Handling Big Datasets for Machine Learning

A.K.SREEJA [1], PREMAJAIN [2]
[1, 2]*Department of Information Science &Engineering, BNM Institute of Engineering & Technology*

Abstract- ***Machine learning with Big Data is, in many ways, different than "regular" machine learning. Big Data is no longer buzzword terminology or cutting edge conceptually; rather, it just is. Big Data is not easily or precisely definable, but it is generally easy to identify when you see it. We are faced with a torrent of data generated and captured in digital form as a result of the advancement of sciences, engineering and technologies, and various social, economic and human activities. This paper presents a review of the challenges of machine learning with big data. Consequently, synthesizing big data frameworks and deep learning is provided. Different types of data sets and perfect data strategy is described. Also the growth of big data and number of tactics that can be used when dealing with very large data files for machine learning is explained.***

Indexed Terms- ***Big data, machine learning, deep learning, Synthesizing, Artificial Intelligence***

I. INTRODUCTION

More than 2.5 quintillion bytes of data are created each day. 90% of the data in the world was generated in the past two years. The prevalence of data will only increase, so we need to learn how to deal with such large data. These Big Data acquire incredible potential in various fields like health care, biology, transportation, energy management, and financial services [1] [2]. While successful applications of machine learning cannot rely solely on cramming ever-increasing amounts of Big Data at algorithms and hoping for the best, the ability to leverage large amounts of data for machine learning tasks is a must-have skill for practitioners at this point. Existing intelligent machine-learning systems are not intrinsically resourceful enough which ends up, in many cases, a mounting portion of this quantity data unexplored and under exploited [3]. A data set is a collection of data. In other words, a data set corresponds to the contents of a single database table, or a single statistical data matrix, where every column of the table represents a particular variable, and each row corresponds to a given member of the data set in question. In Machine Learning projects, we need a training data set. It is the actual data set used to train the model for performing various actions.

Why do I need a data set?
ML depends heavily on data, without data, it is impossible for an "AI" to learn. It is the most crucial aspect that makes algorithm training possible… No matter how great your AI team is or the size of your data set, if your data set is not good enough, your entire AI project will fail! There are many fantastic projects which fail because of not having a good data set despite having the perfect use case and very skilled data scientists.

II. CHALLENGES OF ML ON BIG DATA

Big data creates numerous challenges for traditional ML in terms of scalability, adaptability and usability. Existing work on ML for big data focused on the volume, velocity and variety aspects, but there has not been much work addressing the remaining two aspects of big data: veracity and value. One promising way to handle data veracity is to develop algorithms that are capable of accessing the trustworthiness or creditability of data. The other direction is to develop new ML models that can inference with unreliable or even contradicting data.

Another challenge is that for years, researchers have used machine learning to solve problems without tormenting about whether the situations they were facing met the requirements and the classical statistical assumptions upon which certain methodologies rely [4].With the advent of Big Data,

many of the assumptions upon which the algorithms rely have now been broken, thereby impeding the performance of analytical tasks. In response to those pitfalls, together with the need to process large datasets fast, a number of new machine learning approaches and paradigms have been develop [5]. However, it remains consistently difficult to find the best tools and techniques to tackle specific challenges.

Here are some eye-opening statistics regarding big data:

- More than 16 million text messages are sent every minute
- More than 100 million spam emails are sent every minute.
- Every day, more than a billion photos are uploaded to Google Photos.

Every minute, there are more than a million tinder swipes. Storing this data is one thing, but what about processing it and developing machine learning algorithms to work with it?

III. TYPES OF DATA SET

During Machine learning, we always rely on data. From training, tuning, model selection to testing, we use three different data sets: the training set, the validation set, and the testing set. Validation sets are used to select and tune the final ML model. We might think that the gathering of data is enough but it is the opposite. In every AI projects, classifying and labeling data sets takes most of our time, especially data sets accurate enough to reflect a realistic vision of the market/world.

There are two data sets that we need — the training data set and test data set because they are used for different purposes during AI project and the success of a project depends a lot on them. The training data set is the one used to train an algorithm to understand how to apply concepts such as neural networks and produce results. It includes both input data and the expected output. Training sets make up the majority of the total data, around 60 %. In testing, the models are fit to parameters in a process that is known as adjusting weights. The test data set is used to evaluate how well the algorithm was trained with the training data set. In AI projects, we can't use the training data set in the testing stage because the algorithm will already know in advance the expected output which is not our goal. Testing sets represent 20% of the data. The test set is ensured to be the input data grouped together with verified correct outputs, generally by human verification.

How much data is needed?
All projects are somehow unique that you need 10 times as much data as the number of parameters in the model being built. The more complicated the task, the more data needed.

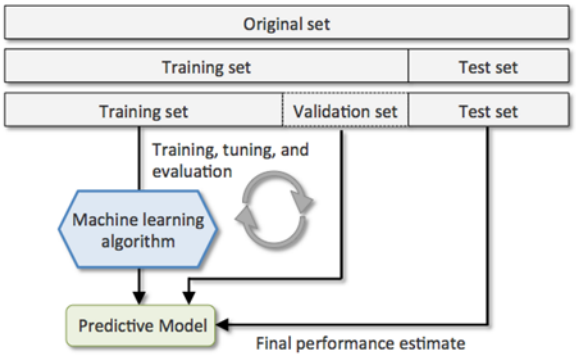

Fig 1.Building dataset for ML

What type data do I need?
What are you trying to achieve through AI? Based on your answer, you need to consider what data you actually need to address the question or problem you are working on. Make some assumptions about the data you require and be careful to record those assumptions so that you can test them later if needed.

I have a data set, what now?
Not so fast! You should know that all data sets are inaccurate. At this moment of the project, we need to do some data preparation, a very important step in the machine learning process. Basically, data preparation is about making your data set more suitable for machine learning. It is a set of procedures that consume most of the time spent on machine learning projects. Even if you have the data, you can still run into problems with its quality, as well as biases hidden within your training sets. To put it simply, the quality of training data determines the performance of machine learning systems.

IV. SYNTHESIZING BIG DATA FRAMEWORKS AND DEEP LEARNING

The majority of big data frameworks software engineers have written in Java whereas the majority of machine learning and particularly deep learning libraries researchers have written in Python. This creates an interesting fault line between the sides. On the one hand we have large data frameworks like Spark, Flink, and Kafka that can rapidly and efficiently process massive datasets, but lack the capability to (easily) train and load the highly successful models aimed specifically at large datasets. On the other side of the equation you have powerful frameworks to easily implement deep architectures; however, these frameworks offer no easy way to integrate these models into large scale data systems. Bridging this fault is not an easy problem as it requires moving code written in Python to the JVM. In addition, there are litanies of dependencies and pre-processing required.

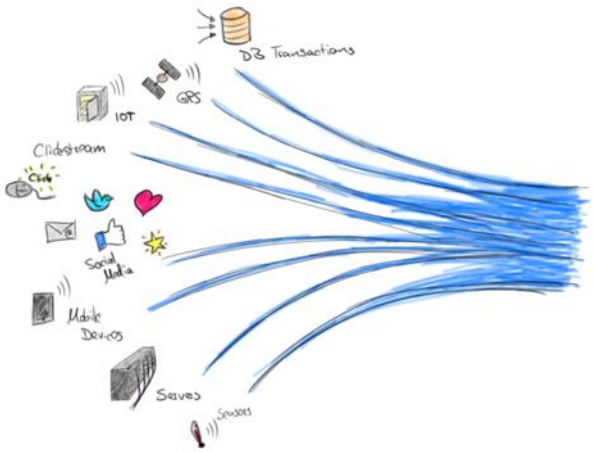

Fig 2.Big data framework and Deep Learning

V. THE PERFECT DATA STRATEGY

The most successful Machine learning projects are those that integrate a data collection strategy during the service/product life-cycle. Indeed, data collection can't be a series of one-off exercises. It must be built into the core product itself. Basically, every time a user engages with the product/service, that we want to collect data from the interaction. The goal is to use this constant new data flow to improve product/service. When we reach this level of data usage, every new customer that we add makes the data set bigger and thus the product better, which attracts more customers, which makes the data set better, and so on. It is some kind of positive circle.

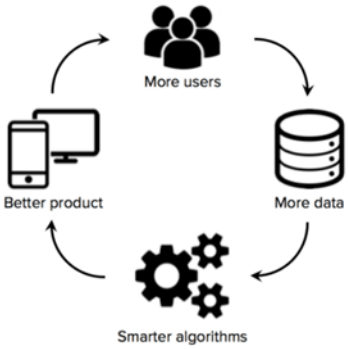

Fig 3.Data Strategy service/product life cycle

- How to deal with large data files for machine learning?

a) Allocate More Memory:
Some machine learning tools or libraries may be limited by a default memory configuration. Check if you can re-configure your tool or library to allocate more memory. A good example is Weka, where you can increase the memory as a parameter when starting the application.

b) Work with a Smaller Sample:
Are you sure you need to work with all of the data?
Take a random sample of your data, such as the first 1,000 or 100,000 rows. Use this smaller sample to work through your problem before fitting a final model on all of your data (using progressive data loading techniques).This is a good practice in general for machine learning to give you quick spot-checks of algorithms and turnaround of results. We may also consider performing a sensitivity analysis of the amount of data used to fit one algorithm compared to the model skill. Perhaps there is a natural point of diminishing returns that you can use as a heuristic size of your smaller sample.

c) Use a Computer with More Memory:
Do you have to work on your computer?
Perhaps you can get access to a much larger computer with an order of magnitude more memory. For example, a good option is to rent compute time on a cloud service like Amazon Web Services that

offers machines with tens of gigabytes of RAM for less than a US dollar per hour.

d) Change the Data Format:
Is your data stored in raw ASCII text, like a CSV file?
Perhaps you can speed up data loading and use less memory by using another data format. A good example is a binary format like GRIB, NetCDF, or HDF. There are many command line tools that you can use to transform one data format into another that do not require the entire dataset to be loaded into memory. Using another format may allow you to store the data in a more compact form that saves memory, such as 2-byte integers, or 4-byte floats.

e) Stream Data or Use Progressive Loading:
Does all of the data need to be in memory at the same time?
Perhaps you can use code or a library to stream or progressively load data as-needed into memory for training. This may require algorithms that can learn iteratively using optimization techniques such as stochastic gradient descent, instead of algorithms that require all data in memory to perform matrix operations such as some implementations of linear and logistic regression. For example, the Keras deep learning library offers this feature for progressively loading image files and is called flow_from_directory. Another example is the Pandas library that can load large CSV files in chunks.

f) Use a Relational Database:
Relational databases provide a standard way of storing and accessing very large datasets. Internally, the data is stored on disk can be progressively loaded in batches and can be queried using a standard query language (SQL). Free open source database tools like MySQL or Postgres can be used, programming languages and many machine learning tools can connect directly to relational databases. You can also use a lightweight approach, such as SQLite. This approach is found to be very effective in the past for very large tabular datasets. Again, you may need to use algorithms that can handle iterative learning.

g) Use a Big Data Platform:
In some cases, you may need to resort to a big data platform. That is, a platform designed for handling very large datasets that allows you to use data transforms and machine learning algorithms on top of it. Two good examples are Hadoop with the Mahout Machine learning library and Spark wit the MLLib library. Nevertheless, there are problems where the data is very large and the previous options will not cut it.

VI. EXPANSION OF BIG DATA

There are many recent examples that can illustrate the tremendous growth in scientific data generation in the literature. It is estimated that there are thousands of wireless sensors currently in place, which generates about a gigabyte of data per sensor per day [6].Besides the environmentalists, a similar challenge facing the climatologists, meteorologists, and geologists today is also making sense of the vast and continually increasing amount of data generated by the earth observation satellites, radars, and high-throughput sensor networks. The World Data Centre for Climate (WDCC) is the world-largest climate data repository, and is also known to have the largest database in the world [7].The WDCC archives 340 terabytes of earth system model data and related observations, and 220 terabytes of data readily accessible on the web including information on climate research and anticipated climatic trends, as well as 110 terabytes (or 24,500 DVD's) worth of climate simulation data. The WDCC data is accessible by a standard web-interface (http://cera.wdc-climate.de). These data are increasingly available in many different formats and have to be incorporated correctly into the various climate change models [8]. Timely and ac-curate interpretation of these data can provide advance warnings in times of severe weather changes, hence enabling corresponding action to be taken promptly so as to minimize its resulting catastrophic damage.

VII. CONCLUSION

This paper has provided a foundation for understanding different ways to handle large data files for machine learning. This big data phenomenon ushers in a new era where human endeavours and

scientific pursuits will be aided by not only human capital, and physical and financial assets, but also data assets. Research issues in machine learning and big data analysis are embedded in multi-dimensional scientific and technological spaces. This paper organizes a review of the challenges of machine learning with big data and specifies types of data set with perfect data strategy. It an attempt to identify the gaps in the work already performed by researchers, thus paving the way for further quality research in handling scalable algorithms for big data.

REFERENCES

[1] W. Raghupathi and V. Raghupathi, "Big data analytics in health care: Promise and potential, HealthInf.Sci.Syst" vol.2, no.1, pp.1-10, 2014.

[2] O. Y. AI Jarrah, P. D. Yoo, S. Muhaidat, G. K. Karagiannidis, and K. Taha, "Efficient machine learning for big data: A review," Bigdata Res., vol 2,no.3,pp.87-93,Sep.2015.

[3] Lina Zhoua, ∗, Shimei Pana, Jianwu Wanga, Athanasios V. Vasilakos," Machine learning on big data: Opportunities and challenges", Elsevier, 12 Jan 2017.

[4] B. Ratner, Statistical and Machine-Learning Data Mining: Techniques for better Predictive Modeling and Analysis of Big Data.Boca Raton, FL: CRC Press,2011.

[5] Alexandra L'Heureux, Katarina Grolinger, and Miriam A. M. Capretz," Machine Learning with Big data: Challenges and Approaches" IEEE Access, Volume 5, June 7, 2017.

[6] Preeti Gupta, Arun Sharma, Rajni Jindal, "Scalable machine-learning algorithms for big data analytics: a comprehensive review",2016.

[7] D.D.P.P. Lamb, R. Jurdak, Csiro ict centre and csiro sensors and sensor networks tcp, online 2009,http://www.csiro.au/.

[8] M. Lautenschlager, Model and Data, Max-Planck-Institute for Meteorology, Hamburg, Germany.