = PyMongo(app, uri="mongodb://localhost:27017/cloudmesh_community_two")
mongo3 = PyMongo(app, uri=
```

```
"mongodb://another.host:27017/cloudmesh_community_Three")
```

12.6.4 Flask-PyMongo Methods

Flask-PyMongo provides helpers for some common tasks. One of them is the `Collection.find_one_or_404` method shown in the following example:

```
@app.route("/user/<username>")
def user_profile(username):
    user = mongo.db.cloudmesh_community.find_one_or_404({"_id": username})
    return render_template("user.html", user=user)
```

This method is very similar to the MongoDB's `find_one()` method, however, instead of returning `None` it causes a 404 Not Found HTTP status [148].

Similarly, the `PyMongo.send_file` and `PyMongo.save_file` methods work on the file-like objects and save them to GridFS using the given file name [148].

12.6.5 Additional Libraries

Flask-MongoAlchemy and Flask-MongoEngine are the additional libraries that can be used to connect to a MongoDB database while using enhanced features with the Flask app. The Flask-MongoAlchemy is used as a proxy between Python and MongoDB to connect. It provides an option such as server or database based authentication to connect to MongoDB. While the default is set server based, to use a database-based authentication, the config value `MONGOALCHEMY_SERVER_AUTH` parameter must be set to `False` [149].

Flask-MongoEngine is the Flask extension that provides integration with the MongoEngine. It handles connection management for the apps. It can be installed through pip and set up very easily as well. The default configuration is set to the local host and port 27017. For the custom port and in cases where MongoDB is running on another server, the host and port must be explicitly specified in connect strings within the `MONGODB_SETTINGS` dictionary with `app.config`,

along with the database username and password, in cases where a database authentication is enabled. The URI style connections are also supported and supply the URI as the host in the MONGODB_SETTINGS dictionary with app.config. There are various custom query sets that are available within Flask-Mongoengine that are attached to Mongoengine's default queryset [150].

12.6.6 Classes and Wrappers

Attributes such as cx and db in the PyMongo objects are the ones that help provide access to the MongoDB server [148]. To achieve this, one must pass the Flask app to the constructor or call init_app() [148].

“Flask-PyMongo wraps PyMongo’s MongoClient, Database, and Collection classes, and overrides their attribute and item accessors” [148].

This type of wrapping allows Flask-PyMongo to add methods to Collection while at the same time allowing a MongoDB-style dotted expressions in the code [148].

```
type(mongo.cx)
type(mongo.db)
type(mongo.db.cloudmesh_community)
```

Flask-PyMongo creates connectivity between Python and Flask using a MongoDB database and supports

“extensions that can add application features as if they were implemented in Flask itself” [151],

hence, it can be used as an additional Flask functionality in Python code. The extensions are there for the purpose of supporting form validations, authentication technologies, object-relational mappers and framework related tools which ultimately adds a lot of strength to this micro-web framework [151]. One of the main reasons and benefits why it is frequently used with MongoDB is its capability of adding more control over databases and history [151].

12.7 WORKBREAKDOWN

- Introduction - Nishad Tupe fa18-523-64
- Learning Outcome - Izolda Fetko fa18-523-60
- MongoDB - Nishad Tupe fa18-523-64
- PyMongo - Izolda Fetko fa18-523-60
- MongoEngine, Flask-PyMongo - Vishal Bhoyar fa18-523-72
- MongoEngine (Peer reviewed) - Izolda Fetko fa18-523-60
- Flask-PyMongo (Peer reviewed) - Nishad Tupe fa18-523-64

13 NATURAL LANGUAGE APPLICATIONS AND CHALLENGES WITHIN BIG DATA FA18-523-61

Jay Stockwell
jaystock@iu.edu
Indiana University
hid: fa18-523-61
github: [blue icon](#)

Keywords : Natural Language Processing, Natural Language Understanding, Natural Language Toolkit, Deep Learning, SyntaxNet, Part of Speech tagging, Hadoop

13.1 ABSTRACT

This paper examines Natural Language Processing (NLP) and its usage in Big Data, the current challenges facing NLP, and solutions to those challenges using new concepts and applications. Organizations have recently begun to harness the immense power of big data and how the concept can prove to be a beneficial component. The term big data used to be a scary term that elicited feelings of consternation and anxiety, but with organizations experiencing exponential growth in data volume, the term has become mainstream and widely accepted. Big data is largely unstructured text and constantly in a state of enormous flux, which is why NLP offers many opportunities to tap into this vast data resource [152]. This paper will explore the evolving, and sometimes challenging relationship between NLP and Big Data, the problems that NLP can solve, which applications are leveraged, and how the data can be transformed into a presentable format for consumption.

13.2 INTRODUCTION

Big data “describes the growing volume of structured and unstructured, multi-source information that is too large for traditional applications to handle [152].” The volume of today’s data is on an unprecedented growth trajectory due to the ample methods to collect and analyze data. The Internet of Things, mobile devices, sensors, cameras, and software logs are just some examples of non-traditional methods of data collection that contribute to the abundance of information today [153]. There’s no end in sight to exponential growth that is projected over the next several years.

Natural Language Processing is a relatively new concept and is gaining momentum in the use of text analysis and presentation. Elizabeth Liddy from Syracuse University provides a great definition:

“Natural Language Processing is a theoretically motivated range of computational techniques for analyzing and representing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications” [154].

Essentially, NLP’s goal is to achieve a level of text analysis and processing that mimicks very closely the way that humans process language. While NLP has truly made significant progress over the last several years, the challenge of deciphering the exact context and inference of text is still an area of which NLP is still improving [154].

13.3 NATURAL LANGUAGE CHALLENGES

As previously mentioned, there are many challenges that face NLP today. Some of the challenges pertain to how NLP parses sentences, deals with cultural differences, language translation, conversational issues (ie. whether the statement is a question or answer), and sentiment. In order for NLP to function in an effective manner within Big Data, there needs to be a process put in place to assist with these issues.

Parsing, the ability to deconstruct a sentence into its parts, is a major challenge facing NLP. Parsing can lead to ambiguity because it can be difficult to detect the correct syntax, and/or the exact interpretation of each word. Prepositions within a sentence can cause confusion because it can be hard to determine which word is being modified. as explained by Armando Viera and Bernadete Ribeiro:

“..the sentence “Alice drove down the street in her car” has at least two possible dependency parses. The first corresponds to the (correct) interpretation where Alice is driving in her car; the second corresponds to the (absurd but possible) interpretation where the street is located in her car. The ambiguity arises because the preposition in can modify either drove or street. [155].”

Part of Speech tagging is another NLP concern. Part of speech tagging is the process of assigning a part of speech definition to a word within a sentence.[156]. Common parts of speech include nouns, verbs, adjectives, and adverbs. Challenges can arise due to the lack of context. There are many words that have multiple meanings, which can lead to ambiguity when computers try to assign a descriptor. For example, the word chair can have multiple meanings; you can chair(verb) or lead a meeting, or you can sit in a chair (noun) [156].

Another challenge is Natural Language Understanding (NLU). NLU pertains to the ability of a computer to comprehend the sentence structure and intended meaning of human languages, which in turn allows humans to fully communicate and interact with machines using real sentences [157]. This concept is gained interest in recent years due to the many possible ways this can be leveraged within applications on a commercial scale.

With the rise of Artificial Intelligence (AI), NLU has risen in complexity due to the extraordinary level of computations involved.

The concept is considered one of the most challenging problems in the AI world. These types of problems require intense computational effort and also require human intervention and resources as they cannot be solved by machines alone. Challenging AI problems are

known as AI-Hard or AI-complete.

"In the field of artificial intelligence, the most difficult problems are informally known as AI-complete or AI-hard, implying that the difficulty of these computational problems is equivalent to that of solving the central artificial intelligence problem—making computers as intelligent as people, or strong AI [158]. To call a problem AI-complete reflects an attitude that it would not be solved by a simple specific algorithm [159]."

In order to NLU to properly function, machines must be programmed to understand text. NLU must follow all the text translation rules that any human would follow if they were reading through any type or text document. Machines must have the ability to mimick human abilities and human intellectual skills such as reason, common sense, and intuition that relate to how humans perceive language and social intelligence concepts [159]. NLU requires an enormous amount of work in order to prove effective. NLU applications require extensive data gathering and subject matter investigation and research in order to properly train the system to perform [160].

13.4 NATURAL LANGUAGE PROCESSING SOLUTIONS

There are some solutions in place to address the challenges facing NLP. With respect to the parsing problem, there is a relatively new application designed by Google called SyntaxNet. SyntaxNet is based on the TensorFlow open source library readily available to users for designing deep learning models. With SyntaxNet, Google employed a normalized neural network model that provides an output of possible syntactical possibilities or hypotheses given a group of words [155]. SyntaxNet runs the model multiple times and discards hypotheses that are ranked lower and appear to be unlikely candidates. As far as parsing, SyntaxNet has developed a reputation for being the best parser, being known to sometimes exceed human accuracy, and recently made available in 40 languages. [155].

The latest version of SyntaxNet can be trained against a separate data set, and individuals have a good deal of freedom in tweaking the parameters of the model to better fit the particular nuances of their datasets. SyntaxNet includes a model built specifically for the English language entitled Parsy McParseface and can be used to analyze English texts right out the box [161].

Members of the Stanford University's Natural Language Processing Group has developed a part of speech tagger that works remarkably well. The application runs on Java and is somewhat memory intensive, requiring upwards of 60-200 Mb of memory to function efficiently, with around 1 Gb recommended in order to train a dataset [162]. The latest download contains three distinct tagger models for the English Language as well as an Arabic, German, Chinese, and French model. These models can be retrained on any language. During an experiment against Penn Treebank WSJ data, the Stanford POS tagger returned an impressive per-position tag accuracy of 97.24% [163].

As NLU continues to become more mainstream, many companies are embedding proprietary NLU algorithms within their products to further enhance their overall NLP capabilities. Some of the companies and their associated applications are Apple (Siri), Google(GoogleNow, Google Search), Microsoft(Cortana), and IBM (Watson, DeepQA) [160]. These products are just the beginning of a new wave of technology that will surely influence how organizations incorporate NLP into their business processes and decisions.

In order for NLP to succeed, it needs to be trained against very large datasets or corpuses. A corpus is a collection of written texts used for research or investigation purposes. Two examples of widely used corpuses are:

"The Google n-gram corpus, a trillion word database containing phrases (up to 5 words long) occurring on public Web pages. The USENET corpus, a 25 billion word (compressed) corpus containing public USENET postings on 47,680 English language, non-binary-file

newsgroups between Oct 2005 and Jan 2010 [164]."

In order to write effective programs against such large big data sets, a new platform has been developed entitled Natural Language Toolkit (NLTK). The platform is completely open source and supported by a strong community of users. Users can leverage the platform to build applications using the Python programming language. NLTK provides user with a simplistic interface that can access over 50 different corpora data sets linguistic and lexical data resources [165]. Users can also leverage several built-in libraries for classification, tokenization, tagging, parsing, semantic reasoning, and powerful wrappers to utilize with NLP libraries [165]. NLTK comes with a comprehensive learning guide to assist users, specifically those with little to no Python experience, to get up to speed with the various syntax commands that are used within NLP.

Some organizations have started to leverage Hadoop, an open source processing framework that works extremely well against very large textual datasets that require enormous amounts of computational power. Hadoop can also act as a data management application providing massive storage space and can handle numerous concurrent processing tasks. Hadoop's ability to work with various kinds of structured and unstructured data make it an ideal application for NLP, as it can handle an exorbitant amount of data from sources such as internet clickstream records, web server application logs, social media sites such as facebook and twitter, customer emails, and sensor information from the internet of things [166]. The aforementioned deep learning algorithms can be set up to run in a Hadoop environment to leverage its extraordinary size and computational power.

In recent years, with the precipitous rise of data science and the use of algorithms for predictive analysis among other areas, one concept in particular, Deep Learning, has emerged as a possible solution to answering some of the aforementioned challenges facing NLP.

"Deep learning (DL) has had a tremendous impact on natural language processing (NLP). After image and

audio, probably this is the area where DL has unleashed the most transformative forces. For example, almost all projects related to NLP at Stanford University, one of the most respected institutions working on this area, involve DL research [155]."

Deep learning, or Deep Neural Networks as it is sometimes referred to, is a branch of AI. The deep neural networks go through a series of transformation steps on the inputted data and leverages the what it learns to build a comprehensive statistical model [167]. The model will continue to run through a series of iterations until it returns an acceptable level of accuracy. Deep learning algorithms can be trained against big data sets just like other standard algorithms and have proven to be effective in this manner. For example, "Trained on movie subtitles, language models are able to generate basic answers to questions about object colors or facts [155]. Deep Learning networks can be coded using the NLTK and Hadoop platforms mentioned above to tackle the numerous challenges facing NLP today.

13.5 CONCLUSION

In conclusion, the NLP field has proven its importance in the data world by allowing for entities to analyze and evaluate many aspects of the different components of language. The challenges that have arisen from leveraging NLP such as parsing and part of speech tagging have been met with the development of new applications such as Google's SyntaxNet, Stanford's Part of Speech tagger, and NLTK. Since NLP is still in its early stages, there will continue to be new challenges, and just like SyntaxNet, new applications will meet these challenges and make NLP even more effective. These applications leverage algorithms such as neural networks and deep learning to facilitate the effective training of datasets. Today's organizations deal with, and store enormous amounts of textual data from many different sources. Because this information is comprised of primarily text, organizations that leverage big data infrastructures and work with this type of data are starting to understand the

implications that NLP provides in evaluating their growing stores of data to detect patterns, connections and trends within their various data sources [168]. Big data storage has become easier to manage due to new open source database management systems such as Hadoop. The volume of NLP data will continue to grow exponentially and new processing and storage management technologies will need to be designed to be not only scalable, but have the capacity to work with ever changing business demands. NLP will continue to grow and we need to be ready to meet the new challenges that the emerging technology presents.

Manek Bahl, Sohan Udupi Rai
mbahl@iu.edu, surai@iu.edu
Indiana University
hid: fa18-523-62 fa18-523-69
paper: [blue link](#)

14.1 INTRODUCTION

Data obtained in real-time from various sources are called Data Streams; processing and extracting insights from such sources is called Big Data Streaming or Real-time streaming analytics. While conventional big-data systems can handle near real-time data by performing micro-batch processing, they have still got latency issues which is not acceptable in some critical applications. We look at various technologies available today for handling data streams and see how each of these deal with the challenges associated with the task. The paper then looks at some real-life examples exploring the implementation of Big data streaming systems in various domains.

Deriving insights from data has always been the key requirement for organizations to gain edge over competitors in the market. But there are certain applications where this needs to be done within a few milliseconds for it to be useful. Recent boom in the field of Internet of Things makes it paramount for analytics systems to be able to deal with such huge data quickly and effectively. This is where Big Data Streaming has gained immense importance in the recent past.

Various domains such as eCommerce, Banking and Finance, Social media generate such data sequentially, and there is a heavy need for this data to be processed as they arrive on a row by row basis to derive actionable insights. These insights enable the companies to understand the recent consumer behavior and act accordingly to

cater to their needs promptly in a timely manner. This is beneficial both for the consumers, who get better service, and for organizations, to be more efficient and proactive in the business decision making [169].

14.2 STREAM PROCESSING VS BATCH PROCESSING

For a long time, the traditional way of dealing with data has been Batch Processing, wherein, the data is collected and continuously stored in a database. Insights are derived from this data by dividing it into batches and then deploying suitable algorithms. The size of the batch could be anything from as short as a day to maybe even a year, depending on the type of insights required. While this type of insight is valuable for the companies to make long term business decisions, there are certain critical applications such as fraud detection, which require analytics to be done in real-time so action can be taken immediately, before the fraudulent transaction can be completed. There arises the need for Stream Processing [170].

Advantages of Stream Processing compared to Batch Processing:

- 1) Data collected from continuous data sources such as traffic sensors, transaction logs and most other IOT sensors is Time Series data and Batch Processing would be tedious since there would be a need for aggregation across batches. Stream processing handles the data without the need for any such aggregation.
- 2) Stream processing reduces the storage requirement of a system. Since analysis is done in real-time, there is no need for the entire generated data to be stored in the system. Only the useful data can be stored.
- 3) Stream Processing requires less hardware capabilities compared to Batch Processing since the amount of data on which analysis is performed is small compared to the large volume of data in Batch Processing [171].

14.3 CHALLENGES IN STREAM PROCESSING

Unlike batch processing, wherein all of the incoming data is first stored and then divided into batches for processing, stream

processing systems must have the capability to handle incoming data as and when it arrives. This leaves room for Data loss. Another factor that comes into play when dealing Stream data is ensuring data serialization. i.e. systems must ensure that messages must be processed in the order in which they were generated by the source. Thus, maintaining data integrity poses a great challenge for Stream processing systems.

Streaming application need to maintain the state and offset to keep track of the last message that was processed. While this is relatively simple to do, it poses a big challenge when dealing with system failure or in the case of vertical scaling since the new version of the application may pose compatibility issues [172].

14.4 BIG DATA STREAMING ARCHITECTURE AND TECHNOLOGIES

Over the recent years, various technologies have been developed for Stream Processing. The architecture adopted by these technologies often differ from one another, making some of these more suitable for some applications than the other. Some of the prominent technologies for Stream Processing and the architecture they use, are discussed below.

14.4.1 Apache Spark

Apache Spark is an open source analytics engine that uses distributed cluster computing framework. It was developed in UC, Berkeley and the codebase was then given to Apache Software Foundation which now maintains it. Spark is known for its high performance and can be used interactively using Scala, R, Python and SQL. It provides an interface to perform various analytics functions such as Machine Learning using MLLib, Streaming using Spark Streaming and Exploratory Analysis using GraphX. Spark Streaming enables us to build streaming data applications which are scalable, fault tolerant. Spark has the capability to ingest data from various sources such as HDFS, Kafka, Flume etc. and then uses complex algorithms to process

the data which can be then stored into a database or can be written out to a file. Spark Streaming receives divides the incoming data stream into batches which are then processed by the Spark Engine to produce a batch of final data. Spark Streaming brings in an ease of operation as it lets us write the streaming jobs in the same way batch jobs. The biggest advantage of using Streaming is that it has the capability to recover lost work as an inbuilt functionality. Moreover, the streaming data can be concatenated with historical data to build interactive applications. Spark uses HDFS or ZooKeeper for high availability. Currently Spark Streaming is widely being used by Yahoo, Uber, Netflix and eBay for providing real time analytics [173].

14.4.2 Apache Storm

Storm is commonly known as the “Hadoop of Real Time Processing” as it processes the streaming data as reliably as Hadoop does for batch processing. It is a distributed framework used for stream processing which was developed by a team at BackType and Twitter and was written in Clojure. The application is designed at a directed acyclic graph with spouts (input streams) and bolts (processing modules) as the vertices and the data flows in the form of tuples. The function of the spouts is to get the input streaming data and pass it to the bolts. Specialized spouts are available to retrieve data from multiple sources however Storm provides an option to create customer spouts as well. The data is then processed in the bolts and as per requirement the processed data is passed to a database or a file system [174]. Like Spark, Storm also provides high fault-tolerance, but it does not have a feature to use the same code for batch and stream processing. Current users of Storm include Twitter, Groupon, Yahoo and Spotify for its ability to create low latency distributed systems for streaming.

14.4.3 Apache Flink

Apache Flink uses a DataFlow model similar to Storm. But the main difference which makes it superior is that unlike Storm, it processes the events as and when they occur rather than processing micro-

batches. This is especially useful in applications where the data stream is sporadic i.e. extremely sparse. This approach reduces amount resources needed to handle the stream. It also eliminates the effort needed to determine the optimal size of micro-batches which is done by trial and error in Storm. Flink is also more flexible compared to Storm due to the simplistic nature of its API. Its SQL API interface makes it very possible for non-programmers to deal with the data stream application. Flink can be setup in either of the two modes – Standalone or Distributed [175].

14.4.4 Apache Kafka

Apache Kafka is an open source platform for Stream processing. Its design is inspired from the Transaction logs maintained by a database management system. It is a distributed system, wherein multiple nodes known as 'Brokers' work together to form a cluster. Its distributed nature ensures high availability and High scalability. Which means that Kafka systems are fault tolerant and provide horizontal-scalability. Horizontal-scalability nature of Kafka is important since it ensures that additional computing power can be provided to the system by adding nodes without disrupting the ongoing operations. The main data structure used by Kafka is a commit log data structure which only allows appends. Once added to the data structure, records cannot be deleted or modified. This ensures that the data are placed in exactly the same order in which the events occurred. This data structure has an added advantage that reads and writes are done in constant time $O(1)$ which drastically increases the speed at which Kafka can handle streaming. Read and write operations can be done simultaneously as there is no 'lock' placed on the data during a write operation. The messages stored in the Kafka nodes are divided into sub- divisions called Topics. Topics are further divided into smaller divisions called Partitions, for improved performance. The commit log-type data structure used in Kafka ensures that all messages in each partition are in the order in which they came in. It also means that Kafka provides excellent performance delivering messages at near network speed without placing the data in RAM, it uses disks to store all its records instead.

Kafka stores the data for a set amount of time during which the consumer can use offsets to pull any record they want. Partitions are replicated and placed in multiple nodes of the system to ensure High-availability [176].

14.4.5 Amazon Kinesis

Kinesis is Amazon's solution to streaming data and analytics. Kinesis Data Stream is one of the key components of Kinesis platform which also includes the Kinesis Video Streams, Kinesis Data Firehose and Kinesis Data Analytics. Data streams reads the data from various input sources in the form of data records. These data records can be analyzed or stored in whichever way the application demands. The data streams start ingesting data within a second of when the data is added. The data then can be sent to any of the built-integrations or can be stored in various third-party stores such as DynamoDB or Cassandra. One advantage that Kinesis Streams offers is scaling of the applications, it allows dynamic changing of the throughput of the stream based on the volume of data expected [177].

14.4.6 Hortonworks Dataflow

Hortonworks DataFlow is an open source distributed platform capable of ingesting, storing and analyzing streaming data from multiple sources simultaneously. It provides a GUI, thus eliminating the need for the end user to have an understanding of programming. Being powered by Apache Nifi, HDF has the ability to ingest data from a variety of Streaming sources with ease. HDFs integration with Apache Ranger ensures excellent data security. HDF uses Apache Kafka as the streaming platform enabling it to process several million transactions per second while allowing users to deploy several Machine learning algorithms. The Streaming Analytics Manager provides users to perform the analytics in a simple visual fashion [178].

14.5 INDUSTRIAL USE CASES OF BIG DATA STREAMING

E-Commerce: Real time transactions can be clustered and used along-with various other features such as product reviews to provide recommendations to the customers based to the latest trends. Alibaba and e-Bay are pioneers of using big data streaming to enhance their business.

Healthcare: Streaming has become an integral part of healthcare industry now with providers analyzing patient records to forecast any future health issues and recommendations to prevent them. Companies such as MyFitnessPal use Apache Spark to clean the data entered by the users to recommend healthy food diet.

Entertainment: Netflix's application monitoring system is largely built on Amazon Kinesis which monitors all the applications within Netflix and tries to detect issues to give a high availability to its customers. Netflix also uses Apache Spark to collect all user activities on the app and analyzes it to come up with personalized recommendations.

Social Media: With various organizations paying so much attention to reviews being posted by users of various Social Websites, it has become increasingly important to be able to analyze the public sentiment. Twitter uses Apache Storm internally on its various applications for anomaly detection to provide high availability to its users.

Banking and Finance: One major use case of big data streaming in financial domain is fraud detection. If a fraudulent transaction is detected, it is imperious that corrective or preventive measures be taken in real time [179].

Mark Miller
mgm3@iu.edu
Indiana University Bloomington
hid:fa18-523-63
github: [blue user icon](#)

Keywords: HID fa18-523-63, Scikit Learn, Machine Learning Methods, Python, Artificial Intelligence

Scikit-learn [180] is Python's inherent machine learning library. It is a robust library that intends uses object oriented programming to implement commonly used machine learning algorithms effectively, efficiently, pythonically, and swiftly. While, for specific purposes, many experts are able to implement their own algorithms that may improve upon the Scikit-learn library, it has sufficient tools and robustness that enable it to be the leading library for machine learning topics within Python.

There are built in libraries to Python that can make these tasks much simpler to understand and to implement, Scikit-learn provides one such solution [180]. Junior machine learning experts are gaining footing in the industry and are able to gain reputation, thanks to the help of many of these libraries.

As data becomes larger and larger with time, experts in the field are needed to perform this operation. Or, better stated, experts are needed to be able to program computers to perform these operations for them. A computer can process streamlined and well-defined data faster than humans in some instances, especially when reviewing the data becomes increasingly tedious. While algorithms may never be as good at recognizing what is in an image quite like a human can, they will become closer and closer to the point where advertising will be even more targeted, devices will understand the wants of their human masters clearly, and effective decisions can be

made with minimal error. Scikit-learn is not the most sophisticated a capable machine learning algorithm out there, but it is effective and easily implemented via Python, which will make it the workhorse of my semester project.

15.1 THE SUPERVISED ALGORITHMS (SOME OF THEM)

- Nearest Neighbors: Nearest neighbors is a machine learning algorithm that makes the decision of one input variable based on training data (making it a supervised algorithm) that most similarly matches itself. Once one of the training sets is identified as the closest match of inputs, it will assign the same category for the test instance. With careful parameter tuning, some scholars believe this to be a better classification method than random forests [181]
- Naive Bayes: Naive Bayes takes a look at Bayesian statistics and makes one majorly naive assumption: all of the inputs are independent of each other. While this is a glaring assumption, due to ease of implementations, most issues that would arise from generally erroneous assumption are not impactful [182].
- Decision trees: Separating on different attributes of the input data, decision trees are one of the more robust machine learning algorithms in that they are able to handle a wide variety of inputs and still maintain their quality [183]. Splitting on each attributes (usually on traits that maximize the entropy of the model, to enhance effectiveness). A branch off algorithm to decision trees are random forests which use many small, randomly chosen (with replacement) trees that can use small subsets of the data to formulate better opinions in a less computationally intensive way.
- Neural network models (supervised) {184} are algorithms that are particularly good at image classification. They obtain different neurons, each of which contributes in the decision making. They are based off of the way a human neural network would work if it could be modeled accurately via

code. each neural network has different levels which contribute to the decision making process.

15.2 THE UNSUPERVISED ALGORITHMS (SOME OF THEM)

There are many unsupervised learning algorithms supported by Scikit, here is highlighted two of them.

- Clustering [185]: Stemming from the k-means clustering, this is designed to group the datapoints based on similarity to others in the same dataset. The inputs are generally strictly numerical but can be n-dimmensional. Clustering converges quickly via iterative methods but is highly sensitive to initialization, making it very important to have domain knowledge and valuable visualization strategies when the data is 3-dimmensional and higher.
- Neural Network (unsupervised) [186]: much like the aforementioned neural networks, sklearn has libraries for unsupervised machine learning algorithms, which don't require training data to make decisions. In this sense, it becomes more of a clustering algorithm than a group identification.

15.3 OTHER METHOD GROUPINGS WITHIN SKLEARN

- Dataset transformations [www-fa18-523-63-scikit-learn-developers]: Oftentimes, data comes mangled and hard to use, requiring the need for effective data wrangling. scikit-learn has methods for feature extraction, preprocessing, random projection, dimmensionality reduction, and more. This makes it a valuable library for more aspects than just the machine learning algorithms themselves.
- Dataset loading utilities [www-fa18-523-63-scikit-learn-developers]: Scikit-learn has the utilities needed to load data as well as read it. There are Application Programming Interfaces for training datasets, real-world datasets,

generated-datasets, and the tools needed to use them effectively.

15.4 FURTHER FUNCTIONALITIES TO SCIKIT

There are more uses and tools in scikit-learn than what I have mentioned here. I couldn't just copy the user guide or man pages for these articles, even though they are good. The user guide contains valuable examples and assists with syntax. You will need basic machine learning understanding to be able to use any of these methods in this library. Once the understanding is there and basic implementations are used, microtuning and enhancing of the algorithms comes quickly and simply to an expert with a good eye. I didn't mention every individual method because there are plenty that I have no experience using and don't understand the nature of the machine learning or other algorithm.

15.5 REAL WORLD APPLICATIONS FOR SCIKIT LEARN

The real world applications are nearly as endless as are the applications for machine learning and artificial intelligence. The trick is getting the data to work together, whether that be through internet of things, internet of computers, internet of people, etc. I have personally used this tool in many ways, ranging from sports analytics to automation of analysis of the stock exchange. Expert knowledge of this library alone can bring six-figure salaries as a machine learning engineer, which many major and minor companies alike choose to employ.

16 NATURAL LANGUAGE TEXT PROCESSING AND LANGUAGE GENERATION FA18-523-67, FA18-523-65

Sahithya Sridhar, Prajakta Patil
sahsrid@iu.edu, patilpr@iu.edu
Indiana University, Bloomington
hid: fa18-523-67, fa18-523-65
github: [blue icon](#)

keywords: Natural language processing, Machine learning, Big Data, Yelp data

"The goal of NLP is to accomplish human-like language processing [187]".

Natural language processing (NLP) forms the link between machine language (binary code) to natural languages (which we humans speak). It borrows elements from artificial intelligence, computer science and information engineering. The success of NLP is highly dependent on how one can successfully program computers to process, analyze and interpret huge amounts of natural language data. Key opportunities/challenges in this inter-disciplinary field involves speech recognition, language interpretation and generation of natural language from a machine representation. With NLP, it is possible for computers to read text, hear speech, interpret, measure sentiment and determine which parts are important [188].

16.1 PURPOSE OF NATURAL LANGUAGE PROCESSING

NLP was initially called NLU (Natural language understanding). A natural language understanding system will be used to paraphrase the input text, translate text into another language, analyze the content of the text and draw conclusions. As NLP still cannot draw inference from the text, it is still dependent on NLU. One example of

such difficulty is understanding if a certain word in a sentence is a noun or a verb. For example, the word leave can be either a noun or a verb depending on the context of the sentence [188].

It is well known [188] that quite a bit of information which exists in our world is very unstructured. The challenge is to make a computer understand this and extract data from it. Humans have been writing down things for thousands of years but computers cannot yet truly understand the language as we do. The trick is to break down the process of understanding English into smaller pieces and understanding each piece separately [188].

According to [187] NLP has two distinct focuses:

- **Language processing:** Produces a meaningful representation by analyzing the language. This is similar to a machine reading the text.
- **Language generation:** Language is produced from representation. This requires a planning capability. This is similar to a machine writing the text.

The text is manipulated for abstractions and indexed for automatic knowledge extraction. Producing text in a desirable format is one of the major research areas in NLP. Structuring of large bodies of textual information to retrieve an information is classified under the natural language text processing [187].

“The central task for natural language text processing systems is the translation of potentially ambiguous natural language queries and texts into unambiguous internal representations, on which matching and retrieval can take place [187]”.

16.2 LEVELS OF NLP

The various important terminologies of NLP are:

16.2.1 Phonology

This refers to the understanding of sound of individual words, and groups of words when spoken together in a sentence of sound. There are three types of rules used here [187]:

- phonetic rules: for sounds within words.
- phonemic rules: for variations of pronunciation when words are spoken together.
- prosodic rules: for fluctuation in stress and intonation across a sentence.

16.2.2 Morphology

This comprises of nature of words where the different parts of the words represent smallest units. A word can be broken down into its constituent morphemes to understand its meaning [187]. This involves understanding suffixes/prefixes attached to the word, whether the word is singular or plural, the roots of the word.

16.2.3 Lexical

This interprets the meaning of the sample word. Words that have only one meaning is replaced with the semantic representation of that word. This requires the word to have a simple or a complex lexicon [187].

16.2.4 Syntactic

This rectifies the grammatical mistakes in a sentence. The output reveals a pattern that has a structural dependency between the words which in turn affects the choice of the parser [187].

16.2.5 Semantic

This determines the meaning of the word by looking at the interactions among word-level meanings in the sentence. An example

of this would be trying to understand if the word honey is used as a noun or an adjective in the given sentence [187].

16.2.6 Pragmatic

This is used to extract information from the text. The main goal is the use of the context over the content of the text for better understanding. This helps in determining how the word is used than what is captured in the plain text of a sentence [189].

16.2.7 Discourse

This focuses on the properties of the text that convey the meaning by making connections between component sentences. This helps to understand how a certain set of sentences convey a part of the story.

“Discourse/text structure recognition determines the functions of sentences in the text, which, in turn, adds to the meaningful representation of the text [187]”.

The current NLP system tends to implement these models that are used for processing the lower levels as they have been well researched and implemented. In addition, the applications do not require them to process at the higher levels[187].

16.3 NATURAL LANGUAGE GENERATION

According to [190], this part of NLP happens in four phases:

- **Identifying the goal:** This defines if we want to generate language in form of written text or spoken words.
- **Evaluating the situation and planning to achieve the goal:** This involves identifying how the goal can be achieved by breaking it into individual tasks.
- **Evaluate the available resources:** This determines whether we want to generate written text or spoken word and also evaluate if this can be achieved with current communication

- devices.
- **Execute the plans as text.**

16.4 APPROACHES TO NLP:

According to [187] some approaches of NLP are as follows:

- **Statistical Approach:** Various mathematical techniques are used here. Observable data is used mainly as evidence. The Hidden Markov model is a widely used statistical approach. This is used in speech recognition, parsing, statistical grammar learning etc.
- **Connectionist:** Models are developed based on the linguistic phenomena. This combines statistical model with other theories.
- **Symbolic Approach:** Performs deep analysis of linguistic phenomena based on the human developed rules and lexicons. Logic or rule based systems and semantic networks are good examples of symbolic approach.

“Symbolic approaches have been used for a few decades in a variety of research areas and applications such as information extraction, text categorization, ambiguity resolution, and lexical acquisition. [187]”

Statistical Approach has been proven to work best on lower levels of NLP. This means analysis on level of parts of words, words and sentences. Symbolic Approach works on both lower and higher levels of processing. Higher level of processing is analysis based on inferences from set of sentences and understanding of whole text in a given document [187].

16.5 RELATED WORK

Significant research has been done on NLP since the 1940s. This has resulted in the development of tools such as Sentiment Analyzer, Parts of Speech Taggers, Emotion Detection, Semantic Role Labelling

etc. This has helped in the creation of real world applications such as Google search, Apple's Siri, Amazon's Alexa etc. Applications such as these have further increased interest in NLP as a useful research topic [187].

Researchers working on tools like Sentiment Analyzer, parts of speech, Emotion detection, Semantic Role Labelling etc have made NLP a good topic for research [190].

- **Sentiment Analyzer:** Analyses the document for positive and negative words. It uses the sentiment lexicon and the sentiment pattern database to analyze the sentiments [190].
- **Parts of speech:** It can efficiently tag and classify words as nouns, adjectives, verbs etc. Parts of speech are assigned to tokenized data. Taggers are already present for the European languages. Research is being done on making parts of speech taggers for other languages like Arabic, Sanskrit [191], Hindi [192] etc.
- **Emotion Detection:** This is just like sentiment analysis, but is used for analysing emotions on social media platforms. It categorizes statements into six groups i.e. anger, disgust, fear, happiness, sadness and surprise based on emotions in the text [190].
- **Sematic Role Labelling:** SRL works by giving a semantic role to a sentence just like assigning roles to words that are arguments of a verb in the sentence. This helps to understand which words indicate action being done, which ones indicate result of action and words that show who is doing the action etc [193].

16.6 APPLICATIONS OF NLP

NLP can be applied into various areas like Machine Translation, Email Spam detection, Information Extraction, Summarization, Question Answering etc. Some more applications of NLP are:

- **Machine Translation:** As the name suggests, Machine

translation is translating a phrase from one language to another with the help of engines like Google Translate. The major challenge in machine translation is in keeping the meaning and the grammatical structure of the translated language intact [190].

- **Text categorization:** This involves splitting the large volume of data into several categories. This usually works by either looking at the email subject, the content or the sender blacklisted by receiver. It is also used to categorize communication so as to forward them to the appropriate department when used in a business setting. [190].
- **Spam Detection:** Various machine learning techniques like Rule Learning, Naïve Bayes, Memory based Learning, Support vector, Decision Trees, Maximum Entropy Model etc. are used to detect the spams. This is very similar to text categorization[190].
- **Chatbot:** A chatbot is a computer trying to mimic human like interaction /communication. We see many applications of these currently on different banking/ecommerce websites. There is lot of on-going research to make this even more capable [190].

16.7 PROBLEMS WITH NLP: LINGUISTIC VARIATION AND AMBIGUITY

There are certain problems in NLP that reduce the efficacy of textual information retrieval. Linguistic variation and ambiguity are some of the problems in NLP. Linguistic variation is an issue when the same words or expressions are used to communicate the idea. Multiple interpretations is one of the main problems with Linguistic variation. Linguistic Variation provokes the omission of certain documents that are relevant and ambiguity implies when a document has duplicate words or words that are not related [194].

Example 1: A notebook was the present that the teacher gave him, when we were present in the class.

Here the word present has different meanings both as an adjective and as a noun. The word present plays different morph-syntactics depending on the situation causing ambiguity problems.

Example 2: He ate food on the car.

Ambiguity is produced here again, as this sentence could mean that he ate the food which was present in the car, or he ate food when he was driving the car.

Example 3: I went to the bank.

Here the word bank could mean a place where we save money and make transactions or the 'bank' of a river.

These examples show that automated process is not easy and that how complex the language is. Statistical processing of NLP and specifically machine learning has improved understanding of learning language by training on text corpus [195].

16.8 NLP IN TEXTUAL INFORMATION RETRIEVAL

When the user gives a query, the following tasks are performed as a part of NLP's textual information retrieval [194]:

- Index is created for the descriptions of a document based on the NLP techniques.
- When a query is given by a user, the system analyses it and transforms it such that it is similar to what is represented in the document.
- The description of each document is compared by the system with the query given by the user, and those documents that have the description close to the user's query are retrieved. There are different methods used to perform job of matching a query and document. Boolean method does this by trying to do an exact match. Vector space model converts the query and documents into vectors that can be stored as matrices. It

then finds the similarity by calculating the cosine angle between the query and the document vector. Another method that is used to perform this is language model. Language model tries to find the probability of the document generating a query. This model depends on training the algorithm on a large corpus of text data/documents.

- The results are shown in the order of the similarity. Google uses PageRank to rank the documents in terms of similarity to query from the user.

16.9 STATISTICAL PROCESSING OF NATURAL LANGUAGE

"This is a very simple focus based on the bag of words. In this approach, all the words in a document are treated as its index terms. Moreover, each term is assigned a weight in function of its importance, usually determined by its appearance frequency within the document. In this way, the word's order, structure, meaning, etc., are not taken into consideration [194]".

The document processing model involves document pre-processing and Parameterization.

- **Document pre-processing:** Prepares the documents by removing those elements that are superfluous. There are three basic phases here:
 - Removing headers, tags etc. from the document which are not for indexing.
 - Tokenization splits text into sentences and sentences into words.
 - Standardizing the text by checking for capitalized or non-capitalized letters, numerals, dates etc. Making all words lowercase helps treat words such as 'Hi' and 'HI' same.
 - Stemming the terms by reducing the words to the roots. This operation removes suffixes, prefixes etc.
- **Parameterization:** Assigns weights to the relevant terms

present in the document.

One of the most used methods to estimate the importance of a term is the TFIDF system (Term Frequency, Inverse Document Frequency).

"It is designed to calculate the importance of a term relative to its appearance frequency in a document as a function of the total appearance frequency for all of the corpus' documents i.e. the fact that a term appears often in one document is indicative that that term is representative of the content only when that term does not appear frequently in all documents. If it appeared frequently in all documents, it would not have any discriminatory value [194]".

Two commonly used techniques in statistical processing are:

- **Detecting N-Grams:** This involves identifying compound words, proper nouns etc., to be able to process them as single words. This is done by determining the probability of the compound words like European union etc. This allows to maintain the sequence of words which is different as compared to the just bag of words which does not maintain order of words.
- **Stopword List:** A list of empty words, with very little semantic values. Deleting these terms avoids duplications and noise [194].

Statistical evaluation in NLP systems is used to evaluate the efficiency, accuracy and robustness. It can be done using the below methods that do it in different ways ???:

- **Descriptive Statistics:** This method calculates Word error rate, Accuracy rate, Recall and Precision
- **Estimation:** This method calculates the confidence interval for true accuracy rate with certain probability.
- **Hypothesis Testing:** When different NLP systems are applied

to same set of data, we want to compare their performance and suggest if one method is better than the other. This method allows us to do it by comparing certain performance parameters between two methods and performing hypothesis testing to tell if there is real statistical significance between the results or not.

16.10 LINGUISTIC PROCESSING OF NATURAL LANGUAGE

In the linguistic process, the words determine how they are related and used together in making grammatical units, sentences etc. Parsers are created and applied to demonstrate the text's syntax structure. The method used to create the parser vary. To determine the semantic structure of the words, certain tools are used. The most often used tool is the lexicographic database WordNet [194].

"This is an annotated semantic lexicon in different languages made up of synonym groups called synsets which provide short definitions along with the different semantic relationships between synonym groups [194].

16.11 NLP FOR BIG DATA

Big data is the most text based content which is constantly growing and is quite unstructured. Every industry generates a large volume of text information, documents, notes, emails, patents, patient information etc. As most of these are text based data, NLP presents an opportunity to take advantage of this situation to reveal patterns and trends [196].

- **Interactions:** Interactive applications are becoming more common these days like Microsoft's Cortana, smart phone assistants, language translation programs etc. These applications use NLP.
- **Business Intelligence:** NLP for big data enables the user to retrieve the documents that they are looking by not limiting

their search to exact keywords. NLP enables them to search using their own words and tries to retrieve documents with that search.

- **Market Research:** With the growth of internet, social network is full of rich, noisy information. The brands and organizations can determine what is said about their products and services by using NLP for their market research analysis.

There are lot of NLP libraries written to process big data. Some of which are:

- **CoreNLP:** It was originally written in java that can also support multiple languages like python. It is well known for its speed and its precise results [197].
- **TextBlob:** Addition of components like sentiment analyzer becomes very easy with Textblob [197].
- **Gensim:** Used best for topic modeling and comparing the document similarity [197].
- **Spacy:** It is a new library which has a very high performance [197].
- **NLTK:** Most commonly used NLP library.

“Natural language tool kit’s (NLTK) modular structure helps comprehend the dependencies between components and get the firsthand experience with composing appropriate models for solving certain tasks [197].”

16.12 NLP IN YELP DATA REVIEW

Yelp is a social networking site that combines business listing with social elements. It helps in finding local businesses such as restaurants where customers can leave feedback on their experience. This feedback helps the other customers of what they might expect from the place. Reviews or feedbacks are in the form of starts. Higher the starts, better the place is. The reviews also help the business to improve their standards in case there is a lower review. We can use

NLP to analyze the text reviews to interpret restaurant reviews on Yelp through a sentiment analysis model.

The first step in NLP depends on the application. Voice based systems like Google Assistant or Alexa translates words into text using Hidden Markov Models. The language and context is then understood through a series of coded grammar rules that rely on algorithms that incorporate statistical machine learning. Another important step is Semantic analysis which helps interpret human sentences logically [195].

Here we will try to predict whether a user liked a local business or not based on their review on Yelp. A simple text classifier will be built based on Python's Pandas, NLTK and Scikit-learn libraries. According to [198], the plan would be to start with a dataset containing 5000 reviews with the following info:

- ID of the restaurant under review
- ID of the posted review
- Date the review was posted
- The star rating provided.
- The text for the review.

NLTK library would be used to process the text and get basic information and insight on the data. The next step would be to visualize the data by utilizing histogram grids for every star rating. The goal would be to identify which feature of the review is useful in finding correlations in the data frame. Once we derive some useful correlations, they could be visualized. According to [199] Challenges faced in NLP text processing are:

- Scalability and portability.
- Certain techniques are too expensive.
- Not very reliable as of now.
- Speech/text processing.

16.13 BUILDING A NLP PIPELINE

It would be really helpful if a computer could understand what the humans are trying to say. NLP helps the computer to read and understand all the data. By applying NLP techniques, we will be able to save a lot of time to the projects. But parsing the English language with a computer has its own complications. Hence we will breakdown the process of understanding english, into small chunks and see how it performs in understanding and giving a correct output [198].

Let's take a paragraph: Delhi is the capital of India and one of the most populous city in Asia. This has been a great settlement for several kings including the Mughals. The original name was Indraprastha.

This paragraph contains several important and useful information. It would be great if a computer could read and understand that Delhi is a city in India, it was ruled by Mughals etc. But to get there we have to train the computer on how to read the sentence [198].

For a computer to understand the text and extract data we need to do some of the following steps [198]:

16.13.1 Sentence Segmentation

First step is to break the text in the paragraph into separate sentences like:

- Delhi is the capital of India and one of the most populous city in Asia.
- This has been a great settlement for several kings including the Mughals.
- The original name was Indraprastha.

By breaking the text in the paragraph into small sentences, it is easy for the computer to read and understand them. We can use NLP pipeline methods to read the sentences and determine what it means.

16.13.2 Word Tokenization

Once we have broken the paragraph into sentences, we can process the individual sentences. We can now break these sentences into separate words or tokens. This is called “Tokenization”. Every word including the punctuation is split apart.

Eg: Delhi, is, the, capital, of, India, and, one, of, the, most, populous, city, in, Asia, .

16.13.3 Predicting parts of speech

Each token is taken individually and the part of speech for that token is determined. Finding out if the word is a noun, verb etc. helps to determine what the sentence is about. Each word is then fed into a part of speech classification model which was trained already by feeding in millions of English sentences to determine the part of speech.

Eg: Delhi is a noun and capital is a noun. So we can determine that the sentence is probably about Delhi.

16.13.4 Text Lemmatization

There may be cases where a same word may appear in different forms. Text Lemmatization helps to determine the base form of each word, so that it will be easy to figure out that the words are the same if they were in different base forms.

“Lemmatization is typically done by having a look-up table of the lemma forms of words based on their part of speech and possibly having some custom rules to handle words that you’ve never seen before [198]”.

16.13.5 Identifying Stop Words

There are lot of filler words like a, the etc. These words are called stop words. The stop words are considered a noise and are usually removed before performing any statistical analysis. Here we

determine how the words are related to each other.

Eg: Delhi, capital, India, one, most, populous, city, Asia, .

16.13.6 Dependency Parsing

“The goal is to build a tree that assigns a single parent word to each word in the sentence. The root of the tree will be the main verb in the sentence . In addition to determining the parent word, the relation that exists between the word is also found out [198].”

16.13.7 Finding noun phrases

The words that represent single idea is grouped together instead of considering every word as a single entity. The information from the dependency parse tree is taken to group the related words together. By combining the non-phrases from the sentence we get:

Eg: Delhi the capital most populous city ...

16.13.8 Named Entity Recognition

The aim of Named Entity Recognition is to detect and label the nouns with real world concepts. A Named Entity can Recognize people's name, location, products, date and time, money etc.

Eg: Delhi, India and Indraprastha represent places on a map. With Named Entity Recognition we will be able to detect that on a map.

16.14 INSTALLATION

1. Install the latest version of Python (avoid the 64-bit versions)
2. Install Numpy (optional)
3. Install NLTK: pip install nltk
4. Install the NLTK packages:

```
import nltk
```

```
nltk.download()
```

16.15 EXAMPLE

Tokenize:

```
from nltk.tokenize import sent_tokenize, word_tokenize  
line = "A quick brown fox jumps over the lazy dog"  
print(word_tokenize(line))
```

```
output: ["A", "quick", "brown", "fox", "jumps", "over", "the", "lazy", "dog"].
```

16.16 CONCLUSION

Based on the user's query and information need, NLP system represents the true meaning of what is expected. The content of the document will be searched in order to retrieve the relevant document based on the search query [187]. Alan Turing famously proposed a test to check intelligence of machines. The test measures if a machine is able to exhibit intelligent behavior/thinking like humans when it is asked same question by a human and its response is compared to the response of other human. This led to field of Artificial Intelligence (AI). NLP is important part of AI as it helps with communication between machine and human.

16.17 TEAM MEMBERS AND WORK BREAKDOWN

- Sahithya Sridhar (fa18-523-67): Introduction, Natural language text processing, Natural Language generation
- Prajakta Patil (fa18-523-65): Related Work, Applications of NLP, NLP in Yelp data review

Ritu Susan Sanjay
rssanjay@iu.edu
Indiana University, Bloomington
hid : fa18-523-66
github: [blue user icon](#)

Keywords: SAS Viya, Data Mining, Cloud Analytic Services, Analytics

17.1 INTRODUCTION

SAS Viya is an in-memory analytics engine, with cloud enables features, providing users with accurate, quick and most importantly reliable insights. The software provides features like authorship, full-versioning, change management and a lineage viewer along with centralized administration enabling tracking of users, servers and job content. [200]

The demand for data scientists is at an all time high. At the simplest level, it is the art of making sense from the never-ending sea of data - simple as that. To better understand this, we can think of developing a data mining model analogous to making a dish. You first scrape together the ingredients; this is our raw data. To improve the taste, you add a pinch of salt and other seasonings or spices; this is the creation of new predictor variables. Finally, you mix it all up together, and have a taste. If it doesn't taste right, you might want to try a couple of other methods until you've perfected the recipe; analogy here referring to iteratively building descriptive (unsupervised) or predictive (supervised) models. And just like in the show Master Chef you often end up competing with other data scientists in developing the best possible recipe, or in this case the best model. [201]

In the above example, just as how the chef needs his sharp knives, a

data scientist needs the right tool. The top rated software being python, spark, R, Matlab to name a few. SAS Viya is the latest enhancement of the SAS platform. In the words of its developers:

"SAS Viya addresses the complex analytical challenges of today, and effortlessly scales to meet your future needs, with cloud-enabled, > elastic in-memory processing, in a high availability, multi-user environment. It is designed to address the new, and increasingly diverse, needs of organizations with methods, access, and deployment that scale to meet burgeoning analytics use cases" [201].

SAS 9 brought with it a user-friendly server-client web service model, all processes governed by a resilient metadata server. It was one of the more popular platform for analytics, albeit a bit expensive, and hence was usually preferred by organizations willing to invest in analytic platforms rather than self-financed analysts. However, the advent of cloud computing blew the whole tech industry out of proportion.

"SAS Viya brings a more resilient, elastic, unified, and accessible architecture, which leverages cloud-friendly microservices and a next generation analytics run-time engine" [202].

SAS Viya enables users to explore data deeper, using the latest innovations in in-memory analytics. The methods available to users is classified in two: Data Wrangling methods and Modeling methods. Data wrangling methods include binning, transformations, SQL, clustering etc., while modelling techniques include everything from regression to text mining to neural networks. One of the prime features of using SAS for these methods is its innate ability to run all above-mentioned methods in-memory and of course take advantage of the parallel processing infrastructure [203].

SAS Viya emphasises a unified experience for data scientists and analysts alike. The new cloud analytical platform, allows

programmers to execute using open-source languages like Python, Java, and Lua. Furthermore, the platform also allows these codes to be written and executed on Jupyter notebooks [204]. Figure 19 shows a detailed view of the SAS Viya 3.4 architecture.

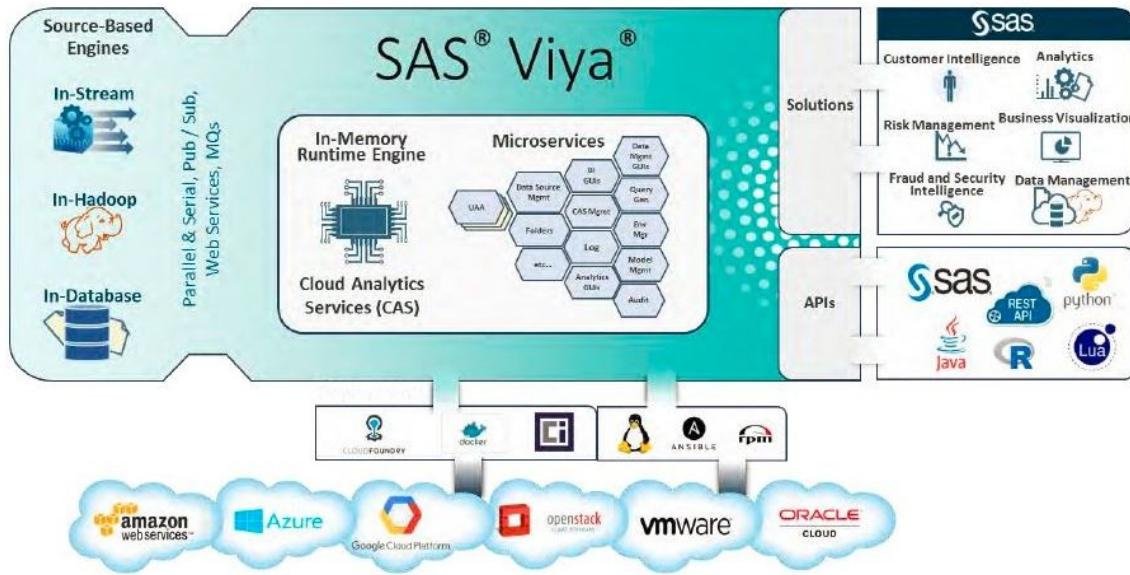


Figure 19: SAS Platform [205]

17.2 SAS VIYA COMPONENTS

SAS Viya 3.4 gives data scientists access to the following components:

17.2.1 SAS Cloud Analytic Services

"CAS goals are to provide an analytics service with a public API accessible by many clients supported by SAS or open-source clients using plug-in modules from SAS. [206]"

SAS CAS is a platform for distributed and high-performance computing with a cloud-based RE. The many features of CAS include data sharing between sessions, security, and fault tolerance (i.e. allowing a node to fail without data loss). CAS was designed to operate fully on-cloud, either as a single host or on a cluster (private or public). CAS uses sessions to track users and offers a full Security

interface to protect data at the file level, as well as the column level. The sessions provide isolation for the user, which protects the integrity of the server. The purpose of connecting to CAS is to execute server requests. A user must create a session to submit a request. The user can connect to the server either through a HTTP-based REST interface or through a ProtoBUF-based binary interface. The user must be authenticated by CAS in order to create a session [207].

The most important aspect of CAS is that all data is stored in the form of tables. These tables may be streamed from a database into the server, ESP stream or loaded from disk. Data (including metadata) in CAS is all stored and accessed through the CASLIB. The caslib is basically a container that usually has one or more instances of the CAS tables [208].

Like all cloud services, SAS Viya too concentrates on fault tolerance. Node failure is inevitable when dealing with multiple number of nodes are implemented in a system. Data is replicated across the cluster, in order to retrieve data in case one of the worker nodes fail. The new system has been dubbed the GCCOMM. The subsystem can detect failure in nodes; the controllers and workers can reconfigure the system, thus restarting the action and allowing the remaining worker nodes to access lost data from the redundant blocks [208].

17.2.2 SAS Studio

In the simplest of words, SAS Studio is an editor designed for both expert and novice programmers to write and execute SAS code in an assisted environment. Users are provided with a single interface to access all data files, programs and libraries (user-defined and in-built). The SAS studio is extremely convenient: there is no local installation involved i.e. once the software is installed you just provide the users with the url to access the software. This centralizes and simplifies regular maintenance. Other features range from (data) table analyzer, sql engine, code snippet library prompts from frequently executed codes, report generation and export in multiple formats including pdf and xml. One important feature to be noted is

that the studio interface is consistent, regardless of where and how the software runs; as the IT infrastructure is modified, the SAS Studio environment is the same [209].

17.2.3 SAS Visual Analytics

“SAS Visual Analytics provides a complete platform for analytics visualization and interactive self-service BI and reporting capabilities are combined with out-of-the-box advanced analytics to enable users to discover insights from any size and type of data > including text” [210].

Popular analytic tools include - goal seeking and scenario analysis, automated forecasting, decision trees, text analytics and network diagrams. [210]

17.2.4 SAS Visual Statistics

“It provides a drag-and-drop web browser interface that empowers multiple users to explore massive data, and then interactively and iteratively create descriptive and predictive models” [211].

Statisticians are most often faced with the challenge of choosing the right model that fits the data. As the amount of data to be analyzed keeps increasing, this can prove to be a very time-consuming task. SAS Visual Statistics allows users to generate reports that detail model comparison summaries, misclassification tables and ROC charts. The software includes an interactive slider that manipulates thresholds preset by the user [211].

17.2.5 SAS Visual Data Mining and Machine Learning

“In addition to innovative machine learning and deep learning techniques for analyzing structured and unstructured data, it integrates > all other tasks in your analytical processes. From data preparation and

exploration to model development and deployment, multiple personas work in the same, integrated environment” [212].

The software allows multiple users to concurrently analyze data (structured or unstructured) with the Model Studio. Every project can be subdivided using visual pipelines, displayed in a logical sequence. The SAS Visual Data Mining and Machine Learning boasts of a significantly reduced runtime, owing to the multicore architecture. Popular models included in the software are: gradient boosting, SVMs, neural networks, Gaussian models, bayesian networks among others [212].

17.2.6 SAS Econometrics

“SAS Econometrics supports a range of econometric model types with a single framework. It’s fully integrated with all of the contributing analytics that coincide with econometrics, and with data preparation, exploration, presentation and reporting capabilities in SAS that are essential to successful econometric analysis” [213].

The in-memory analytics feature of SAS Viya ensures that iterative and repetitive tasks can be run quickly without re-loading data. The software also provides a wide-range of tools for modelling business scenarios. Simulations and forecasting techniques can be easily implemented as the software makes use of the SAS Viya Engine, ensuring high-availability and the ability to code using open-source languages [213].

17.2.7 SAS Visual Forecasting

“SAS Visual Forecasting provides a resilient, distributed and optimized generic time series analysis scripting environment for cloud computing” [214].

"Users can range from analysts responsible for the creation of the forecasts to the managers and directors responsible for overseeing > the forecasting and planning processes" [214].

The software itself recommends the most suitable model and additionally models are selected based not on how well they fit past data but on how well they can be used to predict the future. The interface allows for training and modelling data mining algorithms including neural networks. For example, the Multistage Forecasting node (for regression and time series included) creates a forecast combining signals obtained from different models [214].

17.2.8 SAS Visual Text Analytics

SAS Visual Text Analytics seeks to bring together concepts of NLP [215] along with machine learning and data mining techniques to derive insights from unstructured data. The accuracy of an analytical model may be increased by utilizing a combination of machine learning techniques and rule-based approaches. Users also have the ability to build their own custom search engine, provided by microservice architecture and built in APIs. The text-analytics pipeline makes available five types of nodes : Text Parsing, Concepts, Categories, Sentiment and Topics. The software also features automatic extraction of features identified by topics generated by the machine. [216]

17.2.9 SAS Optimization

SAS Optimization was designed for industry experts who utilize operations research and optimization techniques to create decision-making models to solve problems.

"SAS Optimization provides a powerful, intuitive algebraic optimization modeling language and an array of algorithms. This involves a range of models, including linear, mixed-integer linear, nonlinear,

quadratic, and network optimization, as well as solve constraint satisfaction problems." [217]

Models are executed efficiently since the software runs on the Viya Engine. Notable models on the SAS Optimization are : Local Search Optimization, Constraint Programming, Multistart Algorithm and the Decomposition Algorithm.

17.3 DEPLOYMENT

"SAS Viya has undergone rigorous performance testing with various hardware combinations. In addition to being tested on high- performing Intel Xeon E3-E7 series microprocessors, SAS Viya has also been tested with newer Intel chips, such as Intel Xeon Scalable > Processors. SAS Viya also supports 64-bit AMD chipsets (thirty-two-bit chipsets are not supported)" [218].

It is necessary to note that a separate independent host is needed if SAS 9.4 exists on the system (co-installation is not possible). Also, if the existing SAS software on the system is SAS 9.3, note that many of the features on SAS 9.3 are not supported if the Java version has been updated to Java 8 or plus. The hardware requirements for a programming only environment also differs from a full deployment.

17.3.1 System Requirements

It is first necessary to understand the difference between the two deployment types: full deployment and programming-only deployment. The full deployment includes all the features SAS Viya has to offer and is usually the default mode. However, it is also possible to deploy only a subset of the features; the programming-only deployment excludes the SAS Drive and a number of the graphical features [219]. The hardware requirements for a programming only and a full deployment differ. When determining the specifications of the host, three components are to be kept in mind: CAS server, programming runtime and the service layer [218].

1. CAS Server : Before installation a key point to be taken to account is the required amount of RAM. This may vary depending on activity level of users in the SAS software environment and the data (load) to be processed. However, less than 1 gigabyte of RAM is mandatory for CAS SErver startup [218].
2. Programming Runtime : The CAS license procured determines the number of CPU cores required for your environment. On that note, if the CAS license specifies N cores, then the user is entitled to the same number of cores on their setup. SAS however, specifies a minimum of two cores and at least four cores for optimal performance. The software also necessitates a minimum of 4 gigabyte RAM for the programming environment [218].
3. Service Layer : The service layer primarily are the components required for a full deployment ideally. These also include services that support other SAS softwares. Additionally, it includes supporting services for the SAS VIYa analytics software, namely the Core Services host [218].

17.3.2 Installation

The first step in the installation involves setting up the accounts. The user account for both the CAS as well as the postgreSQL requires the SAS credentials to be specified. Instructions on how to set this up is extensively detailed in the setup manual (the link to the latest version of the same is provided at the end of this section). If the user seeks to set up the full deployment, then changes may be made to modify the postgreSQL settings to specify personal ports and directories. Further to this, the CAS Server Monitor port may also be changed along with modifications to the kerberos [220] settings. The final step after the configuration files have been modified is to simply run the SAS Viya setup batch file through the command prompt folling the regular intructions [221].

After completing the installation of SAS Viya, it is necessary to configure the connection to your identity provider before your users can access SAS Environment Manager and SAS Visual Analytics. The final step then remains to create a backup configuration [222]. Note that this is also different depending on the type of deployment (i.e. programming-only or full deployment).

The latest version of the deployment guide may be accessed at : SAS Viya Deployment Guide [\[22\]](#)

17.4 SAMPLE ILLUSTRATION

This section consists of an example detailing how easy it is to create a model in SAS Viya. For the example, we consider a dataset consisting of the political opinion poll from the Annual National Election Survey, conducted once every four years. Thermometer measures are used in many surveys to rank opinions. These variables range from 0 (very cold, or unfavorable feeling) to 100 (very warm, favorable feeling). This example examines the multivariate relationship of the preference for the Democratic variable against current economy condition, religious attendance and how better off the respondent is compared to the previous year.

Step 1: Start up the SAS Viya Service

SAS Viya starts up with a user friendly interface. The left panel details all about the data and the models that may be created and executed using the engine. While the right panel houses options for modifying or altering data variables and adjusting model parameters. Figure [20](#) shows a screenshot of the start menu.

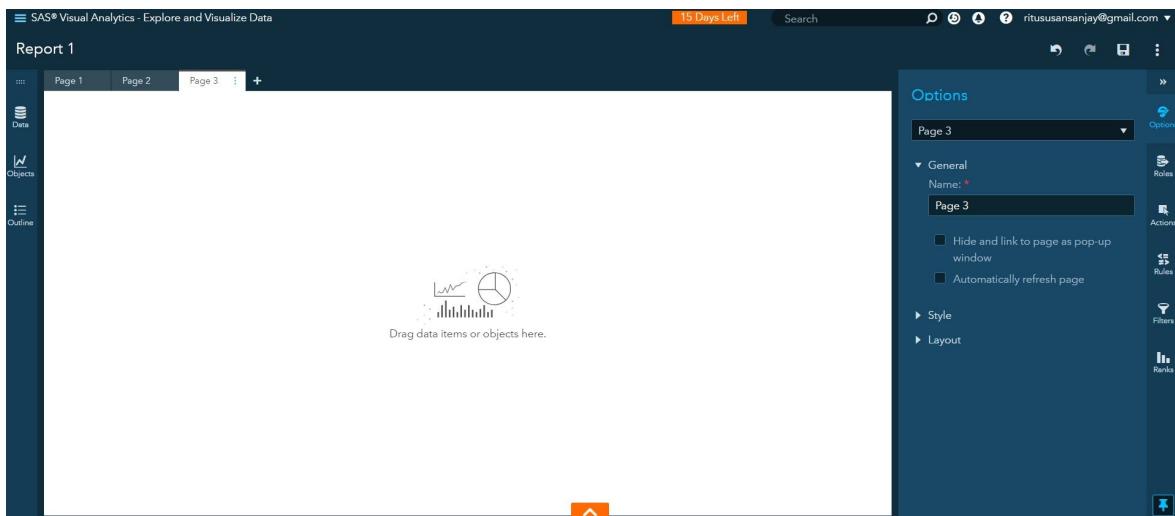


Figure 20: SAS Viya Start Page [223]

Step 2: Import data

Figure 21 shows how datasets may be imported onto the SAS library.

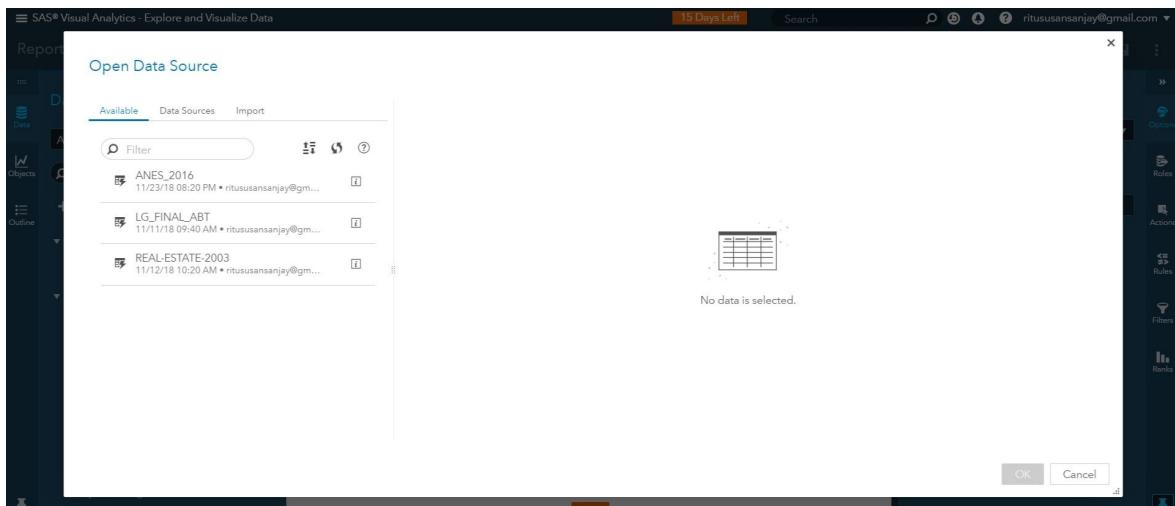


Figure 21: SAS Viya Import Data [224]

Step 3: Add model object i.e. linear regression object

To create a linear regression model, all you have to do is drag and drop the desired object into the analysis screen. Figure 22 demonstrates how a model may be added.

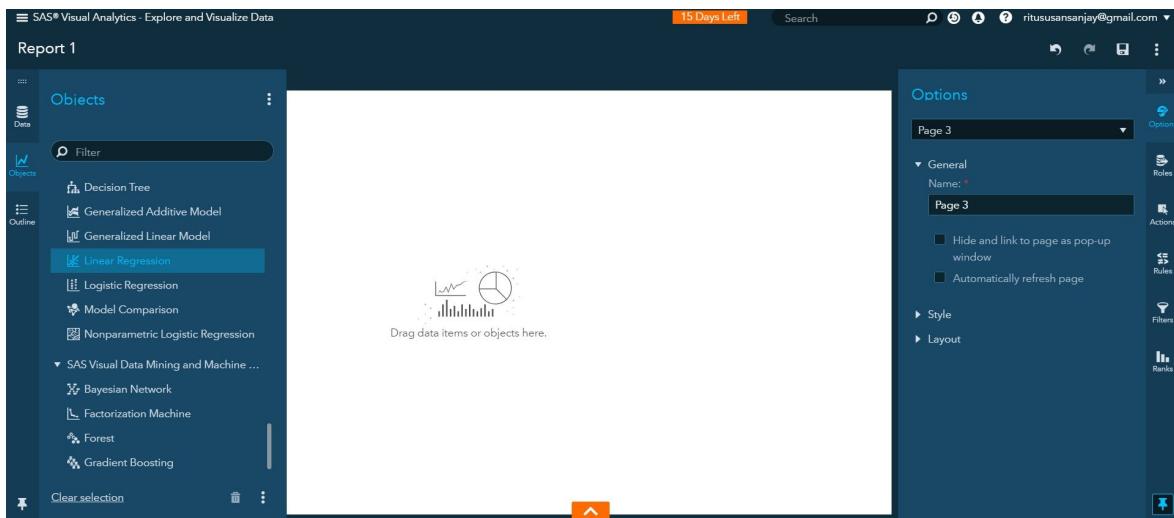


Figure 22: SAS Viya Add Data Object [225]

Step 4: Specify roles on the right options panel

Figure [23](#) demonstrates how roles and rules may be applied to the model.

The screenshot shows the SAS Viya Data Roles interface. At the top, there are icons for search, refresh, help, and user information, followed by the email address "ritususansanjay@gmail.com". Below the header are additional icons for refresh, undo, redo, and a three-dot menu.

The main area is titled "Data Roles" and displays a list of variables under the project "Linear Regression - thermometer_clin...".

- Response:** thermometer_clinton
- Continuous effects:**
 - economy_good_bad
 - better_worse_1year
 - religious_services
- Add:** + Add
- Classification effects:**
 - + Add
- Interaction effects:**
 - + Add
- Partition ID:**
 - + Add

On the right side, there is a vertical sidebar with icons and labels for different roles:

- Options
- Roles
- Actions
- Rules
- Filters
- Ranks

A blue button with a white icon is located at the bottom right of the sidebar.

Figure 23: SAS Viya Add Variable Roles [226]

Step 5: Modify parameters if necessary to improve model

The model can then be interpreted using measures like the adjusted r-square, that predicts approximately 21.5% of the variation in the variable. Looking at other measures like the F-statistic (very high) and p-value (very low) implies that the model is statistically significant. Figure 24 displays the final result.

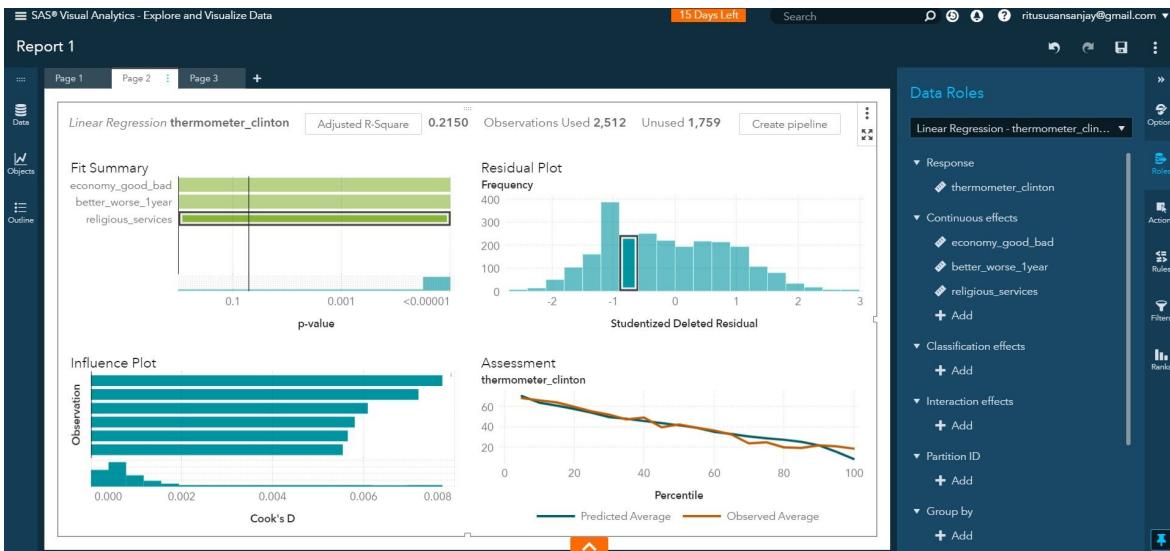


Figure 24: Linear Regression Results [227]

17.5 CONCLUSION

Big data speaks volumes when applied to find solutions to challenging 'everyday' problems. We capture and store far more data than is actually used; the true potentials of big data are just being realized [228]. Today data is the raw material generated and consumed by businesses, governments and scientific researchers. Given the right tools and the computing power, data can open up a whole new world of insights. The new cloud-based analytic software offered by SAS helps create well-defined models and generate results, giving way to new ideas. However, SAS Viya is just one of the many tools among thousands offered today and choosing the right tool depends on the users' goals.

Selahattin Akkas
sakkas@iu.edu
Indiana University
hid: fa18-523-68
github: [blue](#)

- this is a draft, review has not been started due to this
 - second review. No further review possibly before we grade.
-

Keywords: Distributed TensorFlow, TensorFlow

18.1 ABSTRACT

It is non-practical to do computation on a single machine for Big Data applications. Likewise, it is also non-practical to train machine learning algorithms using large datasets on a single machine. One of the widely used Deep Learning framework TensorFlow supports distributed learning. In this paper, Distributed TensorFlow's architecture will be explained.

18.2 INTRODUCTION

18.3 PARAMETER SERVER

18.4 TENSORFLOW CLUSTER

18.5 PARAMETER SERVER

18.6 SHARED VARIABLES

18.7 SYNCHRONUS DATA PARALLELISM

18.8 ASYNCHRONUS DATA PARALLELISM

18.9 IN-GRAPH REPLICATION

18.10 BETWEEN-GRAPH REPLICATION

18.11 ASYNCHRONUS TRAINING

18.12 SYNCHRONUS TRAINING

19 BIG DATA APPLICATION IN RECOMMENDER SYSTEMS

FA18-523-70

Sushmita Dash
sushdash@iu.edu
Indiana University
hid: fa18-523-70
github: [blue icon](#)

Keywords: recommender system, TV genome, KNN classification

19.1 INTRODUCTION

In today's world where people have a very busy lifestyle. They often do not have the time and patience to go through a very vast selection of options available to them. This is applicable in many aspects such as watching TV shows or getting a product online. Here is when our recommendation system comes into play. It plays a critical role in engaging the customers in the online service platforms. Earlier, in order to find a movie or a product that the user likes, they had to tediously browse through media catalogs or product catalogs. Amidst information overload, shorter attention span, and competing content, the only way to grab users' attention is personalization. This is where big data comes into play

There are many daily life examples where we can see the use of the recommender systems. Few examples are as follows:

1. Netflix uses this to show a personalized recommendation of the shows you may like based on the TV programs you have seen before
2. Various other tech giants also use this technology. For example, we often see friend suggestions in Facebook. It will be based on various criteria like how many mutual friends do

- we have. If we have been in any photo together or in the same school etc
3. We can see similar experience when we are browsing through a product catalog in Amazon or any other online shopping website. It will show us similar products based on our taste.

19.2 WHAT IS A RECOMMENDER SYSTEM?

A recommendation engine filters the data using different algorithms and recommends the most relevant items to users. It first captures the past behavior of a customer and based on that, recommends products which the users might be likely to buy or watch [229]. Below is a very simple illustration of how recommender systems work in the context of an e-commerce site.

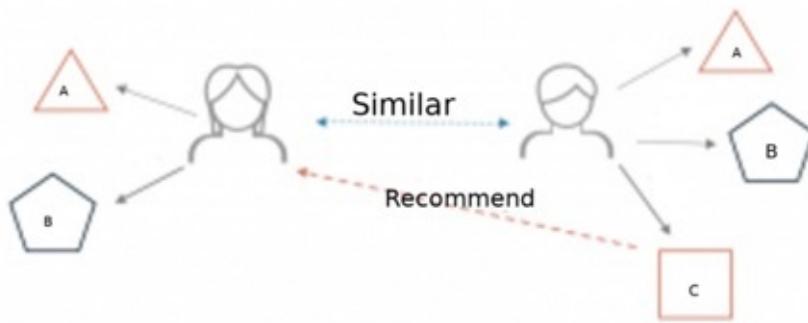


Figure 25: Simple Recommender System[229]

Two users buy the same items A and B from an ecommerce store. When this happens the similarity index of these two users is computed. Depending on the score the system can recommend item C to the other user because it detects that those two users are similar in terms of the items they purchase[230].

19.3 HOW DOES THE RECOMMENDER SYSTEM WORK?

A typical recommendation engine processes data through the following four phases namely collection, storing, analyzing and

filtering[231].

The phases are described below:



Figure 26: Phases of recommendation engine [229]

19.3.1 Collection of Data

The first step in creating a recommendation engine is gathering data. Data can be either explicit or implicit data. Explicit data would consist of data inputted by users such as ratings and comments on products. And implicit data would be the order history/return history, Cart events, Pageviews, Click thru and search log. This data set will be created for every user visiting the site.

Behavior data is easy to collect because you can keep a log of user activities on your site. Collecting this data is also straightforward because it doesn't need any extra action from the user; they're already using the application. The downside of this approach is that it's harder to analyze the data. For example, filtering the useful logs from the less useful ones can be difficult.

Since each user is bound to have different likes or dislikes about a product, their data sets will be distinct. Over time as you 'feed' the engine more data, it gets smarter and smarter with its recommendations so that your email subscribers and customers are more likely to engage, click and buy. Just like how the Amazon's recommendation engine works with the 'Frequently bought together' and 'Recommended for you' tab.

19.3.2 Storing the data

The more training data is available for the algorithms, better the recommendations will be. This means that any recommendations project can quickly turn into a big data project. The storage of the

data depends on whether we are trying to capture user's input or behavior and on factors such as ease of implementation, size of the data, integration with other systems and portability. We can use noSQL database or a standard SQL database, etc. for data storage. When saving user ratings or comments, a scalable and managed database minimizes the number of tasks required and helps to focus on the recommendation. Cloud SQL fulfills both of these needs and also makes it easy to load the data directly from Spark.

19.3.3 Analyzing the data

In order to find items with similar user engagement data, data is filtered using different analysis methods. Some of the ways in which we can analyze the data are:

- Real-time systems can process data as it's created. This type of system usually involves tools that can process and analyze streams of events. A real-time system would be required to give in-the-moment recommendations.
- Batch analysis demands you to process the data periodically. This approach implies that enough data needs to be created in order to make the analysis relevant, such as daily sales volume. A batch system might work fine to send an e-mail at a later date.
- Near-real-time analysis lets you gather data quickly so you can refresh the analytics every few minutes or seconds. A near-real-time system works best for providing recommendations during the same browsing session.

Algorithm:

for each item in product catalog, I_1

 for each customer C who purchased I_1

 for each item I_2 purchased by customer C

 Record that a customer purchased I_1 and I_2

for each item I_2

compute the similarity between I_1 and I_2

Algorithm Complexity: - Worst Case: $O(N^2M)$ - In practice: $O(NM)$, cause customers have fewer purchases

19.3.4 Filtering the data

After collecting and storing the data, we have to filter it so as to extract the relevant information required to make the final recommendations. We need to filter the data to get the relevant data necessary to provide recommendations to the user. We have to choose an algorithm that would better suit the recommendation engine. For example

- **Content-based:** A popular, recommended product has similar characteristics to what a user views or likes.
- **Cluster:** Recommended products go well together, no matter what other users have done.
- **Collaborative:** Other users, who like the same products as another user views or likes, will also like a recommended product. Collaborative filtering enables you to make product attributes theoretical and make predictions based on user tastes. The output of this filtering is based on the assumption that two users who liked the same products in the past will probably like the same ones now or in the future.

Data about ratings or interactions can be represented as a set of matrices, with products and users as dimensions. Assume that the following two matrices are similar, but then we deduct the second from the first by replacing existing ratings with the number one and missing ratings by the number zero. The resulting matrix is a truth table where a number one represents an interaction by users with a product.

Rating matrix		Interaction matrix									
Users	Products					Users	Products				
	1	2	3	4	5		1	0	1	1	0
1	1	2	2	4	4	2	0	1	0	1	0
1	1	3	5	5	5	3	1	0	1	1	1

Figure 27: Rating Matrix [229]

We use K-Nearest algorithm, Jaccard's coefficient, Dijkstra's algorithm, cosine similarity to better relate the data sets of people for recommending based on the rating or product.

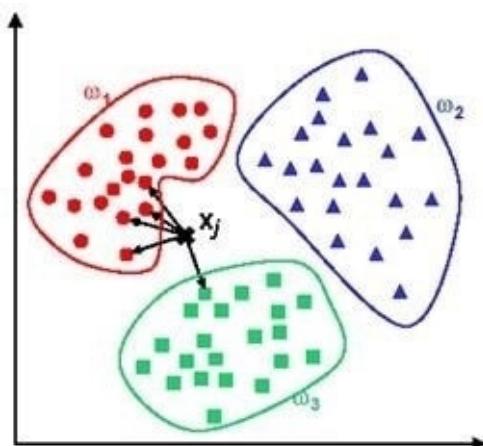


Figure 28: K Nearest algorithm [229]

The above graph shows how a k-nearest algorithm's cluster filtering works. Then finally, the result obtained after filtering and using the algorithm, recommendations are given to the user based on the timeliness of the type of recommendation. Whether real time recommendation or sending an email later after some time.

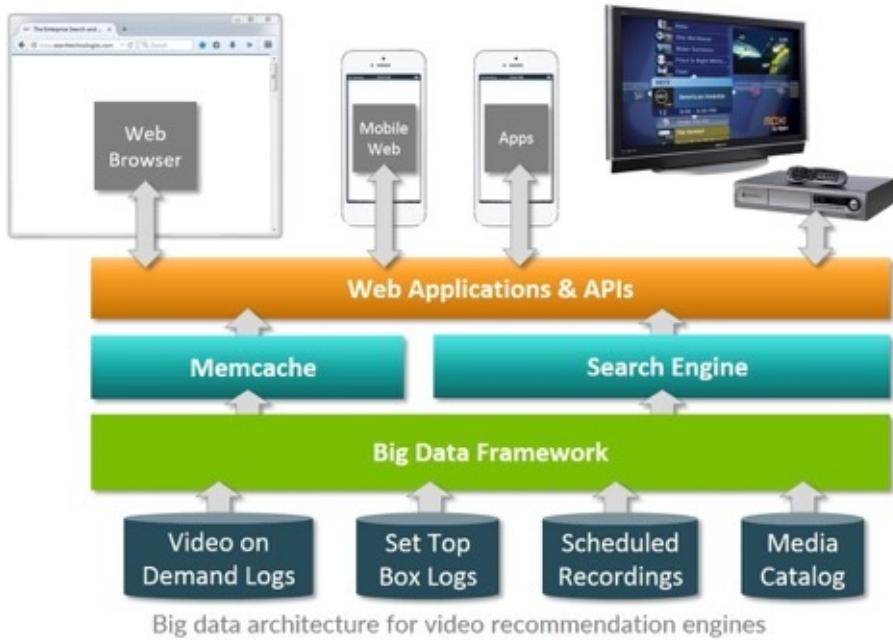


Figure 29: Big Data Architecture Of Recommendation System [230]

19.4 TYPES OF RECOMMENDER SYSTEMS

Recommender systems are among the most popular applications of data science today[232]. They are used to predict the “rating” or “preference” that a user would give to an item. Almost every major tech company has applied them in some form or the other: Amazon uses it to suggest products to customers, YouTube uses it to decide which video to play next on auto play, and Facebook uses it to recommend pages to like and people to follow. What’s more, for some companies -think Netflix and Spotify-, the business model and its success revolves around the potency of their recommendations. In fact, Netflix even offered a million dollars in 2009 to anyone who could improve its system by 10%.

1. **Simple recommenders:** offer generalized recommendations to every user, based on movie popularity and/or genre. The basic idea behind this system is that movies that are more popular and critically acclaimed will have a higher probability of being liked by the average audience. IMDB Top 250 is an

example of this system.

2. **Content-based recommenders:** suggest similar items based on a particular item. This system uses item metadata, such as genre, director, description, actors, etc. for movies, to make these recommendations. The general idea behind these recommender systems is that if a person liked a particular item, he or she will also like an item that is similar to it.
3. **Collaborative filtering engines:** these systems try to predict the rating or preference that a user would give an item-based on past ratings and preferences of other users. Collaborative filters do not require item metadata like its content-based counterparts. Enables users to explore diverse contents, dissimilar to that viewed in the past.

19.5 ALGORITHMS

Content based methods are based on similarity of item attributes and collaborative methods calculate similarity from interactions[233]. Here we discuss few collaborative methods:

19.5.1 K-Nearest Neighbors

- Computes similarity of users
- Find k most similar users to user 'a'
- Recommends movies not seen by user 'a'

The simplest algorithm computes cosine or correlation similarity of rows (users) or columns (items) and recommends items that k—nearest neighbors enjoyed.

19.5.2 Association Rules

Association rules can also be used for recommendation. Items that are frequently consumed together are connected with an edge in the graph. You can see clusters of best sellers (densely connected items

that almost everybody interacted with) and small separated clusters of niche content[233].

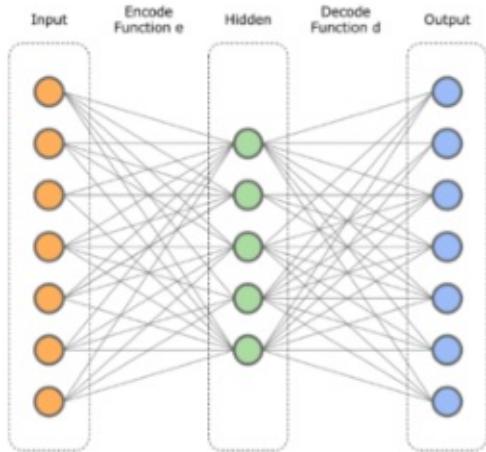
19.5.3 Matrix Factorization

"Matrix factorization models map both users and items to a joint latent factor space of dimensionality f , such that user-item interactions are modeled as inner products in that space. Accordingly, each item i is associated with a vector $q_i \in R^f$, and each user u is associated with a vector $p_u \in R^f$. For a given item i , the elements of q_i measure the extent to which the item possesses those factors, positive or negative. For a given user u , the elements of p_u measure the extent of interest the user has in items that are high on the corresponding factors, again, positive or negative. The resulting dot product, $q_i^T p_u$, captures the interaction between user u and item i —the user's overall interest in the item's characteristics. This approximates user u 's rating of item i , which is denoted by r_{ui} , leading to the estimate [234]."

Most popular training algorithm is a stochastic gradient descent (SGD) minimizing loss by gradient updates of both columns and rows of p a q matrices. SGD updates each parameter independently. Derive the loss function wrt each parameter.

19.5.4 Deep Neural Networks

Rating matrix can be also compressed by a neural network. So called autoencoder is very similar to the matrix factorization. Deep autoencoders, with multiple hidden layers and nonlinearities are more powerful but harder to train. Neural net can be also used to preprocess item attributes so we can combine content based and collaborative approaches.



{#fig: NeuralNetworks} [233]

$$\begin{aligned}
 \phi : X - > Z : x - > \phi(x) = \sigma(Wx + b) := z \\
 \Phi : Z - > Z : z - > \Phi(z) = \sigma(\bar{W}z + \bar{b}) := x' \\
 L(x, x') &= \sum_{i=1}^n \|x_i - x'_i\|^2 \\
 &= \sum_{i=1}^n \|x_i - \sigma(Wz_i + b)\|^2 \\
 &= \sum_{i=1}^n \|x_i - \sigma(W(\bar{W}x_i + \bar{b}) + \bar{b})\|^2
 \end{aligned}$$

Neural Network Equation

19.6 EVALUATION OF RECOMMENDER SYSTEMS

Few methods how the accuracy of a recommender system can be evaluated are as follows:

19.6.1 Validation of Recommender System

- Recommenders can be evaluated similarly as classical machine learning models on historical data
- Users are divided into:
 - Training set - This is fully submitted to the recommender system

- Testing set - This is submitted partially and used to evaluate the recommender

19.6.2 Root mean squared error

- When some observed data is provided, the recommender system is to predict the rating of an unknown user-item pair

"The root-mean-square deviation (RMSD) or root-mean-square error (RMSE) (or sometimes root-mean-squared error) is a frequently used measure of the differences between values (sample or population values) predicted by a model or an estimator and the values observed. The RMSD represents the square root of the second sample moment of the differences between predicted values and observed values or the quadratic mean of these differences[235]."

$$RMSE(model) = \sqrt{\frac{1}{|R_{test}|} \sum_{(u,i,r) \in R_{test}} (model(u, i) - r)^2}$$

RMSE equation

19.6.3 Top N Recommendations

"The explosive growth of the world-wide-web and the emergence of e-commerce has led to the development of recommender systems—a personalized information filtering technology used to identify a set of N items that will be of interest to a certain user. User-based Collaborative filtering is the most successful technology for building recommender systems to date, and is extensively used in many commercial recommender systems. Unfortunately, the computational complexity of these methods grows linearly with the number of customers that in typical commercial applications can grow to be several millions. To address these scalability

concerns item-based recommendation techniques have been developed that analyze the user-item matrix to identify relations between the different items, and use these relations to compute the list of recommendations. In this paper we present one such class of item-based recommendation algorithms that first determine the similarities between the various items and then used them to identify the set of items to be recommended. The key steps in this class of algorithms are (i) the method used to compute the similarity between the items, and (ii) the method used to combine these similarities in order to compute the similarity between a basket of items and a candidate recommender item. Our experimental evaluation on five different datasets show that the proposed item-based algorithms are up to 28 times faster than the traditional user-neighborhood based recommender systems and provide recommendations whose quality is up to 27% better[236]."

$$\text{Precision on Top - N} : \text{Precision}(u) = \frac{|\text{Recommended}(u) \cap \text{Testing}(u)|}{|\text{Recommended}(u)|}$$

$$\text{Recall on Top - N} : \text{Recall}(u) = \frac{|\text{Recommended}(u) \cap \text{Testing}(u)|}{|\text{Testing}(u)|}$$

$$\text{Serendipity, DCG} : \text{DCG} = \sum_{i=1}^p \frac{2^{rel_i} - 1}{\log_2(i + 1)}$$

Top N Recommendation equation

19.7 ACKNOWLEDGEMENT

I am thankful to Dr Gregor Von Laszewski to help me complete the project and the paper for the Big Data Applications and Analytics course

20 IOT AND BIG DATA: APPLICATIONS AND FUTURE TRENDS

FA18-523-71 FA18-523-59

Uma Kota, Jatinkumar Bhutka
umabkota@iu.edu, jdbhutka@iu.edu

Indiana University Bloomington

hid: fa18-523-71 fa18-523-59

github: [blue icon](#)

spaces before some of the citations missing, mentioned before

Keywords: IoT, Big Data, Analytics, Smart devices fa18-523-59, fa18-523-71.

20.0.1 Introduction

Since its inception, internet has been all about collaboration between people across the globe. Games, social media, images, movies etc., available in the internet were created by people for people. It caused a revolution in the way people connected with each other and it's now woven into their lives in one or the other way. With the rise in ubiquitous computing, Internet of things has taken this technology to the next level. Physical objects were now introduced and connected to the digital world. The term IoT was coined by Kevin Ashton who envisioned a future where computers could be connected to things and by leveraging the data collected could manage the things with little human intervention [237].

In today's bigdata world, Internet of things (IoT) has established itself in different fields of life by making processes more efficient and robust. As a result, with increase in digital connections between physical objects and the internet, data generation rate is going up and the availability of vast amount of data has opened doors for different kinds of analyses with the help of the services provided by big data technologies. IoT allows a device to connect with different

types of things like electronic devices, software's and sensors which exchange continuous streams of data, without any human intervention. The features offered by IoT allow companies to analyze their collected data and use it for business intelligence. Also, it can be useful to generate various models that can improve daily routine experience of the users. By 2020, Gartner has expected the IoT to connect over 20.4 billion things together ranging from mobile devices, vehicles, robots, and various industrial equipment etc [238].

The huge amount of data generated by IoT platform is not self-sufficient to generate insights and improve processes, it needs to be handled, processed, managed, integrated, analyzed in association with big data technologies, in a scalable, cost-effective way and more importantly in real time. Hence, digital world networks of physical objects and big data technologies can collaboratively be used to achieve complex tasks in the field of health care, manufacturing Industries, energy conservation, home automation, transportation system, education and research to improve quality of life.

20.0.2 Architecture

Interactions between digital mediums require special architecture and there are many architectures for IoT. The most common IoT Ecosystem architecture is as shown in Figure [30](#). It consists of seven different layers [239].

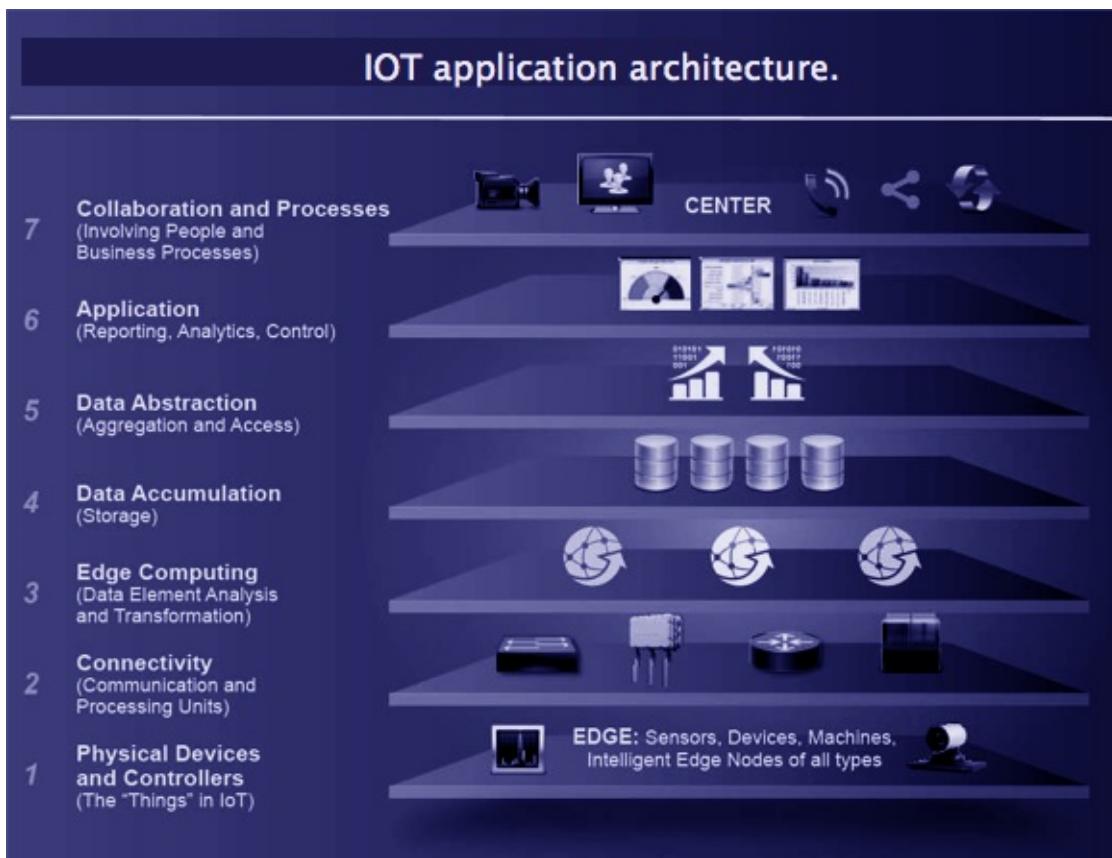


Figure 30: IoT Architecture[239]

- **Physical Devices and controllers:** The lowest layer of this Internet of Things (IoT) architecture is comprised of physical devices and controllers. The layer consists of devices such as electronic gadgets, sensors and activators which input the needed data from various sources.
- **Connectivity:** Next layer is termed as connectivity layer which takes care of communication between sensors and processing units. These processing units try converting input data from sensors into an understandable format with help of certain protocols and perform selection on data that is to be processed further into the IoT architecture creating thresholds for incoming data.
- **Edge computing:** Edge Computing does the analysis of data elements. Their transformation is achieved by analyzing and filtering data before it is processed further thereby reducing

huge computing and networking costs. The incoming raw data from sensors and activators can be selectively sent for further analysis. Also, one of the major concerns of Internet of Things (IoT) architecture is Data Security. Edge computing overcomes this to a considerable extent by feeding some of the sensitive data into sensor devices. Also, devices known as edge devices are coming into existence which will help in analytics and computation purposes, delivering data deliverable at a much faster speed in a robust manner. These edge devices will further help in maintaining connectivity with connected devices at source thereby allowing us to have a luxury of new smart devices.

- **Data Accumulation:** Data Accumulation is largely done in distributed frameworks as incoming data is in humongous volumes and variety while being input at a great velocity. Data is distributed into small sets of data using key/pair values just as in the case of data tuples, mapped and then reduced to small chunks of data before processing it further. Also, these days, organizations are heading towards PaaS (Platform as a Service) as a cloud platform for hosting data publicly and at the same time securing the data, thus customizing it to their needs.
- **Data Abstraction:** Different translation rules are brought into place for securing connectivity of specific devices. Also, a single model for data abstraction is created and provided to all the devices of a specific service thereby achieving integration of various devices.
- **Application:** The Application layer consists of reporting and analytics part of the architecture. All the efforts that were put into data accumulation, abstraction, storage, transformation, cleansing, preparing smaller chunks will be benefited only if proper analytics is performed and strategic Business Intelligence reports are generated out of this data.
- **Collaboration and processes:** Collaboration and processes

layer is the user interface layer where people i.e., the end users and the business processes come into picture. In the end, it is the customer who engages with the business processes and as it brings both of them together, it's named as collaboration and processes layer.

20.0.3 Big Data and IoT Together

Big Data analytics helps analyze data sourced from Internet of Things (IoT) which has multiple devices connected to it. These devices can be sensors, activators, websites, social media etc. Internet of Things (IoT) has had its advancements and applications in fields such as automobile industry, health-care, transportation & logistics, education, commercialized residences etc., and incoming data from these domains is in the order of billions of gigabytes per day, at the same time it is largely diversified. Also, the velocity of the inward data flow is extremely high. A Big Data platform then takes this generated data as its input and stores it into files. Since the input data is unstructured or semi-structured, frameworks used to store this data in an intermediate place are largely distributed. Different big data tools can be used for storing this huge data such as Hadoop, Apache Hive etc. One of the most prominent ones used in the industry today is the Hadoop Architecture which has Hadoop Distributed File System (HDFS) and MapReduce as its two components to process data into small chunks for report generation and analysis purposes. The data is in a way captured, integrated, mapped into different data sets of tuples and then processed to warehouses for storage. The data warehouses help store the legacy data in them and allow generating reports on the desired data at any given point in time [240]. This is shown in Figure [31](#).

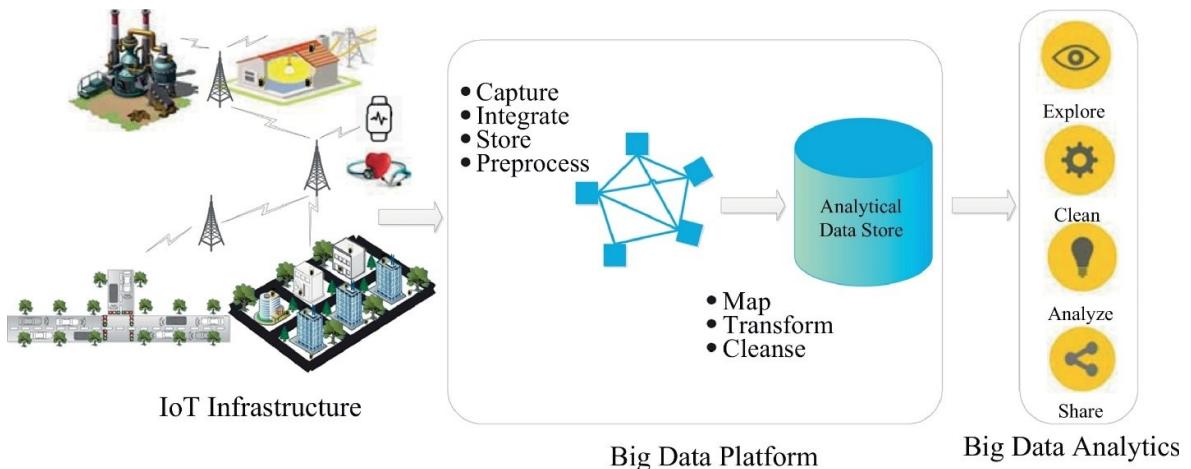


Figure 31: Big Data and IoT[240]

Big data helps businesses in coming up with inferences, insights and actionable recommendations by analyzing the unstructured or semi-structured data. Billions of devices are expected to be connected to the internet and hence for the functionalities of these devices to be held, data is needed. IoT will be a major data source for all the analyses that will be performed by companies for growth and effective decision making, with the nexus of big data and the IOT data from the connected devices making it cheaper and faster. The process of IoT with Big Data is seen in Figure 32.

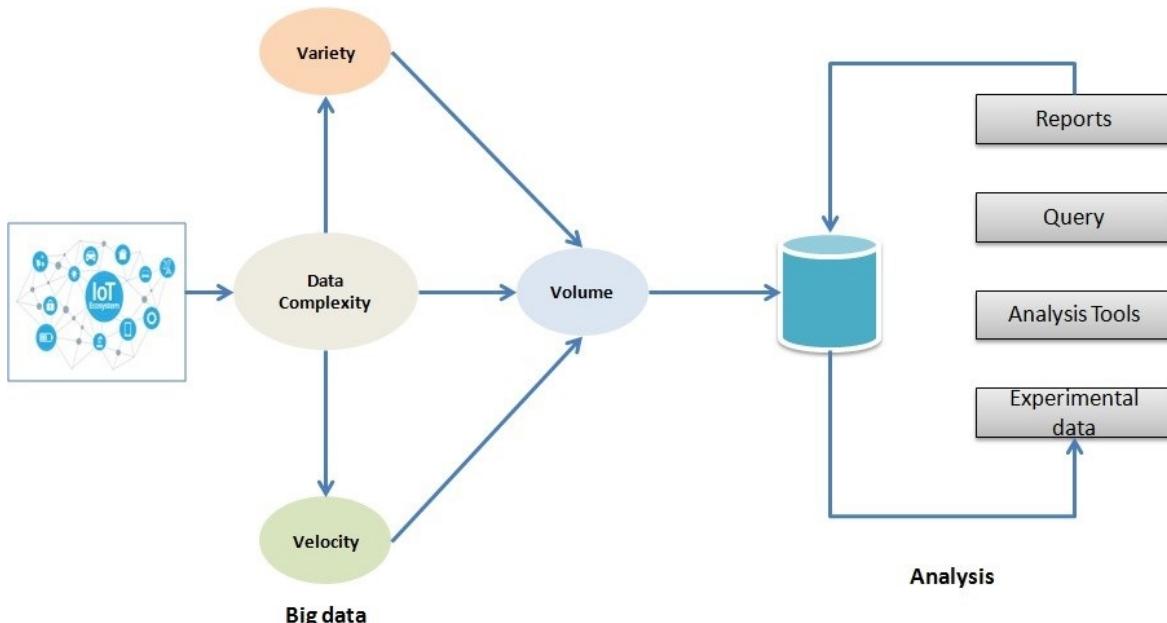


Figure 32: IoT Process[241]

- Data, which is sourced from various connected devices or from disparate data repositories is collected and stored in a big data platform.
- A big data system is often referred to as a distributed database as incoming data is sourced from different data repositories and is then stored in the form of flat files, in case of a Hadoop big data platform it's stored in the Hadoop Distributed File System (HDFS). Another big component of the Hadoop platform is Hadoop MapReduce which is used to process independent data chunks efficiently. The Hadoop MapReduce framework works on a MapReduce algorithm where data in form of files is converted into a data set using key/value pairs analogous to dictionaries and tuples. The output of a Map is then given as an input to the reduce stage where different data sets of tuples are converted into smaller chunks of data in order to be processed further.
- Reports are then generated by taking data from the concerned data warehouses using the Business Intelligence Report generation tools as per client/user specifications. The same is accomplished through complex query writing into databases. Therefore, big data platform paves way for this disparate data, collected from different 'collected devices' using the Internet of Things (IoT), to be leveraged in deriving different business trends and business insights.

20.0.4 Impacts of IOT on Big data

The Internet of Things (IoT) has impacted the Big data platform in a great deal and there is going to be a significant impact which lies ahead as well. Today, Humongous amounts of data are generated every hour by devices connected to the IoT and in the future there are going to be more and more devices connected to the IoT invariably making it difficult for data collection and analytics platforms to effectively and efficiently keep producing results [242], [243].

- Big Data Storage: With data being sourced from various disparate data sources and connected devices as is the case working with Internet of Things (IoT), the data storage platforms need to be very flexible as the incoming data is in the order of billions of terabytes. Also, the rate at which data is input to a data storage platform is unlike the natural data speed. One of the main features the companies seek is hosting data in cloud systems so that the data is kept publicly secure and as per the company requirements. PaaS (Platform as a Service) is one such model the companies are eyeing, to serve their purpose of effectively handling the continuous inward flow of diversified data and hosting them on cloud as PaaS provides this functionality.
- Big data Security: Major threats as far as big data security is concerned are to the Data Mining solutions as they provide the strategic solutions to the business. Hence it becomes increasingly important to secure them against the security breaches caused by fake users. There needs to be proper encryption of authentication measures of the users. Also, it can be resolved with the use of metadata (data about data) as to who accessed what type of data and so on. The other prominent security issue with Big Data is of the distributed frameworks that are used for data storage purposes. Hadoop being one such open source big data platform where more systems can be prone to the security issues as the data processed is distributed over many distributed frameworks.
- Big data Analytics: The analytics of big data consists of a lot of challenges as the inward flow of data is quite unstructured considering it is from different connected devices like the sensors and websites. Hence big data consists of three V's: Volume, huge volumes of data sourced from sensors is to be analyzed. Variety, Data is unstructured and is in the form of 3D data, 2D data, log files etc. Velocity, the speed with which the data is processed which is nothing but real-time processing and continuous data processing. Hence, we see

data complexity in Big Data Analytics.

- Tools of big data: In order to draw effective insights and inferences from the data, proper big data tools need to be in place. To make sure pivotal information is extracted in form of insights from the data while securing it, different Big data technologies are leveraged. The commonly used Big Data tools are Apache Hadoop, Apache Hive, Storm, Cloudera, Qubole etc.

20.0.5 Challenges

While IoT seems to be promising, there are many challenges associated with it. With increase in number of data sources and with these sources storing a large variety of data including private data, there is an increase in security concerns all over. Poor connectivity also seems to be an issue when the number of connected devices is in a large number. The business models that are implemented are expected to be robust and coming up with feasible, cost effective and efficient models is still a challenge. With time there may be compatibility issues between the technologies in the system as new standards may be implemented which will thereby increase the number of hardware and software devices connected. Data reliability is also a growing concern as the available data if corrupted will be leading to bad decisions. Increase in connected devices may also limit the speed with which data is collected [242], [244].

20.0.6 Applications

The Internet of Things and Big Data coming together play an important role in commercial, industrial and other applications to offer better data insights and inferences. Some of the most common application are shown in the figure Figure [33](#).



Figure 33: IoT Application [245]

- **Application in Manufacturing Industries:** The machinery that is enabled with the Internet of Things (IoT) assists in transmitting operation related information to the managers which helps them identify the areas of optimization and the process of automation. It is also used in facilitative management of machine tools expected to function in specified ranges of temperature. Sensors enabled by IoT can help monitor such machines sending alerts as soon as sensor senses some deviation from the original specifications. Operating machinery in the specified specification range in turn increases the efficiency of operations, diminishes the downtime of machines, thereby, helps reducing the costs and conserving energy. Another application of IoT in the manufacturing domain is the Production Flow monitoring where production lines monitoring is enabled from refining to the packaging of final products. If there is any discrepancy found in operations, then the IoT enabled monitoring of processes helps in adjusting them and managing them better from the cost of operation point of view and avoiding wastes during the production process. Further, IoT helps in tracking and tracing of inventory while delivering the products on a global basis which helps in managers getting a clear idea of the current supply chain process. We can ensure the safety of the workers in any given project by monitoring the performance indicators thereby reducing the injury rate and any relevant loss to the organization. IoT significantly helps in

quality control over the life cycle of a manufacturing process [246].

- Application in Health care: The Internet of Things (IoT) has advanced its applications in the field of Healthcare by a great deal right from monitoring remotely to getting accustomed to medication. It has served as a helping hand to the doctors to make significant progress in patient's health. The hospitals have started using Internet of Things (IoT) ensuring every patient gets maximum benefits and stays healthier no matter how small the disease to be cured is. The IoT has guided the healthcare industry in improving the implementation of healthcare processes in a myriad of ways.
 - IoT for managing inventory: There is a considerable amount of change in the way the hospitals have taken advantage of the IoT inventory management for the inventory control in the warehouses and the pharmacies.
 - IoT for optimization of healthcare workflow: Adoption of wireless infrastructure have helped manage the throughput and perform an analysis of the existence of bottlenecks, if any, in the system and that we could get rid of the same.
 - IoT for integrating the medical devices: Different ways are being looked at for integrating the devices like Fitbits with a view to obtain more data about the patient and be taken care of. Neel Ganguly, vice president and CIO at JFK Health System in Edison, New Jersey mentioned of JFK Health System starting to use devices like blood pressure cuffs, glucometers etc., with the intention of collecting data about the various signs of the patients which helped them serve the patients in a more effective manner. Just like IoT has security concerns in most of its extensive applications, the healthcare industry is no different. But continuous

efforts are being made to get rid of the existing barriers as well as the potential barriers [247].

- Application in Energy Management of Buildings: The traditional building management systems look after buildings power sourced systems. The new Smart building energy management systems with help of IoT sensors offer a lot more than monitoring these systems. The data collected is also leveraged to conduct analysis on the raw data which is then transformed into insightful reports that can help make the systems way more efficient. Sensor devices that track weather and traffic data can also be connected to this system and their data can be utilized in adjusting a building's lighting, temperature etc., in real-time. The operating costs can be cut by 25% with help of these systems [248].

20.0.7 Use cases

Now, IoT and Big data is everywhere. Due to the flexibility and scalability of big data, industries in recent times have started adapting to the use of IoT. Some of the industrial use cases include,

- Caterpillar: An IoT pioneer Caterpillar, one of the leading equipment makers is a perfect example of how it has reaped the benefits of using Internet of Things (IoT). They have eased out process of identification of levels of fuels for the personnel operating the machines and the timings of replacement of the air filters using the Internet of Things (IoT) along with Augmented Reality (AR) [249].
- IoT based Asset Tracking System in Transportation & Logistics: Number of assets were able to be tracked down by the existing tracking systems and the system use to breakdown reaching its threshold after the number of assets tracked increased exponentially. Since the devices used for tracking were from diversified manufacturers, it became increasingly difficult to analyze, draw inferences from the

data and provide Business Intelligence reports. The solution to this was Internet of Things (IoT) which helped built a server for processing robust messaging. Geo-fencing was brought into implementation using IoT which would allow a trigger to be made on an asset entering or leaving a area. Also, an engine with respect to analytics was developed with a view to provide better data insights and meaningful Business Intelligence reports [250].

- Smart metering using IoT: For measuring the consumption of gas, energy and water in buildings, a device capable of working with Internet called a smart meter is used. It has helped overcome the disadvantages of measuring just the overall consumption by allowing to record the level of consumption of each of the resources used thereby benefitting consumers monetary wise. In this way, Internet of Things (IoT) has helped improve the process of forecasting and binding the consumption of power [250].
- Predictive Maintenance using IoT: Millions of dollars can be saved by the companies by keeping the equipment's and assets going all the time with the help of sensors and data analytics thereby preventive maintenance [250].
- Airline Management using IoT: IoT in airline industry can be seen everywhere from baggage tracking to cabin climate control. With help of IoT, the industry is trying to increase customer satisfaction by reducing the complaints like lost baggage, flight delays and service issues as well as in cutting unnecessary operational costs. Airways like Virgin Atlantic, Etihad are manufacturing planes that are connected with help of IoT devices. Data of half a terabyte is expected to be produced over a flight journey, this will be analyzed to get the information regarding mechanical issues etc., before they may even occur. With help of such technology the flight delays, safety issues will be restricted. Delta was one of the first airlines to introduce Radio Frequency Identification

(RFID), a baggage tracking technology that leverages IoT. With help of RFID, the customers can always access their baggage location from their phone while travelling with the airlines. IoT is also being utilized in navigation of flights such that aircraft always takes an optimal route thereby optimizing the fuel use. This is being implemented by Air Asia with help of GE to cut fuel costs. Jet Blue with help of IoT has automated the check in process for customers, with a ticket and seat being issued directly to every customer not needing any customer inputs by analyzing their preferences from their data [251].

20.0.8 Future Trends

- Retail Industry: IoT promises to help retailers take better business decisions while helping them better the customer satisfaction. Omni-channel experience is going to be the future of retail and is feasible with IoT. The retail industry in future will be seamlessly integrating online stores with brick and mortar stores to make it easier for the customer to find their products while simultaneously connecting with shoppers in a more personal way. For example: Alerts can be sent to the customers when their favored location is out of a certain product with other recommendations etc [252].
- Health-care: With a proactive technology in place gathering a person's daily data, the onset of any clinical trail can be predicted there by reducing the number of emergency cases. IoT enabled technologies can make this scenario feasible in the future with help of a network of wearable devices, sensors, monitors etc., set up for monitoring health. With help of Monitoring technologies like telepresence, checkups can be done from any part of the world and the number of hours spent in hospitals can be cut down. Also, the data gathered by these technologies can help transform the check-in process by automatically sharing the past data of patient with health professionals [253].

20.0.9 Conclusion

With data coming from disparate data sources and diversified connected devices, it has given the big data analysts option to distinguish and filter data and just process the useful chunks of data further to the data warehouses and then generate Business Intelligence reports from there on. With the emergence of IoT, it has become increasingly possible to reach out to the customers at any given time of the day. Devices like Fitbits and smart devices have made it possible for the doctors to get timely updates of the patient's doing about his health thereby allowing the physicians to intervene whenever needed and provide mannerly treatment and diagnosis to their patients. Businesses have been reaping the benefits of the advancement in technology by having the luxury of digitally connecting to their clients even from the most remote locations in the world. In the years to come, the amalgam of advancement in technology i.e. the Internet of Things, big data will give more likelihood to businessmen in gathering even the most detailed information of their client and customers thereby giving themselves a better opportunity to improve their business from the perspective of effectively managing the monetary department and providing a better customer satisfaction which will invariably up their business a great deal [254].

With time and the advancements in the field of IoT and big data it can be concluded that they are two sides of the same coin bolstering one another. The Nexus of Internet of Things and Big Data can be seen as the future of business intelligence, health-care and many other industries.

20.0.10 Acknowledgments

The authors would like to thank Dr. Gregor von Laszewski for his support and suggestions to write this paper.

format incorrect (fixed)

Wang Tong

wangton@iu.edu

Indiana University Bloomington

hid:fa18-523-73

github: [https://github.com/wangton](#)

species before some of the citations missing, mentioned before in class (fixed)

keywords: Big data, healthcare, healthcare informatics, unstructured data

21.1 INTRODUCTION

Big Data is used in the analysis of healthcare informatics, which consists of complex data sets that are hard to store, resolve, organize, process, and interpret. The benefits obtained from the application of Big Data in the healthcare system include reduced costs incurred in the delivery of healthcare services, optimal management of information, prevention of the side effects of drugs on patients, the discovery of treatments for incurable diseases, and improved service delivery. The challenges faced in the use of Big Data in the healthcare system include problems with analyzing unstructured data, focusing on correlations rather than causality, privacy issues which act as barriers in the collection of information, data security due to hacking, and reluctance by health care centers to share patient data due to competition.

Big Data is the vast amounts of complex information which require advanced technology to assist in their capture, storage, distribution,

management, and analysis. The Big Data characteristics can be explained using the 6Vs, which are velocity, volume, value, variety, variability, and veracity. The former implies the data sets are generated at very high speeds. The volume stands for large amounts of data sets that require analysis. The value suggests the vital information that is contained in the data to be analyzed. The variety stands for all types of data sets such as structured, semi-structured, and unstructured. The variability aspect entails the changes that occur during the processing of data. The veracity of the data sets collected implies that it is consistent and it can be trusted. The Big Data evolution can be attributed to the technological innovations of artificial intelligence, the internet of things, the fifth generation of the internet, and advancements in machine learning. The emergence of Big Data is essential to the growth of all fields. Therefore, the application of Big Data in the healthcare system will help to increase the speed, volume, accuracy, and efficiency in the analysis of the complex healthcare records.

21.2 REQUIREMENTS OF THE ELECTRONIC HEALTH RECORDS

The electronic health records (EHR) is a system that has been developed by software engineers and contains information regarding the health care system. The data provided in the EHRs include patient information, medications, diagnoses of diseases, procedures, signs and symptoms of conditions, visit dates and times, and clinical notes of the healthcare practitioners. The data stored by the EHR system can either be structured or unstructured. The former include administrative data such as the demographics of the patients, diagnoses, and procedures. Unstructured data mainly entails the clinical notes that are taken by the doctor, and are the most efficient way for clinical documentation because they rely on human intuition. The unstructured data is difficult to analyze because of the format used since it may contain grammatical errors, abbreviations related to medicine, and spelling errors. The large amount of information about the patients and healthcare in the EHR system has led to the need for computer-based methods to help in the organization,

analysis, and interpretation of the data sets. Hence, the adoption of Big Data in the EHR system has been essential due to the high speed involved in the analysis of the extensive medical health records.

21.3 THE ARCHITECTURE OF THE ELECTRONIC HEALTH RECORD

The data in the electronic healthcare record system can be categorized into various forms that include genomic, clinical notes, behavior, patient sentiments, clinical reference, and administrative. Genomic data is a category which contains the DNA sequence, gene expression, and genotyping [255]. Clinical notes are unstructured data sets that include a diagnostic testing report, medical images, patient discharge summaries, and the doctor's notes [255]. The patient sentiments contain personal information about them such as their demographics. Clinical reference data is information from text-based applications such as journals, articles, clinical research, websites, and product information. Administrative data involves information from the health care center that allows others from outside the context to understand the medical records [255]. Therefore, through the understanding of the various categories of data in the electronic healthcare record system, a proper analysis can be made on the available information.

In addition, the EHR system contains the aspect of interoperability, which implies the ability to share data between various health center facilities. The descriptive information about the patient's demographics, pharmaceuticals, and diagnoses need to be recorded in a way that allows sharing across different healthcare facilities. The institutions can implement internal encoding mechanisms that assist in the analysis of the data collected before sharing of the information with other health center facilities. Additionally, administrative data should be included in the EHR system to allow for an understanding of the information contained in the records outside the context in which they were recorded. However, the main hindrance to the health records data interoperability is the lack of a similar data

standard being used by the different healthcare centers. The barrier creates the need to establish a systematic widely accepted data-encoding scheme that will enable hospitals to share descriptive information about patients contained in their EHR system. Thus, data interoperability provides for the ease of sharing EHR by hospitals, which allows research to be easily conducted.

Furthermore, the EHR system also makes it possible for data mining, which is the extraction of the information contained in the electronic health records. It entails the use of various techniques, which include regression analysis, classification, associate rule learning, and temporal data mining. Regression analysis involves the estimation of the correlation between the independent and the dependent variables. Regression linear model fitting can be used if the dependent variables adopt distributions such as binomial, normal, and Poisson. Classification involves techniques such as k-nearest neighbors and the decision trees. It is effective in clinical applications. It involves the assigning of a new observation to a class through building statistical models. Associate rule learning uses the numerous associations among clinical variables to predict the occurrence of phenomena. Temporal data mining is based on the fact that it is hard to generalize the outcome of a treatment to a given patient because of the difference in the clinical variables. Therefore, the vast amounts of complex data contained in the EHR allow for the extraction of the data using data mining.

21.4 IMPLEMENTATION OF BIG DATA IN ELECTRONIC HEALTH RECORD

The implementation of Big Data in the EHR system has improved the ease of capture, storage, organizing, interpretation, and the analysis of the data sets. Furthermore, it has made the system more reliable, accurate, and efficient in service delivery when compared to the traditional system that was being used to keep the medical records. The benefits gained from the implementation of Big Data in the electronic health records outweigh the challenges faced. Therefore,

the implementation of Big Data in EHR is a necessity that will help to make the health care system efficient in service delivery.

21.5 BENCHMARK OF BIG DATA IN EHR SYSTEMS

The benchmark of Big Data in the electronic health record system involves the analysis of the benefits, opportunities, and the challenges faced in the implementation process. The benefits of adopting Big Data in the EHR system include precision medicine, reduced costs incurred in treatment, and the optimization of the workflow in the healthcare center. The challenges faced include data hacking, privacy issues and the analysis of unstructured data. Thus, despite the difficulties in implementing Big Data in EHR systems, its adoption is inevitable by the hospitals.

21.5.1 Benefits of Big Data in EHR Systems

The first benefit of the adoption of Big Data in electronic health records is precision medicine. Through the evolution of Big Data, high-performance genome analysis technology was invented which allows for the collection of large amounts of genomic data. Additionally, new analytical algorithms have been designed to help analyze the genomic data collected [256]. The high-performance genome analysis technology enables the healthcare facilities to compare the genomic data of one patient to a large population of other individuals [256]. The comparison helps to establish the emergence of rare diseases. Moreover, it not only helps the healthcare facilities to develop a diagnosis of the illness but also prevent its spread to other members of the population. Nevertheless, it allows for the improved efficiency in service delivery due to the systematic collection and the analysis of genetic data. Other benefits obtained from the concept include the selection of the best treatments, avoidance of the side effects of diagnoses to the patients, and the elimination of ineffective therapies. Therefore, precision medicine has been crucial in improving efficiency in healthcare operations.

Moreover, the adoption of Big Data in the electronic healthcare records has reduced the costs incurred for medication. The technology has allowed the health care centers to notice the early signs of diseases such as cancer [256]. The earlier recognition allows the individuals to meet the low costs incurred in treating the primary phase rather than the high bills needed for the final period of the disease. Furthermore, the health facilities have reduced the expenditures incurred in the research of new drugs for the treatment of diseases because of the ease of access to information required for research purposes in the EHR. At the same time, the hospitals can administer treatment on a performance basis whereby the practitioners receive money because of the outcome of the diagnoses rather than the time spent treating a patient [256]. Hence, the adoption of Big Data in the health care system has helped to reduce the costs incurred by both the hospital and the patients during the treatment of diseases.

Besides, the adoption of Big Data in the healthcare facilities has optimized workflows in healthcare centers. It has enabled various departments within the hospital to share information using the electronic healthcare register and reduce the movements from one department to another [256]. Additionally, the practitioners can communicate their findings on patients with each other and establish the best selection of treatment that will help in the diagnoses of disease. The communication of the departments using the EHR allows for better utilization of the resources of the healthcare facility. Moreover, it also brings higher efficiency in service delivery by the health care facility to the patients [256]. Thus, the optimization in the workflow within a healthcare facility is a benefit obtained due to the implementation of Big Data in the EHR system.

21.5.2 Challenges of Big Data in EHR Systems

The first challenge faced in the implementation of Big Data in healthcare systems is the analysis of unstructured data. The test results, scanned documents, X-ray images, and progress notes are some of the examples of unstructured data found in the EHR of a

healthcare facility [255]. The large volumes of unstructured data, such as medical imaging, are difficult to handle by the EHR systems. Additionally, extracting potentially useful information from the unstructured data is difficult. It is also problematic to understand the informal clinical notes because of the context in which they were written [255]. Thus, analysis of unstructured data is a key challenge faced in the implementation of Big Data in electronic health records.

Moreover, privacy issues are another challenge faced in the adoption of Big Data by the electronic healthcare records. The hospital's ethical considerations demand the protection of the patient information, and it also restricts the sharing of the data without the consent of the individuals. The aspect of privacy concerns hinders data interoperability among various healthcare facilities that would have aided in the efficiency of service delivery by the hospitals. The reason is Big Data requires data interoperability. Nevertheless, the healthcare facilities can employ privacy protection mechanisms to help protect the information of the patients from loss or unauthorized access.

Another challenge faced with the implementation of Big Data in the EHR is the aspect of data hacking. Healthcare facilities have incurred high costs due to the leakage of data [255]. Apart from being sued by the patients whose medical information is leaked, the hospitals also face the loss of their patients due to a lack of trust. Additionally, the hospital is held at ransom by the hackers who demand money in exchange for the information obtained. Nevertheless, the healthcare facilities have adopted the use of biometrics, such as fingerprints and voice recognition software, to reduce the leakage of data and improve protection of information about the patients from the electronic health record system [255]. Hence, hacking is a menace that hinders the implementation of the Big Data in the EHR system.

21.6 CONCLUSION

The adoption of Big Data in EHR is inevitable because of the benefits associated with its approval. The advantages related to Big Data

include best selection of treatment for diagnoses, reduced costs incurred, optimal management of information, prevention of the side effects of drugs on patients, and discovery of treatments for incurable diseases. Despite the benefits associated with the concept, it faces challenges such as privacy issues, difficulty in the analysis of unstructured data, and data hacking. Further studies need to be carried out to establish on the ways to minimize the challenges encountered in the implementation of Big Data. Therefore, the application of Big Data in the health care systems is essential for an improvement in service delivery.

Yeyi Ma

yeyima@umail.iu.edu

Indiana University Bloomington

hid:fa18-523-74

github: [blue user icon](#)

Keyword: Privacy

22.1 INTRODUCTION

Big data as a smart technological aggregate of database technologies utilized at the software and hardware level and its current be used in the business, technology, governmental circles and etc. Benchmarks that have been developed in order to standardize big data, it is revealed that the technology is relatively young to employ most of these benchmarks. In respect to privacy, big data poses a major threat to peoples entitlement to confidential data by design. As such, it is necessary that big data is regulated.

Big data refers to the study and the application of complex data sets in application software and hardware. Big data is associated with certain qualities of data such as variety, velocity, volume, veracity, and value. The computer and information age has made big data even more practical and relevant in processing information. Today, big data entails the use of computerized systems which execute predictive analytics and user behavior analytics. These analytics extract value from data sets. Most processed data is available in large volumes. However, that is not a huge problem when working with big data. To computer systems, processing a single record will take the fraction of a second. Processing a million records will not take a million seconds. Instead, it might take a few extra seconds. Modern computer systems utilize economies of scale in a manner that cannot

be replicated in any other domain.

The most important characteristic of big data is that it analyses large data sets in a manner that is intuitive. This explains why big data has become so popular in nearly every field in the society: government, internet search, fin-tech, business informatics, urban informatics, medicine, and travel. Big data poses a threat to privacy since the end user does not have the centralized infrastructure required to make it work flawlessly. There is always a billion-dollar company out there with access to individuals' private information for every smart device out there.

22.2 ACADEMIC THEORY

22.2.1 Requirements

One of the main reasons why big data is growing rapidly is the increase in the popularity of devices which fall under the internet-of-things. Therefore, gathering the data necessary for processing is relatively easy [257]. Over the past 4 decades, the technological ability to store data per capita has nearly doubled. This will only continue to grow over the next several decades.

The greatest concern for millennials and Gen-Z is who should stay in control of the big data initiatives. This raises the question of privacy. Trusting big companies such as Facebook, Apple, Google, Amazon, and Microsoft is not considered to be a credible solution. These companies own most of the big data infrastructure in the world. Moreover, they have a monopoly over all the internet-of-things devices such that they can essentially shut down all the competition by making their services and devices cheaper while harvesting user information.

The leaders of these companies are not elected by the public and are out to maximize profits. This is dangerous given that they already have a monopoly over modern communication but manage to operate as private entities capable of ideological, cultural, and

religious bias. Moreover, they are not answerable to anyone other than their investors who usually only care about the bottom line. Big data promises the world numerous benefits over the next several decades. However, this comes at the cost of giving up liberties enshrined in the constitutions of most countries that have a Bill of Rights.

22.2.2 Architecture

Big data architecture is not new in the 21st century. Database management systems were popular in the 1990s and were offered by a few big companies. A company such as Wintercorp became famous for issuing intuitive big data repositories in the form of reports in this era. At the time, the biggest hard disks were only 2.5 GB. This means that the definition of the term big data keeps evolving in accordance with Kryder's Law[258]. The company has installed more of these data stores over the past 10 years. Their largest database store is more than 50 PB.

Other companies such as LexisNexis Group have been involved in developing the architecture which defines big data. In 2000, LexisNexis created a C++ system for distributing data and enforcing simple querying. This dialect uses a technology called "apply schema" in order to infer the schema of data under query.

Another important organization that has been utilizing and developing big data includes CERN[259]. This organization has been collecting big data for decades. The data is analyzed using supercomputers. In 2004, Alphabet Inc. published a paper called MapReduce which employs the same structure as CERN. The paper proposed using parallel processing to enhance speed and accuracy. In MapReduce, queries get split and distributed over nodes[260]. They are then processed in a parallel manner in what is called a Map step. The Apache open source project is one of the first to implement the MapReduce paradigm in a project called Hadoop. However, the MapReduce paradigm had certain limitations. As such, it was necessary to develop the Apache Spark. This new paradigm added

the ability to add many operations as opposed to a single map.

Today, the most popular big data architecture is the MIKE 2.0. This is an open approach to Information System Management[261]. It acknowledges and addresses the need to keep revising the implications of big data. These insights were captured in a paper called “Big Data Solution Offering”. Studies highlight that using multiple layers in big data is one of the ways of addressing the speed problems that have persisted in big data. In manufacturing, big data architecture is implemented as 5C. This stands for conversion, cyber, connection, configuration, and cognition. Another vital aspect of the big data architecture is data lake. This is a technology which makes it possible to shift focus from centralization to a shared model in respect to the changing information system dynamics. It also makes it possible to segregate data within a data lake in order to minimize the overhead time.

22.2.3 Implementation

The main components of big data can be summarized into three categories: analyzing techniques, databases, and visualizations. The data analysis techniques include A/B testing, natural language processing, and machine learning. On the other hand, databases include any technologies employed in the process of data storage. This covers cloud computing as well as business intelligence technologies. Visualization includes graphs, charts, videos, and other technologies which are used for displaying the output data.

In some cases, it becomes necessary to represent multidimensional big data as cubes or tensors. To do this, it is necessary to introduce an array of database sensors for high-level query and storage support. Other technologies that are necessary when implementing big data in an institution include subspace learning, tensor-based algorithms, distributed filing, HPC infrastructure, cloud infrastructure, data mining, the world wide web, and distributed databases. It is important to note that in spite of all the improvements seen in big data architecture, machine learning is relatively elementary. There

are numerous challenges to machine learning. While most of these challenges are technical, a few of them are social in regards to the interaction between individuals, the law, and profitability.

Companies that have implemented big data utilize the latest and greatest when it comes to computing and storage. This means that there is a significant barrier to full utilization of big data at an individual or SME level. Companies that have embraced big data employ direct attached storage such as solid state drives and high-speed SATA inside parallel processing nodes for speed. It is quite rare for a company utilizing big data to use storage area networks or network arranged storages. These two are perceived to be slow, expensive, and difficult to use. If there is an existing technology in the market that performs faster, it needs to be implemented in big data.

Big data has numerous applications in a variety of fields. By 2010, the big data subsector was worth at least \$100 billion. It was growing at the rate of 10% per year. This was about twice the rate of growth of the software industry in the same year. Many developed economies in the world are continuously using data-intensive technologies. Many people in the world have access to the internet thanks to the development in computer software and hardware[262]. The effective capacity of the world to exchange information is growing at a rate that is unprecedented. By 2014, internet traffic reached 667 exabytes by predictions. Big data is being utilized in international development, manufacturing, healthcare, media, education, the internet of things, information technology, and insurance.

22.2.4 Big Data and the Web

Big data is commonly associated to the internet of things since both concepts are based on smart data collection and manipulation. The number of people who rely on the internet for their work, entertainment, communication, travel planning, and education is increasing every year. The development of the internet is responsible for driving this shift from traditional tools to digital ones. Big data can be seen as an augmentation of the internet. This is because it creates

a link between the physical world and software. Most efficient programs run with a connection to the internet for the purposes of collaboration, communication, and sharing resources. With more and more links to big data, there is a clear conduit between the world wide web and the physical world.

One the most important improvements that has taken place in technology over the past several years is the improvement in user interface that is presented to technology users. Big data seeks to fill the gap between the user and the software that takes instructions from various devices by collecting feedback without too much work from the user. This explains why the internet of things is becoming the most important component of big data. The internet of things allows physical objects to embed operating systems in order to collect data in real-time. When such information accumulates to volumes that can significantly alter or influence the manner in which an object works, big data components such as DBMS take over the records and process them yielding output that can be used adjust certain parameters of object, an apparatus, or a system. While the internet of things addresses the user interface solely, big data goes a step further and provides a processing platform that can then be used to develop useful output. Big data also allows systems to consume the feedback generated without the intervention of a human.

22.3 BENCHMARK AND PRIVACY

Like any other viral and potentially useful technology, big data has various benchmarks. These benchmarks are tailored to the new technology since the existing ones have proven to be ineffective. In spite of this, the benchmarks that have been proposed have not been robust enough: BigBench, HiBench, AMP, and CloudSuite[263]. Each of these benchmarks has various merits and demerits. With all these options, the problem remains that big data is not mature enough to be tested in a standard environment. As such, it is prudent for an institution to benchmark their big data implementation using the usage scenario as opposed to these tools.

The factors that make big data efficient are the same ones that make it a major privacy risk. Big data analytics is dumb, meaning that it does not have preconceived notions about the subject on which data is being collected and processed[264]. This means that it is capable of collecting more accurate information than people perceive. For instance, big data can be used in medicine to diagnose in an objective manner. It can gather data and come up with recommendations that have been obvious but ignored due to preconceived notions, ideology, or human expectations. When it comes to big data, using algorithmic black boxes can be dangerous. Such implementations leave machines to decide what to make of inferences without the possibility of human intuition. Multinational companies that are implementing big data and using it to sell their services have handled this problem to a great extent. They have manual overrides to minimize the overreach of big data. However, these companies are not public utilities since they have a profit agenda meaning that privacy is hardly a priority.

22.4 CONCLUSIONS

Big data is replacing the raw information age that was ushered in by the growth of the internet as well as internet-based technologies in the late 20th and early 21st century. Big data makes use of numerous technologies to ensure that data is harvested in real-time and utilized in the same fashion. With the internet getting embedded in different home and personal appliances via embedded operating systems, big data will continue to become more popular. The applications of the technology are likely to outdo those of the bare internet given that user interface is one of the key areas that big data seeks to revolutionize. The applications will be seen in business, government, technology, services, travel, academia, and other sectors. However, big data poses a big threat to privacy since it eliminates the regulative human factor from the equation in an effort to offer fluid services whenever it is implemented. It is necessary that policymakers create regulations which protect technology consumers without killing innovation such as big data.

Abhishek Rapelli
arapelli@iu.edu
Indiana University
hid: fa18-523-79
github: [abhi_rapelli](#)

23.1 KEYWORDS

HID fa18-523-79, QlikView, Associative In-Memory Technology, QlikView Server, QlikView Publisher

23.2 INTRODUCTION

QlikView is a popular software that was based on different approach for data discovery and analytics compared to other traditional analytical and visualization software tools. The uniqueness of QlikView is that rather than first building a query and then fetching the results, it forms associations in the data once it is loaded and then the user is prompted to explore and analyze the data with. This simplification of data exploration and the enhanced user interface has made it one of the most popular choice for traditional business analytics and reporting for businesses. QlikView, besides QlikSense is one of the software tools that was developed and maintained by Qlik software technologies company for Business Intelligence, analytics and visualization. Over the years, with more advancements and simplifications in BI tools, QlikView has gained its prominence for small to medium scale business analytics applications due to its simplicity, easiness and handiness. It adapted to the changes in the BI tools experience that businesses and Industries wanted, like for example Predictive analytics ability, advanced visualization and smart suggestions. Majority of the business analytical tasks require routine reporting and dashboard creation for presentation to managers, which is most times very repetitive and mundane. QlikView makes

such work much simpler and easy for them [265].

23.3 ARCHITECTURE

Before moving forward to deploy and use QlikView software, it is very much important to understand the architecture of QlikView, its products and the components. The QlikView deployment has three main infrastructure components. They are QlikView Developer (QVD), QlikView Server (QVS) and QlikView Publisher (QVP). The QVD is a desktop application for Windows operating system for designers and developers for performing operations like data retrieval, storage and processing, and also to make graphical user interface (GUI). The QVS is a component that handles the interaction or communication between the clients and QlikView applications and components. It also loads QV applications into the main memory and helps in running the user selections. The QVP is collects and loads the data from various sources, in different formats ranging from XML to CSV. It also reduces the QV application and then distributes the data to the QVS server. It is always good to separate the two components of QVS and QVP, since both function differently, handle the memory and CPU differently and have completely different roles [266].

Broadly speaking, QlikView's architecture can be viewed as the combination of two main components. They are the front end and the back end. The front end's function is to visualize the processed data whereas the back end's function is to provide security and publication mechanism for the new user documents.

23.4 THE FRONT END

The front is the component that facilitates users to interact with the data and documents stored for data processing through the QlikView server, anywhere and at any time. The QlikView Publisher in the back end creates QlikView documents of the user, which are contained in the QlikView server of the front end. These files are stored in the formats of QVW, meta, and shared. The interaction or communication

between the user or client and the server is managed through HTTPS, or QlikView Proprietary, also called QVP protocol. The QlikView server also handles the security of the client.

23.5 THE BACK END

The back end is the inner component of the QlikView where the QlikView documents that are created by using QlikView Developer are stored and protected. The files can be script documents for data extraction from various sources like SQL scripts, the data that is in binary form and stored within the QVD files. The most important component of the back end is the QlikView Publisher, which is responsible loading and distribution of data. The back end functions within the windows environment and required special privileges and permissions for its functioning.

23.6 ASSOCIATIVE IN-MEMORY TECHNOLOGY

Associative In-Memory technology is used by QlikView for the analyzing and processing the data. The advantage of this technology is that it stores only unique entries at a time in the memory and rest all are pointers to the main data. This makes it very fast and allows to store large amounts of data than other traditional methods. The performance and scaling is key to QlikView as the user is directly connected to the CPU of the QlikView, and hence this technology is used for speed and scaling.

The QlikView's System resources consists of computation components like CPU, RAM, etc. The QlikView Server and QlikView Publisher are the two components that use and handle these resources differently as they serve different purposes and exhibit different roles. Let us look at how these two utilize and handle these resources.

23.7 QLIKVIEW SERVER (QVS)

23.7.1 CPU

The QlikView Server consists of multiple CPU cores that are multi-threaded and optimized. The available cores are used linearly for processing the QV files. The QVS manages the usage of these cores for computation of these files on real-time basis by monitoring and allocating the cores efficiently. It leverages the processor for dynamic aggregation creation quickly and shows the results to the end user intuitively. Typically, the actual data that is stored in the memory or RAM is in unaggregated raw form. Thus, to perform aggregation operation on real-time bases very quickly on huge sets of data, efficient and dynamic use of computing power is needed. If the processing power is not sufficient, it may lead to waiting and the processes are done based on priority, where a waiting list is created for allocation to the cores based on priority and cores availability. This is a linear and sequential processing model.

23.7.2 Memory

The QlikView documents and files are stored in the main memory RAM, which is the primary storage for all files. The data that is stored in the RAM can either be in unaggregated form or aggregated form. QlikView memory is based on Snapshot technology, where it is continuously refreshed through a process known as reloading a QlikView document. As the QlikView document is loaded, it will establish connectivity to the data sources that are to be analyzed and the unaggregated data to be extracted and for compression, which is then stored as .QVW format in the persistent disk storage.

When a user opens an analytical application, QlikView loads the .QVW file from the persistent disk storage into the RAM. Further, QlikView only relies on the dataset that is loaded into the RAM at that time and does not care about the other data in the database. This is done because QlikView would be able to handle the process quickly, resulting in instantaneous real time response to the end user. Thus, all the data that has to be processed is placed in the RAM. RAM is the important deciding factor for QlikView about how much data it can

handle at once. A Windows Operating System will consume a RAM of 500 to 1000 MB, while QlikView Server process requires a RAM of at least 100 MB on QVS.exe process.

23.7.3 Data Compression

For effective usage of RAM and quicker real time processing, we need to forgo redundant and repeating data. This can be achieved by loading on distinct data points into the RAM instead of all the data points. This not only reduces the RAM usage and scales up to accommodate huge dataset but also makes the process very quick due to non-redundancy in the dataset. This process of reducing the data points is called Data Compression and is handled by the QlikView server.

23.8 QLIKVIEW PUBLISHER

23.8.1 CPU

QlikView Publisher is a database load engine and it creates thread for every database connection to facilitate hundred percent utilization of cores during processing. The number of cores being utilized for a process is equal to the number of databases being loaded for the process. The QVS and QPS are not placed on the same server because both use and handle CPU cores in different way as explained earlier.

23.8.2 Hard Drive

The QlikView creates a data repository of historical data that the end users have loaded from various applications. This is also known as data cache. The main advantage of this kind of data model is that it reduces the database communication and reduces the time lag. The main disadvantage of this is that the disk space needed to source files is very high. It is recommended to have at least 150 GB space of SAN drive.

23.8.3 Memory

As we noted earlier, as QlikView Publisher is a database load engine and a file distribution service. It is not just an analytics service engine. Compared to the QlikView Server, QlikView Publisher is not memory intense and hence memory is not a consideration or a factor for determining the size of the service to accommodate huge QlikView Publisher instances [267].

23.9 USES AND ADVANTAGES

QlikView is widely used by businesses for Business Intelligence, analytics, visualization and reporting task to monitor business performances on daily, weekly, month, quarterly or yearly basis. It can be used both for on-site reporting as well as client-side remote reporting on real time through internet. Hence, it can be accessed at any time or anywhere through internet. From user point of view, it is very simple and conformable UI, that is drag and drop based and hence not at all difficult or hard to master and operate quickly. The reports and visualizations can be interactive, which makes it feel good and easy for understanding to the viewer. All these advantages make it one of the most preferred BI tools in the industry [268].

23.10 CONCLUSION

QlikView being a web-based software tool is very simple, reliable and robust and access-able for various BI applications. It is easy to learn, work on and present compared to traditional software available in this segment. Yet, it has got lot of short coming too, where it cannot be used for large scale data analytics like Big-Data and it does not support advanced machine learning based predictive analytics. However, for small to medium-large level data sets, it is one of the best software that is available for business analytics and visualization tasks.

24 UTILIZING PYTHON MATPLOTLIB PACKAGE FOR DATA VISUALIZATION OF IN CANCER CLINICAL TRIALS

FA18-523-80

- the +@ is not done corre tly you forgot the @, see sample.
(Updated to add @ symbol in the two places that it was missing)

Evan Beall
ebbeall@iu.edu
Indiana University
hid: fa18-523-80
github: [ebbeall](#)

Keywords: Python, Visualization, Matplotlib, Cancer, Oncology

24.1 INTRODUCTION

Cancer clinical trial research is an ever-evolving field. The pharmaceutical companies that carry out these trials employ individuals of various backgrounds to carry these trials out successfully. The individuals required to run these trials have backgrounds that range from scientific, business, operational, etc. In order for these various teams to work together efficiently, the business units involved in this pursuit will need to be able to communicate effectively. Individuals from these varied business units will have different specializations and prior knowledge. To accommodate the varied backgrounds between groups communication can be aided by visualizations. Visualizations provide a method that allows all business units involved regardless of background to have a general understanding of research progress.

In the last several years, cancer research data has been ported into

Electronic Data Capture (EDC) systems [269]. These EDC systems function only as databases to have data entry performed and hold data. These systems have no native way of creating visualizations. Due to the lack of native visualization capability, data needs to be extracted and run through software that can quickly and effectively create visualizations of large data sets. One such tool is the Matplotlib package that is available via Python programming.

Cancer research is beginning to utilize big data technology like Matplotlib to help to analyze, standardize, and communicate results. Matplotlib is a open source library within the Python environment. This environment provides simple but extremely powerful 2D and 3D visualizations of large amounts of data [270]. Matplotlib is currently not widely used in clinical trial research but could become an extremely powerful tool within the clinical trial ecosystem. This tool would be especially helpful for scientists and data managers to be able to display their trial progress to those that do not come from a scientific background. The types of visualizations available within this library range from basic scatterplots that require little outside knowledge to interpret; all the way up to EEG plotting capabilities that require specific medical knowledge to interpret. An example of the EEG plotting capabilities can be seen in Figure [34](#).

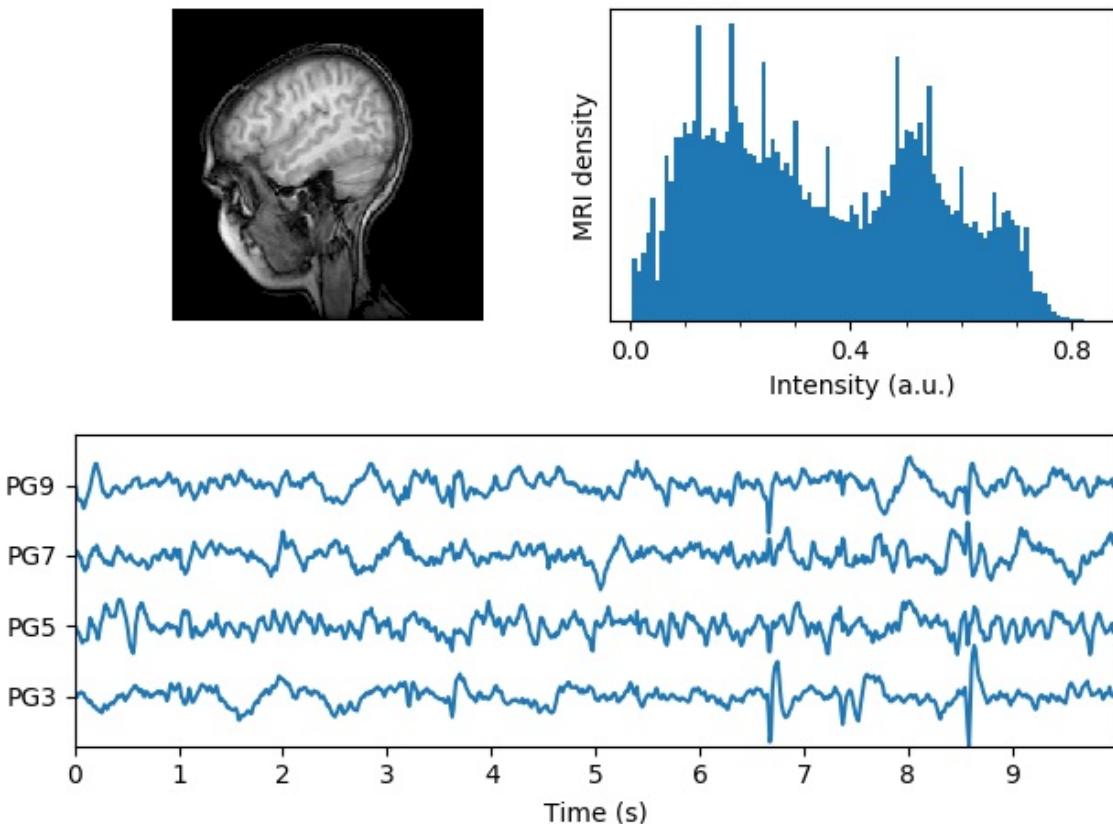


Figure 34: Matplot EEG [271]

24.2 ARCHITECTURE OF ELECTRONIC DATA CAPTURE SYSTEMS

Database structure of an electronic data capture system can vary widely across industry pharmaceutical companies [269]. The data present in each of these systems will vary based on several factors. These factors include: phase of trial, therapy being investigated, type of lesion being research, etc. While a trial is actively accruing and treating patients, the data within the EDC system is unstructured and it is nearly impossible to create comparisons between other trials [269]. The goal of this research is to prove that these new drugs provide benefit to the general public and allow patients to glean those benefits.

In the United States the FDA is an organization in place to guard the general public from harmful and useless new treatments being brought to market. Every new treatment resulting from a clinical trial

needs to submit its data to the FDA to gain approval to market the drug. To accomplish this, the clinical trial research community has created a structure that standardizes data once it has been extracted from the electronic data capture systems used in a clinical trial. This standardization structure is called Study Data Tabulation Model or SDTM [272]. SDTM provides a standard structure for both human clinical and nonclinical studies to be submitted to the FDA for approval. In 2004, SDTM was chosen by the FDA as the standard that would be utilized for all submissions for drug approval [272].

Clinical trials can vary widely regarding what observations are collected throughout the life of the trial. SDTM provides a set of defined variables that each of these observations will need to fit into. Each of these observations are broken down by topic, timing, qualifiers, and identifiers depending on the type of observation that is being assessed [272]. Each observation is then sorted into a domain. Domains are groups of variables or observations that are related by a topic-specific commonality or scientific commonality. In general, each domain correlates to a corresponding dataset, however, some domains can be spread across multiple datasets. Examples of datasets that are used in Oncology clinical trials are: DM (Demographics), AE (Adverse Events), etc [272]. Utilizing SDTM coded databases allows for data coming out of electronic data capture systems to be compared. In turn, this allows for the FDA to compare across clinical trials to assess the scientific backing of each submission to the FDA. The FDA is then able to better analyze the efficacy of the research and if the general public would benefit from having this product available in the American healthcare marketplace. This standardization also allows for comparisons to be drawn between research done all over the world.

24.3 MATPLOTLIB USE CASE FOR CLINICAL TRIALS VISUALIZATIONS

Oncology clinical trial research and clinical trial research in general generates an enormous amount of data. Clinical trials can include

thousands of patients with health data for multiple years at a time. Managing this massive amount of data is not a small task. Thousands of individuals are involved in the procurement, entry, cleaning, and manipulation of these databases before a clinical trial can be called a success or failure. The cost to bring a drug to market is currently estimated to be about 648 million dollars. Along with this, it is estimated that the cost to run one clinical trial depending on its phase could be anywhere between 10 million to 40 million dollars [273].

When this many people from differing backgrounds are working together to carry out each of these trials, it is necessary to have a concise and universally understood way of communicating trial progress and goals. This is where matplotlib's visualizations tools can be an extremely powerful asset. The visualizations created via Python's matplotlib library would provide researchers a way to communicate with other business units in ways that do not require oncology knowledge. Utilizing visualization tools allows for workers with different backgrounds to communicate in a universal manner. Some examples of visualizations that may be helpful during trials would be identifying how many patients are responding to treatment positively, showing a graph of how long patients are remaining on the investigation therapy, or showing a graph of the characteristics of tumors that are being studied. A visualization that plots survival status of patients on treatment can be seen in Figure [35](#) below:

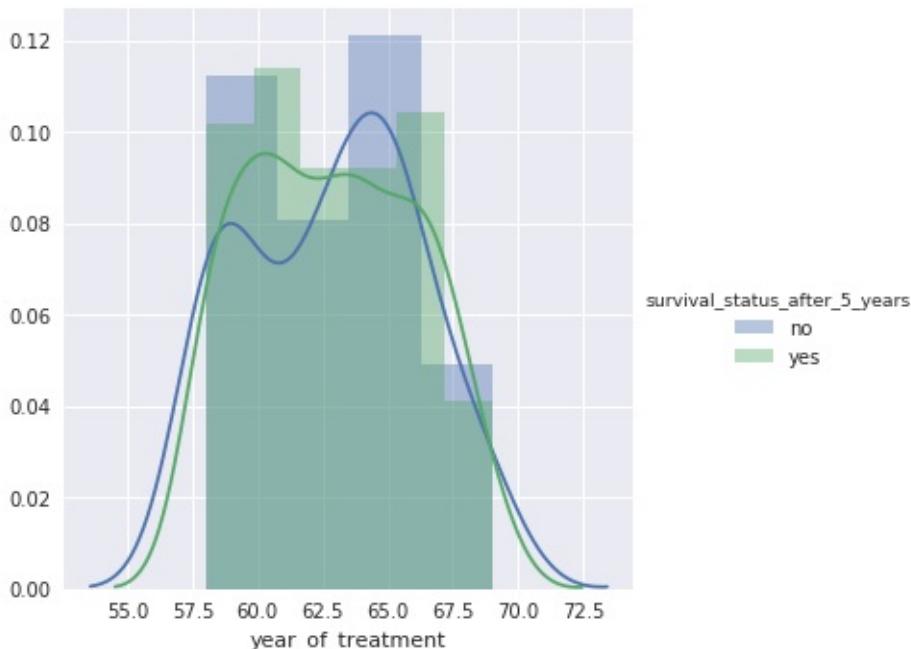


Figure 35: Survival Status [274]

Communicating outcomes such as those seen in Figure 35 early on in the study can inform other business' units decisions. The survival status layout shown in the image above can be interpreted by the medical team to determine patients benefit gained from the study. Medical staff might be able to determine trial futility earlier on in the study utilizing visualizations such as this. These individuals are then able to pass this information on to other internal teams while utilizing the visualization to communicate their analysis. Following this, the budgeting departments might be able to make decisions on where to increase or decrease funding. Increasing the availability of visualizations to everyone involved in a clinical trial would allow for better communication and potentially save millions of dollars throughout the life of a study. These visualizations would provide a method for each business unit to have robust oversight throughout the trial.

Most of the current electronic data capture systems are built in SQL based relational databases. One such SQL based system is an Oracle based database called Inform [269]. Databases like Inform do not provide great tracking or visualization tools for people of all

backgrounds to be able to understand how research is progressing. Most individuals involved in clinical trials research are not well versed in data manipulation, statistical analysis, or querying databases. This is the most critical reason why accessibility of visualization tools would be a large boon to large pharmaceutical companies allowing all business units to be able to communicate with each other during the clinical trial process. Visualization tools will allow for earlier analysis of clinical trial research to occur by all business units such as budgetary, medical, and legal. The cancer research industry uses statistical teams to perform deep analysis, however, utilizing this staff is expensive and time consuming. Each time statistical analysis is run on the entire trial, it requires fully committing statistical colleagues to run analysis on every aspect of the trial. By instead utilizing visualization tools, each business unit would be able to quickly run analysis on specific aspects of the trial. Visualization tools would provide a quick glance at specific areas of interest allowing all business units to quickly determine if the study/research is going as planned. If these units have further questions regarding trial progress, then a full analysis can be performed on the study. This could save time and money for the company as a whole. Also, having a robust oversight plan would allow for mistakes and problems to be caught even earlier.

24.4 MATPLOTLIB ARCHITECTURE

Matplotlib's architecture is split into three different layers. The layers involved in producing a plot with the matplotlib library are the backend layer, artist layer, and scripting layer [275]. These layers are developed in a stack orientation in which the layers can talk to the layers below them, but the lower layers are not aware of those layers above them [275]. The backend layer involves FigureCanvas, Renderer, and Event classes. The FigureCanvas class involves the area where the plot will be drawn. The renderer class actually does the drawing of the plot. Finally, the Event class handles keyboard or mouse events that might occur during the drawing process. The Artist Layer is the middle layer of the Matplotlib stack. Within this layer is

the Artist class. This class allows the user to create and customize the plot. Allowing for the system to understand the drawable area on the canvas and what type of titles and types of plots should be drawn. An example of how the Artist layer functions is displayed in both Figure [36](#) and Figure [37](#)

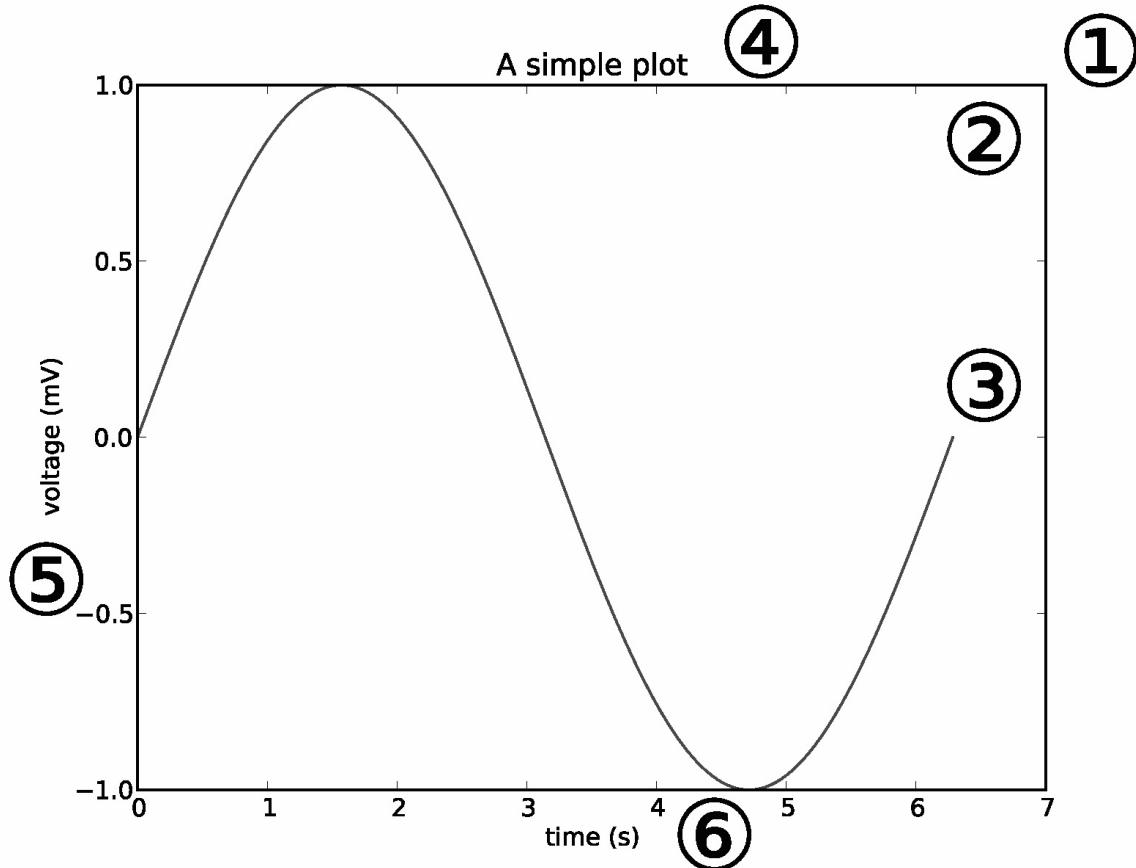


Figure 36: Artist Plot [275]

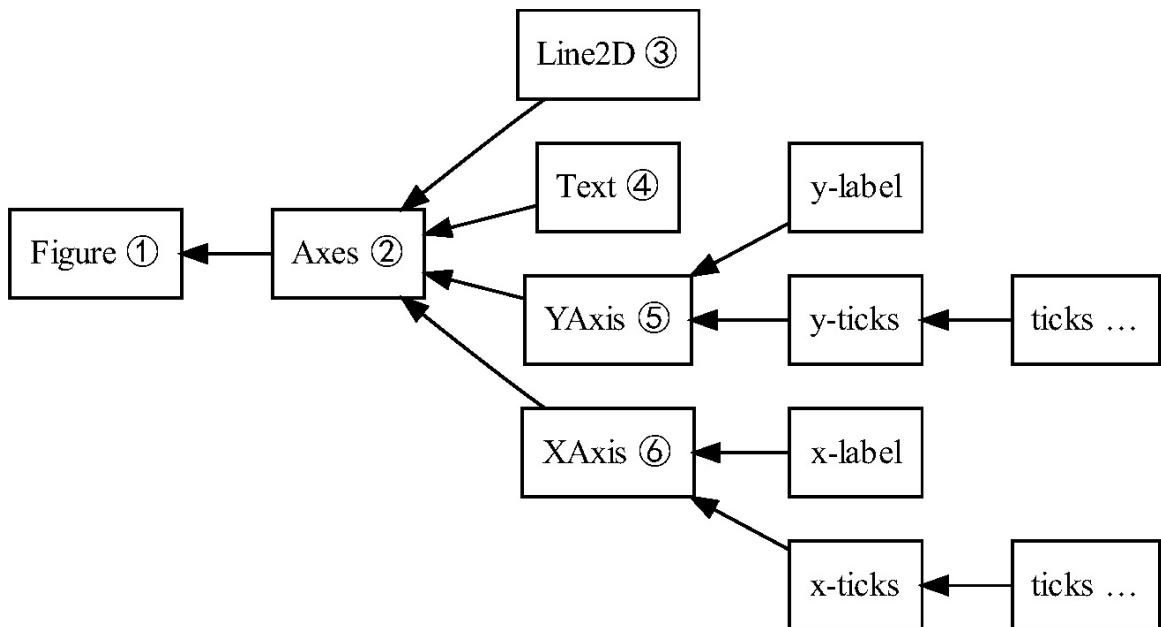


Figure 37: Artist Flow [275]

Figure [36](#) displays the final output plot that is being generated by the Artist layer. Figure [37](#) shows the hierarchy of how this finished plot is built within the artist layer.

Finally, the Scripting Layer utilizes the pyplot API to allow the user to speak to the backend layer and make choices about what is represented in the plot that is created. This is the layer that the user will interact with and create code to influence the other layers of the library.

24.5 MATPLOTLIB FEATURES FOR CLINICAL TRIAL RESEARCH

Matplotlib is a massive library that allows for easy creation of basic plots, while also providing the functionality to create very intricate and powerful visualizations depending on your skill with the library. The most utilized package within the library is pyplot. The usefulness of the Matplotlib library is found in the versatility of visualizations that can be created. Very basic analysis can be run and presented in laymens terms to those individuals that are not versed in clinical trials research (such as the public) or more advanced visualizations can be run for internal teams.

Some plots and features that are available within the Matplotlib package are: line plots, multiple subplots, histograms, data handling, three-dimensional plotting, streamplotting, tables, scatter plots, gui widgets, log plots, EEG plots, etc [271]. There are several applications of the offered matplotlib features that would be extremely useful when running analysis in the middle of a clinical trial. Specific examples are gui widgets. This powerful tool would allow individuals with no knowledge of data handling to model outcomes if specific criteria were to happen. In oncology clinical trials and example would be: if a certain number of patient's cancer progress within a certain time point, they would choose to discontinue the trial. Another visualization that would be helpful throughout these trials would be basic line, bar, and histograms. These types of graphs require the least amount of external knowledge to interpret. With these types of charts it is easy to show how research is trending. Researchers would

be able to visualize how tumor size is increasing/decreasing in the patient population over the course of a trial. Other business units would be able to clearly see if tumor size to understand and make changes based on these results.

24.6 CONCLUSION

Carrying out an oncology clinical trial successfully results in a massive output of data. This data needs to be analyzed throughout the course of the trial in order to ensure goals are being met. By analyzing the data output frequently, it will allow all business units involved to make better decisions to help both the patients and the company carrying out the trial. Current electronic data capture systems do not natively have functionality to create visualizations of the unstructured data within them. However, by utilizing the SDTM standard chosen by the FDA to give the data structure and combining it with the matplotlib Python package these problems could be remedied. The visualization created by the matplotlib package would provide pharmaceutical companies with a great way to communicate both internally and externally throughout the course of a trial. The visualizations available within matplotlib provide plotting functionality for basic to advanced plotting depending on the audience that they are intended for. Utilizing the matplotlib package throughout the course of the trial could allow pharmaceutical companies to make better decisions for patients, budgetary decisions for themselves, and provide progress updates for potential investors. These factors and much more make a great argument to start to implement the matplotlib package's visualization tools into standard practice for clinical trials research.

Harika Putti
haputti@iu.edu
Indiana University
hid: fa18-523-81
github: [blue icon](#)

25.1 INTRODUCTION

IBM Cognos [276] is a business intelligence suite that can help users to provide powerful insights to drive better business decisions. Business intelligence is an all-encompassing term that includes tools, infrastructure and applications that help in analyzing, evaluating and visualizing data. With help of BI software, one can envision relationships within the data that help organizations make knowledgeable business decisions. Using business intelligence tools, organizations can integrate their data and create reports, dashboards, metrics and scorecards to gain insights. Business intelligence suites by definition need to incorporate techniques like data processing, data modelling, querying, visualizing among other things. IBMs Cognos Business Intelligence is one among the many suites that has an extensive set of options ranging from exploration, modelling and querying to data visualization. The Cognos BI suite has multiple components including report studio, event studio, metric studio, framework manager, workspace among many others.

Keywords: hid fa18-523-81, business intelligence, BI, cognos, analytics, reports, queries

25.2 HISTORY

Cognos was a consulting and performance management company that was founded in 1969. It was acquired by IBM into its Infosphere product line in 2008 when companies like SAP, Oracle, Microsoft were fighting to become leaders in the BI market. In 2005, the company had released its Cognos 8 suite which introduced tools such as the Report studio, Query studio, Analysis studio and many others. After the acquisition by IBM, IBM Cognos 10 was released that had the capability to incorporate SPSS predictive analytics, historical and real-time analysis, better, faster and more flexible way of generating reports and dashboards. The next version of Cognos was the IBM Cognos Business Intelligence 10.2.2 that had the ability to integrate Microsoft Office with Cognos [277]. The latest edition of Cognos is the IBM Cognos Analytics 11.0 which is a state-of-the-art Analytics tool. This version of Cognos is a very powerful BI tool with ability to connect to Hadoop, an in-built AI assistant and smart data-discovery

25.3 ARCHITECTURE

Cognos has a 3-tiered architecture. Each tier separated by network firewalls.

- Tier 1: Web servers and Gateways
- Tier 2: Applications
- Tier 3: Data and Content store

Figure [38](#) shows the architecture for the Cognos Business intelligence suite.

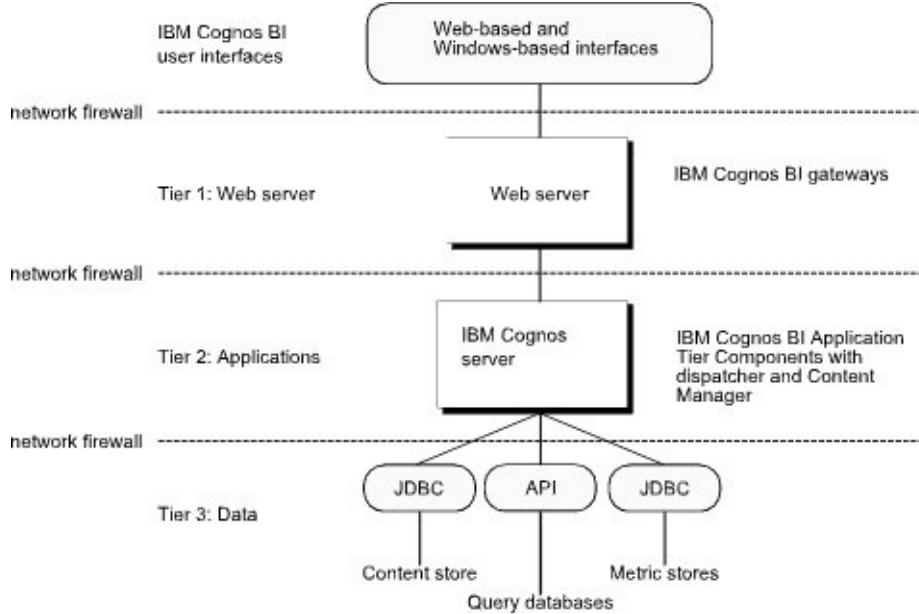


Figure 38: Architecture for cognos [278]

25.4 COMPONENTS

The various components of Cognos Business intelligence are [279]-

Component	Application
IBM Cognos Connection	Publishing, managing, and viewing content
IBM Cognos Insight	Managed workspaces
IBM Cognos Workspace	Interactive workspaces
IBM Cognos Workspace Advanced	Ad hoc querying and exploring data
IBM Cognos Report Studio	Creating reports
IBM Cognos Event Studio	Event management and alerting
IBM Cognos Metric Studio	Metrics and Scorecards
IBM Cognos for Microsoft Office	Cognos in Microsoft Office
IBM Cognos Query Studio	Ad hoc querying

IBM Cognos Analysis Studio	Exploring Data
IBM Framework Manager	Creating metadata models
IBM Cognos Transformer	Multi-dimensional data modeling

25.4.1 IBM Cognos Connection

IBM Cognos Connection is a web-based user interface that is used to access various cognos services such as Query Studio, Report Studio, Analysis Studio, Metric Studio etc. Using IBM Cognos connection one can access all the available reports and perform different operations such as create, run, schedule and access the reports. The admin has the ability to set up access permissions, manage data sources and provide individual and group memberships with in the interface. Users can personalize the connection as per their choice.

25.4.2 IBM Cognos Insight

Insight [280] is an individual entity with-in the Cognos family that incorporates a range of analytic abilities like ad-hoc querying and analyzing what-if scenarios. It is very user-friendly since one can perform data analysis, do off-hand querying, create dashboards with utmost ease. It's almost like tableau where you can drag and drop the data files and insight automatically creates crosstabs and charts using something called smart metadata. Insight has the ability to create natural hierarchies between the data it receives to create OLAP cubes which makes it easier for the users to slice and dice the data as they prefer. Insight also comes with write back functionality that provides the ability to create models using prior data or new data that the user provides.

25.4.3 IBM Cognos Workspace

Workspace [281] is a web-based tool that can be used to create interactive dashboards known as workspaces using existing or new reports with in the IBM Cognos connection. This allows the users to create insightful visuals that can convey as much meaning from the

information at a time as possible. It contains widgets and filters that users could use while creating the dashboards.

25.4.4 IBM Cognos Workspace Advanced

Business Intelligence is an area with a lot of options to choose from i.e. which studio, which data package etc. Most of the Cognos BI users are perplexed about the time wasted in navigating between different studios. Cognos Workspace Advanced was designed to bridge the gap between various studios. With the advent of Cognos Workspace Advanced, the redundancy of having three different studios is clearly noticed [282]. The advanced Cognos was designed in a way that it can amalgamate all the services on one platform where the users can use multiple services at the same time. It provides a single interface for querying and analysis and is very flexible in terms of presentation options.

25.4.5 IBM Cognos Report Studio

Report Studio [283] is a report composing instrument that proficient report creators and designers use to fabricate complex, various page reports against numerous databases. It's the most used component in the IBM Cognos Business intelligence suite. With Report Studio, you can make any reports that your association requires. Report Studio offers a variety of services such as creating and formatting report using grouping, headers, footers, and other formatting options. Report Studio also enables focusing reports by filtering data and using prompts. The report studio also aids in adding value to the reports by performing different manipulations. It enhances the visual appeal of the reports though advanced formatting and exceptional data highlighting.

25.4.6 IBM Cognos Event Studio

The Cognos Event Studio [284] is predominantly used to keep track of the events in an organization to ensure that the decision-makers are notified of the upcoming events to make timely and effective

decisions. An event is generally a situation that can create some impact on the business. When there are significant changes in the data, an event is detected to be taking place and agents are placed within this framework to detect the occurrence of events in organizational data. The sole purpose of an event studio is to notify the decision-makers about an event which is otherwise inconspicuous. When these agents are activated by the changes in data, the activation gets passed as notification to for further action such as sending an e-mail, adding information to the portal and running reports.

25.4.7 IBM Cognos Metric Studio

This service of IBM cognos helps to manage the performance of an organization by measuring the metrics at all the levels of the organization through creation of scorecards. The sole purpose behind creating score cards is to put performance indicators alongside the organization's main performance measures. These scorecards then can be used to link to reports containing related information. These score cards can be customized using metric studio which helps in monitoring and analyzing metrics and projects throughout the organization [285]. Metric Studio helps in converting an organization's strategy into relevant and measurable goals. That align every employee's actions with a strategic plan. An environment rich of scorecards is analogous to a quick review document that shows how successful is an organization and where the flaws are to work and improve upon. It helps the decision-makers at all levels to react and plan based on the reports generated from the comparison of performance against targets while also simultaneously making a note of the current status of the business.

25.4.8 IBM Cognos Query Studio

This service of IBM Cognos is preliminarily for people with little or no training, one can quickly design, create and save reports that are not covered by the organizational reports. One can use query studio for ad-hoc reporting and can view data in hierarchies, create crosstab

views and filter, sort, suppress and group the data easily without having to create any complex reports. One cannot define the properties of a data object like we can do in report studio, one cannot create multi-query or multipage reports either. Report studio offers a lot more visualizations and templates. Query studio is generally used by customers to quickly create reports that can be used as a reference to create higher level reports by the developer [286].

25.4.9 IBM Cognos Analysis Studio

Analytics studio [287] can be used to intelligently manipulate the data to understand the relationships within it. It provides support for filtering, calculating, sorting and analyzing the data. It can be used to comprehend patterns and inconsistencies, look at information, for example, points of interest to outlines, or real outcomes to planned outcomes evaluate execution by concentrating on the best or most exceedingly bad outcomes. It helps in multidimensional examination and investigation of expansive data sources. IBM Cognos Analytics is intended to enable one to report and dissect an organization's performance rapidly and effortlessly.

25.5 Cognos Analytics

Cognos Analytics [288] is a futuristic tool which can provide analytic solutions with ease. It's very user-friendly and has the ability to perform machine learning, pattern detection and smart visualization. It also enables sharing visualizations and reports on platforms like slack.

25.5.1 Big data and Cognos

With the increase in the amount of information that an organization can gather, the need for an integration between big-data and business intelligence tools has also increased. IBM having recognized that, created the possibility to connect to Hadoop databases.

25.6 CONCLUSIONS

In conclusion, Cognos is IBM's business intelligence tool that can be used for a thorough scrutiny of performance of a business. This product is intended to empower business clients without programming prowess to extract corporate information, analyze and create reports using that information. Cognos empowers even a layman to create professional reports which are otherwise created only using years of expertise. Cognos provides exceptionally good implementation and deployment options that support scalability to grow along with the organization. It integrates well with other applications such as its seamless ability to integrate email and pdf functionality thereby allowing one to schedule reporting data deliveries. Cognos facilitates connection to multiple databases which can be stored within cognos' content store.

26 IBM WATSON CONSTRUCTION AND ITS SERVICES

FA18-523-82

Pavan Kumar Madineni

pmadineni@iu.edu

Indiana University

hid: fa18-523-82

github: [🔗](#)

some text reads like advertisement, slight improvement in grammar desired

26.1 INTRODUCTION

IBM Watson [289] is basically an artificial intelligence that is bringing rapid changes to the way the world works while simultaneously making businesses faster, smarter and more secure. This AI system is helping businesses utilize artificial intelligence to work at scale providing unparalleled business advantage. As the world continues to become more social, the data is ought to grow, and competitive advantage is to those who utilize the data better than their peers and can directly connect it to their business outcomes and other useful pursuits. Watson enables businesses to personalize customer experiences by streamlining processes, minimizing risks associated and kindling innovation. Watson is helping millions of engineers to seamlessly process huge volumes of data across different disciplines of a business thereby aiding to predict the decline in business, point of break down and proactively fixing them. Therefore, to play a competitive role in the field of business, there is an immense need for tools and processes, which help to collect all the required data generated across numerous platforms easily, to store, manage, manipulate, aggregate, analyze and integrate the data which help in making insightful business decisions. Watson has its services spread across varied disciplines from healthcare to automotive to telecom to education. It helps banks to deploy artificial models that act as virtual

agents trained on thousands of customer inquiries helping them to provide expert service to large number of customers at one and a half times faster pace. Watson not only ensures faster pace of working but also transforms workflows.

Keywords: hid fa18-523-82, Artificial Intelligence, Machine Learning, Watson, Transfer Learning, Data Visualization

26.1.1 Watson's Methodology of Working

Watson ensures the work is accomplished in a smarter way by centralizing the data which enables the teams and business to connect to data whether the data is on the storage tapes within the organization or in the cloud or on any online file storage without any disruption and hassles [290]. With each day passing, the businesses are relying more heavily on artificial intelligence. So, the businesses demand an intelligence system that does not compromise on transparency of operations to confirm if the recommendations given are trustworthy. Watson is committed to provide the expected transparency through not making any biased recommendations while simultaneously ensuring the variance in the recommendations is also within the permissible limits. Watson as an intelligence system follows three main thumb rules in order to increase the ease of doing business; they are reducing disruptions, enriching customer interactions and making confident recommendations. All these features make Watson the most sought-after intelligence system owing to the ability of Watson to deliver smarter and more productive work [291].

26.1.2 Role of Artificial Intelligence in Building Watson

Artificial Intelligence is the process that gives a machine the power to learn adapt to new inputs after thoroughly analyzing the already known inputs along with their solutions to make better informed decisions. Machine learning is a branch of artificial intelligence that

uses computer algorithms to analyze the trends in the data and make smart and intelligent decisions based what these algorithms observe in the data. This is how numerous machine learning models are built on numerous platforms across numerous disciplines [292]. A machine learning model is generally of the form of an input output device that takes in a few observed samples of inputs and provides an output after implementing pertinent computer algorithm on these inputs. A sample machine learning model can be to predict flu outbreaks based on the volume of tweets mentioning flu-related keywords, recognizing the patterns in human mobility by analyzing the mobile phone call records, or a future forecasting model that can be used to forecast the financial success of a movie by studying the page views statistics of the Wikipedia articles about a movie. All these models or predictive examples in common illustrate the concept of quantifying and measuring human activity at a collective level to understand and build better human societies through a computational framework. Watson broadly falls under one such computational framework that works on real data to make informed decisions and build smarter and more secure human societies [293].

26.1.3 Role of Machine Learning in Building Watson

Among the existing machine learning techniques, deep learning is probably the most sophisticated machine learning technique. Watson utilizes deep learning framework to create an artificial neural network that can perpetually learn from various inputs determining whether decisions made are correct while constantly improving the quality and accuracy of results. This is what enables Watson to learn even from unstructured data that is available from the society such as photos, videos and audio files. Deep learning also enables Watson's natural language understanding capabilities thus allowing it to learn by deconstructing sentences and then analyzing and identifying the concepts and underlying relations of those sentences. Once these relations are discovered, it can decipher the context and intent of what the original sentences would want to convey [294]. Any artificial intelligence in general also needs to understand the specific language and terminology in order to decipher the jargon pertinent to that

particular domain and industry but this process is absolutely cumbersome using the traditional artificial intelligence models and also requires huge volume of data and computing power to run these algorithms unintermittently for long durations. Watson however simplifies this process drastically by applying a technique called transfer learning.

26.1.4 Role of Transfer Learning in Building Watson

Transfer learning is a machine learning technique where a model trained on one task is re-trained and on another related task. This process ensures improvement of learning in a new task through the transfer of knowledge from a related task that has already been trained and learned. Transfer learning also reduces the overall training time by alleviating the need to train the algorithm from scratch which can be achieved by feeding Watson with knowledge from an already trained model. Watson's transfer learning architecture comprises a three-layered artificial intelligence model. The bottom layer constitutes an out-of-the-box general knowledge like Wikipedia for artificial intelligence. This layer provides the model the basic knowledge about the domain the model is trying to get trained and learn. The middle layer is prepackaged with knowledge requisite to specific domains and industries. This layer takes care of jargon specific to domain thereby removing ambiguity associated with terms which have different meanings in different contexts. The top layer is where personalized learning takes place [295]. The model will be fed with all the training data the model is intended to learn. The model is now potentially knowledgeable to understand a company's specific risk and behavioral attributes. Transfer learning is thus an integral part of how Watson is able to accelerate business operations at a rapid pace thereby dramatically reducing the operating costs.

26.2 IBM WATSON AND ITS SERVICES

IBM Watson is one such intelligent data analysis and visualization

service on the cloud that lets anybody pose different questions and provides answers to the questions posed in natural language [296]. IBM Watson provides all the tools and services necessary to work with your data and build machine learning models at one place which makes analyzing data much simple be it a novice or an expert. IBM Watson offers numerous services namely Watson Analytics, Visual Recognition, Natural language Understanding, Speech to Text, Text to Speech, Tone Analyzer, Language Translator, Machine Learning etc [291].

26.2.1 IBM Watson Analytics

IBM Watson Analytics is one such service that enables extracting intricate data flow processes as well as convoluted relationships between different fields of data. The analytics services offered by IBM Watson not only enables you to discover novel insights about an organization but also swiftly create and share highly informative and illustrative dashboards and infographics. The Watson Analytics is also capable of analyzing data directly from social platforms like Twitter and output the sentiment of the customer or learn the best time to tweet to enhance the span of the audience. It also enables multiple users to collaborate and share visualizations and dashboards with each other.

26.2.1.1 Watson Analytics on Health Care

One of the few domains that extensively needs an artificial intelligence system about which everyone is concerned about is health care. The current approach to tackling different health problems today is flawed and grave concerns are exhibited over improvements in health care industry. Watson has partnered to build solutions that shall allow the larger health care community which involves both individual patients as well as larger health populations to be benefitted as the participants share and apply data-driven insights in real-time [297]. The medical data is growing exponentially each year with data flowing in from numerical sources such as medical and clinical research, various sensors, personal fitness

trackers etc. The health industry is unable to keep up with this staggering growth of medical data. The IBM Watson health cloud brings together huge volumes of medical data into one centralized platform on the cloud. This data is then used to apply a fusion of traditional analytical techniques and Watson's advanced machine learning techniques to churn out valuable insights out of it. This is possible because of ecosystem environment in which Watson Health Cloud operates i.e. it has multiple contributors to keep the system functioning smoothly. Watson Health Cloud uses huge volumes of data, knowledge pertaining to data and perspectives and opinions of researchers and domain experts as contributors to its ecosystem. Watson has customized this ecosystem even for personalized use which make the health cloud more dynamic and efficient. Watson is also well-known for the way it handles highly confidential data such as patient related information. It makes sure to remove all personal identifiers associated with the data that is being uploaded to the cloud. This process is called De-identification which is very crucial to create safe and secure cloud environment [298]. Watson's extraordinary machine learning abilities combined with its interactive ecosystem as well as its secure de-identification has made it one of a kind health care analysis tools which is transforming the health care industry substantially.

26.2.2 IBM Watson Machine Learning

IBM Watson Machine Learning is another powerful service which is integrated to IBM Watson Studio that enables users to perform two fundamental operations of machine learning i.e. training and scoring. Training is the process of teaching an algorithm the underlying trends and behavior of data by feeding labelled data to the algorithm. A trained algorithm learns coefficients of mathematical expressions which represent the behavior of the data in the best possible way. Scoring is the process of predicting an output using the coefficients learned from the trained algorithm which is otherwise called as a predictive model. These predictions made by the algorithms after scoring enables data scientists to collaborate with data engineers to further explore the data and gain deeper insights out of it.

26.3 CONCLUSION

To sum up, IBM Watson is an ideal tool if one wants to perform their own analysis but perplexed where to begin. IBM Watson is a smart data discovery tool that enables you to leverage all the state-of-the-art data analytics and visualization techniques to draw valuable insights out of data almost instantly. It automates the processes such as data preparation, predictive modeling, and data visualization which are otherwise very hectic and tedious processes [299]. Watson is today considered the best artificial intelligence for most of the enterprises. It facilitates faster learning from smaller chunks of data. It is explicitly designed to make sure that data flows only from bottom to the top so that this structure aids in transfer learning process by sharing knowledge in the bottom two layers while still holding a firm control over the top layer.

27 SMART HOME IoT SENSORS FOR RASPBERRY PI

FA18-523-84

Adam Hilgenkamp

ahilgenk@iu.edu

Indiana University - Bloomington

hid: fa18-523-84

github: [blue user icon](#)

incomplete

Keywords: IoT Sensors, Raspberry Pi, Thermostat, Smart Home

27.1 PROJECT LOCATION

As an alternative to a formal paper I am contributing to the Raspberry Pi epub. A link to the contribution can be found below.

[Sensor Contributions](#)

27.2 OUTLINE

27.2.1 Contribute to sensor specific sections

- DS18B20 Temperature Sensor
- DHT11 Humidity Sensor
- 5v Relay Module
- Light Sensor
- Touch Sensor
- LCD Screen

TODO

- Motion Sensor
- Sound Sensor
- Maybe sync to Alexa?

27.2.2 Sample Project

- Learning Objectives: Understand how different IoT sensors work. Integrate sensors into an application.
- Introduction: Introduction of what the project will be building
- What you need: list of items needed to replicate the project
- How to set up each sensor: Overview of how the collection of sensors is wired with reference to wiring for each individual sensor.
- Using the sensors in an application: The code and set up needed to read the data and do something with it. Will have all the code in the document.
- Conclusion and next steps: Reference to setting up cassandra database and building an application to display the data in charts.

28 BIG DATA ANALYTICS IN E-COMMERCE

523-85

FA18-

Bo Li
bl15@iu.edu
Indiana University Bloomington
hid: fa18-523-85
github: [blue icon](#)

Keywords: E-commerce, Consumer Behaviors, Python, TensorFlow, Big Data, Deep Learning

This paper is too short to justify an abstract

This paper has very few references. We think this topic is there not more out there

maybe ask someone to help with grammar

28.1 ABSTRACT

In recent years, online shopping has become a more popular way of consuming. Traditional retailers are eager to find a way to maintain revenue, while online retailers are also finding ways to extend the market. The rise of 'big data' has impacted on marketing research and practice a lot. As technology developed, we have huge data on consumer behaviors which could be very detailed and accurate, but how to mine the data is a problem. In this article, we talk about the application of big data in the consumer behaviors data, subsequently discuss how the TensorFlow can translate the data into valuable conclusions in consumers behaviors research.

28.2 INTRODUCTION

The Big Data has a common definition, Big Data always comes with 3V: volume, velocity, and veracity. The Internet allows us to do almost all work online and keep records of our actions. If you listen to a song in the playlist, maybe iTunes will record it as part of your individual activity log, which could be the dataset that explores your interest. If you often use Uber to commute between your home and your company, maybe they could picture your daily life including the places that you have spent time in. If you use your device to safari on the internet, your action of clicking on several links could also be recorded and researched since the action contains you using habits and preferences [300]. The online shopping data includes the consumer's all kinds of information: age, job, education, catalog preferences, price sensitivity, etc. But some of them are not presented to us directly, mining the consumer behaviors is an appropriate way to get access to the hidden information. How could we mine something meaningful to explore consumer behaviors and provide valuable insight? The answer lies in several using cases and the understanding of market research and also human psychology research.

28.3 TECHNOLOGY BACKGROUND

TensorFlow is an open source software library for numerical computation using data flow graphs [301]. TensorFlow could help developers to transform from code to graph, which could benefit developer in understanding their work, and the term tensor, is generated in the process, as the tensor will go from the beginning to the end of the graph, so the technology is called TensorFlow. The process of computation could be done in CPU or GPU, as we know, in blockchain, GPU works better than CPU since the fundamental design of GPU fits better in the computation of mining coins. TensorFlow also has a data visualization module called TensorBoard, which contains the common drawing tools as well as some useful templates for the developers to visualize their data. There is no doubt that the graph will be more clear than codes especially when the structure of data is very complex. And the graph could give the readers a direct

presentation of the data, which is worthwhile since it could reduce the communication cost between different developers.

There is a team in Google called Brain team, which is the initial developer of TensorFlow, there original purpose of the development of this module is to improve the efficiency of machine learning. For example, if the deep learning task is to predict a result based on a training dataset, the more layers you have, the more accuracy you will have. But more layers will cost much more time, so TensorFlow is created to solve the problem. One more important thing is that in the previous version of TensorFlow, it began to support distributed computing which means more resources could be deployed in the process so the efficiency will be boosted.

To support more developers, Python API and C APIs are also available in TensorFlow. One thing that must be pointed out is that, although TensorFlow supports many kinds of languages, the Python API is the most efficient one since Python does better in the feedback process which focuses on improving the model. And there are also more examples in Python since most of the machine learning work is done by Python.

28.4 BIG DATA APPLICATIONS IN E-COMMERCE

Since the design of the website is getting more complex than before, the users may conduct different operations in different pages of the website, but most of them are very import to provide an essential information for us to find the customer's preferences. >"There is a way to achieve that which is called four rights. Talk to the right audience, through the right channel, with the right message, at the right time" [302].

"Customer acquisition: Marketing will target high-value customer segments identified by behavior analytics and study behavior patterns to determine the best potential offers. Customer engagement: Behavior patterns will be used to generate personalized next-

best, cross-sell and up-sell offers, while behavioral customer segmentation will be used for more general customer marketing offers. Customer retention: Behavior patterns will be used to detect possible customer churn and generate next-best retention offers" [302].

The strategic meaning of big data is that deploying professional analysis on those meaningful datasets generated from the E-commerce trading. The improvement in the value of data is the most important part of the benefit of using big data, especially compared with the previous situation, that the data useless since the huge amount. Some meaningful things are hidden behind the big data, mining them is the main task of the application. In the E-commerce domain, users have generated huge amount data about their every action: browsing products, clicking on the details of products, adding the products into the wishing list, adding the products into the cart, delete the products from the wishing list, clicking dislike product button and querying more information to the seller of the products. Such information has been stored in the log document, but it is too massive and fragmented to analysis it with limited technologies and techniques.

Big data could help the enterprise to have a more profound understanding by analyzing users behavior data, which allows the enterprise to establish strategies with more specific aims. This could make the enterprise to be competitive in the market and win more consumers' hearts. For example, the user may want to buy a guitar for himself, and he has browsed several kinds of guitars and could hardly to make his decision. The big data technologies could base on his operation history to find his acceptable price level and the specific version of guitar (such as with or without pick up system), then we can put such requirements into our database to find the appropriate guitars and push those potential choices to the user's interface, which could realize an improvement in sales transformation rate.

TensorFlow could be a kind tool to analyze consumers behaviors

since the model of recommendation is doable in TensorFlow. Due to the feature of distributed computation, the efficiency of the model is good enough for a small scale recommendation system. To have a more profound understanding of user behavior, a whole lifecycle of the user is needed to be established for the analysis of user behavior.

28.5 FEATURES OF USER BEHAVIOR IN THE E-COMMERCE PLATFORM

The online platform is the main platform for e-commerce which lists the product in different ways and provides the whole chain of finishing the trade. Different from traditional commerce, online e-commerce has some special features which could be used in the big data.

There are fewer limitations for consumers in the B2C pattern since the online platform can run for 24 hours if it is well maintained. Consumers are able to conduct any operation (browsing, selecting, finishing the trade) at any time in anywhere.

The trading cost is much less than the traditional commerce pattern. For the consumer, time cost, transportation cost and delivery cost are lower than the traditional commerce pattern. Their trading action is much simplified by the online shopping system, the trade can be done by several clicks on the mobile device.

The online product can offer a more attractive price due to the advantages of the internet. Comparing with the traditional commerce, online sellers have less item to pay for maintaining the shop. There is a lot of extra costs for the real store.

Customized service. The recommendation system is able to recommend the most wanted goods for each customer based on their user behavior and the big data technology, which is the traditional commerce cannot achieve due to the cost. The customized service can benefit a lot in the transformation between browsing and buying.

More kinds of product and no space limitations. Since the information of the product is much smaller than the product itself, and the online store can exhibit them all at the same time, so there are more choices presented for the consumers.

The information is easy to get. Every item in the online platform has been labeled by the system, so the search of the item is very convenient for consumers, the cost is significantly lower than the traditional commerce.

28.6 APPLICATIONS

Due to the hotness of machine learning and deep learning, there are a lot of applications in every domain. Search ranking and recommendation are the most common two applications. >“Recommendation systems in particular benefit from specialized features describing past user behavior with items” [303].

Just like search ranking, recommendation systems also have a problem of the balance between memorization and generalization. Memorization can be seen as the representing of the relationship between the products and users, which can be extracted as vectors. Generalization is to generate rare feature combinations in order to serve for the recommendation systems [304].

TensorFlow, with so many advantages in machine learning, is very appropriate for the recommendation system. Since the features of products could be learned by multi-labels classification, and the user's features could be learned in his historical actions in the online platform, which has the record of every consumer's trading history. When we have both features of products and users, we can establish a recommendation system by matching the two objects. Besides, the model can be judged by the dot product between the two vectors.

To establish such a recommendation system, we need to fit the TensorFlow since it is tensor that flows in the whole model. The transformation from dataset to tensor is a necessary step to conduct

the model. And the users', as well as the products' character tensors, need to be transformed into the presentations of users and products by the embedding function. The next step is to generate the recommendation by the pair the presentations of users and products. Such pair of presentations contains the most match user and product calculated by the model, the vectors in the model contain all the information of the user and the product. The last step is to compare the generated score and the actual comment from users to define the result's quality, which is called loss function [305].

TensorRec scores recommendations by consuming user and item features (ids, tags, or other metadata) and building two low-dimensional vectors, a "user representation" and an "item representation". The dot product of these two vectors is the score for the relationship between that user and that item—the highest scores are predicted to be the best recommendations.

The representation function in TensorRec can be set up by developer's preferences, it could extract the features of users as well as products. It can be very convenient for developers to set the parameters independently since the scenario varies in different cases [305].

28.7 CONCLUSION

Information is booming in recent years, data and internet techniques are spreading in everywhere, with the significant effect on consumers' deciding pattern and purchasing pattern. The digital economic on the internet has become the focus of all the domain. For the biggest group in the digital economics in the internet, online consumers are the focus in the specific domain. How to draw the picture of the users and get the key feature of their behaviors have become a hot topic.

Benefitted by the internet, we have all the records of most of the online activities of users, but the relationship between those actions and the user's features is ambiguous. Basing on the TensorFlow, we are able to use a low-dimensional vector to represent the user's

features as well as the products' features. The algorithm allows us to extract the key point of users as well as products, which provide a base for the recommendation of the product.

The big data technology can help us to mine the black box of the relationship between the actions and the features. Several factors which measure the user's preferences can represent the user, and those factors are also the key parameters in the model. Once we get a clear picture of the user, we are able to customize the recommendation, which can not only improve the user's experience and also improve the revenue of the online retailer.

Jeff Liu
liujeff@iu.edu
Indiana University
hid: fa18-523-86
github: [blue user icon](#)

29.1 INTRODUCTION

ERP (Enterprise resource planning)[306], a systemized management theory based on information technology, has become into an important and popular modern enterprise management tool for providing a management platform for decision-making operations for enterprises. A good ERP system, it is just a set of software but a management idea. It can not only fully adapt to the management and business processes of the enterprise, but also achieve rapid deployment and challenges in technology. SAP)[307] is a leading ERP software, most of the world's top 500 are in use, although the SAP license, maintenance updates and related training will cost a lot of money, but it comes up with the improvement of operational efficiency and information processing cost saving, so SAP becomes a first choice for the business operation of large enterprises. SAP (Systems Applications and Products in Data Processing) is ERP software, from the back end to the company management level, from the factory warehouse to the storefront, from the computer desktop to the mobile terminal, SAP provides ERP solutions, and can provide comprehensive services for enterprises of various industries and different levels.

29.2 IMPLEMENTATION

SAP Modules and Functions: There are 2 Types of SAP ERP Modules. Number one is Functional Modules and second one is Technical Modules. All SAP Modules integrated with each other with

functionality and provide us best solution for Business[308]. Most important SAP Modules that Bunnies implement for their business are

1. SAP FICO module
2. Human Resource Management (SAP HRM), also known as Human Resource (HR)
3. Production Planning (SAP PP)
4. Material Management (SAP MM)
5. Financial Supply Chain Management (SAP FSCM)
6. Sales and Distribution (SAP SD)
7. Project System (SAP PS)
8. Financial Accounting and Controlling (SAP FICO)
9. Plant Maintenance (SAP PM)
10. Quality Management (SAP QM) security module[309].

SAP Functions:

1. SAP Business Objects provides comprehensive business intelligence capabilities that give users the ability to make effective and informed decisions based on solid data and analysis results. All users from high-level analysts to ordinary business users have access to the information they need, with less IT support[310].
2. SAP CRM can help you reduce costs and improve decision making while helping enterprises differentiate to gain a long-term competitive advantage. It helps to increase the competitive advantage and bring higher profits[311].
3. SAP Business Objects Information Management provides comprehensive information management capabilities to help deliver consolidated enterprise data in a timely and accurate manner, both structured and unstructured. it helps users provide data for key action plans such as business transaction processing, business intelligence, data warehousing, data migration, and master data management
4. SAP Business Objects helps you leverage the value of your company's data and make your business more agile and competitive by increasing your organization's collaboration, insight and confidence[310].
5. SAP ERP is one of the top five suites of SAP Business Suite and the most powerful core suite of SAP in the market. The SAP ERP application software supports the basic functions of the business process and operational efficiency of

the enterprise and is customized to their specific needs [310]. 6. SAP HR supports the entire process of recruiting, deploying, developing, motivating, and ultimately leaving valuable employees, improving these processes from the beginning to the end[312]. 7. SAP PLM, one of the core suites in the SAP Business Suite, provides collaborative engineering, custom development, project management, financial management, quality management and more throughout the product and asset lifecycle. 8. SAP Supply Chain Management is a member of the SAP Business Suite. The suite uses modular software that works with other SAP and non-SAP software to enable organizations to perform basic business upgrades. 9. SAP Supplier Relationship Management SRM is a sub function of the SAP Business Suite business application. This integrated suite expands the value of SAP Business Suite by automating the process of commodities and services from purchase to payment[311].

29.3 CONCLUSIONS

In summary, SAP is an ideal EPR tool for big companies. SAP system is very expensive, but the system is also very powerful, and can be adjusted differently for each customer's different needs. First, SAP's did good at customer management and carefully examine the customer's relevant information, asset status and so on. Secondly, SAP's most powerful and outstanding function is sales management, from order opening, process determination, cost analysis, performance tracking, delivery arrangements, accident handling, payment tracking, a series of powerful and powerful Features.

30 TBD FA18-523-88

format incorrect

github: [cloud](#)

author missing

use the md format

check the TOC

31.1 INTRODUCTION

Each organization has unique needs when it comes to Big Data. These needs cannot be described with one defined structure alone, and likewise, the information that they use does not come with defined data types. Because of this, there is the need for the Big Data Platform. Big Data is gaining more popularity because of its ability to connect to a number of devices in the so-called Internet of Things (IoT), producing a huge dump of data that needs to be transformed into information assets. It is also very popular to buy additional on-demand computing power and storage from public and private clouds to perform intensive data-parallel processing. These things not only create the way for Big Data expansion but also boosts security and privacy issues. Big Data security is the process of securing data and their processes both within and outside the organization. Big Data deployments are valuable targets for intruders and, because of this, security becomes a never ending concern for any organization. A single unauthorized user gaining access to an organization's big data could in and of itself acquire all the valuable information that the company possesses which could result not only in monetary loss but also be detrimental to its business and to its brand name. In current trends, security teams work towards continuously monitoring networks, hosts and application behavior across their organization's data. Traditional methods of securing firewalls are no longer enough to secure a company's data assets and Big Data platforms need to be secured with a mix of both traditional and newly developed security tools, as well as big data analytics for monitoring security throughout the life of the platform [313].

31.2 WHAT IS BIG DATA

Big Data, by definition of its name, is an extensive variety and heavy volume of data that can be entered or transferred at high velocity, and include data sets coming from dynamic sources of data and applies technologies to analyze these data sets. It is a term usually used to define huge and complex data sets that do not fit into any traditional system. Most recently, the term Big Data tends to refer to the use of predictive, user behavior analytics, or certain other advanced data analytics that extract value from data sets. These analytics provide more insights about the data which indeed help businesses understand their trends which will eventually, in good theory, help their growth [314].

For example, a company that works with waste management, can collect data on the waste production and human activities from very diverse sources, then interpret the findings of Big Data to make optimal decisions [315].

31.3 BIG DATA NEEDS BIG SECURITY

The amount of data collected by organizations and individuals around the world is growing on a daily basis, and the volume of the data being collected is expected to continue to grow exponentially. It is believed that the 90% of the data we have currently have in the world has been collected in the past few years. Velocity, volume and variety of Big Data comes results in privacy, security and compliance issues as well. Some of the data stored in Big Data platforms is very sensitive and regulations need be put in place, strictly controlling specific aspects of the data and who has access to the data. Proper measures have to be taken to control any weaknesses to cyber threats.

There are requirements for security measures already in place. Big Data platforms are subject to compliance mandates by government and industry regulations, including GDPR, PCI, Sarbanes-Oxley (SOX), and HIPAA [316]. These measures place regulations on company

practices and implementations that ensure proper data security and monitoring. These regulations are mandatory, and failing to comply could result in severe penalties, from heavy fines to legal actions.

While these requirements are important, traditional security mechanisms that have been in place for securing structured static data are no longer sufficient. With technological advances also comes a need to continually assess weaknesses in the new systems, to protect itself from new cyber threats and hacking strategies, and to create user friendly platforms for client that do not compromise the data being collected or stored. These developments are often far ahead of regulation, and individual entities need to be continually monitoring and enhancing their platforms to ensure protection of its data and systems. Big Data needs bigger security to protect its data, applications and infrastructure. Securing data not only protects the brand, reduces costs and avoids any legal issues, it also helps in retaining the brand name and increases revenue and growth [317].

31.4 BIG DATA SECURITY CHALLENGES

Recent adoption of cloud storage has increased the amount of data collected by organizations and hence it has become of vital importance to secure these data platforms as well. Data security issues are generally caused by the lack of proper tools and measures provided by traditional anti-virus software. Routine security checks to detect patches are no longer enough to handle real time influxes of data. Streaming real time data demands a great amount of attention focused on security and privacy solutions. Databases are no longer static. Big Data security's motto is to restrict unauthorized users and intruders from getting into a platform and also to block the encryption of data both in-transit and at-rest. The adoption of cloud storage creates a need to pay particular attention to the in-transit, or the continually expanding and modifying databases. Big Data security tools must be in place at all stages of data i.e. on incoming data, data stored in the platform and also on the data that goes out to other applications or outside party [318].

31.4.1 Access Control

Access control, in the context of Big Data, is controlling who can access data by using security settings. The different platforms that use Big Data need to be able to identify critical data, data origination and also who has access to the data. In this capacity, data access is not only protecting from external access, but also protecting data from those who have internal access as well [319].

User access should be controlled via a policy-based approach that automates access based on user and role-based settings. This manages different level of approvals in order to regulate who has access to the critical data and to protect the big data platform against inside attacks [320].

31.4.2 Audit Control

Big Data analytics can be used to analyze different types of logs in order to identify malicious activity. It also can regularly audit all the working directories inside the organization in order to check for any unauthorized access to any sensitive or privacy data. In reality, not all attacks are identified in the exact moment when the attack occurs. In order to perform a root cause analysis of the incident, data security professionals need to have access to audit logs which allow them to trace attacks back to the point of entry, exact time, modifications or weaknesses. In case of data breach, some firms are required to turn over their audit logs to stakeholders and possibly affected companies and heavy fines are imposed for failure to comply [317].

31.4.3 Real Time Compliance Control

Real time security monitoring is always very challenging due to the number of false positive alerts generated by security programs. Because of the frequency of false positive alerts, they are usually ignored. Big Data analytics may help provide more meaningful insights that could result in real time detection .

31.4.4 Non Relational Databases Privacy

Non Relational Databases are still not fully matured. This poses a severe threat to securing the data and it is often difficult for security and governance team to keep up with the demand. NoSQL databases primarily focus on how to handle high volume of data without paying much attention to their security needs.

31.4.5 End-Point Input Validation

Many organizations collect their data from End-Point devices. It is very important to ensure that data coming from these devices is not infected. Proper steps must be taken to make sure data is coming from an authentic source and it is legitimate. Incoming data from End-Point devices such as smart phones is growing tremendously and filtering or validating data from these sources is a very big challenge [317].

31.4.6 Securing Transaction Logs and Data

Data in any organization may be stored at various levels (tiers) of the storage structure depending on the need and usage of the data. Increase in the transfer of data within the organization enforce for the need of auto-tiering for Big Data storage whereas auto-tiering does not maintain the log of where the data is stored and hence security is a big concern.

31.4.7 Securing Distributed Framework

Distributed framework enforces parallelism. This means that data is distributed across multiple nodes to achieve faster processing of large volumes of data. This increases the security concern of the framework and the data that exists there. Most companies use a distributed framework like MapReduce in which mappers read and compute and reducers combine the output from each mapper. If mappers are not secured, there is the chance of data being compromised [320].

31.4.8 Data Provenance

It is very important to know the original data that is coming to the platform so that we can better classify them. Data Origin should be consistently monitored but in reality due to the high volume it is becoming a big concern for data security. Provenance metadata is growing significantly as well and protecting metadata is very crucial for any organization [320].

31.5 BIG DATA SECURITY STAKEHOLDERS

In the digital era, the traditional way of securing the data, changing passwords frequently, firewall protection is just not enough to keep up with the growth of data produced by Internet of Things(IoT), Smart Devices, Bring Your Own Devices (BYOD) and several customer friendly apps that is coming out everyday. "Even though end user has the biggest responsibility with securing his own data, unfortunately, end users are not fully aware of the cyber security issues and they do not have the appropriate knowledge to discover the world wide web in complete safety" [321].

Big Data deployment is not possible to handle by any single business unit or with single tech team. It involves several business units, infrastructure, information technology, security, compliance, programmers, testers and product owners are all involved in big data deployment. They are all responsible for Big Data Security. Information Technology and Security team is responsible for drawing the policies and procedures. Compliance officers together with security team will protect compliance, such as automatically encrypting personally identifiable information before it is easily accessible. Administrators will automate these process to protect their environment. Even though every organizations have their policies and control laid in place to protect their biggest asset, phishing attacks can come in any form as a simple email. Frequent internal audit within the company can help us periodically check if all privacy, security and compliance are all in place.If not, proper measures can be taken right away to avoid any legal issues.

"The average annualized cost of cyber crime based upon a representative sample of 237 organizations in six countries by Ponemon Institute in their 2016 Cost of Cyber Crime Study and the Risk of Business Innovation sponsored by Hewlett Packard Enterprise is 9.5million U.S. dollars" [322]. In any organization, loss of information is the most expensive consequence of a cyber crime. The cyber attack may results in business disruptions, data or information loss, loss of revenue, damage to equipment and last but not the least it damages the brand. So it is big time to protect and secure the big data and the environment from all angles.

31.6 BEST PRACTICES FOR SECURING BIG DATA

There are three fundamental principles used in defining security goals: confidentiality, integrity, and availability. Confidentiality is the ability to keep sensitive data safe from third parties and unauthorized access. Integrity in this context means to avoid unauthorized modification of the data. Finally, availability means always being able to access the data and resources. These three concepts are known as the CIA triad, and is used as base principles when discussing and designing security practices [323].

To meet these goals, there are four main branches of security that apply to Big Data: Authentication, Encryption, Data Masking and Access Control [324].

31.6.1 Authentication

Because of its nature (large sizes of data, linking different sources, sharing access with third parties, etc), some of Big Data's features are highly susceptible to different privacy, security and welfare risks [325].

Privacy can be defined as the condition of confidentiality, protecting information from third parties. To support privacy, there have been different Authentication methods that both verify and validate entities who attempt to access the information. This ensures that only

authorized entities are able to access the data or resources.

With Big Data, it is important to choose a proper authentication method, with the least computation complexity as possible, to allow dynamic security solutions within large Data Centers and also to avoid incrementing the traffic unnecessarily. Choosing an overbearing authentication method can cause both delay and storage issues. Because of this, it is important to tailor the security to the needs of the specific network ???.

31.6.2 Cryptography

There are multiple understandings of how data moves through stages, also known as Data Life cycles. Cryptography- define in terms of security. CITA.

From the perspective of cryptography, there are three phases in the Data Life Cycle: Data in Transit, Data in Storage, and Data in Use. Different cryptography techniques will be implemented depending on which stage of the life cycle the data is in [323].

There are different cryptographic tools that not only keep data secure at each point in its life-cycle, but also enable richer use of the data. The main tool is Encryption. Encryption takes pieces of data in plain text and use a cryptographic key to produce a version of the data that can only be read using the cryptographic key. Without the key, the information is illegible. There are two types of encryption: secret key encryption and public key encryption. Secret key encryption is when the same key is used for both encrypting and decrypting data. There are scenarios when one of the keys can be made public. For instance, if the locking key is kept private but the unlocking one is made public, this security can be used to prove authenticity [323].

There are different standards for encryption. The most well known and commonly used is Advanced Encryption Standard (AES). This standard sets guiding principles to ensure that data is encrypted in a manner that meets security needs and allows the recovery of original data [323].

31.6.3 Data Masking

By definition, Big Data works with large volumes of heterogeneous data sets using software to manage the data and to provide predictive analysis. Data masking works by replacing sensitive data with non-sensitive values, yet preserves the data integrity. For instance, replacing names with code names, or social security numbers with a key number. By doing this, different parties can access information without putting sensitive data at risk [326].

Five laws for data masking have been developed by Securosis Research. The first law is that data masking should not be reversible. This means that the data should not be unmasked easily using reverse engineering. The second law is that data that has been masked has to represent the original data set. For example, it has to belong in the same context. The third law states that data masking should maintain application and database integrity. This means that the process of data masking should not modify or affect the data in the databases in a negative way. The fourth law emphasizes that non-sensitive data can be masked, but it should not be masked if it can not be used to make sensitive data vulnerable. For instance, when masking information about a person, it is correct to mask the person's name, email address and social security number, but other information like gender, or favorite colour, would be useless to mask. Finally, data masking must be a repeatable process, using a standard to reproduce the steps taken to mask the data, allowing to troubleshoot possible problems in the process [326].

31.6.4 Access Control

As it was explained in the Challenges section, Access control, allows some entities to access the data or resources, while denying its use to other entities. Through security settings.

Some authors add that the inferences drawn from data should also be a cause for concern, because they can identify traits and patterns that could expose vulnerabilities. They propose that organizations

who use the protected data should disclose their decisions criteria in order to apply access control in a broader spectrum. By doing so, it would be sufficient to diminish privacy concerns by de-identifying the data, or denying access to certain parts of the data that could be used to make entities or data vulnerable. Some of these authors say that by doing this, it would not only reduce the privacy risk, it would also salvage large amounts of data for alternative use. This de-identification can also be achieved through data masking, pseudonymization, aggregation, among other methods [327].

31.6.5 Physical Security

It is always better to build and deploy Big Data platforms in their own data center. If deployed in a cloud, the organization must diligent to ensure that the cloud provider's data center is physically well secured. Access should be restricted to strangers and staff who have no official responsibilities in the designated areas or interacting with the data sources. Data centers should be properly monitored at all times and video surveillance and security logs are important tools to achieve this.

31.7 FUTURE OF BIG DATA SECURITY

To think about Future of Big Data Security, it is necessary to engage the conversation of what the trends are in Big Data and what technologies are expanding and changing the horizon. There are many new technologies and solutions that are shaping the future of the Big Data, but because of the length and focus of this document, there are three main areas that will be covered: Virtualization and Cloud Computing, IOT Security, External Password Vaults and Penetration Tests.

31.7.1 Virtualization and Cloud Computing

Virtualization is a way of deploying resources at multiple levels, such as hardware, network infrastructure, application and desktop

centralized managing and using dynamically the physical resources. This makes the system flexible and less costly than traditional environments and giving management new tools to optimize the use of resources [328].

Since virtualization can be developed in so many levels, including cloud computing and by multiple service providers, it is natural that the system requirements of users and organizations move towards a variety of solutions that may include Infrastructure as a Service (IaaS) frameworks from public clouds such as the ones offered by Amazon, Microsoft, Google, Rackspace, HP, among others, or even Private clouds, maintained and many times even set up by internal IT departments [329].

These cloud computing technologies are being used to solve data-intensive problems on large-scale infrastructure. Thus, integrating big data technologies and cloud computing for data mining, knowledge discovery, and decision-making [330].

31.7.2 IOT Security

The Internet of Things (IoT) is the name given to the large network of physical devices that does not match the typical concept of computer networks, this includes all kinds of objects. The large and growing amount of devices and diverse uses given to them, makes IoT generates very important Big Data streams. Making it necessary to develop new systems and data mining techniques for this new paradigm [331].

In this IoT paradigm, each new opportunity opens doors to new threats as well. This makes it necessary to develop techniques to ensure trust, security and privacy. Different Authors write about the possible ways to face these challenges, and some, they consider three main axes to articulate the solutions: Effective security - used in very small embedded networks, context-aware privacy and user-centric privacy, and the third one is the systemic and cognitive approach for IoT security - where the interaction between people and the IoT can

be envisioned as a set of nodes and tensions [332].

All this to say that in order to approach privacy and security in this new paradigm, many new theories and techniques have been developed since old security products and techniques may not suffice the needs of the different IoT users and communications.

31.7.3 External Password Vaults

Password vaults are applications that store multiple passwords and encrypt them storing them in a database [333].

There are small Password Vaults that can be stored locally on a system, or larger options that can be integrated into larger systems, providing additional security options, like generating real time temporary passwords for effective password rotation (I.E. Cyberark External Password Vault) [334].

These techniques are key to articulate authentication and a proper data access while using multiple services such as Cloud infrastructure and IoT.

31.7.4 Penetration Tests

After applying all the security techniques and strategies, and after putting in place all necessary security and privacy policies, the most important step is validating the strength of the security of the system. For some time, companies have started to perform tests that consist on simulating an attack from the perspective of an attacker, this method is known as Penetration test and it allows to actively evaluate and assess the security of a system [335].

The tester identifies the threats faced by an organization from hackers and suggest changes to improve the security and minimize the vulnerabilities and close the possible loop holes in the network [335].

31.8 CONCLUSIONS

Big Data as a constantly evolving and ever changing branch of information technologies resembles an ecosystem that since it covers gathering data from so many sources, processing it and generating new information, there will be many entities and interests involved that will need to be protected. The features of Big Data such as Volume, Variety and Velocity bring new challenges to security and privacy protection. To protect the integrity and availability, security providers and local IT departments, will have to diversify their security and privacy strategies and policies, in order to keep pace with the growth and evolution of this new ecosystem.

The authors would like to thank Dr. Gregor von Laszewski for his support and suggestions in writing this paper.

31.9 WORK BREAKDOWN

- Uma Kugan Research for Section Big Data Needs Big Security, Big Data Security and Challenges.
- Andres Castro Benavides Research for Section Best practices and Future
- Editing:: Andres Castro Benavides and Uma Kugan, Gregor von Laszewski

REFERNCE

- [1] G. von Laszewski, "Sample project." Report, Oct-2018 [Online]. Available: <https://github.com/cloudmesh-community/proceedings-fa18/blob/master/project-report/report.md>
- [2] M. Scherocman, "Top 5 benefits of microsoft azure sql database." website, 2016 [Online]. Available: <https://www.interlink.com/blog/entry/top-5-benefits-of-windows-azure-sql-database>
- [3] Microsoft, "Azure sql database purchasing models | microsoft docs." website, 2018 [Online]. Available: <https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/sql-database/sql-database-service-tiers.md>
- [4] Microsoft, "Welcome to azure cosmos db." website, 2018 [Online]. Available: <https://docs.microsoft.com/en-us/azure/cosmos-db/introduction>
- [5] Microsoft, "SLA for azure cosmos db." website, 2018 [Online]. Available: https://azure.microsoft.com/en-us/support/legal/sla/cosmos-db/v1_2/
- [6] A. Ali, "Getting started with azure sql data warehouse - part 1." website, 2017 [Online]. Available: <https://www.databasejournal.com/features/mssql/getting-started-with-azure-sql-data-warehouse-part-1.html>
- [7] Microsoft, "What is polybase?" website, 2018 [Online]. Available: <https://docs.microsoft.com/en-us/sql/relational-databases/polybase/polybase-guide?view=sql-server-2017>
- [8] J. P. Hoang, "Common isv application patterns using azure sql data warehouse." website, 2017 [Online]. Available: <https://blogs.msdn.microsoft.com/sqlcat/2017/09/05/common-isv-application-patterns-using-azure-sql-data-warehouse/>

[9] Microsoft, "What is azure hdinsight and the apache hadoop technology stack." website, 2018 [Online]. Available: <https://docs.microsoft.com/en-us/azure/hdinsight/hadoop/apache-hadoop-introduction>

[10] Microsoft, "What is stream analytics?" website, 2018 [Online]. Available: <https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-introduction>

[11] Microsoft, "A closer look at azure data storage gen2." website, 2018 [Online]. Available: <https://azure.microsoft.com/en-us/blog/a-closer-look-at-azure-data-storage-gen2/>

[12] Microsoft, "What is azure data lake analytics?" website, 2018 [Online]. Available: <https://docs.microsoft.com/en-us/azure/data-lake-analytics/data-lake-analytics-overview>

[13] Microsoft, "Introduction to azure data factory." website, 2018 [Online]. Available: <https://docs.microsoft.com/en-us/azure/data-factory/introduction>

[14] Microsoft, "Transform data in azure data factory." website, 2018 [Online]. Available: <https://docs.microsoft.com/en-us/azure/data-factory/transform-data>

[15] AWS, "Navigating GDPR Compliance on AWS." Web Page, Sep-2018 [Online]. Available: https://d1.awsstatic.com/whitepapers/compliance/GDPR_Compliance.pdf

[16] J. White, "Private vs. Public Cloud: What's the Difference?" Web Page, 2018 [Online]. Available: <https://www.expedient.com/blog/private-vs-public-cloud-whats-difference/>

[17] V. D. Somma, "Openstack compliance with GDPR." Web Page, 2018 [Online]. Available: https://archive.fosdem.org/2018/schedule/event/vai_openstack_gdpr/

- [18] RedHat, "Privacy Statement." Web Page, May-2018 [Online]. Available: <https://www.redhat.com/en/about/privacy-policy>
- [19] V. Badola, "What is Cloud Foundry? Key benefits and a real use case." Sep-2015 [Online]. Available: <https://cloudacademy.com/blog/cloud-foundry-benefits/>
- [20] N. Nimalsiri, "Introduction to Cloud Foundry." Jan-2016 [Online]. Available: <http://nanduni.blogspot.com/2016/01/introduction-to-cloud-foundry.html>
- [21] C. Foundry, "Case Studies." 2018 [Online]. Available: <https://www.cloudfoundry.org/case-studies/>
- [22] A. Bloom, "Beyond the Twelve-Factor App." Jun-2016 [Online]. Available: <https://content.pivotal.io/blog/beyond-the-twelve-factor-app>
- [23] Cloud Security Alliance, "About." website, 2018 [Online]. Available: <https://cloudsecurityalliance.org/about/>
- [24] Cloud Security Alliance, "About us." website, 2018 [Online]. Available: <https://www.linkedin.com/company/cloud-security-alliance/>
- [25] E. Messmer, "Cloud security alliance formed to promote best practices." Website, Mar-2009 [Online]. Available: <https://www.computerworld.com/article/2523598/security0/cloud-security-alliance-formed-to-promote-best-practices.html>
- [26] Cloud Security Alliance, "Chapters." website, 2018 [Online]. Available: <https://cloudsecurityalliance.org/chapters/>
- [27] Cloud Security Alliance, "Guidance." website, 2018 [Online]. Available: https://cloudsecurityalliance.org/guidance/#_overview
- [28] Cloud Security Alliance, "STAR." website, 2018 [Online]. Available: https://cloudsecurityalliance.org/star/#_overview
- [29] Cloud Security Alliance, "CCSA." website, 2019 [Online]. Available:

Why Obtain the CCSK?

- [30] Cloud Security Alliance, “CCSP.” website, 2018 [Online]. Available: https://cloudsecurityalliance.org/education/ccsp/#_overview
- [31] Cloud Security Alliance, “Global consultancy.” website, 2018 [Online]. Available: https://cloudsecurityalliance.org/global-consultancy/#_overview
- [32] Cloud Security Alliance, “Groups.” website, 2018 [Online]. Available: https://cloudsecurityalliance.org/research/#_groups
- [33] KNIME, “KNIME integrations.” Web page, 2018 [Online]. Available: <https://www.knime.com/knime-software/knime-integrations>
- [34] A. Vidhya, “Building your first machine learning model using knime (no coding required!).” Web page, 2017 [Online]. Available: <https://www.analyticsvidhya.com/blog/2017/08/knime-machine-learning/>
- [35] KNIME, “JSON processing.” Web page, 2018 [Online]. Available: <https://www.knime.com/whats-new-in-knime-211#JSON>
- [36] U. Sewwandi, “Guided analytics using knime analytics platform.” Web page, 2018 [Online]. Available: <https://towardsdatascience.com/guided-analytics-using-knime-analytics-platform-b6543ebab7e2>
- [37] KNIME, “KNIME workflow hub.” Web page, 2018 [Online]. Available: <https://www.knime.com/whats-new-in-knime-36#knime-workflow-hub>
- [38] KNIME, “Distributed executors in the next major version of knime server.” Web page, 2018 [Online]. Available: <https://www.knime.com/blog/distributed-executors-in-the-next-major-version-of-knime-server>
- [39] KNIME, “KNIME analytics platform.” Web page, 2018 [Online]. Available: <https://www.knime.com/knime-software/knime-analytics->

platform

- [40] KNIME, “KNIME quickstart guide.” Web page, 2018 [Online]. Available: <https://forge.epn-campus.eu/svn/edna/trunk/deprecated/rcp-knime/org.edna.workbench.target/knime2.1.2/org.knime.workbench.l>
- [41] KNIME, “KNIME on amazon web services.” Web page, 2018 [Online]. Available: <https://www.knime.com/knime-software/knime-aws>
- [42] KNIME, “KNIME on microsoft azure.” Web page, 2018 [Online]. Available: <https://www.knime.com/knime-software/knime-azure>
- [43] Apache, “Kafka.” Web Page [Online]. Available: <https://kafka.apache.org/>
- [44] T. P. Neha Narkhede Gwen Shapira, Kafka: The definitive guide, First. O'REILLY, 2017 [Online]. Available: <https://www.confluent.io/wp-content/uploads/confluent-kafka-definitive-guide-complete.pdf>
- [45] Apache, “Apache kafka.” Web Page, 2018 [Online]. Available: <https://www.apache.org/dyn/closer.cgi?path=/kafka/2.1.0/kafka-2.1.0-src.tgz>
- [46] Apache, “Kafka.” Web Page [Online]. Available: <https://issues.apache.org/jira/browse/KAFKA-6855>
- [47] The Apache Software Foundation, “Apache nifi.” Web page, Oct-2018 [Online]. Available: <https://nifi.apache.org/>
- [48] A. DOKAEVA, “How to make etl simple and intuitive with nifi.” Web page, Mar-2018 [Online]. Available: <https://issart.com/blog/how-to-make-etl-simple-and-intuitive-with-nifi/>
- [49] S. Maarek, “Introduction to apache nifi (hortonworks dataflow - hdf 2.0).” Presentation [Online]. Available: <https://www.udemy.com/apache-nifi/>

- [50] A. Bridgwater, “NSA ‘nifi’ big data automation project out in the open.” Web Page, Jul-2015 [Online]. Available: <https://www.forbes.com/sites/adrianbridgwater/2015/07/21/nsa-nifi-big-data-automation-project-out-in-the-open/#68cdd7dc55d6>
- [51] Apache NiFi Team, “Apache nifi overview.” Web page, Oct-2018 [Online]. Available: <https://nifi.apache.org/docs.html>
- [52] hortonworks, “Analyze transit patterns with apache nifi.” Web page, Oct-2018 [Online]. Available: <https://hortonworks.com/tutorial/analyze-transit-patterns-with-apache-nifi/section/1/>
- [53] S. Gupta, “Creating custom processors and controllers in apache nifi.” Web page, May-2018 [Online]. Available: <https://medium.com/hashmapinc/creating-custom-processors-and-controllers-in-apache-nifi-e14148740ea>
- [54] Apache NiFi Team, “Apache nifi downloads.” Web page, Oct-2018 [Online]. Available: <http://nifi.apache.org/download.html>
- [55] Apache NiFi Team, “Getting started with apache nifi.” Web page, Oct-2018 [Online]. Available: <https://nifi.apache.org/docs/nifi-docs/html/getting-started.html#downloading-and-installing-nifi>
- [56] V. Anand, “Using nifi to simplify data flow & streaming use cases @ mastercard.” Presentation [Online]. Available: <https://dataworkssummit.com/san-jose-2018/session/using-nifi-to-simplify-data-flow-streaming-use-cases-mastercard/>
- [57] S. Vaid, “Streaming analytics with opentext magellan.” Web page, Aug-2018 [Online]. Available: <https://blogs.opentext.com/streaming-analytics-with-opentext-magellan/>
- [58] Compose, “A real use case with nifi, the swiss army knife of data flow.” Presentation [Online]. Available: <http://mybbt.bbtconsulting.com:8069/slides/slide/a-real-use-case-with-nifi-the-swiss-army-knife-of-data-flow-121>

- [59] Ford, "Real time streaming architecture at ford." Presentation, Jun-2017 [Online]. Available: https://www.slideshare.net/Hadoop_Summit/real-time-streaming-architecture-at-ford
- [60] A. Paszke et al., "Automatic differentiation in pytorch," 2017.
- [61] NumPy, "NumPy." Website [Online]. Available: <http://www.numpy.org/>
- [62] PyTorch, "What is pytorch?" Website [Online]. Available: https://pytorch.org/tutorials/beginner/blitz/tensor_tutorial.html
- [63] Wikipedia, "PyTorch." Website [Online]. Available: <https://en.wikipedia.org/wiki/PyTorch>
- [64] TensorFlow, "Get started." Website [Online]. Available: <https://www.tensorflow.org/>
- [65] Keras, "Keras: The python deep learning library." Website [Online]. Available: <https://keras.io/>
- [66] Caffe, "Caffe." Website [Online]. Available: <http://caffe.berkeleyvision.org/>
- [67] Chainer, "Get started." Website [Online]. Available: <https://chainer.org/>
- [68] MXNet, "Apache mxnet (incubating)." Website [Online]. Available: <https://mxnet.apache.org/>
- [69] Microsoft, "CNTK." GitHub [Online]. Available: <https://github.com/Microsoft/CNTK>
- [70] DL4J, "Quick start." Website [Online]. Available: <https://deeplearning4j.org/>
- [71] Torch, "Torch." Website [Online]. Available: <http://torch.ch/>

- [72] Wikipedia, “C programming language.” Website [Online]. Available: [https://en.wikipedia.org/wiki/C_\(programming_language\)](https://en.wikipedia.org/wiki/C_(programming_language))
- [73] LuaJIT, “The luajit project.” Website [Online]. Available: <http://luajit.org/>
- [74] Wikipedia, “Deep learning.” Website [Online]. Available: https://en.wikipedia.org/wiki/Deep_learning
- [75] J. Brownlee, “What is deep learning?” Website [Online]. Available: <https://machinelearningmastery.com/what-is-deep-learning/>
- [76] Wikipedia, “Artificial neural network.” Website [Online]. Available: https://en.wikipedia.org/wiki/Artificial_neural_network
- [77] M. Mayo, “WTF is a tensor?!?” Website [Online]. Available: <https://www.kdnuggets.com/2018/05/wtf-tensor.html>
- [78] A. M. Giancarlo Zaccone Md. Rezaul Karim, “Computational graphs.” Website [Online]. Available: https://www.packtpub.com/mapt/book/big_data_and_business_intelligible_graphs
- [79] Wikipedia, “Automatic differentiation.” Website [Online]. Available: https://en.wikipedia.org/wiki/Automatic_differentiation
- [80] Wikipedia, “Backpropagation.” Website [Online]. Available: <https://en.wikipedia.org/wiki/Backpropagation>
- [81] PyTorch, “Autograd mechanics.” Website [Online]. Available: <https://pytorch.org/docs/stable/notes/autograd.html>
- [82] V. Rao, “Get started with pytorch.” Website [Online]. Available: <https://developer.ibm.com/articles/cc-get-started-pytorch/>
- [83] PyTorch, “QUICK start locally.” Website [Online]. Available: <https://pytorch.org>
- [84] PyTorch, “GET started.” Website [Online]. Available:

<https://pytorch.org/get-started/locally/>

[85] PyTorch, "WELCOME to pytorch tutorials." Website [Online]. Available: <https://pytorch.org/tutorials/index.html>

[86] D. Mwiti, "Introduction to pytorch for deep learning." Website [Online]. Available: <https://heartbeat.fritz.ai/introduction-to-pytorch-for-deep-learning-5b437cea90ac>

[87] D. Mesquita, "How pytorch gives the big picture with deep learning." Website [Online]. Available: <https://medium.freecodecamp.org/how-pytorch-gives-the-big-picture-with-deep-learning-e4a0f372f4b6>

[88] D. Mesquita, "README." Website [Online]. Available: https://github.com/dmesquita/understanding_pytorch_nn

[89] kdnuggets, "PyTorch or tensorflow?" Website [Online]. Available: <https://www.kdnuggets.com/2017/08/pytorch-tensorflow.html>

[90] Scipy, "Scipy." Website [Online]. Available: <https://www.scipy.org/>

[91] A. Paszke, "CREATING extensions using numpy and scipy." Website [Online]. Available: https://pytorch.org/tutorials/advanced/numpy_extensions_tutorial.htm

[92] Siftery Discover, "PyTorch alternatives." Website [Online]. Available: <https://siftery.com/pytorch/alternatives>

[93] Wikipedia, "Serialization." Website [Online]. Available: <https://en.wikipedia.org/wiki/Serialization>

[94] J. Yangqing et al., "Caffe: Convolutional architecture for fast feature embedding," arXiv preprint arXiv:1408.5093, 2014.

[95] Wikipedia, Website [Online]. Available: https://en.wikipedia.org/wiki/Deep_learning

[96] "Theano." Web page [Online]. Available:

<http://deeplearning.net/software/theano/introduction.html>

[97] S. Yangqing Jia Evan, "Caffe | deep learning framework." Web page [Online]. Available: <http://caffe.berkeleyvision.org/>

[98] Wikipedia, "Convolutional neural network." Website [Online]. Available: https://en.wikipedia.org/wiki/Convolutional_neural_network

[99] Wikipedia, Website [Online]. Available: https://en.wikipedia.org/wiki/Graphics_processing_unit

[100] Wikipedia, "Computer vision." Website [Online]. Available: https://en.wikipedia.org/wiki/Computer_vision

[101] Wikipedia, "Artificial intelligence." Website [Online]. Available: https://en.wikipedia.org/wiki/Artificial_intelligence

[102] Wikipedia, "Signal processing." Website [Online]. Available: https://en.wikipedia.org/wiki/Signal_processing

[103] Wikipedia, Website [Online]. Available: https://en.wikipedia.org/wiki/Natural_language_processing

[104] Wikipedia, "Perceptron." Website [Online]. Available: <https://en.wikipedia.org/wiki/Perceptron>

[105] "Docker - overview." Web page [Online]. Available: https://www.tutorialspoint.com/docker/docker_overview.htm

[106] "What is the Jupyter Notebook? — Jupyter Notebook 5.0.0.dev documentation." Web page, 27-Feb-2017.

[107] Wikipedia, "CUDA." Web page, Feb-2017 [Online]. Available: <https://en.wikipedia.org/wiki/CUDA>

[108] Wikipedia, "MATLAB." Website [Online]. Available: <https://en.wikipedia.org/wiki/MATLAB>

[109] Wikipedia, "Artificial neural network." Website [Online].

Available: https://en.wikipedia.org/wiki/Artificial_neural_network

[110] E. Shelhamer, J. Donahue, J. Long, Y. Jia, and R. Girshick, "DIY deep learning for vision: A hands-on tutorial with caffe." 2017 [Online]. Available:

https://docs.google.com/presentation/d/1UeKXVgRvxg9OUdh_UiC5G

[111] Wikipedia, "HDF5." Website [Online]. Available: https://en.wikipedia.org/wiki/Hierarchical_Data_Format

[112] P. P. Subhasis Das Hou Yunqing, "Quora-answer-3." Website [Online]. Available: <https://www.quora.com/What-are-the-pros-and-cons-of-Caffe-the-deep-learning-framework>

[113] Facebook, "Caffe2." Web page [Online]. Available: <https://github.com/pytorch/pytorch/tree/master/caffe2>

[114] C. WODEHOUSE, "Should you use mongodb? A look at the leading nosql database." Web Page, 2018 [Online]. Available: <https://www.upwork.com/hiring/data/should-you-use-mongodb-a-look-at-the-leading-nosql-database/>

[115] Guru99, "Introduction to mongodb." Web Page, 2018 [Online]. Available: <https://www.guru99.com/mongodb-tutorials.html#1>

[116] MongoDB, "Https://www.mongodb.com/." Web Page, 2018 [Online]. Available: <https://docs.mongodb.com/manual/introduction/>

[117] M. Papiernik, "How to install mongodb on ubuntu 18.04." Web Page, Jun-2018 [Online]. Available: <https://www.digitalocean.com/community/tutorials/how-to-install-mongodb-on-ubuntu-18-04>

[118] J. Ellingwood, "Initial server setup with ubuntu 18.04." Web Page, Apr-2018 [Online]. Available: <https://www.digitalocean.com/community/tutorials/initial-server-setup-with-ubuntu-18-04>

[119] MongoDB, Databases and collections, 4.0 ed. New York, New

York, USA: MongoDB Inc, 2008 [Online]. Available: <https://docs.mongodb.com/manual/core/databases-and-collections/>

[120] J. M. Craig Buckler, "Using joins in mongodb nosql databases." Web Page, Sep-2016 [Online]. Available: <https://www.sitepoint.com/using-joins-in-mongodb-nosql-databases/>

[121] MongoDB, Lookup (aggregation), 3.2 ed. New York City, New York, United States: MongoDB Inc, 2008 [Online]. Available: <https://docs.mongodb.com/manual/reference/operator/aggregation/lookup/>

[122] MongoDB, MongoDB package components - mongoexport, 4.0 ed. New York City, New York, United States: MongoDB Inc, 2008 [Online]. Available: <https://docs.mongodb.com/manual/reference/program/mongoexport/>

[123] MongoDB, Security, 4.0 ed. New York City, New York, United States: MongoDB Inc, 2008 [Online]. Available: <https://docs.mongodb.com/manual/security/>

[124] MongoDB, "MongoDB atlas." Web Page, 2018 [Online]. Available: <https://www.mongodb.com/cloud/atlas>

[125] I. MongoDB, "PyMongo 3.7.1 documentation." Web Page, 2008 [Online]. Available: <https://api.mongodb.com/python/current/api>

[126] A. J. J. Davis, "Announcing pymongo3." Web Page, Apr-2015 [Online]. Available: <https://emptysqua.re/blog/announcing-pymongo-3/>

[127] M. Dirolf, "PyMongo." Web Page, Jul-2018 [Online]. Available: <https://github.com/mongodb/mongo-python-driver>

[128] N. Leite, "MongoDB and python." Web Page, Mar-2015 [Online]. Available: <https://www.slideshare.net/NorbertoLeite/mongodb-and-python>

[129] V. Oleynik, "How do you use mongodb with python?" Web Page, Mar-2017 [Online]. Available: <https://gearheart.io/blog/how-do-you-use-mongodb-with-python>

[use-mongodb-with-python/](#)

[130] I. MongoDB, “Installing / upgrading.” Web pages, 2008 [Online]. Available: <http://api.mongodb.com/python/current/installation.html>

[131] R. Python, “Introduction to mongodb and python.” Web Page, 2016 [Online]. Available: <https://realpython.com/introduction-to-mongodb-and-python/>

[132] W3Schools, “Python mongodb create database.” Web Page, 1999 [Online]. Available: https://www.w3schools.com/python/python_mongodb_create_db.asp

[133] I. MongoDB, “PyMongo 3.7.1 documentation.” Web Page, 2008 [Online]. Available: <https://api.mongodb.com/python/current/tutorial.html>

[134] N. O’Higgins, PyMongo & python. O'Reilly, 2011 [Online]. Available: <http://img105.job1001.com/upload/adminnew/2015-04-07/1428393873-MHKX3LN.pdf>

[135] I. MongoDB, “PyMongo 3.7.1 documentation.” Web Page, 2008 [Online]. Available: <https://api.mongodb.com/python/current/examples/aggregation.html>

[136] MongoDB, “PyMongo 3.7.2 documenation.” Web Page, 2008 [Online]. Available: <https://docs.mongodb.com/manual/reference/operator/aggregation-pipeline/>

[137] MongoDB, “PyMongo 3.7.2 documenation.” Web Page, 2008 [Online]. Available: <https://docs.mongodb.com/manual/core/map-reduce/>

[138] MongoDB, “PyMongo v2.0 documentation.” Web Page, 2008 [Online]. Available: https://api.mongodb.com/python/2.0/examples/map_reduce.html

[139] MongoDB, “PyMongo 3.7.2 documenation.” Web Page, 2008

- [Online]. Available:
<https://api.mongodb.com/python/current/examples/copydb.html>
- [140] MongoEngine, “MongoEngine user documentation.” Web Page, 2009 [Online]. Available: <http://docs.mongoengine.org/>
- [141] Wikipedia, “Object-relational mapping.” Web Page, May-2009 [Online]. Available: https://en.wikipedia.org/wiki/Object-relational_mapping
- [142] MongoDB, “Flask-mongoengine.” Web Page, 2008 [Online]. Available: <http://docs.mongoengine.org/guide/defining-documents.html>
- [143] MongoEngine, “User guide: Document instances.” Web Page, 2009 [Online]. Available: <http://docs.mongoengine.org/guide/document-instances.html>
- [144] MongoEngine, “2.1 installing mongoengine.” Web Page, 2009 [Online]. Available: <http://docs.mongoengine.org/guide/installing.html>
- [145] MongoEngine, “2.2 connection to mongodb.” Web Page, 2009 [Online]. Available: <http://docs.mongoengine.org/guide/connecting.html>
- [146] MongoEngine, “User guide 2.5. Querying the database.” Web Page, 2009 [Online]. Available: <http://docs.mongoengine.org/guide/querying.html>
- [147] wikipedia, “Flask (web framework).” Web Page, 2010 [Online]. Available: [https://en.wikipedia.org/wiki/Flask_\(web_framework\)](https://en.wikipedia.org/wiki/Flask_(web_framework))
- [148] MongoDB, “Flask-pymongo.” Web Page, 2008 [Online]. Available: <https://flask-pymongo.readthedocs.io/en/latest/>
- [149] MongoDB, “Flask mongoalchemy.” Web Page, 2008 [Online]. Available: <https://pythonhosted.org/Flask-MongoAlchemy/>
- [150] MongoDB, “Flask-mongoengine.” Web Page, 2008 [Online].

Available: <http://docs.mongoengine.org/projects/flask-mongoengine/en/latest/>

[151] Wikipedia, “Flask (web framework).” Web Page, Oct-2018 [Online]. Available: [https://en.wikipedia.org/wiki/Flask_\(web_framework\)](https://en.wikipedia.org/wiki/Flask_(web_framework))

[152] “NLP for big data: What everyone should know?” Web page [Online]. Available: <https://www.expertsystem.com/nlp-big-data-everyone-know/>

[153] Wikipedia, “Big data.” Web page [Online]. Available: https://en.wikipedia.org/wiki/Big_data

[154] E. Liddy, “Encyclopedia of library and information sciences,” 2nd ed., Marcel Decker, Inc., 2001 [Online]. Available: <https://surface.syr.edu/cgi/viewcontent.cgi?referer=&httpsredir=1&article=1019&context=cnlp>

[155] A. Vieira and B. Ribeiro, Natural language processing and speech. In: Introduction to deep learning business applications for developers. Apress, Berkeley CA., 2018.

[156] Robin, “Part-of-speech tagging.” Web page, Dec-2009 [Online]. Available: <http://language.worldofcomputing.net/pos-tagging/parts-of-speech-tagging.html#>

[157] Gartner, “Natural-language understanding.” Web page, 2018 [Online]. Available: <https://www.gartner.com/it-glossary/nlu-natural-language-understanding/>

[158] S. C. Shapiro, “Encyclopedia of artificial intelligence,” 2nd ed., S. C. Shapiro, Ed. John Wiley & Sons, Inc., 1992, pp. 54–57 [Online]. Available: <https://cse.buffalo.edu/~shapiro/Papers/ai.pdf>

[159] Wikipedia, “AI-complete.” Web page [Online]. Available: <https://en.wikipedia.org/wiki/AI-complete>

[160] M. Clark, “Understanding nlu: A cheatsheet for beginners.” Web

page, Apr-2017 [Online]. Available:
<https://info.contactsolutions.com/digital-engagement-blog/understanding-nlu-cheat-sheet-for-beginners>

[161] S. Petrov, "Announcing syntaxnet: The world's most accurate parser goes open source." Web page, May-2016 [Online]. Available: <https://ai.googleblog.com/2016/05/announcing-syntaxnet-worlds-most.html>

[162] "Stanford log linear part-of-speech tagger." Web page [Online]. Available: <https://nlp.stanford.edu/software/tagger.shtml>

[163] K. Toutanova, D. Klein, C. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in Proceedings of hlt-naacl 2003, 2003, pp. 252-259 [Online]. Available: <https://nlp.stanford.edu/~manning/papers/tagging.pdf>

[164] N. Madnani and J. Lin, "Natural language processing with apache hadoop and python." Web page, Mar-2010 [Online]. Available: <https://blog.cloudera.com/blog/2010/03/natural-language-processing-with-hadoop-and-python/>

[165] N. Project, "Natural language toolkit." Web page, 2017 [Online]. Available: <https://www.nltk.org/>

[166] M. Rouse, "Hadoop." Web page, Apr-2018 [Online]. Available: <https://searchdatamanagement.techtarget.com/definition/Hadoop>

[167] M. Rouse, "Deep learning (deep neural network)." Web page, 2018 [Online]. Available: <https://searchenterpriseai.techtarget.com/definition/deep-learning-deep-neural-network>

[168] E. Systems, "NLP for big data: What everyone should know?" Web page, Aug-2016 [Online]. Available: <https://www.expertsystem.com/nlp-big-data-everyone-know/>

[169] Amazon, Web page [Online]. Available:

<https://aws.amazon.com/streaming-data/>

[170] G. Vaseekaran, "Big data battle : Batch processing vs stream processing." Web page, Oct-2017 [Online]. Available: <https://medium.com/@gowthamy/big-data-battle-batch-processing-vs-stream-processing-5d94600d8103>

[171] S. Perera, "A gentle introduction to stream processing." Web page, Apr-2018 [Online]. Available: <https://medium.com/stream-processing/what-is-stream-processing-1eadfca11b97>

[172] R. Vadai, "Challenges processing data in real-time using conventional big data solutions." Web page, Mar-2017 [Online]. Available: <https://codelook.com/challenges-with-processing-data-in-real-time-using-conventional-big-data-solutions-bb602b33da0c>

[173] "Spark streaming." Web page [Online]. Available: <https://spark.apache.org/streaming/>

[174] A. C. Oliver, "Storm or spark: Choose your real-time weapon." Web page, Dec-2014 [Online]. Available: <https://www.infoworld.com/article/2854894/application-development/spark-and-storm-for-real-time-computation.html>

[175] K. Khare, "What makes apache flink the best choice for streaming applications?" Web page, Apr-2018 [Online]. Available: <https://hackernoon.com/what-makes-apache-flink-the-best-choice-for-streaming-applications-fc377858a53>

[176] S. Kozlovski, "Thorough introduction to apache kafka." Web page, Dec-2017 [Online]. Available: <https://hackernoon.com/thorough-introduction-to-apache-kafka-6fbf2989bbc1>

[177] "Amazon kinesis data streams." Web page [Online]. Available: <https://aws.amazon.com/kinesis/data-streams/>

[178] "Hortonworks dataflow (hdf)." Web page [Online]. Available:

<https://hortonworks.com/products/data-platforms/hdf/>

[179] "Top 5 apache spark use cases." Web page, Jun-2016 [Online]. Available: <https://www.dezyre.com/article/top-5-apache-spark-use-cases/271>

[180] S.-l. developers, "User guide," None [Online]. Available: http://scikit-learn.org/stable/user_guide.html

[181] S. Li, E. J. Harner, and D. A. Adjeroh, "Random knn feature selection - a fast and stable alternative to random forests," BMC Bioinformatics, vol. 12, no. 1, p. 450, Nov. 2011 [Online]. Available: <https://doi.org/10.1186/1471-2105-12-450>

[182] Q. Wang, G. M. Garrity, J. M. Tiedje, and J. R. Cole, "Naïve bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy," Applied and Environmental Microbiology, vol. 73, no. 16, pp. 5261–5267, 2007 [Online]. Available: <https://aem.asm.org/content/73/16/5261>

[183] S. K. Murthy, "Automatic construction of decision trees from data: A multi-disciplinary survey," Data Mining and Knowledge Discovery, vol. 2, no. 4, pp. 345–389, Dec. 1998 [Online]. Available: <https://doi.org/10.1023/A:1009744630224>

[184] L. K. Hansen and P. Salamon, "Neural network ensembles," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 10, pp. 993–1001, Oct. 1990.

[185] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 24, pp. 881–892, Jul. 2002 [Online]. Available: <doi.ieeecomputersociety.org/10.1109/TPAMI.2002.1017616>

[186] J. F. Martins, V. F. Pires, and A. J. Pires, "Unsupervised neural-network-based algorithm for an on-line diagnosis of three-phase induction motor stator fault," IEEE Transactions on Industrial

Electronics, vol. 54, no. 1, pp. 259–264, Feb. 2007.

[187] E. D. Liddy, “Enhanced text retrieval using natural language processing,” Bulletin of the American Society for Information Science and Technology, vol. 24, no. 4, pp. 14–16, 1998.

[188] website [Online]. Available:
https://www.sas.com/en_us/insights/analytics/what-is-natural-language-processing-nlp.html

[189] G. G. Chaudhary, “Natural language processing,” Dept. of Computer and Information Sciences University of Strathclyde, Glasgow G1 1XH, UK, 2003.

[190] D. Khurana, A. Koli, K. Khatter, and S. Singh, “Natural language processing: State of the art, current trends and challenges,” CoRR, vol. abs/1708.05148, 2017.

[191] N. Tapaswi and S. Jain, “Treebank based deep grammar acquisition and part-of-speech tagging for sanskrit sentences,” in Software engineering (conseg), 2012 csi sixth international conference on, 2012, pp. 1–4.

[192] P. Ranjan and H. V. S. S. A. Basu, “Part of speech tagging and local word grouping techniques for natural language parsing in hindi,” in Proceedings of the 1st international conference on natural language processing (icon 2003), 2003.

[193] M. Palmer, D. Gildea, and P. Kingsbury, “The proposition bank: An annotated corpus of semantic roles,” Computational linguistics, vol. 31, no. 1, pp. 71–106, 2005.

[194] M. Vallez and R. Pedraza-Jimenez, “Natural language processing in textual information retrieval and related topics,” 2011.

[195] “Natural language processing.” website [Online]. Available:
https://en.wikipedia.org/wiki/Natural_language_processing

[196] chirag, “NLP for big data: How natural language processing is

poised to revolutionise big data analytics." website, Jun-2017 [Online]. Available: <https://huddle.eurostarsoftwaretesting.com/nlp-for-big-data-how-nlp-will-revolutionise-big-data-analytics/>

[197] V. Fedak, "5 heroic tools for natural language processing." website, Jan-2018 [Online]. Available: <https://towardsdatascience.com/5-heroic-tools-for-natural-language-processing-7f3c1f8fc9f0>

[198] A. Geitgey, "Natural language processing is fun!" website, Jul-2018 [Online]. Available: <https://medium.com/@ageitgey/natural-language-processing-is-fun-9a0bff37854e>

[199] T. Mills, "What is natural language processing and what is it used for?" website, Jul-2018 [Online]. Available: <https://www.forbes.com/forbes/welcome/?toURL=https://www.forbes.com/sites/forbestechcouncil/2018/07/02/what-is-natural-language-processing-and-what-is-it-used-for/&refURL=https://www.google.com/&referrer=https://www.google.com>

[200] SAS Institute, SAS® Viya® 3.4: Overview. SAS Institute Inc, 2018.

[201] Randy Guard, "Discovering SAS Viya : Special Collection," SAS Global Users Group Proceedings, pp. 8-10, 2017.

[202] M. Schneider, "SAS Viya: What It Means for SAS Administration," SAS Global Users Group Proceedings, 2017.

[203] Jonathan Wexler and Susan Haller and Radhikha Myneni, "An Overview of SAS Visual Data Mining and Machine Learning on SAS Viya," SAS Global Users Group Proceedings, 2017.

[204] Xiangxiang Meng and Kevin Smith, "I Am Multilingual: A Comparison of the Python, Java, Lua, and REST Interfaces to SAS Viya," SAS Global Users Group Proceedings, 2017.

[205] SAS, "SAS Platform." 2017.

[206] J. Pendergrass, "The Architecture of the SAS® Cloud Analytic

Services in SAS® Viya™,” SAS Global Users Group Proceedings, pp. 10–18, 2017.

[207] J. Pendergrass, “The Architecture of the SAS® Cloud Analytic Services in SAS® Viya™,” SAS Global Users Group Proceedings, pp. 1–3, 2017.

[208] J. Pendergrass, “The Architecture of the SAS® Cloud Analytic Services in SAS® Viya™,” SAS Global Users Group Proceedings, pp. 4–9, 2017.

[209] SAS, SAS® Studio. SAS Institute Inc, 2018.

[210] SAS, SAS® Visual Analytics. SAS Institute Inc, 2018.

[211] SAS, SAS® Visual Statistics. SAS Institute Inc, 2018.

[212] SAS, SAS® Visual Data Mining and Machine Learning. SAS Institute Inc, 2018.

[213] SAS, SAS® Econometrics. SAS Institute Inc, 2018.

[214] SAS, SAS® Visual Forecasting. SAS Institute Inc, 2018.

[215] Wikipedia, “Natural Language Processing.” Webpage, Nov-2018.

[216] SAS, SAS® Visual Text Analytics. SAS Institute inc, 2018.

[217] SAS, SAS® Optimization. SAS Institute Inc, 2018.

[218] SAS, SAS® Viya® 3.4 on Windows: Deployment Guide. SAS Institute Inc, 2018.

[219] SAS, “Getting ready to install SAS Viya 3.4 on Windows? Then read on.” Webpage, May-2018.

[220] MITKerberos Consortium, “ABOUT - THE MIT KERBEROS CONSORTIUM.” Webpage, 2017.

[221] SAS, SAS® Viya® 3.4 on Windows: Deployment Guide. SAS Institute Inc, 2018.

[222] SAS, SAS® Viya® 3.4 on Windows: Deployment Guide. SAS Institute Inc, 2018.

[223] "SAS Viya Start Page." Screenshot, Nov-2018.

[224] "SAS Viya Import Data." Screenshot, Nov-2018.

[225] "SAS Viya Add Data Object." Screenshot, Nov-2018.

[226] "SAS Viya Add Variable Roles." Screenshot, Nov-2018.

[227] "Linear Regression Results." Screenshot, Nov-2018.

[228] Victor Mayer-Schonberger and Kenneth Cukier, Big Data : A Revolution That Will Transform How We Live, Work and Think. John Murray, 2013.

[229] M. Techlabs, "How does a recommendation engine really work?" Oct-2017 [Online]. Available: <https://towardsdatascience.com/how-does-a-recommendation-engine-really-work-656bdf12a5fc>

[230] R. Banik, "Displaydetect.py." github, Jan-2018 [Online]. Available: <https://www.datacamp.com/community/tutorials/recommender-systems-python>

[231] P. Sharma, "Comprehensive guide to build a recommendation engine from scratch (in python)." Jun-2018 [Online]. Available: <https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-recommendation-engine-python/>

[232] Google Developers, "Using machine learning on compute engine to make product recommendations." Web Page, Nov-2018 [Online]. Available: [5.https://cloud.google.com/solutions/recommendations-using-machine-learning-on-compute-engine#storing_the_data](https://cloud.google.com/solutions/recommendations-using-machine-learning-on-compute-engine#storing_the_data)

[233] P. Kordik, "Machine learning for recommender systems." github, Jan-2018 [Online]. Available: <https://medium.com/recombee-blog/machine-learning-for-recommender-systems-part-1-algorithms-evaluation-and-cold-start-6f696683d0ed>

[234] R. B. Yehuda Koren and C. Volinsky, "MATRIX factorization techniques for recommender systems," Recommender-Systems-[Netflix], Aug. 2009.

[235] Wikipedia, "Machine learning for recommender systems." github, Jan-2018 [Online]. Available: https://en.wikipedia.org/wiki/Root-mean-square_deviation

[236] G. Karypis, "Evaluation of item-based top-n recommendation algorithms." github, Jan-2018 [Online]. Available: <https://www.semanticscholar.org/paper/Evaluation-of-Item-Based-Top-N-Recommendation-Karypis/0739fad62026ca36f101a36f29d53630207a5748>

[237] Wikipedia, "Kevin ashton." Wikipedia, Aug-2018 [Online]. Available: https://en.wikipedia.org/wiki/Kevin_Ashton

[238] Gartner, "Gartner." Web Page, 2017 [Online]. Available: <https://www.gartner.com/en/newsroom/press-releases/2017-02-07-gartner-says-8-billion-connected-things-will-be-in-use-in-2017-up-31-percent-from-2016>

[239] M. Ligade, "Architecture for iot applications." Web Page, 2016 [Online]. Available: <https://medium.com/@maheshwar.ligade/architecture-for-iot-applications-d50ece031d38>

[240] E. Ahmed et al., "The role of big data analytics in internet of things." Paper, 2017 [Online]. Available: https://www.researchgate.net/publication/317617290_The_role_of_big

[241] A. Verma, "Internet of things and big data – better together." Web Page, 2018 [Online]. Available:

<https://www.whizlabs.com/blog/iot-and-big-data>

[242] K. Khan, "Future iot and big data." Web Page, 2017 [Online]. Available:

https://www.researchgate.net/publication/316890756_Future_IOT_and_Big_Data

[243] A. Monnappa, "How big data is powering the internet of things (iot) revolution." Web Page, 2017 [Online]. Available: <https://www.simplilearn.com/how-big-data-powering-internet-of-things-iot-revolution-article>

[244] A. K. Zainab Alansari Nor Barul and J. Alshaer, "Challenges of internet of things and big data integration." Web Page, 2018 [Online]. Available: <https://arxiv.org/ftp/arxiv/papers/1806/1806.08953.pdf>

[245] A. Grilo, "Internet of things: An introduction." Web Page, 2018 [Online]. Available:

<https://fenix.tecnico.ulisboa.pt/downloadFile/1689468335603255/SER-IoT.pdf>

[246] newgenapps, "8 uses, applications, and benefits of industrial iot in manufacturing." Web Page, 2017 [Online]. Available: <https://www.newgenapps.com/blog/8-uses-applications-and-benefits-of-industrial-iot-in-manufacturing>

[247] Techtarget, "A guide to healthcare iot possibilities and obstacles." Web Page, 2017 [Online]. Available:

<https://searchhealthit.techtarget.com/essentialguide/A-guide-to-healthcare-IoT-possibilities-and-obstacles>

[248] I. Innovation, "How iot technology is changing building energy management systems." Web Page [Online]. Available: https://internet-of-things-innovation.com/insights/the-blog/how-iot-technology-is-changing-building-energy-management-systems/#.W_ZYzuhKhPY

[249] B. Marr, "IoT and big data at caterpillar: How predictive maintenance saves millions of dollars." Web Page, 2017 [Online]. Available: https://www.forbes.com/sites/bernardmarr/2017/02/07/iot-and-big-data-at-caterpillar-how-predictive-maintenance-saves-millions-of-dollars/#.W_ZYzuhKhPY

[and-big-data-at-caterpillar-how-predictive-maintenance-saves-millions-of-dollars/#50d19cc97240](#)

[250] IoTONE, "Accelerating the industrial internet of things." Web Page, 2018 [Online]. Available: <https://www.iotone.com/usecases>

[251] M. Drummond, "5 great ways airlines are using the internet of things." Web Page, Aug-2016 [Online]. Available: <https://w3.accelya.com/blog/5-great-ways-airlines-are-using-the-internet-of-things>

[252] D. Newman, "The iots impact on the future of retail." Web Page, Feb-2018 [Online]. Available: <https://www.forbes.com/sites/danielnewman/2018/02/20/the-iots-impact-on-the-future-of-retail/#711cacb07b1a>

[253] G. Christopher, "IoT centred healthcare." Web Page, Jul-2016 [Online]. Available: <https://www.computerworlduk.com/iot/iot-centred-healthcare-system-3643726>

[254] F. Online, "The future of iot and big data." Web Page, 2018 [Online]. Available: <https://financesonline.com/future-iot-big-data>

[255] L. Wang and C. A. Alexander, "Big data in medical applications and health care," American Medical Journal, 2015 [Online]. Available: <http://thescipub.com/pdf/10.3844/amjsp.2015.1.8>

[256] T. H. subgroup, "Needs, opportunities and challenges of the health sector," Big Data Value Association, 2016 [Online]. Available: <http://www.bdva.eu/sites/default/files/Big%20Data%20Technologies%20-%20Needs%20Opportunities%20Challenges%20of%20the%20Health%20Sector.pdf>

[257] F. Provost and T. Fawcett, "Data science and its relationship to big data and data-driven decision making," Published Online, vol. 1, no. 1, pp. 51–59, 2013 [Online]. Available: <https://doi.org/10.1089/big.2013.1508>

[258] C. Walter, "Kryder's law," Scientific American, pp. 32–33, Aug. 2005 [Online]. Available:

<https://www.scientificamerican.com/article/kryders-law/>

[259] V. Marx, "Biology: The big challenges of big data," International Journal of Science, 2013 [Online]. Available: http://pic.b.qs1401.com/42548/pdf/bigbioldata_nature13.pdf

[260] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," Commun. ACM, vol. 51, no. 1, pp. 107–113, Jan. 2008 [Online]. Available: <http://doi.acm.org/10.1145/1327452.1327492>

[261] J. S. Ward and A. Barker, "Undefined by data: A survey of big data definitions," 2013 [Online]. Available: <https://arxiv.org/pdf/1309.5821.pdf>

[262] B. Wellman and M. Gulia, "Net surfers don't ride alone: Virtual communities as communities," Communities and cyberspace, 1997 [Online]. Available: <http://groups.chass.utoronto.ca/netlab/wp-content/uploads/2012/05/Net-Surfers-Dont-Ride-Alone-Virtual-Community-as-Community.pdf>

[263] C. L. Lei Wang Jianfeng Zhan and B. Qiu, "BigDataBench: A big data benchmark suite from internet services." 20th IEEE International Symposium On High Performance Computer Architecture (HPCA-2014), Orlando, Florida, USA, Feb-2014 [Online]. Available: <https://arxiv.org/pdf/1401.1406.pdf>

[264] S. Rattay, "Profiling algorithms and content targeting - an exploration of the filter bubble phenomenon," Master's thesis, Malmö University, 2014 [Online]. Available: https://muep.mau.se/bitstream/handle/2043/21535/Rattay_Exploratio sequence=2

[265] Wikipedia, "Qlikview." Website [Online]. Available: <https://en.wikipedia.org/wiki/Qlik>

[266] E. Mathews, "QLIKVIEW architecture and system resource usage." Website [Online]. Available: <https://www.quora.com/What-is-QLIKVIEW-architecture-and-system-resource-usage>

[QlikView-and-what-is-the-future-of-ones-career-in-QlikView](#)

- [267] QlikView, “QLIKVIEW architecture and system resource usage,” QlikView complete architecture, 2011.
- [268] Edureka, “Understand the power of qlikview’s click-visualization.” Website [Online]. Available: <https://www.edureka.co/blog/qlikview-tutorial/>
- [269] Shah, “Electronic data capture for registries and clinical trials in orthopaedic surgery: Open source versus commercial systems,” Clin Orthop Relat Res, vol. 10, p. 2664, Jul. 2010.
- [270] Tutorialspoint, “Python - matplotlib.” Web Page, Nov-2018.
- [271] J. D. Hunter, “Matplotlib: A 2d graphics environment,” Computing In Science & Engineering, vol. 9, no. 3, pp. 90–95, 2007.
- [272] CDISC, “Study data tabulation model (sdtm).” Web page, Nov-2018 [Online]. Available: <https://www.cdisc.org/standards/foundational/sdtm>
- [273] C. Mattina, “Bringing drugs to market costs less than previously thought, study finds,” Associated Journal of Managed Care, Sep. 2017 [Online]. Available: <https://www.ajmc.com/newsroom/bringing-drugs-to-market-costs-less-than-previously-thought-study-finds>
- [274] G. Karthik, “Haberman’s Cancer Survival: Visual Exploratory Data Analysis using Python.” Webpage, Mar-2018 [Online]. Available: <https://medium.com/@gokulkarthikk/habermans-cancer-survival-visual-exploratory-data-analysis-using-python-e7dcbae7ac01ed>
- [275] M. D. John Hunter, The Architecture of Open Source Applications, vol. II. lulu.com, 2008.
- [276] I. K. Centre, “IBM cognos business intelligence.” Website, 2017 [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SSEP7J_10.2.2/cor

[277] Wikipedia, “Cognos.” Website, 2007 [Online]. Available: <https://en.wikipedia.org/wiki/Cognos>

[278] I. K. Center, “IBM cognos architecture.” Website, 2014 [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SSEP7J_11.0.0/cor

[279] I. Community, “IBM cognos 8 bi.” Website, 2016 [Online]. Available: https://www.ibm.com/developerworks/community/blogs/8d7e4a2b-2364-4719-9f4e-aa9e24db7465/entry/ibm-cognos-bi-suite?lang=en_us

[280] I. K. Center, “IBM cognos insight.” Website, 2016 [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SSEP7J_10.2.2/cor

[281] I. K. Center, “IBM cognos workspace.” Website, 2017 [Online]. Available: https://www.ibm.com/support/knowledgecenter/SSEP7J_10.2.1/com.it

[282] I. K. Center, “IBM cognos workspace advanced.” Website, 2017 [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SSEP7J_10.2.2/cor

[283] I. K. Centre, “Understanding report studio.” Website, 2017 [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SSRL5J_1.0.1/com

[284] I. K. Center, “Event studio user guide.” Webiste, 2017 [Online]. Available: https://www.ibm.com/support/knowledgecenter/SSEP7J_11.0.0/com.it

[285] BMC, “IBM cognos bi metrics server.” Website, 2015 [Online]. Available: <https://docs.bmc.com/docs/display/Configipedia/IBM+Cognos+BI+Met>

[286] I. K. Center, “IBM cognos query studio.” Website, 2017 [Online].

Available:

https://www.ibm.com/support/knowledgecenter/en/SSEP7J_10.2.2/cor

[287] I. K. Center, "IBM cognos analysis studio." Website, 2017 [Online]. Available:

https://www.ibm.com/support/knowledgecenter/en/SSEP7J_11.0.0/cor

[288] I. K. Centre, "IBM cognos analytics 11.0 documentation." Website, 2018 [Online]. Available:

https://www.ibm.com/support/knowledgecenter/SSEP7J_11.0.0/com.ik

[289] IBM, "IBM watson." Website [Online]. Available:

<https://www.ibm.com/watson/>

[290] I. Watson, "Build with watson." Website [Online]. Available:

<https://www.ibm.com/watson/developercloud/>

[291] IBM, "Getting started with watson analytics." Website [Online].

Available: https://community.watsonanalytics.com/wp-content/uploads/2017/11/wa_tutorial-3.pdf?cm_mc_uid=96448751331815393204336&cm_mc_sid_50200000=7336

[292] I. Cloud, "AI openscale." Website [Online]. Available:

<https://www.ibm.com/cloud/ai-openscale>

[293] I. Watson, "How does ibm watson work." Youtube [Online].

Available: <https://www.youtube.com/watch?v=r7E1TJ1HtM0&t=163s>

[294] I. Cloud, "Watson machine learning." Website [Online]. Available:

<https://www.ibm.com/cloud/machine-learning>

[295] IBM, "IBM watson documentation." Website [Online]. Available:

https://console.bluemix.net/developer/watson/documentation?cm_mc_uid=71151039716715369550371&cm_mc_sid_50200000=1890

[296] I. Watson, "IBM watson services." Website [Online]. Available:

<https://www.ibm.com/watson/products-services/>

[297] I. Watson, "IBM watson health." Website [Online]. Available:

<https://www.ibm.com/watson/health/index-1.html>

[298] I. Watson, "How it works: IBM watson health." Yotube Video [Online]. Available: https://www.youtube.com/watch?v=ZPXF5e1_HI

[299] Wikipedia, "Watson (computer)." Website [Online]. Available: [https://en.wikipedia.org/wiki/Watson_\(computer\)](https://en.wikipedia.org/wiki/Watson_(computer))

[300] S. C. Matz and O. Netzer, "Using big data as a window into consumers' psychology," Current Opinion in Behavioral Sciences, vol. 18, pp. 7–12, 2017.

[301] M. Abadi et al., "Tensorflow: A system for large-scale machine learning." in OSDI, 2016, vol. 16, pp. 265–283.

[302] Datameer, "Six ways to create better customer behavior analytics." online, Feb-2018 [Online]. Available: <https://www.datameer.com/blog/six-ways-create-better-customer-behavior-analytics/>

[303] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in Proceedings of the 10th acm conference on recommender systems, 2016, pp. 191–198.

[304] H.-T. Cheng et al., "Wide & deep learning for recommender systems," in Proceedings of the 1st workshop on deep learning for recommender systems, 2016, pp. 7–10.

[305] xingangzhongzhinao, "A recommendation system based on tensorflow." online, Jan-2018 [Online]. Available: <https://blog.csdn.net/shebao3333/article/details/78966926>

[306] wikipedia.org, "ERP." Web page, Nov-2018 [Online]. Available: https://en.wikipedia.org/wiki/Enterprise_resource_planning

[307] SAP, Inc., "SAP: Software solutions | business applications and technology." Web page, Nov-2018 [Online]. Available: <https://www.sap.com/index.html/>

[308] Saudi ERP & Website Solution Blog, "Complete list of sap erp modules." Web page, Jun-2015 [Online]. Available: <https://solutiondots.com/blog/sap-erp-modules/>

[309] Eshna Verma, "Overview of sap modules." Web page, Nov-2018 [Online]. Available: <https://www.simplilearn.com/sap-modules-sap-fi-sap-co-sap-sd-sap-hcm-and-more-rar111-article>

[310] SAP, Inc., "Business intelligence software." Web page, Nov-2018 [Online]. Available: <https://www.sap.com/products/bi-platform.html>

[311] SAP, Inc., "CRM and customer experience." Web page, Nov-2018 [Online]. Available: <https://www.sap.com/products/crm-commerce.html>

[312] SAP, Inc., "Core hr and payroll." Web page, Nov-2018 [Online]. Available: <https://www.sap.com/products/human-resources-hcm/core-hr-payroll.html>

[313] J. Moura and C. Serrao, "Security and privacy issues of big data," CoRR, vol. abs/1601.06206, pp. 1–2, 2016 [Online]. Available: <https://arxiv.org/ftp/arxiv/papers/1601/1601.06206>

[314] A. Hey, S. Tansley, and K. Tolle, The fourth paradigm: Data-intensive scientific discovery. REDMOND, WASHINGTON: Microsoft Research, 2009 [Online]. Available: <https://books.google.com.my/books?id=oGs\AQAAIAAJ>

[315] V. Yenkar and M. Bartere, "Review on 'data mining with big data,'" International Journal of Computer Science and Mobile Computing, vol. 3, no. 4, pp. 97–102, 2014.

[316] C. O'Neill, "Big data needs big security. Here's why." Imperva, Redwood Shores, California, 2017 [Online]. Available: <https://www.imperva.com/blog/2017/02/big-data-needs-big-security-heres/>

[317] S. Rajan, W. van Ginkel, and N. Sundaresan, "Top ten big data

security and privacy challenges," Cloud Security Alliance, 2012 [Online]. Available:

https://downloads.cloudsecurityalliance.org/initiatives/bdwg/Big_Data

[318] C. Taylor, "Big data security," Datamation, Foster City, California, 2017 [Online]. Available: <https://www.datamation.com/big-data/big-data-security.html>

[319] A. RAHMANI, A. AMINE, and M. R. HAMOU, "A mathematical model of access control in big data using confidence interval and digital signature," Computer Science & Information Technology, vol. 5, pp. 183–198, 2015.

[320] P. Buttler, "Big data needs big security. Here's why." Dataconomy, Berlin, Germany, 2017 [Online]. Available: <http://dataconomy.com/2017/07/10-challenges-big-data-security-privacy/>

[321] Realdolmen, "Cyber security." Belgium, 2017 [Online]. Available: <http://www.realdolmen.com/en/blog/who-responsible-for-data-security-your-company>

[322] P. R. Department, "Ponemon institute© research report." Traverse City, Michigan, 2016 [Online]. Available: <https://www.ponemon.org/local/upload/file/2016%20HPE%20CCC%20>

[323] A. Hamlin, N. Schear, E. Shen, M. Varia, S. Yakoubov, and A. Yerukhimovich, Cryptography for big data security. Boca Raton, Florida: Taylor & Francis CRC Press, 2016, pp. 241–288.

[324] K. Abouelmehdi, A. Beni-Hssane, H. Khaloufi, and M. Saadi, ""Big data security and privacy in healthcare: A review,"" "Procedia Computer Science", vol. 113, no. Supplement C, pp. 73–80, 2017 [Online]. Available:

<http://www.sciencedirect.com/science/article/pii/S1877050917317015>

[325] N. Kshetri, "Big data' s impact on privacy, security and consumer welfare," Telecommunications Policy, vol. 38, no. 11, pp. 1134–1145,

2014.

[326] R. Archana, R. S. Hegadi, and T. Manjunath, "A big data security using data masking methods," Indonesian Journal of Electrical Engineering and Computer Science, vol. 7, no. 2, pp. 449–456, 2017.

[327] O. Tene and J. Polonetsky, "Big data for all: Privacy and user control in the age of analytics," Nw. J. Tech. & Intell. Prop., vol. 11, p. xxvii, 2012.

[328] K. Padmini, "Securing data management based on key technologies in cloud computing," International Journal of Advance Research in Computer Science and Management Studies, vol. 3, no. 2, pp. 165–172, 2015.

[329] G. von Laszewski, F. Wang, H. Lee, H. Chen, and G. C. Fox, "Accessing multiple clouds with cloudmesh," in Proceedings of the 2014 acm international workshop on software-defined ecosystems, 2014, pp. 21–28 [Online]. Available: <http://doi.acm.org/10.1145/2609441.2609638>

[330] R. Kune, P. K. Konugurthi, A. Agarwal, R. R. Chillarige, and R. Buyya, "The anatomy of big data computing," Software: Practice and Experience, vol. 46, no. 1, pp. 79–105, 2016.

[331] A. Bifet, "Mining internet of things (iot) big data streams." in SIMBig, 2016, pp. 15–16.

[332] A. Riahi, E. Natalizio, Y. Challal, N. Mitton, and A. Iera, "A systemic and cognitive approach for iot security," in Computing, networking and communications (icnc), 2014 international conference on, 2014, pp. 183–188.

[333] R. Chatterjee, J. Bonneau, A. Juels, and T. Ristenpart, "Cracking-resistant password vaults using natural language encoders," in Security and privacy (sp), 2015 ieee symposium on, 2015, pp. 481–498.

[334] G. S. Nelson, "Practical implications of sharing data: A primer on data privacy, anonymization, and de-identification," in SAS global forum proceedings, 2015, p. XXIII.

[335] C. N. Shivayogimath, "AN overview of network penetration testing," International Journal of Research Engineering and Technology, vol. 3, no. 7, pp. 408–413, 2014.