AN APPROACH FOR REAL-TIME VALIDATION OF THE LOCATION OF BIODIVERSITY OBSERVATIONS CONTRIBUTED IN A CITIZEN SCIENCE PROJECT

Maryam Lotfian [1,2]*, Jens Ingensand[1], Maria Antonia Brovelli[2]

[1]University of Applied Sciences and Arts Western Switzerland,School of Business and Engineering,
School of Business and Engineering Vaud, Institute INSIT, 1400,Yverdon-les-Bains, Switzerland.
(maryam.lotfian, jens.ingensand)@heig-vd.ch

[2]Department of Civil and Environmental Engineering, Politecnico di Milano,
Piazza Leonardo da Vinci 32,20133 Milano, Italy.
(maryam.lotfian, maria.brovelli)@polimi.it

Commission IV, WG WG IV/4

KEY WORDS: Citizen Science, Machine Learning, Automatic Data Validation, Real-Time Feedback, Biodiversity

ABSTRACT:

The number of citizen science (CS) projects has grown significantly in recent years, owing to technological advancements. One important aspect of ensuring the success of a CS project is to consider and address the challenges in this field. Two of the main challenges in CS projects are sustaining participation and improving the quality of contributed data. This research investigates how incorporating Machine Learning (ML) into CS projects can help to address the aforementioned challenges. A biodiversity CS project is implemented to accomplish this, with the goal of collecting and automatically validating location of observations, as well as providing participants with real-time feedback on the likelihood of observing a species in a specific location. The findings indicated that, on the one hand, automatic data filtering simplifies data validation, and on the other, real-time feedback can increase volunteers' motivation to continue contributing to a CS project.

1. INTRODUCTION

The participation of general public in scientific projects, known as Citizen Science (CS), has been around for centuries, however the term CS was coined in the 1990s and has grown in popularity since then [Vohland et al., 2021]. Since a large number of CS projects involve public data collection for scientific projects, CS aims to assist the members of academic institutes in obtaining data and information from citizen scientists that would otherwise be difficult to obtain [Cohn, 2008].

Recent advances in technology, particularly mobile technology, has resulted in development of a large number of CS mobile/web applications in various domains [Schade and Tsinaraki, 2016]. This increase in the number of CS projects has resulted in the collection of large amounts of data [Dalby et al., 2021] in many fields, most notably biodiversity [Kullenberg and Kasperowski, 2016]. Regardless of the increase in the number of CS projects, one of the most important aspects of CS is knowing how to keep a CS project successful. Thus, two major questions must be addressed: How to motivate citizens to contribute to CS (public engagement)? and Is the data collected useful for scientific projects (data quality)? Several studies have been conducted to answer these two questions [Lotfian et al., 2020], yielding interesting outcomes and frameworks for others to consider before designing their CS project. However, the aforementioned questions continue to be a source of concern in CS projects, with researchers looking for new approaches to finding answers. Machine Learning (ML), which presents new opportunities in CS, is a recent focus for answering these questions [Lotfian et al., 2021,Ceccaroni et al., 2019].

One of the major challenges in ML is a lack of sufficient labeled data to train the algorithms [Keshavan et al., 2019]. As previously stated, the growth of CS leads to big data collection, which can be a way of addressing the lack of sufficient data for ML algorithms, thereby forming a partnership between CS and ML. Nevertheless, aside from CS's assistance in providing input data for ML algorithms, what are the benefits of this partnership for CS? The majority of studies in which CS and ML are combined, have focused on using citizens' contributions to collect labelled data for ML algorithms, but to the best of our knowledge, only a few studies have focused on using ML algorithms to address challenges in CS projects.

Keeping the foregoing in mind, in this research we aim at focusing on the integration of ML in CS towards sustaining participation and automating data validation. To do so, a biodiversity CS project is implemented with the goal of collecting and automatically validating the location of species observations using species distribution models generated with ML algorithms. Furthermore, in this project, real-time feedback is generated for participants as a result of machine predictions, such as the likelihood of observing a species in a specific location and species habitat characteristics. The goal is to analyze the effect of real-time machine-generated feedback on motivating participants to continue contributing to CS projects. Finally, a user experiment is carried out to evaluate this approach.

The following is how this article is structured: The following section presents data quality in CS. Following that, biodiversity data validation using species distribution modeling is presented, followed by a presentation of our case study and the results of our user experiment. The main findings and conclusions are then presented.

* Corresponding author

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-4-W1-2022-271-2022 | © Author(s) 2022. CC BY 4.0 License.

2. DATA QUALITY IN CITIZEN SCIENCE

When it comes to data quality assurance, several factors must be considered, such as accuracy, timeliness, completeness, accessibility and so on. The literature on data quality in CS is mostly project-specific, and a framework or general guidance on dealing with data quality is lacking, even in projects in similar domains [Balázs et al., 2021]. Balazs et al. [Balázs et al., 2021] argue that in order to reuse data from CS projects, a protocol for ensuring a minimum standard of data quality across different CS projects is required. The authors define four aspects to evaluate CS data: data quality, data contextualization, data reuse, and data interoperability. Data quality refers to ensuring the validity and reliability of the data, data contextualization refers to communicating how a specific data set is created, for example by providing metadata, data reuse refers to clarification on data ownership and future accessibility, as well as using open data and open standards, and finally data interoperability refers to the development of a standard system to simplify data reuse across various projects and systems.

Data quality assurance or data validation in CS projects is mainly done by experts in the field [Adriaens et al., 2021]. While the use of expert knowledge is critical in a CS project, relying solely on expert review has its own drawbacks, such as the validation task being time-consuming, lack of sufficient volunteer experts, and a large time gap between the moment volunteers make a contribution and receiving feedback (if they receive any) that can demotivate volunteers. Although there are new ways to automate data validation, this is still in its early stages, and more research should be conducted to determine how to optimize data validation automation or which factors are more important in data validation automation. As mentioned, researchers are increasingly focusing on the use of ML to address a variety of scientific challenges, and this is also becoming a recent focus in CS projects. Therefore, in the section that follows, we present how we used ML algorithms to validate biodiversity observations in our case study.

3. BIODIVERSITY DATA VALIDATION USING MACHINE LEARNING

The goal of this section is to present how to validate biodiversity observations (with a particular emphasis on birds species) by generating species distribution models using existing data from other CS projects (e.g., eBird). Before introducing our case study, we will first go over the specifics of species distribution modeling, explaining what it is, how it can be generated, and what types of data are required.

3.1 Species Distribution Modeling (SDM)

SDM is a class of numerical models that explain how the presence or absence of a species at a given location is related to environmental (e.g. temperature, precipitation, etc.) and landscape characteristics (e.g. landcover, elevation, slope, etc.) [Elith and Leathwick, 2009]. SDM was initially based on linear regressions, but as modeling advances and new algorithms have been introduced, it has advanced to use new modeling techniques and algorithms [Wintle et al., 2005]. The same is true for advancements in the data used to generate SDM. Initially, ecologists had access to limited geospatial data such as latitude, longitude, or elevation; however, advances in GIS and the availability of new tools and software such as widely accessible satellite images, the possibility of obtaining 3D terrain models, and so on have made it easier to obtain a broader range of data to use for SDM [Elith and Leathwick, 2009]. To generate SDM two sets of data are required:

- Species occurrence data: the locations (often point-based) where the species has been observed, which are collected in a variety of ways, including natural museum records, field observations by biologists, and crowdsourcing and CS projects.
- Environmental variables: environmental variables include both climate data such as temperature and precipitation as well as landscape characteristics such as elevation, slope, soil type, land cover, and so on.

The steps to generate SDM are as follows:

1) Data preparation

2) Choose an algorithm

3) Feed and train the algorithm using the input data

4) Evaluate the performance of the algorithm

5) Predict species distribution over the whole study area

The species data set contains both presences (where the species is observed) and sometimes absences (where the species is not observed). Some algorithms only require presence data, whereas others require both presence and absence data. Because obtaining true absences is extremely difficult, one approach is to generate pseudo-absences (or artificial absences).

SDM algorithms are classified into four types: profile models, statistical regression models, ML models, and geographical models. Each of these categories contains one or more algorithms for investigating the species-environment relationship. Statistical models and machine learning models have received the most attention in the literature of these four categories. In this research we have generated SDM using the four algorithms of Naive Baysian (NB), Random Forest (RF), Balanced RF, and Deep Neural Network (DNN).

DNN: Artificial Neural Network (ANN) refers to the algorithms which are inspired by the interconnected networks of neurons in biological brain [Abraham, 2005]. The processing elements of neural networks known as nodes or artificial neurons, receive input signals and using the connection weights the output is generated [Abraham, 2005]. The network improves learning each time by adjusting the weights. The architecture of a ANN includes an input layer, one or more hidden layers, and an output layer. DNN is an ANN with multiple hidden layers.

RF: RF [Breiman, 2001], as the name suggests, is an ensemble model of several decision trees that, like decision trees, can be used for classification and regression problems. RF fits many decision trees to subsets of training data. Then, each tree creates a classification, which is referred to as the tree votes for that class, and the class with the most votes is chosen as the final prediction from among all the trees in the forest. In classification problems, the class with the most votes is chosen as the model prediction, and in regression problems, the average of the terminal node values is calculated.

Balanced RF: Imbalanced data sets are those in which the records of various classes are distributed unevenly, or in other words, one class label has many records while another class label has few records. This class imbalance can cause ML models to be biased towards the majority class [He and Garcia, 2009], meaning that the model can have higher accuracy on the majority class and perform poorly on the minority class [Kaur et al., 2019]. The default RF produces bootstrap samples by randomly sampling the training data without regard for class labels. As a result, some bootstrap samples may contain very few or no examples of the minority class. One solution is to use one of sampling methods on the bootstrap samples. There are various sampling methods, but we used undersampling in this study, which randomly removes or selects a subset of samples from the majority class [Mohammed et al., 2020]. We used a Python package called *Imbalanced-learn* to train a Balanced RF classifier.

Naive Bayesian: NB is a supervised algorithm based on the Bayes theorem, which is named after the philosopher Thomas Bayes [Bayes' theorem, n.d.]. Bayes' theorem describes the probability of an event occurring based on prior knowledge of the conditions associated with that event [Bayes' theorem, n.d.]. Equation 1 illustrates the Bayes Theorem.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{1}$$

To generate SDM, NB models the probability of observing a species (A) given a set of environmental variables (B).

3.2 Our case study: BioSenCS application

BioSenCS is a biodiversity CS project that we developed with the following objectives:

- Simplify data validation by automatically validating observations
- Provide participants with real-time machine-generated feedback
- Encourage public engagement as a result of automatic feedback
- Increase participants' knowledge about biodiversity using machine-generated feedback
- Improve data quality as a result of automatic feedback

BioSenCS is implemented in a Django framework[1], which is a Python-based free and open-source web framework, and a PostgreSQL[2]/PostGIS[3] database is used for constructing our data models and preserving the collected observations. The high-level architecture of BioSenCS application is illustrated in figure 1, and the source code is available on GitHub[4].

One of the main goals of this project was to apply an automatic validation or filtering of the observations. The validation process is illustrated in figure 2 and it works as follows: When a user submits an observation to the application, the observation goes through the automatic filtering process, and if the observation fails the automated filter criterion, it is flagged as unusual.

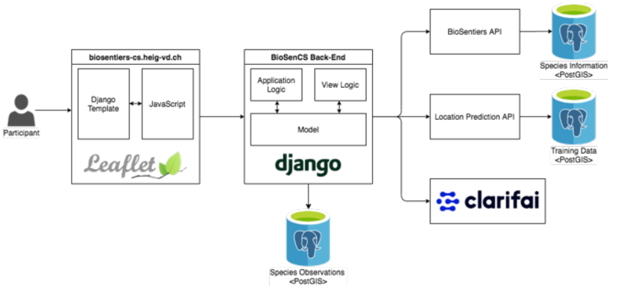


Figure 1. The high-level architecture of BioSenCS application

In this case, the user will receive feedback (first feedback) explaining why the species was flagged as an outlier, and there are two possible outcomes: first, the user can modify the observation and resubmit it using the information in the machine-generated feedback, or second, the user can keep the observation as is and confirm the submission. In the second scenario, the observation will be forwarded to the final expert validation, and if more information is needed, the expert will send the user additional feedback (second feedback). Therefore, our two objectives here are to reduce the number of observations that must be controlled by the expert and to simplify the data validation task, and on the other hand to give real-time feedback to the participants regarding their observation towards keeping them motivated and sustaining their participation to the project. The next section presents the process of automatic location validation in BioSenCS.

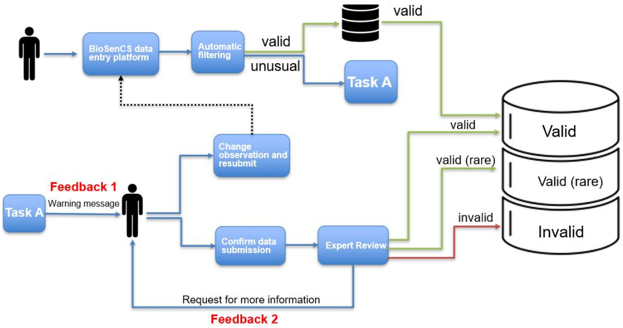


Figure 2. The automatic data validation procedure applied in BioSenCS

3.3 Location validation in BioSenCS

To perform location validation, we determined how the environmental variables surrounding the observation location corresponded to the species habitat characteristics. To accomplish this, we generated the distribution of the species in relation to the environmental variables in our study area. Thus, we used the previously mentioned information about SDM (Section 3.1), particularly with regards to the required data set and the ML algorithms to generate SDM. Thereby, we present the data set we used with the steps on data preparation, the algorithms we trained to generate SDM, the evaluation and results of the algorithms, and finally, we discussed how we used the generated SDMs to validate the location of a new observation and to provide real-time feedback to the participants.

3.4 Data preparation for SDM

Species Data: For location validation, the bird species data set from eBird platform are used. With nearly 600 million bird observations from all over the world as of January 2019,

[1] https://www.djangoproject.com/
[2] https://www.postgresql.org/
[3] https://postgis.net/
[4] https://github.com/mlotfian/Biosentiers-CS-functionality

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-4-W1-2022-271-2022 | © Author(s) 2022. CC BY 4.0 License.

eBird [Sullivan et al., 2014] is one of the largest CS projects for collecting bird observations. We obtained only the validated observations for Switzerland from January 2016 to July 2020. Moreover, we have limited the species to those with at least one hundred distinct observation points. As a result, we obtained 322778 records (out of 400450 total) with 101 species as a result of data filtering.

Generating pseudo-absence data: In this research, we used a method of randomly sampling absences by taking into account the spatial extent around each presence point. We generated pseudo-absences with radius of 5 kilometers around each presence point, and we sampled pseudo-absences outside of these limits. For each species, we randomly sampled 5000 pseudo-absence points. Figure 3 illustrates an example of presence/pseudo-absences for an species called Carrion crow[5] with distance of 5 kilometers from the presence points.

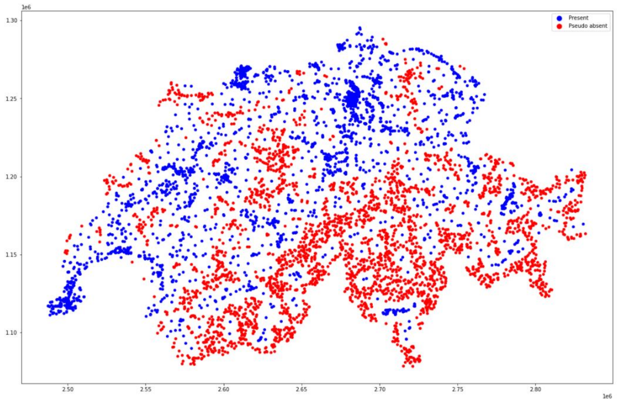


Figure 3. Presences (blue points) and Pseudo-absences (red points) for Carrion Crow in Switzerland

Adding environmental variables: For each record (presence or absence) we created a neighbourhood of size $2km^2$ around the point, and we computed and extracted the environmental variables within this zone. In this research, we have used 19 environmental variables, 16 of which are landscape proportions (ratio of the land cover classes in the defined zone of $2km^2$) which are extracted from CORINE land cover[6], and the remaining 3 are average elevation, average slope, and average NDVI (Normalized Difference Vegetation Index).

Therefore, for each species, a data set was created which included all of the presence/pseudo-absence points as well as the environmental variables within the neighbourhood of $2km^2$ around each point. Data set generation for all 101 species took around 8 hours.

A spatial five-folds cross-validation approach was used to split the data into training and validation data sets before training the algorithms.

3.5 Evaluation and comparison of the algorithms

As mentioned, we trained four algorithms of NB, RF, Balanced RF, and DNN. To build and train the algorithms, we used free and open-source ML libraries including *scikit-learn* , *imbalanced-learn*, *Tensorflow*, and *Keras*.

There are various metrics to evaluate ML algorithms such as classification accuracy, F1 score, AUC (Area Under "ROC" Curve) [Fawcett, 2006], logarithmic loss, etc. To evaluate the performances of our algorithms we used AUC.

For each species, we computed the average AUC over the five folds for all the four algorithms. Figure 4 illustrates the box plots of the variations of AUC within the trained algorithms for all the species. From the box plots we can observe that DNN has a higher AUC median (0.86) compared to the other algorithms, however it's performance is not consistent through all the species. Balanced-RF, with an AUC median of 0.82, outperforms default RF (median = 0.74) and NB (median = 0.75), and performs relatively better across all species than the other three algorithms. Furthermore, for some species where the other three algorithms performed poorly (AUC less than 70%), Balanced-RF outperforms the others. Figure 5 depicts this variation, and it shows that default RF performs the worst for these species, NB and DNN perform similarly, and Balanced-RF performs better for all of them.

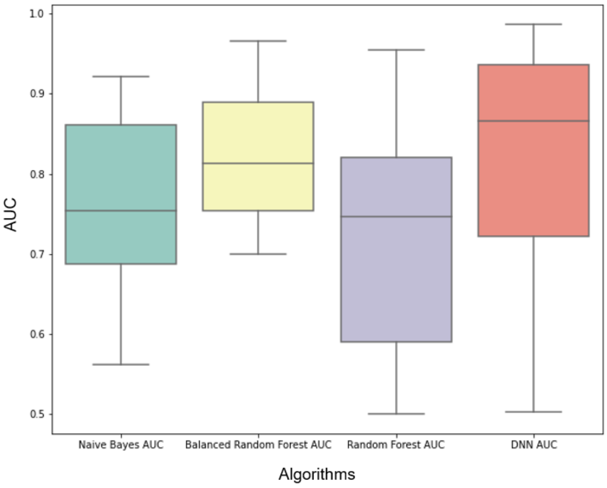


Figure 4. The box plots comparing the AUC among the four trained algorithms to generate SDM

After deciding on Balanced-RF, we used Gini Index [Han et al., 2016] to assess which environmental variables had the greatest influence on the model's ability to predict species occurrences. In other words, we ranked the importance of the environmental variables; the higher the importance, the greater the impact of the variable on model predictions. We used the *sklearn* library to extract the variable importance from the Balanced-RF models using the *feature_importances_*[7] attribute. Figure 6 illustrates the average importance of all the environmental variables for all species, and indicates that, average elevation was an important variable in predicting the distribution of all the species.

Finally, for each species, we obtained two output distributions maps: a binary classification, and a map of probability of occurrence of the species over the whole Switzerland. Figure 7 illustrates the maps of binary classification and probability of occurrence of Common kingfisher species.

[5] Carrion crow: https://www.vogelwarte.ch/en/birds/birds-of-switzerland/carrion-crow

[6] https://www.eea.europa.eu/data-and-maps/data/copernicus-land-monitoring-service-corine

[7] https://scikit-learn.org/stable/auto_examples/ensemble/plot_forest_importances.html

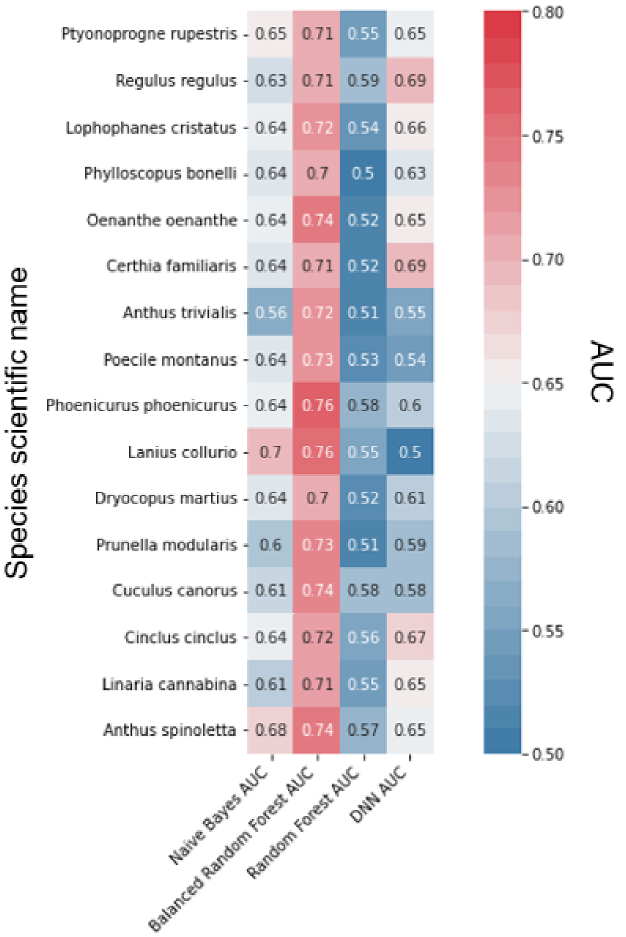


Figure 5. Comparison of the algorithms for the species where NB, RF, and DNN have AUC below 70%, Balanced-RF performs better for such species

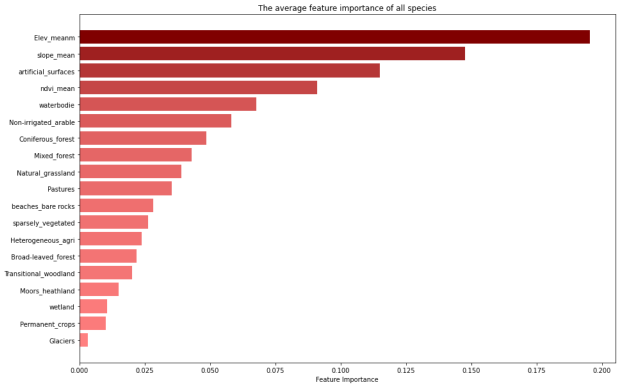
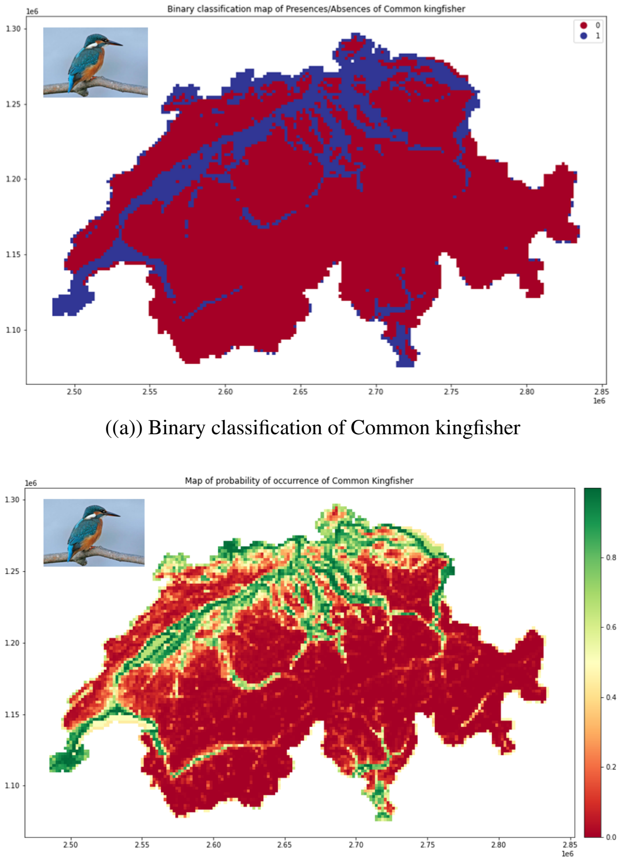


Figure 6. Average environmental variable importance derived from Balanced-RF for all species

4. BIOLOCATION API

The models trained on Balanced-RF were saved in our server with a specific format(`.pkl`) using the Python module called *Pickle*[8]. In order to use the models to validate new observations in the BioSenCS application, we developed an API (Application Programming Interface) that aims to validate new observations while also providing user-centered suggestions on the top-five high-probable species that can be observed near the user's location. To implement the API we used Flask[9], which is a micro web framework written in Python. Figure 8 illustrates the architecture of our API called BioLocation to validate location of new bird observations. The API includes an endpoint for obtaining species names, an endpoint for predicting species probability of occurrence, and an endpoint for suggestion, all of which are explained further below.


Binary classification map of Presences/Absences of Common kingfisher
1e6
0
1
Map of probability of occurrence of Common Kingfisher
1.30
1.25
1.20
1.15
1.10
2.50
2.55
2.60
2.65
2.70
2.75
2.80
2.85
0.8
0.6
0.4
0.2
0.0

((a)) Binary classification of Common kingfisher

((b)) Classification of probability of occurrence of Common kingfisher

Figure 7. Maps of binary classification (a), and probability of occurrence (b) of Common kingfisher within Switzerland

Validation: The overview of the location validation process is presented in figure 9, and details of the steps are as follows:

1) The participant selects the location of observation and adds species name. The location and species name are then passed as the parameters of the BioLocation API with the *predict* endpoint.

2) A neighborhood of $2km^2$ is extracted around the added location.

3) The environmental variables in that neighborhood (proportion of land cover classes, average elevation, average slope, and average NDVI) are computed and a JSON file is created (See figure 10).

4) Based on the species name added by the participant, the environmental variables are passed to the loaded SDM model for that species.

[8] `https://docs.python.org/3/library/pickle.html`

[9] `https://flask.palletsprojects.com/en/2.0.x/`

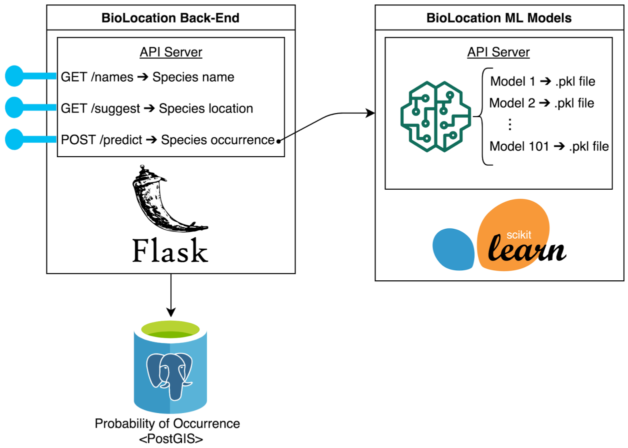


Figure 8. The architecture of BioLocation API

5) The model takes the environmental variables and predicts the presence or absence of the species as well as the likelihood of observing the species in the defined neighborhood.

6) The prediction of the model is then used to validate the observation and provide the participant with real-time feedback on the predicted probability of occurrence as well as information about the species' habitat characteristics.

The generated feedback is intended to either simply provide additional information to the participant if the probability of occurrence of species in the added location is higher that 50 percent, or to propose to the participant to confirm the validity of an observation if the probability of occurrence is less than 50 percent (See figure 11) and in this case to flag the observation in BioSenCS database in a Boolean attribute named FlagLocation. Once the participant receives the feedback, she/he decides whether or not to alter the observation.

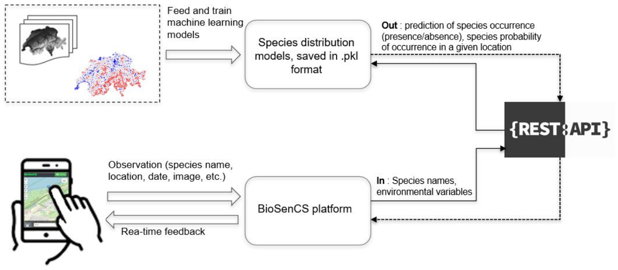
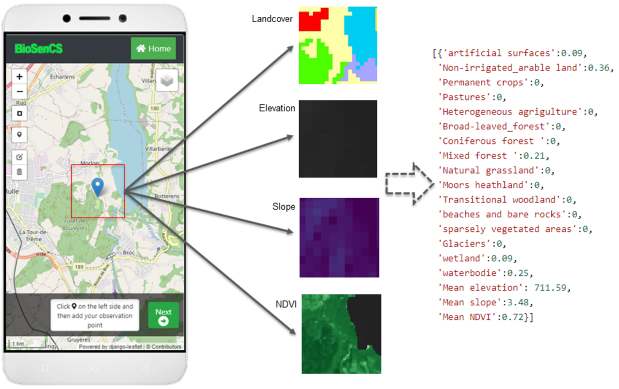

Feed and train machine learning models
Species distribution models, saved in .pkl format
Out : prediction of species occurrence (presence/absence), species probability of occurrence in a given location
{REST:API}
Observation (species name, location, date, image, etc.)
Rea-time feedback
BioSenCS platform
In : Species names, environmental variables

Figure 9. The process of automatic location validation and real-time feedback generation

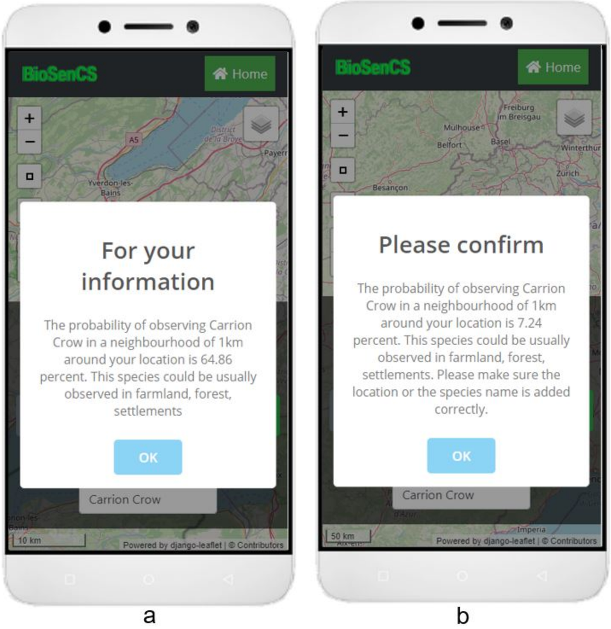
Suggestion: The API also provides suggestions to participants based on their location, such as possible species that can be observed within a 1km radius of the participant's location. Thus, whenever the participant's location is passed to the API's suggest endpoint, the top five species with the highest probability in that neighbourhood are queried from the database, and the results are passed to the participant as a list of species names.

Name: When submitting a bird observation, the participant has three options: either the participant does not know the name of the species, the participant is unsure and checks the suggestion list for the name of the species, or the participant knows the name and writes the name in a text field with an auto-complete function that includes species common names in English taken from the BioLocation database. Figure 12 illustrates the three options to add the name of the bird species.



```
[{'artificial surfaces':0.09,
 'Non-irrigated_arable land':0.36,
 'Permanent crops':0,
 'Pastures':0,
 'Heterogeneous agrigulture':0,
 'Broad-leaved_forest':0,
 'Coniferous forest ':0,
 'Mixed forest ':0.21,
 'Natural grassland':0,
 'Moors heathland':0,
 'Transitional woodland':0,
 'beaches and bare rocks':0,
 'sparsely vegetated areas':0,
 'Glaciers':0,
 'wetland':0.09,
 'waterbodie':0.25,
 'Mean elevation': 711.59,
 'Mean slope':3.48,
 'Mean NDVI':0.72}]
```

Figure 10. Extraction of environmental variables in a neighbourhood of 1km around the location of observation added by the participant



For your information

The probability of observing Carrion Crow in a neighbourhood of 1km around your location is 64.86 percent. This species could be usually observed in farmland, forest, settlements

OK

Carrion Crow

Please confirm

The probability of observing Carrion Crow in a neighbourhood of 1km around your location is 7.24 percent. This species could be usually observed in farmland, forest, settlements. Please make sure the location or the species name is added correctly.

OK

Carrion Crow

Figure 11. Location feedback if probability of occurrence of species is higher (a) and lower (b) than 50%

5. USER EXPERIMENT

Finally, we tested the BioSenCS application within a three weeks period to collect user feedback about the application interface and to explore the view of the participants regarding receiving automatic feedback. Among the 224 users who visited BioSenCS application, only 38 users created a user account, and only 14 out of these 38 users contributed to the project. In addition, among the 14 contributors, three of them were very active each collecting at least 40 observations, four participants were contributing from time to time between 10 to 20 total observations each, and the rest of the contributors contributed mostly only one day or maximum two days during the experiment period with less than 10 observations each. This participation pattern is very known in VGI and CS projects with participation pattern to OpenStreetMap being among one of the most known examples (See [Wood, 2014] for *The Long Tail of OpenStreetMap*). In addition to the number of participants, during the user testing period 230 observations were collected, with 160 of them being birds, 36 flowers, 19 trees, and 15 butterflies.

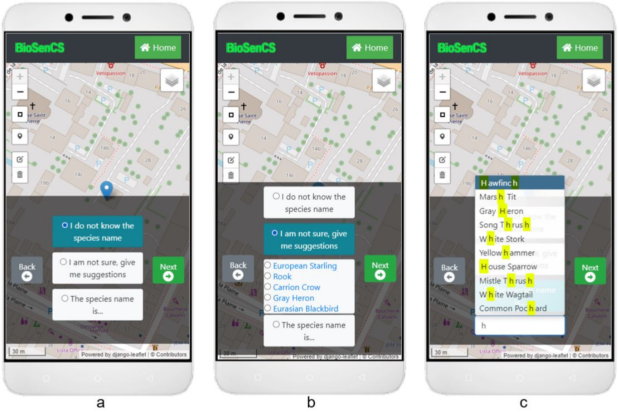


Figure 12. Adding bird species name in BioSenCS application: User does not know the species name (a), participant ask for suggestions to add the species name (b), and participant knows the species name and type the name and an auto-complete of bird names is offered (c)

After the testing period, a questionnaire was sent to the participants to obtain first some general information about their past experience contributing to VGI and CS projects and their views on data sharing in such projects, and second to obtain feedback regarding their experience with the BioSenCS application in terms of their frequency and number of contributions to the app, their views on user friendliness of the interface, the real-time feedback, their motivations to contribute, and finally their feedback on how to improve the application. The questionnaire was created with the Sphinx software[10] and included a variety of question types[11]). Among the contributors, 10 people answered to the questionnaire, while a small sample for statistically testing the validity of the answers, provided us with the necessary information to initially understand how the feedback was affecting the participants' motivations and data quality.

View on receiving real-time feedback: We asked our participants to what extent they found the information in the feedback useful, and whether receiving feedback increased their motivation to contribute to the project. The questions were also asked on a 5-point Likert scale, with the average score for the usefulness of the feedback (1: not at all useful, 5: very useful) and the role of feedback in increasing motivation (1: not at all motivating, 5: very motivating) being 3.33 and 3.5, respectively (See Figure 13).

Besides that, we investigated whether the frequency with which the application is used is related to the scores assigned to the two questions about automatic feedback. We assigned scores to the frequency of app use: only once during the three weeks: 1, once a week on average during the three weeks: 2, and twice a week on average during the three weeks: 3. To explore the correlation, we ran a Pearson test using *SciPy*[12]. The correlation coefficients between the frequency of app use and the usefulness of feedback and the role of feedback in increasing motivation were 0.79 and 0.49, respectively, indicating a strong correlation. However, due to the small sample size, the test was not statistically significant, with p-values of 0.059 for the correlation between frequency of use and usefulness of feedback and 0.34 for the correlation between frequency of use and role of feedback in increasing motivation.

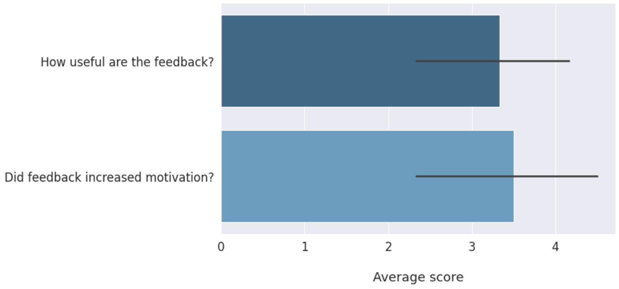


Figure 13. Average scores given to two questions regarding receiving automatic feedback. minimum score 1, and maximum score 5

Furthermore, in order to evaluate the impact of feedback on improving data quality, we investigated whether there was any correlation between the number of flagged observations (O_F) and total number of contributed observations (O_T) per user to see if feedback was a factor in encouraging volunteers to collect higher quality data.

The correlation between the ratio of flagged observations to total number of observations (O_F/O_T) and the total number of observations (O_T) per user was examined. The correlation indicated a statistically significant negative correlation with a value of -0.63 and a p-value of 0.036, indicating that participants who contributed more had fewer flagged observations. This is to say that the participants either used the feedback to improve their observation before submitting it (for example, checking to see if the location pin was correctly added) or they learned to provide higher quality data.

6. CONCLUSIONS

Recent technological advancements have resulted in an increase in the number of CS projects in a variety of scientific areas [Schade et al., 2020]. Despite the large number of CS projects, not all of them are successful [Cox et al., 2015]; therefore, keeping a CS project successful necessitates cautious consideration of the challenges that exist in this field, the two main challenges being increasing public engagement and improving data quality. Several studies have focused on these two challenges, conducting surveys to understand the motivations of volunteers to contribute to CS and evaluating data quality criteria [Leocadio et al., 2021]. Notwithstanding the existing literature, there is still a need to develop new approaches based on new technologies to address these two challenges and lead to more successful CS projects. Therefore, the objective of this research was to focus on addressing the aforementioned challenges using the integration of ML in CS projects.

Accordingly, we investigated the role of ML in automatically filtering and validating citizens' contributed data in addition to providing machine-generated feedback to participants. To that end, we developed BioSenCS, a CS project that invites participants to collect biodiversity observations with the objective of automatically validating collected biodiversity data using ML algorithms. To do so, we generated species distribution models and used them to automatically verify the location of an observation based on the likelihood of observing a species in a specific location. The location validation was done in real-time,

[10] https://en.lesphinx-developpement.fr/sphinx-logiciels-2/sphinx-declic/

[11] https://enquete.heig-vd.ch/SurveyServer/s/INSIT/BioSentiers-CS/questionnaire.htm

[12] https://www.scipy.org/

and the participants received real-time feedback on the probability of observing the species as well as information on species habitat characteristics based on the model's predictions.

The results from the user experiment indicated that participants with a higher number of contributions found the real-time feedback to be more useful in learning about biodiversity and stated that it increased their motivation to contribute to the project. Besides that, as a result of automatic data validation, only 10% of observations were flagged for expert verification, resulting in a faster validation process and improved data quality by combining human and machine power.

The integration of ML and CS is still in its early stages, and more research is needed to evaluate various aspects of this integration. As a future continuation of our research, we aim to investigate how our proposed approach here can be expanded to other CS projects besides the field of biodiversity.

REFERENCES

Abraham, A., 2005. Artificial neural networks. *Handbook of measuring system design*.

Adriaens, T., Tricarico, E., Reyserhove, L., De Jesus Cardoso, A., Gervasini, E., Lopez Canizares, C., Mitton, I., Schade, S., Spinelli, F.-A., Tsiamis, K., 2021. Data-validation solutions for citizen science data on invasive alien species. *Publications Office of the European Union: Luxembourg*.

Balázs, B., Mooney, P., Nováková, E., Bastin, L., Arsanjani, J. J., 2021. Data quality in citizen science. *The Science of Citizen Science*, 139.

Bayes' theorem, n.d. Page Version ID: 1042113059.

Breiman, L., 2001. Random forests. *Machine learning*, 45(1), 5–32.

Ceccaroni, L., Bibby, J., Roger, E., Flemons, P., Michael, K., Fagan, L., Oliver, J. L., 2019. Opportunities and risks for citizen science in the age of artificial intelligence. *Citizen Science: Theory and Practice*, 4(1).

Cohn, J. P., 2008. Citizen Science: Can Volunteers Do Real Research? *BioScience*, 58(3), 192-197. https://doi.org/10.1641/B580303.

Cox, J., Oh, E. Y., Simmons, B., Lintott, C., Masters, K., Greenhill, A., Graham, G., Holmes, K., 2015. Defining and Measuring Success in Online Citizen Science: A Case Study of Zooniverse Projects. *Computing in Science Engineering*, 17(4), 28-41.

Dalby, O., Sinha, I., Unsworth, R. K. F., McKenzie, L. J., Jones, B. L., Cullen-Unsworth, L. C., 2021. Citizen Science Driven Big Data Collection Requires Improved and Inclusive Societal Engagement. *Frontiers in Marine Science*, 8, 432. https://www.frontiersin.org/article/10.3389/fmars.2021.610397.

Elith, J., Leathwick, J. R., 2009. Species distribution models: Ecological explanation and prediction across space and time. *Annual Review of Ecology, Evolution, and Systematics*, 40, 677-697.

Fawcett, T., 2006. An introduction to ROC analysis. *Pattern recognition letters*, 27(8), 861–874.

Han, H., Guo, X., Yu, H., 2016. Variable selection using mean decrease accuracy and mean decrease gini based on random forest. *2016 7th ieee international conference on software engineering and service science (icsess)*, IEEE, 219–224.

He, H., Garcia, E. A., 2009. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, 21(9), 1263–1284.

Kaur, H., Pannu, H. S., Malhi, A. K., 2019. A systematic review on imbalanced data challenges in machine learning: Applications and solutions. *ACM Computing Surveys (CSUR)*, 52(4), 1–36.

Keshavan, A., Yeatman, J. D., Rokem, A., 2019. Combining citizen science and deep learning to amplify expertise in neuroimaging. *Frontiers in neuroinformatics*, 13, 29.

Kullenberg, C., Kasperowski, D., 2016. What is citizen science?–A scientometric meta-analysis. *PloS one*, 11(1), e0147152.

Leocadio, J. N., Ghilardi-Lopes, N. P., Koffler, S., Barbiéri, C., Francoy, T. M., Albertini, B., Saraiva, A. M., 2021. Data Reliability in a Citizen Science Protocol for Monitoring Stingless Bees Flight Activity. *Insects*, 12(9). https://www.mdpi.com/2075-4450/12/9/766.

Lotfian, M., Ingensand, J., Brovelli, M. A., 2020. A framework for classifying participant motivation that considers the typology of citizen science projects. *ISPRS International Journal of Geo-Information*, 9(12), 704.

Lotfian, M., Ingensand, J., Brovelli, M. A., 2021. The Partnership of Citizen Science and Machine Learning: Benefits, Risks, and Future Challenges for Engagement, Data Collection, and Data Quality. *Sustainability*, 13(14). https://www.mdpi.com/2071-1050/13/14/8087.

Mohammed, R., Rawashdeh, J., Abdullah, M., 2020. Machine learning with oversampling and undersampling techniques: overview study and experimental results. *2020 11th International Conference on Information and Communication Systems (ICICS)*, IEEE, 243–248.

Schade, S., Tsinaraki, C., 2016. Survey report: data management in Citizen Science projects.

Schade, S., Tsinaraki, C., Manzoni, M., Berti Suman, A., Spinelli, F. A., Mitton, I., Kotsev, A., Delipetrev, B., Fullerton, K. T., 2020. Activity Report on Citizen Science ? discoveries from a five year journey. *Publications Office of the European Union, Luxembourg*.

Sullivan, B. L., Aycrigg, J. L., Barry, J. H., Bonney, R. E., Bruns, N., Cooper, C. B., Damoulas, T., Dhondt, A. A., Dietterich, T., Farnsworth, A. et al., 2014. The eBird enterprise: an integrated approach to development and application of citizen science. *Biological Conservation*, 169, 31–40.

Vohland, K., Land-Zandstra, A., Ceccaroni, L., Lemmens, R., Perelló, J., Ponti, M., Samson, R., Wagenknecht, K., 2021. *Editorial: The Science of Citizen Science Evolves*. Springer International Publishing, Cham, 1–12.

Wintle, B. A., Elith, J., Potts, J. M., 2005. Fauna habitat modelling and mapping: a review and case study in the Lower Hunter Central Coast region of NSW. *Austral Ecology*, 30(7), 719–738.

Wood, H., 2014. The Long Tail of OpenStreetMap.