

Poseidon – water infrastructure status prediction system for communities

DHAVAL BHAILALBHAI PATEL, University of Regina, Canada

The project Poseidon leverage the data science and machine learning techniques to develop a novel solution for water crisis problem by predicting the status of water infrastructure of different communities across Tanzania. The developed model is deployed on a django based webapp to increase its global outreach. The project is develop to support UN sustainable goals of clean water and sanitation, good health and well-being, partnership for goals and sustainable communities and cities. One of the proposed method predicts the status of water infrastructure with training precision of 99 percent.

CCS Concepts: • **Software and its engineering** → Community oriented Design; Designing software; • **Human-centered computing** → Collaborative and social computing ; • **Computing methodologies** → Machine Learning; Parallel computing methodologies.

Additional Key Words and Phrases: Stewardship, Separating the concerns, Mis-information handling, Digital Habitat, Agile SDLC model with Bazaar Approach, Data transformation into knowledge, Gamification, sustainable development

ACM Reference Format:

Dhaval Bhailalbhahi Patel. 2021. Poseidon – water infrastructure status prediction system for communities . 1, 1 (June 2021), 7 pages. <https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

1 INTRODUCTION

The key objective of this paper is to propose solution for solving water crisis issue in African country of Tanzania. According to UN report on water for life[1], more than half of the people living in Tanzania are without safe drinking water, there is mind boggling funding gap of 61 percent to develop current infrastructure, an average person has to travel more than 30 minutes to get access to clean drinking water and 4000 children death every year from water born disease. For a country like Tanzania facing major water crisis better managing their current water infrastructure is of paramount importance. The proposed solution revolves around the **community centered design** using **false-consensus effect** where solution goal is to design digital habitat for actual needs of the community. The remainder of this document is presented as follows: section 2 presents the proposed methodology; section 3 presents results and section 4 discuss the future work and conclusion.

1.1 Community and stakeholders

To create a digital habitat that is habitable and thriving for its users a comprehensive study was under taken to understand communities in Tanzania and their orientations. The inference from the study [2] are, the current water infrastructure is managed by ministry of water, the **north star customers** for project is communities across Tanzania, the computer literacy rate is low in the

Author's address: Dhaval Bhailalbhahi Patel, dpf761@uregina.ca, University of Regina, 3851 Retallack Street, Regina, Saskatchewan, Canada.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2021 Association for Computing Machinery.

XXXX-XXXX/2021/6-ART \$15.00

<https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

country thus we expect bulk of the customer base comprised of **lurkers**, the future work suggests creation of chat forum for more **prosumer-based** collaborative approach moving forward and welcoming contributions from **innovators**. Stakeholders includes communities of Tanzania as NorthStar customers, water infrastructure management Government agency, various local and global NGO's.

1.2 Tools and Technologies

The software development life cycle for project Poseidon is based on **Agile** approach. The agile SDLC for this project consist of **3 sprints of one week each resulting in 3 MVP**. Using the **Bazaar ideology**, technology and license used to develop the **end-end solution** were **open-source** for supporting collaborative work. This technology includes python programming language and its various open-source modules, Anaconda development environment, Jupiter notebook Kernel running on Visual Studio code editor, createely and photo-shop for creating documentation. GitHub for version control, Django library deployed in anaconda virtual environment for hosting web application. The approach uses machine learning algorithm of random forest to predict the status of water infrastructure of the community in Tanzania.

2 PROPOSED METHODS

Every machine learning model needs data to make prediction. To leverage the machine learning technique to transform **raw data into knowledge** we required a non-synthetic dataset to solve real-world problem and not to limit proposed method by using synthetic data. After browsing for the dataset on various data platforms like Kaggle, data driven, etc. The selected dataset that was used to develop the predictive model was based on data available from Government of Tanzania - Ministry of water and hosted by data driven organization as a **public dataset**. This selection was in accordance of bazaar approach used in this project. For the purpose of **separating the concerns** at various development levels prediction model and web interface are developed differently and if there are any future improvement in the model it can be easily deployed on webapp by simply uploading the model file into webapp. This feature also consider **UFFFAA** [4] approach in which flagging the shortcoming of the current model can be overcome by framing knowledge, fixing knowledge and later assuring knowledge in later versions of upcoming models also preserving knowledge from previous versions as well.

2.1 Dataset

The dataset comprised of 40 features with total of 74,000 listed rows of data in total. The 40 features include, amount-tsh, date-recorded, funder, gps-height, installer, longitude, latitude, wpt-name, num-private, basin, subvillage, region, region-code, district-code, lga, ward, population, public-meeting, recorded-by, scheme-management, scheme-name, permit, construction-year, extraction-type, extraction-type-group, extraction-type-class, management, management-group, payment, payment-type, water-quality, quality-group, quantity, quantity-group, source, source-type, source-class, waterpoint-type, waterpoint-type-group. The data provided is in **raw CSV format** with many missing values for the features like funder, scheme-name and permit which contributed 3635, 28166 and 3056 values respectively.

The label to be predicted has three possible values

- functional: the waterpoint is operational and there are no repairs needed
- functional needs repair: the waterpoint is operational, but needs repairs
- non functional: the waterpoint is not operational

2.2 Feature Engineering

The feature engineering plays an important role in designing the model. Basically, all machine learning algorithms use some input data to create outputs. This input data comprises of features, which are usually in the form of structured columns as discussed in dataset. Algorithms require features with some specific characteristic to accurately predict the result.

The main objective for using feature engineering can be summarized as follow

- To prepare the proper input data set compatible with the machine learning algorithm requirements
- For improving the performance of machine learning model

To develop additional input parameters in the data features were engineered. Some of these features are

- Age of the pump is a key feature but it was missing from the original dataset which was computed using the values from date-recorded and construction year
 $\text{features['age']} = \text{features['date-recorded']} - \text{features['construction-year']}$
- Population served per age of the pump
 $\text{features['pop/year']} = \text{features['population']}.replace(0:1) / \text{features['age']}$

2.3 Imputation

Imputation is the process of replacing the missing values. They pose a huge challenge in creating data pipelines and adversely affect the accuracy of the model. The missing value in the selected dataset was replaced by mean values of respective columns which also helps to create normal distribution of values for these features which is ideal for obtaining better overall accuracy.

2.4 Encoding

The encoding is performed for categorical data as it is difficult for any machine learning algorithm to understand the categorical data. This process simply changes the data with cardinality more than 150 values into numerical format thus enabling grouping of data without any data loss and rest of the low cardinality variables were ordinally encoded in with string values were simply replaced by a numeric value representing that class of value. This was performed to build pipelines architecture using SKlearn module.

2.5 Scaling

The scaling helps to bring different values to the same numeric range so that a machine learning algorithm can compare different features and identify important features from the feature group. For scaling, transform functionality of SKlearn module was used.

2.6 Parallely Implemented random forest classifier model

The prediction model is based on random forest classifier algorithm implemented using parallel programming. The model is hyper-tuned using gridsearchCV method of SKlearn. The hyper-tuned parameters for the model only consist of max depth thus resulting cluster of decision trees with max depth 25. There are many methods available for hyper-tuning but to make the model simpler and decrease the training time only max depth was used as tuning it drastically impacted the accuracy. The n-tier which controls the number of combinations is set to 5 thus covering a wider search space. The value of CV is set to 5 for stopping model from over-fitting the data.

The project has two versions of model with same hyper-tuned parameters.

- **Version 1**[Deep Model] was trained using 40 input parameters developed in Sprint 1. It has 322 computed important features to make predictions

- **Version 2**[Shallow model] was trained using 8 input parameters developed in Sprint 2 It has 9 computed important features selected using base model evaluation by using SAS miner tool to make predictions.

2.7 Flow Diagram of the proposed method

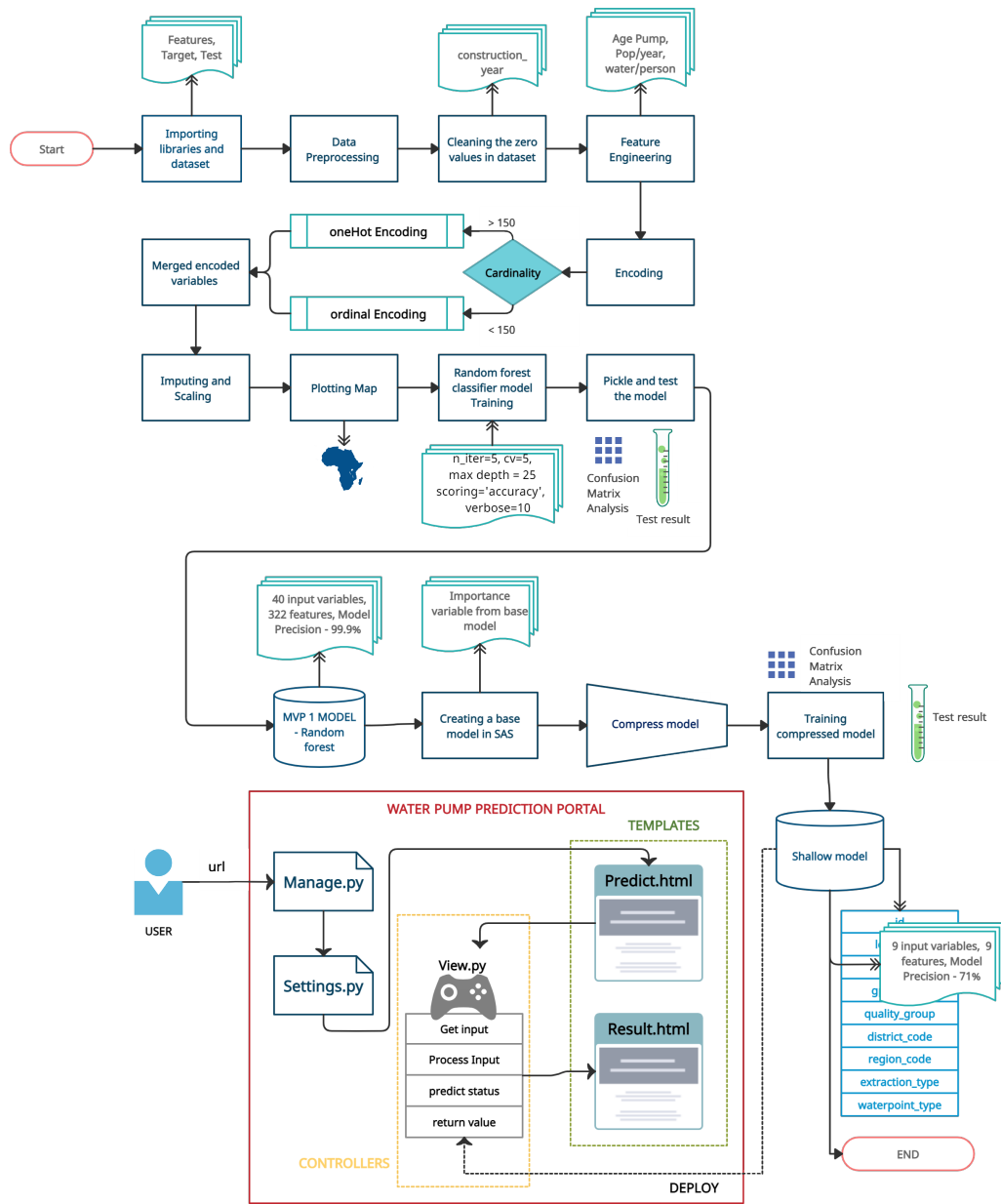


Fig. 1. Flow Diagram of project Poseidon

2.8 Web Portal

To increase the global outreach of the model and to create a software based on **people centric approach** instead of product-based approach, I have deployed the prediction model on a django based web application so as to test the model with real world data. This approach would also help to improve the product based on feed backs from people about its accuracy.

The django webapp have 2 primary components controller and views. Users interact with the views implementing controller as a backend. The concept of **separations of concern** is used to separate the model from the controller code. This result in a smooth mechanism to deploy and test models with different configuration and hyper-tuning parameters with ease. **Version 2** model was considered for testing purpose for the application due to time constraints between each sprints.

2.9 Map plot

To create a better user experience the status of the pump was plotted using geopandas and matplotlib modules of python on map of Tanzania. The red spot represent water pump that are not functional, green represent water pump that are functional and yellow represent water pump that are functional but need repair

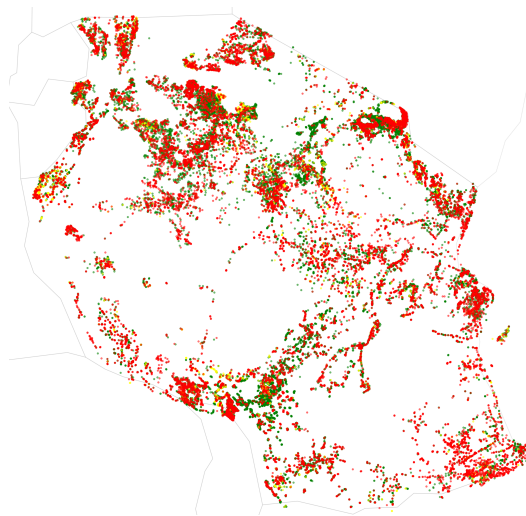


Fig. 2. Plot illustrating status of various water pump across Tanzania

3 RESULTS

The developed models were compared using confusion matrix analysis. The criteria used for evaluation of the developed models are precision, recall, f1 score and accuracy.

We can observe from table 1 and table 2 that version 2 have lower accuracy than version 1. It was expected as version 2 only uses 8 features to predict the value as compared to original 40 features used by version 1. We can observe a **trade off between accuracy and model size** while comparing accuracy of the two versions of model for this dataset.

The front end design of the web application as displayed in **figure 3** is responsive and designed to minimise **gulfs of interaction** for users. The website has a simple form design with precursors for aiding in filling the form. The button clearly state what action is to be expected once it is clicked.


Table 1. Version 1 [Deep Model] Training evaluation matrix

Predicted outcome	Precision	Recall	F1-score	Accuracy
Functional	0.98	1.00	0.99	0.99
Needs Repair	1.00	0.95	0.98	0.99
Not-Functional	1.00	0.99	0.99	0.99

Table 2. Version 2 [shallow Model] Training evaluation matrix

Predicted outcome	Precision	Recall	F1-score	Accuracy
Functional	0.60	0.97	0.74	0.63
Needs Repair	0.00	0.00	0.00	0.63
Not-Functional	0.82	0.27	0.41	0.63

The predicted result clearly provide prediction with accuracy of the information thus handling **misinformation** in the application.



WATER PUMP STATUS PREDICTOR

💧 what is the status of the water pump with your input characteristics ? 💧

Predict

✖ NON FUNCTIONAL [Precision 71 percent]

Fig. 3. User interface of web portal

4 FUTURE WORK AND CONCLUSION

The present document provide insight into predictive machine learning approach to provide a novel solution for solving water crisis in Tanzania. The proposed method discuss various design aspects and technology concepts used to develop two versions of predictive model and webapp used to deploy these models. The result from the study prove that version 1 - Deep model has better accuracy than version 2 - Shallow model.

The future work include functionality of chat forum, announcement page for NGO, Donate page for the community, Award section for recognizing efforts from community members, live sensor data integration for providing real time status prediction and deploying a deep model on web interface. Using concepts of gamification i.e. guessing which pump will be non operational today, can help to rise awareness in the community to forge a stronger resolution in working towards solving water crisis through collaborative efforts of communities.

ACKNOWLEDGMENTS

- To Government of Tanzania - ministry of water for providing the dataset.
- To Datadriven organization for hosting the dataset
- To Dr.Timothy Maciag for reviewing my idea.

REFERENCES

- (1) United Nations. (n.d.). Water for Life. Retrieved June 20, 2021, from <https://www.un.org/>
- (2) Patel, D. (n.d.). Activity 1 - ENSE 885. Retrieved June 20, 2021, from <https://github.com/Dhaval-B-Patel/ENSE-885—spring-2021/blob/522763601394b79914e59ee50c9aba06bdad1a8d/Activity>
- (3) Macaig, T, 2021, Class Notes, People Centered Design, University of Regina, delivered May 2020
- (4) Macaig, T, 2021, Class Notes, Topics in Computer-Supported Collaborative Work, University of Regina, delivered May 2021
- (5) Towards Data Science . (n.d.). Fundamental Techniques of Feature Engineering for Machine Learning. Retrieved June 20, 2021, from <https://towardsdatascience.com/feature-engineering-for-machine-learning-3a5e293a5114>
- (6) Towards Data Science . (n.d.). Hyperparameter Tuning the Random Forest. Retrieved June 20, 2021, from <https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74>