Miss-identification detection in citizen science platform for biodiversity monitoring using machine learning

Zakaria Saoud, Colin Fontaine, Grégoire Loïs, Romain Julliard, Iandry Rakotoniaina

Miss-identification detection in citizen science platform for biodiversity monitoring using machine learning

Zakaria Saoud*, Colin Fontaine, Grégoire Loïs, Romain Julliard, Iandry Rakotoniaina

Centre d'Ecologie et des Science de la Conservation, UMR 7204 CNRS-MNHN-SU, Mus'eum national d'Histoire naturelle, 61 rue Buffon, 75005 Paris, France

ARTICLEINFO

Keywords:
Citizen sciences
Machine learning
Data validation
Data mining SPIPOLL

ABSTRACT

In the recent years, several citizen science platforms for biodiversity monitoring have emerged. These platforms represent a powerful tool for collecting biodiversity data for researchers and increasing the knowledge of participants. Typical biodiversity data are species names observed at a given time and place by numerous participants. The use of photos to document observations allows data validation, in particular validation of species identification, a key aspect needed for the quality control of such databases. However, the increasing amount of data collected represents a major challenge given the limited number of co-opted experts dedicated to data validation. Therefore, detecting miss identifications can be very helpful to focus the limited expert workforce on dubious identifications. In this paper, we test various machine learning approaches to detect miss-identifications in such databases based on various features extracted form the history of validated observations. The proposed model can be used to automate the data validation process in the SPIPOLL platform.

1. Introduction iNaturalist,[1] eButterfly,[2] BirdLab, HerpMapper,[3] iSpot[4] and SPIPOLL[5] are few from many Citizen science (CS) platforms dedicated to biodiversity observation. These platforms allow collecting biodiversity data from numerous participants, and contribute to raise public awareness and increase knowledge on biodiversity issues. For example: eButterfly collect butterfly pictures from many volunteers, to provide data about butterfly abundance, in order to determine how climate change may be impacting butterfly distribution. BirdLab allows participants to collect bird pictures and play a collaborative game in order to understand the behaviors and the feeding strategy of birds. HerpMapper aims to gather and share information about reptile and amphibian observations across the planet. iSpot and iNaturalist iSpot aim to collect data of any creature in nature across large temporal and geographic scales. In these platforms, people upload their observations of wildlife and help each other to identify it. Each user can change its identifications many times forming an identification history. In addition to other available tools, iNaturalist offers an automated species identification computer vision tool. Similarly to previous CS, SPIPOLL collect data of flowering plants and insect pollinating it from many participants, in order to study changes in pollinator assemblages across space and time. In the SPIPOLL, after taking picture of pollinators on flowers and sharing it on the website, users are asked to identify each photographed insect, using an interactive identification key containing more than 600 insect names. Users can change their identifications, following the advices from the comments or the suggestions of other participants. After that, experts validate or correct the identifications. Data quality represents a critical aspect for the success of any CS project (Bonney et al., 2009). Improving data quality represents a big challenge in order to increase researchers' confidence in the gathered data from a big numbers of participants with varying levels of expertise. In particular, it is difficult for a limited number of co-opted experts to validate the gathered data manually. In addition, studies (Deguines et al., 2018) have shown that users increase their expertise while participating and that the speed of learning depends on the insect, as difficulty of identification varies

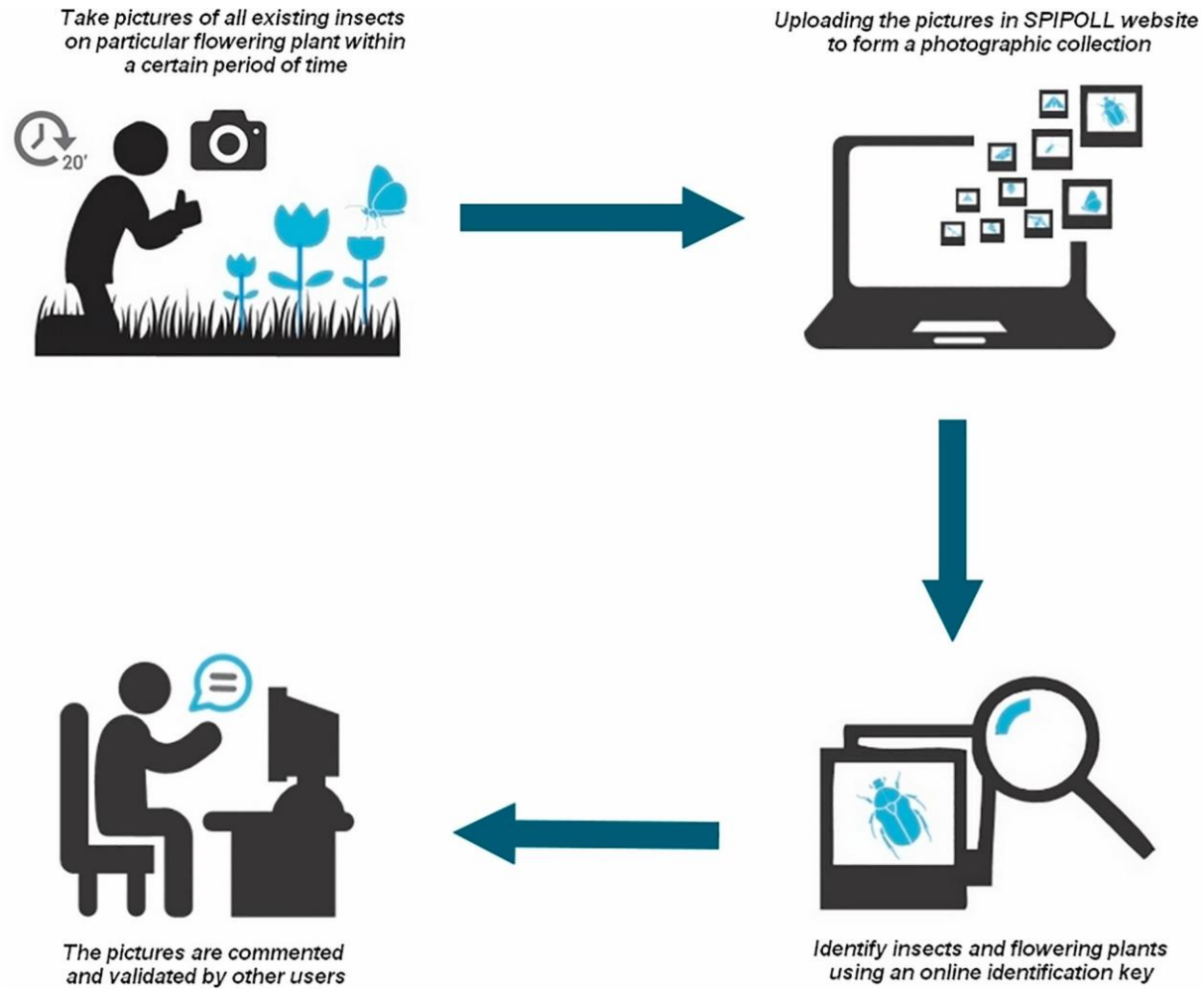

Fig. 1. The general process of the SPIPOLL.

among insects. As a consequence, it might be useful to detect potential miss-identification and to focus the limited expert workforce on them. To do so, we trained several ML algorithms to predict if the identification is correct or not in order to validate the obtained data. These algorithms are trained using a set of extracted features from the set of users' observations. The following of this paper is organized as follows: Section 2 provides an overview of the related work in the area of answer predictions. Section 3 presents the general structure of the SPIPOLL website. Section 4 introduces the details of our prediction model. Section 5 describes the experimental setup and obtained results. Finally, we provide some concluding remarks in Section 6.

2. Related work

In recent years, various works have been done in the area of answer predictions. Most of these works consist of using a set of features from the historical forum data to train a machine-learning algorithm to predict the best answer for a posted question. The chosen answer is supposed to be the best among multiple answers. The main difference between the works lies in the chosen features. Jenders et al. (Jenders et al., 2016) proposed a machine-learning model to predict the correct answer on the Massive Open Online Courses (MOOCs) platform. They have used the forum thread, the users' participation and the textual content of the questions and answers to train the model. Yang et al. (Yang et al., 2011) analyzed the set of not answered questions of Yahoo! Answers platform, to extract features from different perspectives, such as: topic feature, asker history feature, question time and question length features. These features have been used to train a machine-learning model to predict the best answer. Shah et al. (Shah and Pomerantz, 2010) proposed a model for predicting the answers quality for community-based question answering (CQA) platforms. They extract various features from questions, answers and the users who posted them, such as the length of the answer, the length of the question, number of comments for the question and the achieved level of the answerer. They have trained a number of classifiers using these features, to select the best answer. Zhu et al. (Zhu et al., 2009) proposed a multi-dimensional model for estimating the quality of answers in social Question & Answer (Q&A) sites in the context of eLearning. The proposed model has been trained using several dimensions (features) such as the informativeness of the answer, the completeness of the answer, the readability of the answer and the truthfulness of the answer. Liu et al. (Liu et al., 2008) proposed a model for asker satisfaction prediction using three different families of classification algorithms. These algorithms have been trained using several features such as question-answer relationship, asker user history, answerer user history and textual features. Tian et al. (Tian et al., 2013) developed a model for predicting the best answer on (CQA) platforms. The model was trained using three principals' features: the answer context, the question-answer relationship and the answer content. The experiments were conducted using a dataset from the Stack Overflow platform. Zhou et al. (Zhou et al., 2012) exploited three categories of user profile information to improve the answer ranking prediction process on (CQA) services. Three user profile information categories have been defined: level-related, engagement-related and authority-related. Unlike the previous approaches, we used different kind of features such as the environment features and the observation features and time features. Rather than using a CQA platform, we apply our prediction model to validate the biodiversity data gathered in a CS platform, the SPIPOLL.

3. The general structure of the SPIPOLL

SPIPOLL is an SC platform created by the National Museum of Natural History (MNHN) and the Office for Insects and their Environment (Opie), to collect data on flowers and pollinating insects in metropolitan France. The collected data improve the users' knowledge about insect pollinators and allow scientists to assess the abundance variations of pollinator communities. In the SPIPOLL, each user (observer) is asked to take pictures of all insects visiting chosen flowering plant, for a certain period of time. The collected pictures of insects and flowering plant are then uploaded on the SPIPOLL website to form a photographic collection. Nowadays, the SPIPOLL database contains more than 31,329 photographic collections and 307,719 insects' pictures. After data collection, observers are asked to identify insects and flowering plants, using an online identification key. Finally, a group of entomologists from the OPIE validate the identifications. In the SPIPOLL, users can also comment pictures and collections, and add doubts in the identified photos if they are not sure about identifications. Observers can change their initial identifications if they think that it was not correct. Fig. 1 represents the general process of the SPIPOLL.

4. Our prediction model

4.1. Prediction problem

4.1.1. Problem definition

In our proposed model, we forge the prediction problem as a supervised learning binary classification problem. Definition: "True identifications Prediction". Given a training set of validated observations, the prediction task aims to explore the newly posted observations by the observers and predict whether they will be good identified or not. For our training data, the identification is considered true, if it is identical with the validation.

4.1.2. Classification algorithms

We use eight algorithms provided in the scikit-learn library (Pedregosa et al., 2011) to perform classification experiments: (1) Naïve bays (NB): A simple probabilistic classification method based on Bayes' theorem. (2) Decision Trees (DT): which creates a prediction model based on decision tree. Each node in this tree represents a feature and each leaf represents an outcome. (3) Support Vector Machines (SVM): which uses a separating hyperplane to make the classification. (4) Stochastic Gradient Descent classifier (SGD): a linear classifier that uses SGD for training. (5) Logistic Regression CV (LRCV): this uses a cross-validation estimator. (6) Random Forest (RF): this creates a set of decision trees and merges them together to obtain a more accurate prediction model. (7) K- nearest neighbor (KNN): this is based on features similarity between the neighbors to classify a given data point. (8) Multi-layer Perceptron classifier (MLP): a neural network classification model.

4.2. Features description

We exploit the provided data on the SPIPOLL platform, to extract various features. We distinguish insect feature, user features, observation features, time features, and environment features. All of these features are available at observation time. The detailed list of features is reported in Table 1. Except user features, identification history features, insect feature and the number of days between the picture validation and the picture identification, we transform the other features into binary features (dummy variables). Each distinct category from these features will represent an attribute in our training dataset.

Table 1
The detailed list of features: Insect feature (IF), User features (UF), Time features (TF), Observation features (OF), Identification history features (HF) and Environment features (EF).

Features	Description
User features (UF)	
UF: User expertise	Proportion of true identifications per target insect.
UF: Number of observations	Number of observations per target insect.
UF: Total observations	Total number of observations for the whole insects.
Time features (TF)	
TF: Observation season	The season of the observation.
TF: Observation time	Time of the day of the observation.
TF: Observation delay	Number of days between the identification and the actual day (for the training data is the number between the identification and the validation).
Observation features (OF)	
OF: Observation protocol	Duration of the observation session.
OF: Insects count	Number of existing insects on the collection.
OF: Camera type	Smartphone or professional camera.
Identification history features (HF)	
HF: Identification sequence.	The rank of the target identification on the re-identification rank
HF: Doubt presence	The presence or absence of doubt on the observation.
Environment features (EF)	
EF: Flower type	Spontaneous or planted.
EF: Flower shade	The presence or absence of shade on the flower.
EF: Temperature	Weather temperature (10–20°, 20–30° …etc.).
EF: Wind	The wind speed (strong, low, continuous …etc.).
EF: Habitat type	The environment type (rural, garden, urban …etc.).
Insect feature (IF)	Identification ease score of the insect.

Insect feature describe the identified insect in the target picture. This identification is given by the observer who has taken and uploaded the target picture in the SPIPOLL platform. This feature gives information about the ease score of the identified insect *tx*. This score is high when the insect is easy to identify and is low when it is hard to identify. This score is calculated as follows:

$$ease(tx) = \frac{Number\ of\ tx\ pictures\ with\ true\ identifications}{Total\ number\ of\ tx\ validated\ pictures} \quad (1)$$

User features describe the user who has identified the target picture. These features give information about the user expertise on the target insect, the total number of user observations for the whole insects and the number of user observations for the identified insect. The expertise of the user n (U_n) for the insect m (tx_m) is calculated as follows:

$$Expertise(U_n, tx_m) = \frac{Number\ of\ correct\ identifications\ posted\ on\ tx^m\ by\ U_n}{Totale\ number\ of\ identifications\ posted\ on\ tx_m\ by\ U_n} \quad (2)$$

Time features give information about the timing of the posted pictures. Among these features, there is the season of the observation and the time of the day of the observation. We suppose that the picture difficult to identify will require more time to be validated than other pictures. Hence, we have added as feature, the number of days between the picture validation and the picture identification.

Observation features include the observation protocol feature, the insect count feature and the camera type feature. The observation protocol represents the duration of each observation session. It can be flash for short observation session and long for long observation session. The insects count represents the number of insects included in the collection of the target picture. The camera type can be smartphone or professional camera. Pictures taken with professional camera are supposed to be more clear and easy for identification than smartphone pictures.

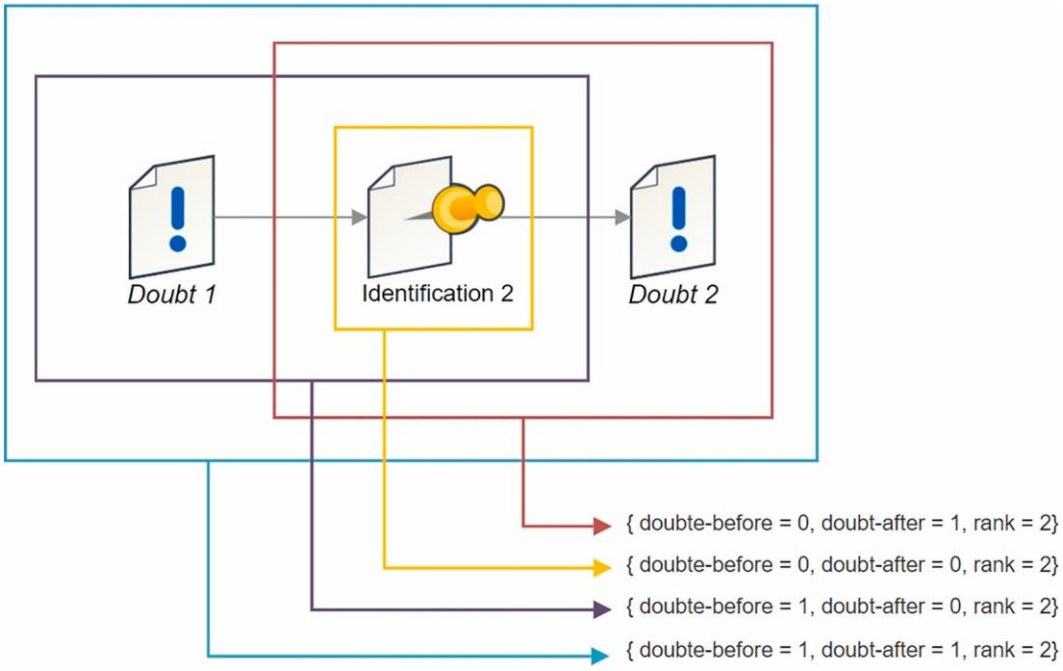

Fig. 2. Lines extraction process from an identification between two doubts.

Identification history features include the identification rank feature, the doubt presence feature. In the SPIPOLL, each picture can be re-identified many times by the observer. The identification rank represents the rank of the target identification on the re-identification sequence before the final validation. For example, the ranks of the first and the second identification will be one and two respectively. In the SPIPOLL, users can add doubt on each picture if they think that it contains wrong identification. Hence, we have decided to add this important information to our other features. For each re-identification sequence, the doubt can occur before or after each identification in the sequence. Hence, we have separated the doubt feature into two features: 1- The "doubt-before" feature: to describe any doubt happened before the identification. 2- The "doubt-after" feature: to describe any doubt happened after the identification. Each identification on the re-identification sequence will be inserted to data in different ways. For an identification I1 that is between two doubts, four different lines will be inserted to data as follows: 1- line 1: without doubts, 2- line 2: with "doubt-before, 3- line 3: with "doubt-after" and 4- line 4: with "doubt- after" and "doubt-before". Fig. 2 shows the lines extraction process from an identification between two doubts.

Environment features give information about the environment of the observation and the weather condition during the time of the observation. These features include the flower type feature, flower shade feature, the temperature feature, the wind feature and the habitat type feature. The flower type can be spontaneous or planted. The flower type can be spontaneous or planted. The flower shade cans affect the identification process. We believe that it is easy to identify the insect when the picture is brighter and without shadow. The temperature represents the temperature of the weather during the observation session. The wind is an important environmental factor that has high impact on the clearness of the picture. Picture taken during a session with high wind speed have probably low quality than picture taken in stable weather. Habitat type represents the environment type where the picture has been taken.

Table 2
Average precision values of the personalization approach.

	Accuracy	Precision	Recall	F1
NB	0.71	0.74	0.72	0.73
DT	0.93	0.93	0.93	0.93
SVM	0.90	0.91	0.91	0.91
SGD	0.74	0.73	0.53	0.61
LRCV	0.91	0.92	0.91	0.91
RF	0.89	0.90	0.90	0.90
KNN	0.82	0.83	0.83	0.83
MLP	0.71	0.73	0.71	0.72

5. Experiments

5.1. Experimental setup

We now describe the metrics used for the evaluation, the datasets, and methods compared in the experimental results of Section 5.

5.1.1. Datasets

The experiments were based on the SPIPOLL dataset described. This dataset contains 31,329 collections, 307,719 pictures (observations), 76,288 comments and 1455 users. This data been collected from a sample of the SPIPOLL database from April 2010 to October 2017. Among the 307,719 observations, there are 155,560 validated observations. 68% of the validated observations have been identified correcly by users.

In our study, we use only the set of validated observations. 70% of data will be used for training and 30% of data will be used for testing.

5.1.2. Evaluation metrics

We evaluate the performance for each classification based on four evaluation metrics: Precision, Recall, accuracy and F1 score. The precision measures the fraction of the predicted identifications that are correct. The recall measure the fraction of all correct identifications that were correctly predicted by the system. The F1 score is the geometric mean of Precision and Recall measures. The accuracy is the proportion of correct predictions to the total number of input samples. Table 2 reports the accuracy, the precision, the recall, and the F1 score of the classification algorithms, by using all features set. For the KNN algorithm we choose k = 3 because it gave better results than the other cases.

Table 3
Accuracy and F1 score according the used features set. G1: includes UF and IF. G2: includes TF and EF. G3: includes OF and HF. G4: includes HF, UF and IF. G5: includes OF, TF and UF.

	Metric	UF	TF	OF	HF	EF	IF	G1	G2	G3	G4	G5
NB	Accuracy	0.53	0.76	0.65	0.69	0.65	0.65	0.53	0.76	0.69	0.54	0.76
	F1	0.59	0.82	0.79	0.80	0.79	0.79	0.59	0.82	0.80	0.60	0.82
DT	Accuracy	0.91	0.79	0.65	0.69	0.66	0.78	0.91	0.78	0.69	0.93	0.78
	F1	0.93	0.85	0.79	0.81	0.78	0.83	0.93	0.83	0.81	0.95	0.83
SVM	Accuracy	0.90	0.75	0.65	0.69	0.65	0.68	0.89	0.76	0.69	0.90	0.74
	F1	0.92	0.81	0.79	0.81	0.79	0.79	0.92	0.82	0.81	0.92	0.80
SGD	Accuracy	0.44	0.74	0.65	0.68	0.65	0.67	0.67	0.71	0.69	0.50	0.75
	F1	0.32	0.80	0.79	0.81	0.79	0.79	0.79	0.76	0.81	0.42	0.81
LRCV	Accuracy	0.89	0.76	0.50	0.69	0.55	0.68	0.90	0.76	0.69	0.90	0.76
	F1	0.92	0.82	0.57	0.80	0.62	0.77	0.91	0.81	0.80	0.92	0.82
RF	Accuracy	0.93	0.76	0.65	0.69	0.65	0.70	0.93	0.65	0.69	0.93	0.65
	F1	0.95	0.82	0.79	0.80	0.79	0.78	0.95	0.79	0.80	0.95	0.79
KNN	Accuracy	0.84	0.76	0.43	0.69	0.60	0.75	0.84	0.79	0.58	0.85	0.79
	F1	0.87	0.82	0.49	0.81	0.70	0.82	0.88	0.84	0.67	0.89	0.84
MLP	Accuracy	0.65	0.62	0.65	0.69	0.66	0.70	0.65	0.45	0.69	0.48	0.54
	F1	0.79	0.65	0.79	0.81	0.78	0.79	0.79	0.40	0.81	0.56	0.57
Average	Accuracy	0.76	0.74	0.60	0.68	0.63	0.70	0.79	0.70	0.67	0.75	0.72
	F1	0.78	0.79	0.72	0.81	0.75	0.79	0.84	0.75	0.78	0.77	0.78

For the MLP we choose 30 hidden layers because it gave better results than

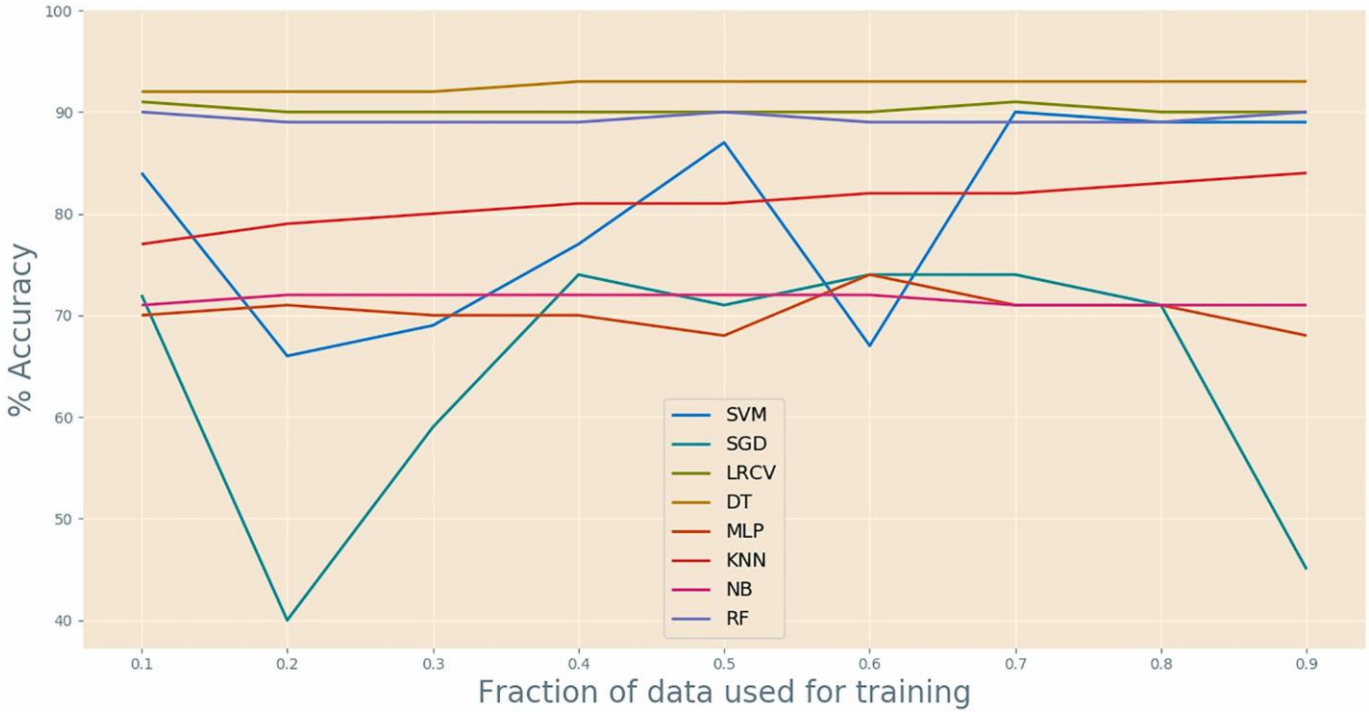

Fig. 3. Accuracy depending on the fraction of used data for training.

the other cases. From the obtained results, we can see that most algorithms give good performance for the prediction of observation quality. Result indicated that DT algorithm had better performance than the other algorithms. This is due to the fact that the tree structure of the DT improves the classification process and protect against skewed distributions. In addition, DT can easily handle with mixture of continuous and categorical features.

5.2. Experimental results

We have also reported the importance of each features group in the prediction process in Table 3. From this table we can see that the difference in accuracies and F1 scores among the whole algorithms was very significant when we use different set of features.

For all algorithms, user features (UF) contribute more than the other features groups for increasing the prediction accuracy. The UF allow the RF and the SVM algorithm to give their highest accuracy (F1 = 0.95 for RF and F1 = 0.92 for SVM). This suggests that identifications given by expert users are likely to

be true. With UF and G1 features, the RF algorithm performs better than DT. This could be because of the increasing of users' observations and expertise with time, from biological point of view. From a statistical point of view, RF algorithm aggregates many decision trees and takes average of many answers given by individual models. The time features (TF) also contribute significantly, suggesting that observation with true identification is usually taken in specific season, period of time and a small observation delay. Most algorithms give similar and tolerable accuracy with using the insect feature (IF) and the Identification history features (HF). This suggests that observation with true identification is usually for insects easy to identify and contains less number of doubts. For most algorithms, the accuracy increase when we combine the user features (UF) with the insect feature (IF). The combination of the observation features (OF) the time features (TF) and the Environment features (EF) give tolerable accuracy but less than accuracy when we use only the TF. This is due to negative effect of the EF and the OF on the TF in the prediction process. The combination of the UF, the IF and HF allows the DT, the SVM, the LRCV, the RF and the KNN algorithm to give their best performance. This suggests that these three feature sets can be used for the prediction process, rather than using the whole feature set.

We have also studied the effect of changing the fraction of data used for training on the performance of the classification algorithms in the prediction process. By varying the amount of data used for training and testing, we obtained different accuracies for each algorithm in each case. Fig. 3 shows the obtained accuracies according the fraction of used data for training for each algorithm. From this figure, we can see that the DT, the RF and the LRCV algorithm outperform the other algorithms, regardless of the amount of training data. Uncommonly, after using only 10% of data (15,556 entries), these algorithms are capable to reach a high accuracy (near or superior than 90%). These three algorithms with the NB algorithm have also a stable performance comparing to the other algorithms that have unstable performance with the increase of the amount of training data. This proves the efficiency of the DT, the RF, the LRCV and the NB with binary classification problems. The SVM algorithm reaches its high and stable performance from 70% of used data. The SVM outperforms the RF algorithm with 70% of used data and give similar performance with the RF with 80% of used data. The performance of the RF algorithm saturates very quickly (from 40% of used data), with the increase of the amount of training data. The SGD gives its best performance with 40%, 60% and 70%. The MLP gives its best performance with 60% of data.

6. Conclusion

In this paper, we presented, to our knowledge, the first approach for automating data validation in citizen science platforms for biodiversity monitoring. The gathered insight from this study can be applied to optimize the validation process in different citizen science platforms, by choosing the best features and the best algorithms. Several kinds of features have been used to train different machine learning algorithms. Each group of features has been grouped in one. These features have been extracted from the set of users' observations, the characteristics, the environment and time of the observation and the insects' identification ease. The trained machine learning algorithms have been used to predict if the identification is true or not in order to validate the obtained data. Our experiment has shown that the user features and the environment features are important for the prediction task. The obtained results have shown also that the most of used algorithms give high prediction accuracy, particularly the decision tree algorithm, which outperforms the other algorithms. In our future work, we plan to develop similar model to predict the answers quality in a crowd sourcing platform.

Declaration of Competing Interest

None

Acknowledgments

This study was supported by l'Agence Nationale de la Recherche (ANR), the National Museum of Natural History (MNHN) and the Office for Insects and their Environment (Opie). We thank Mathieu De Flores (Opie) for his precious help with all the entomological work and Gr´egoire LO IS for providing us the SPIPOLL data. Further, we are grateful to all active SPIPOLL users who helped with data collection: Barbara Mai, Gilles Jardinier, Sagittaire06, MichelMarly, Ascalaf07, Janmar, Prisca and Leonlebourdoun.

References

Bonney, R., Cooper, C.B., Dickinson, J., Kelling, S., Phillips, T., Rosenberg, K.V., Shirk, J., 2009. Citizen science: a developing tool for expanding science knowledge and scientific literacy. BioScience 59 (11), 977–984.

Deguines, N., de Flores, M., Loïs, G., Julliard, R., Fontaine, C., 2018. Fostering close encounters of the entomological kind. Front. Ecol. Environ. 16 (4), 202–203.

Jenders, M., Krestel, R., Naumann, F., 2016. Which answer is best?: Predicting accepted answers in mooc forums. In: Proceedings of the 25th International Conference Companion on World Wide Web. International World Wide Web Conferences Steering Committee, pp. 679–684.

Liu, Y., Bian, J., Agichtein, E., 2008. Predicting information seeker satisfaction in community question answering. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, pp. 483–490.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al., 2011. Scikit-learn: machine learning in python. J. Mach. Learn. Res. 12 (Oct), 2825–2830.

Shah, C., Pomerantz, J., 2010. Evaluating and predicting answer quality in community qa. In: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, pp. 411–418.

Tian, Q., Zhang, P., Li, B., 2013. Towards predicting the best answers in community- based question-answering services. In: Seventh International AAAI Conference on Weblogs and Social Media.

Yang, L., Bao, S., Lin, Q., Wu, X., Han, D., Su, Z., Yu, Y., 2011. Analyzing and predicting not-answered questions in community-based question answering services. In: Twenty-Fifth AAAI Conference on Artificial Intelligence.

Zhou, Z.-M., Lan, M., Niu, Z.-Y., Lu, Y., 2012. Exploiting user profile information for answer ranking in cqa. In: Proceedings of the 21st International Conference on World Wide Web. ACM, pp. 767–774.

Zhu, Z., Bernhard, D., Gurevych, I., 2009. A Multi-Dimensional Model for Assessing the Quality of Answers in Social Q&a Sites (PhD thesis).