UDK 004.75

# ANOMALY DETECTION USING AUTOENCODER FOR DATA QUALITY MONITORING IN CLOUD

***C.S. Dzik***
*Utech Solutions, software and data engineer*

***I.I. Pilecki***
*Candidate of physico-mathematical sciences, Senior scientific researcher, Associate Professor of Informatics Department of the BSUIR*

*Utech Solutions LLC, USA*
*Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus*
*E-mail: constantine.dzik@gmail.com*

**Abstract.** Many works are dedicated to solving the data quality problem, a number of standards have been developed, but the problem has not been solved for decades. Moreover, the problem now requires a more complex solution due to processing large amounts of unstructured data in the cloud. This work presents the original project Autoencoder that focuses on the technology of analysis, detection and forecasting poor-quality data transmission based on machine learning and the use of neural networks.

**Keywords:** anomaly detection, autoencoder, artificial neural network, MLP, deep learning, AWS, S3, unsupervised learning.

***Introduction.*** In today's world, cloud computing is a rapidly evolving technology that many organizations are adopting to enable their digital digital transformation. Cloud technology is opening up new competitive opportunities for companies globally and re-defining how they do business. The cloud makes resources, applications, platforms, and data available anytime, anywhere. Transferring data from outdated systems to the cloud enables you to scale your business, make your data productive, and make it more accessible.

Today most of the company's operations and strategic solutions rely heavily on the cloud data, so data quality is becoming an increasingly important characteristic. Data quality issues that arise when data and data applications are transferred to the cloud have a particular position among the challenges companies face. Cloud computing assumes new types and resources for potential data quality errors. In general, poor quality data can affect productivity, total and overall return on investment.

***Materials and Methods.*** Before discussing data quality errors in the cloud, let's define what the term "data quality" means.

According to data quality experts, data is of high quality when it satisfies the requirements of its intended use. In other words, companies know that they have good quality data when they are able to use it to communicate effectively with their constituents, determine clients' needs, and find effective ways to serve their client base [5-8].

This data quality definition is broad enough to help companies with varying products, markets, and missions to understand if their data is up to standards.

Data quality is not good or bad, high or low. It is a range or an indicator of operability of the data that passes through a company. Data quality management ensures the context-dependent process of improvement of suitability of the data, which is used for analysis and decision-making. The goal is to provide the vision of the "health" of the data by applying different processes and technologies to the increasingly complex data sets [5-7].

Quality of Data for business:

1. Consistency,
2. Completeness,
3. Reliability,
4. Accuracy,
5. Actuality

Data Quality as a process at the company level:

1. Analysis
2. Data profiling
3. Establishing data quality target indicators
4. Design and development of the data quality rules
5. Monitoring the correspondence of the actual data quality indicators to the target indicators
6. Improvement
7. Implementation of the rules in the data integration platform

First, good data quality management provides the framework for all the business initiatives, establishes the structure for all the units of the company for ensuring compliance with the data quality rules.

Second, accurate and up-to-date data provide a clear idea of the day-to-day operations of your company; therefore, you can be sure of the top and bottom indicators that all the data use. Data quality management also reduces unnecessary expenses.

Third, data quality management for compliance with the requirements and risk objectives. Good data management requires clear procedures and communication as well as good basic data

In our data quality study we focus on the data quality availability attribute.

The term "data availability" refers to the ability to ensure that the required data are always available at a particular location and time in an organization's IT infrastructure, even if an error occurs. In practice, this means that the data that are not available if necessary parameters are useless. In fact, the situation described is even worse as data become more than just useless: if systems are configured on the basis of data availability, a catastrophic chain response of a system failure can occur; the data that users are counting on are missing, or worse, the data are in the system, but they are out of date or otherwise compromised [5-8].

Availability is defined as the degree of convenience for users to obtain data and related information, which is divided into the three elements of accessibility, authorization, and timeliness.

Availability indicators:

1. Within a given time, whether the data arrive on time
2. Whether data are regularly updated
3. Whether the time interval from data collection and processing to release meets requirements

Instead, we want to be sure that when we take advantage of the cloud to help data managing, we define data quality parameters at the same time.

The most obvious and compelling way to achieve the goal is to make sure we perform automatic data quality checks for all our data, wherever they are - in the cloud or elsewhere. We must always perform an on-site data quality check.

Virtually every company that works with data has a certain data quality (DQ) monitoring system. Some companies even hire an entire department that deals with the issue. This option is very expensive. In addition, most data quality checks are hard-coded and rule-based. In the event of a failure, the system notifies you of the risk indicator. Such rules are often critical to business continuity. For example, we cannot have a missing customer ID or a "risk profile" variable with an incorrect value. As the amount of data grows, you cannot specify a rule for work with each attribute; not to mention the difficulty of working with hard-coded multidimensional control checks.

The best option is automated DQ (data quality) checks using Machine Learning to detect anomalies that we don't even need to explicitly program.

Anomaly detection (or outlier detection) is the identification of rare items, events or observations which raise suspicions by differing significantly from the majority of the data. Typically, abnormal data may be associated with a problem or rare event such as data quality, bank fraud, health problems, structural defects, faulty equipment, etc. This relationship is interesting in terms of the possibility to identify data points that can be considered anomalies, since the detection of such events is interesting from the point of view of sustainable business development [1].

This brings us to one of the key goals of this study: how to determine whether data points are normal or abnormal? In some simple cases, this can be determined at once (see figure below):

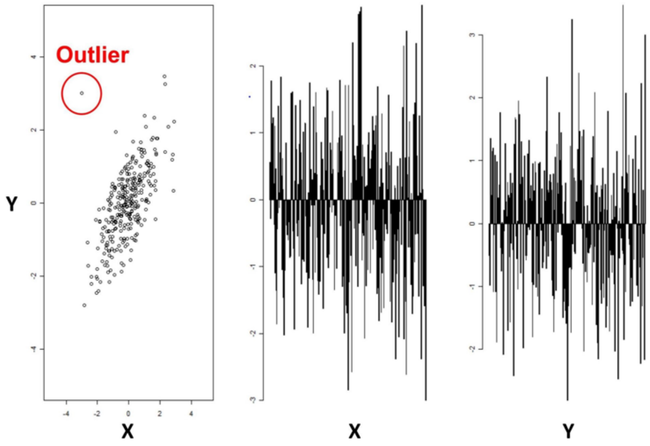


Figure *1.* – Anomaly detection for two variables

Let us consider the case of two-dimensional data (X and Y): it is quite easy to visually identify anomalies through data points located outside the typical distribution. However, looking at the numbers on the right, it is not possible to identify the outlier directly from investigating one variable at the time.

An autoencoder is a type of artificial neural network used to learn efficient data codings in an unsupervised manner. The aim of an autoencoder is to learn a representation (encoding) for a set of data, typically for dimensionality reduction. Along with the reduction side, a reconstructing side is learnt, where the autoencoder tries to generate from the reduced encoding a representati on as close as possible to its original input [3].

In terms of architecture, the simplest form of an autoencoder is a feedforward, non-recurrent neural network very similar to the many single layer perceptrons which makes a multilayer perceptron (MLP) – having an input layer, an output layer and one or more hidden layers connecting them – but with the output layer having the same number of nodes as the input layer, and with the purpose of reconstructing its own inputs.

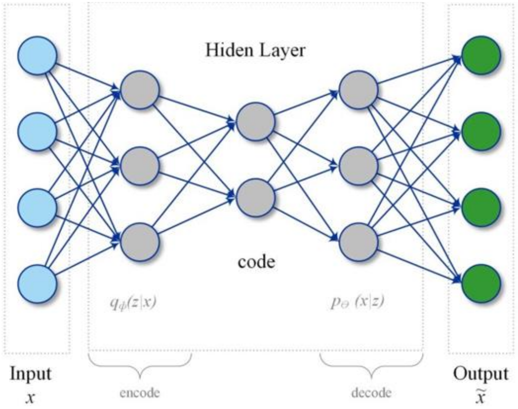


Figure *2*. – Autoencoder network

Autoencoder anomaly detection – solve data quality issues without specifying them.

Autoencoders are neural networks that model their own input. In other words, autoencoders try to learn the identity function. They consist of two parts. Firstly, the encoder reduces the dimensionality of the input data. After that, the decoder tries to reconstruct the input from these reduced dimensions. Autoencoders are relevant for DQ because they can model whole data tables. They map all relevant fields of a data table to the input and the output layer of the network. Hidden layers between input and output layers learn the regular behaviour of the data. Of course, we need to have “good” data to train the model [2].

Imagine something unusual happened to the data, and we didn’t expect that. Even if the system conforms to all the classic hard-coded rules, an autoencoder trained to handle regular data will be completely malfunctioning and predicting incorrect data outputs. We will observe a large reconstruction error – a significant difference between predicted and actual values – and detect a data anomaly. The data doesn’t behave in the way that the autoencoder learned.

Such function is very useful when something new is happening in a dataset. For example, there are suddenly much more missing values (NA) than usual, or levels of categorical variables change or shift. The autoencoder allows you to omit the encoding of these rules. This feature also helps you to identify the source of the problem. The described properties of autocoders allow the detection of anomalies at an early stage of their occurrence and reduce the cost of manual research [4].

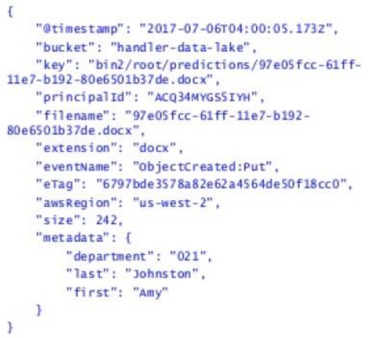
***Approach and Results.*** Neural Network Architecture [3]:

1. Trained with Keras/TensorFlow using Adam - an optimized version of backpropogation
2. Trained to minimize mean squared error between input vector and the output one:

$$MSE = \frac{1}{n}\sum_{i=0}^{n}(X_i - \hat{X}_i)^2$$

3. Three hidden layers, with number of units.
4. Activation functions: Tanh and ReLu

S3 object metadata:



```
{
    "@timestamp": "2017-07-06T04:00:05.173Z",
    "bucket": "handler-data-lake",
    "key": "bin2/root/predictions/97e05fcc-61ff-
11e7-b192-80e6501b37de.docx",
    "principalId": "ACQ34MYGS5IYH",
    "filename": "97e05fcc-61ff-11e7-b192-
80e6501b37de.docx",
    "extension": "docx",
    "eventName": "ObjectCreated:Put",
    "eTag": "6797bde3578a82e62a4564de50f18cc0",
    "awsRegion": "us-west-2",
    "size": 242,
    "metadata": {
        "department": "021",
        "last": "Johnston",
        "first": "Amy"
    }
}
```

Figure *3.* – Programm cod

Data preparation takes the input metadata and output the same records with additional features extracted from the input attributes. Features example: unique id for each file, file size, file size percentage change, timestamp etc.

The general idea behind unsupervised anomaly detection approaches is to find an approximate model that can capture the normal behavior of complex systems. The approximate model can then be used to flag anomalies if the deviation of the predicted behaviors of the trained model from the actual observation exceeds some certain threshold. Training on the normal data, the autoencoder is expected to produce higher reconstruction error for the abnormal inputs than the normal ones, which is adopted as a criterion for identifying anomalies (Figure 3). On Figure 4 we have two columns – time range difference and file size difference. Orange line represent the training curve, while blue line represents new input curve. Green dots indicate normal data points, while red dotes indicates potential anomalies. As we can see on Figure 4, unsupervised anomaly detection approaches Autoencoder allows us to identify unusual patterns and behaviors.

***Conclusion.*** The main purpose of unsupervised deep learning suggested approach is ability to detect anomalies based on load patterns for data files in S3. By using metadata from Objects in S3 and performs transformations and formatting on top of the metadata so it can be consumed by Autoencoder Artificial neural network. In the future our plan to apply the proposed technique for various applications. Also the plan to conduct a more in-depth theoretical analysis of the proposed technique.

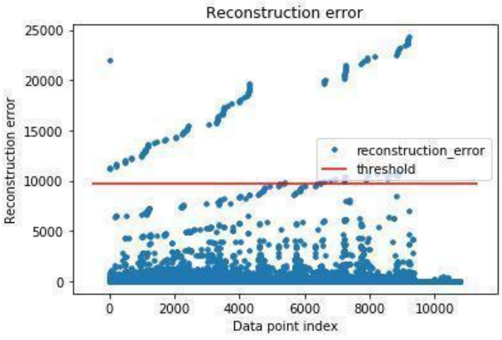


*Figure 4.* – Model allows to detect anomalies

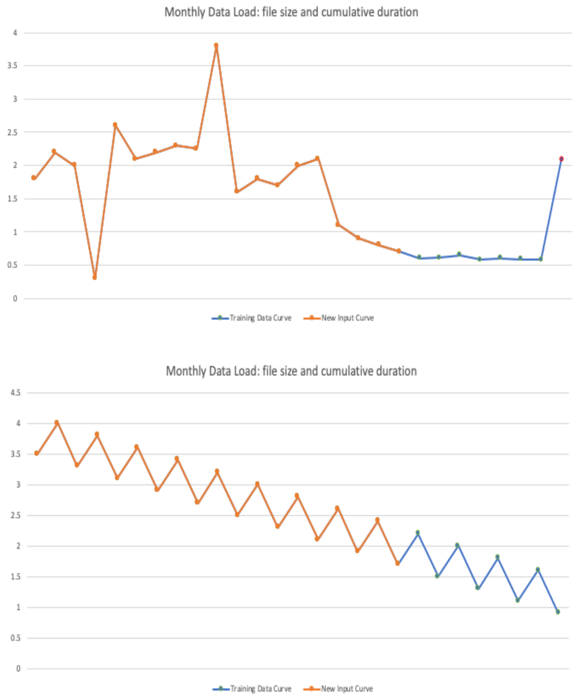


*Figure 5.* – Identify unusual patterns and behaviors

Benefits of the offered model:

1. Unsupervised learning: no need to know the labels of anomalies before learning. We are using an artificial neural network - Autoencoder algorithm. That allows us to identify unusual patterns or behaviours and label output with the results.
2. Able to take all factors that may cause anomalies into count at the same time.
3. Able to take multiple files together to run at the same time.

### ***References***

[1]. Jason Brownlee, Deep Learning for Natural Language Processing, 2017.

[2]. Jason Brownlee, Machine Learning Mastery With Python Understand Your Data, Create Accurate Models and Work Projects End-To-End, 2016.

[3.] Giancarlo Zaccone, Md. Rezaul Karim, Ahmed Menshawy, Deep Learning with TensorFlow, 2017.

[4]. Chun Chat Tan, AUTOENCODER NEURAL NETWORKS: A Performance Study Based on Image Reconstruction, 2009.

[5]. ISO 8000-2.

[6]. ISO/TC 8000-110.

[7]. ISO/TC 8000-120.

[8]. https://en.wikipedia.org/wiki/Data_quality.