Expert Systems With Applications 233 (2023) 120955

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Towards improving prediction accuracy and user-level explainability using deep learning and knowledge graphs: A study on cassava disease

Tek Raj Chhetri [a,b,*], Armin Hohenegger [a], Anna Fensel [c,d], Mariam Aramide Kasali [e], Asiru Afeez Adekunle [f]

[a] Semantic Technology Institute (STI), Department of Computer Science, Universität Innsbruck, Innsbruck, 6020, Austria
[b] Center for Artificial Intelligence (AI) Research Nepal, Sundarharaincha-09, Nepal
[c] Wageningen Data Competence Center, Wageningen University & Research, Wageningen, The Netherlands
[d] Consumption and Healthy Lifestyles Chair Group, Wageningen University & Research, Wageningen, The Netherlands
[e] Department of Plant Physiology and Crop Production, College of Plant Sciences, Federal University of Agriculture, Abeokuta, Nigeria
[f] Kwara Agricultural Network, Ilorin, Kwara, Nigeria

ARTICLE INFO

Dataset link: Survey Data: Towards Improving Prediction Accuracy and User-Level Explainability Using Deep Learning and Knowledge Graphs: A Study on Cassava Disease (Original data)

Keywords:
Explainable AI (XAI)
Agricultural sustainability
Knowledge graphs
Deep learning
Cassava

ABSTRACT

Food security is currently a major concern due to the growing global population, the exponential increase in food demand, the deterioration of soil quality, the occurrence of numerous diseases, and the effects of climate change on crop yield. Sustainable agriculture is necessary to solve this food security challenge. Disruptive technologies, such as of artificial intelligence, especially, deep learning techniques can contribute to agricultural sustainability. For example, applying deep learning techniques for early disease classification allows us to take timely action, thereby helping to increase the yield without inflicting unnecessary environmental damage, such as excessive use of fertilisers or pesticides. Several studies have been conducted on agricultural sustainability using deep learning techniques and also semantic web technologies such as ontologies and knowledge graphs. However, the three major challenges remain: (i) the lack of explainability of deep learning-based systems (e.g. disease information), especially to non-experts like farmers; (ii) a lack of contextual information (e.g. soil or plant information) and domain-expert knowledge in deep learning-based systems; and (iii) the lack of pattern learning ability of systems based on the semantic web, despite their ability to incorporate domain knowledge. Therefore, this paper presents the work on disease classification, addressing the challenges as mentioned earlier by combining deep learning and semantic web technologies, namely ontologies and knowledge graphs. The findings are: (i) 0.905 (90.5%) prediction accuracy on large noisy dataset; (ii) ability to generate user-level explanations about disease and incorporate contextual and domain knowledge; (iii) the average prediction latency of 3.8514 s on 5268 samples; (iv) 95% of users finding the explanation of the proposed method useful; and (v) 85% of users being able to understand generated explanations easily—show that the proposed method is superior to the state-of-the-art in terms of performance and explainability and is also suitable for real-world scenarios.

1. Introduction

With one-third of the global population (2.37 billion) already experiencing moderate or severe food insecurity (UN, 2021) and a rapidly expanding global population, which is expected to reach 9–10 billion by 2050 (Sharma et al., 2020), the agriculture sector is under immense pressure to increase food production. This pressure is further exacerbated by climate change, which has led to a decline in soil quality and the occurrence of numerous diseases such as paddy stackburn (Ayoub Shaikh et al., 2022; Chen et al., 2021), which have a significant effect on the economy. According to the United Nations Food and Agriculture Organisation, plant diseases cost the global economy over $220 billion annually (Food and Agriculture Organization of the United Nations, 2021).

Today, modern technologies, specifically artificial intelligence (AI), are used to combat the issue of plant disease. This is due to the predictive capacity of AI technologies, which enables early identification of potential diseases and prompt preventative measures, thereby reducing loss. As a result, a large number of studies (see Section 2)

* Corresponding author at: Semantic Technology Institute (STI), Department of Computer Science, Universität Innsbruck, Innsbruck, 6020, Austria.
E-mail addresses: Tek-Raj.Chhetri@uibk.ac.at (T.R. Chhetri), Armin.Hohenegger@student.uibk.ac.at (A. Hohenegger), anna.fensel@wur.nl (A. Fensel), kasalimariam8@gmail.com (M.A. Kasali), afeezasirua@gmail.com (A.A. Adekunle).

https://doi.org/10.1016/j.eswa.2023.120955
Received 18 April 2023; Received in revised form 13 June 2023; Accepted 5 July 2023
Available online 8 July 2023

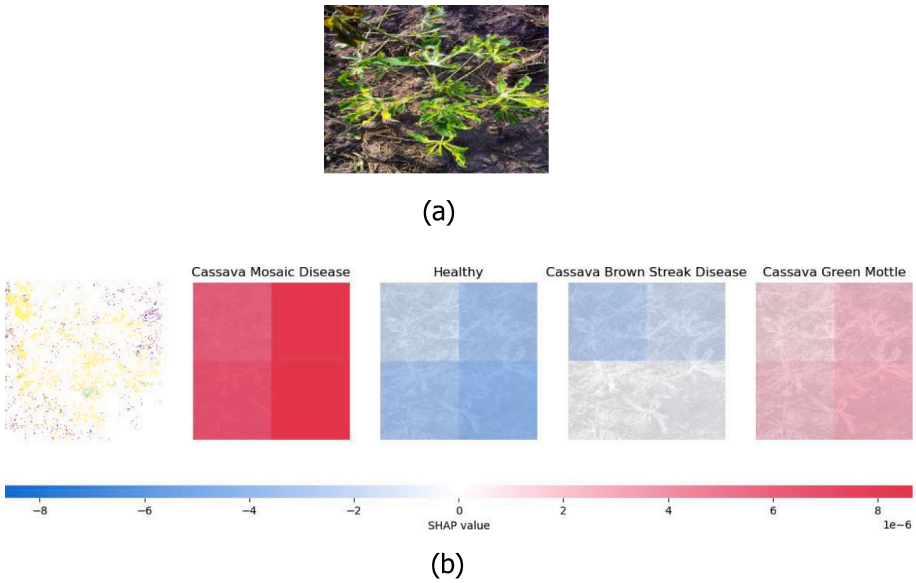

Fig. 1. Explainability using SHAP. **(a)** Input image with disease label "Cassava Mosaic Disease" **(b)** The leftmost image with different colours highlights the features that had an impact on the prediction. The four images on the right, labelled "Cassava Mosaic Disease", "Healthy", "Cassava Brown Streak Disease", and "Cassava Green Mottle", are the top 4 predictions.

have been conducted on plant disease detection and classification using AI technologies, particularly deep learning (DL), a sub-field of machine learning (ML). There has been a substantial improvement in the prediction performance of DL, with an increased prediction accuracy of 90 percent (or even higher in some cases) for disease detection and classification (Emmanuel et al., 2023). However, DL-based techniques suffer from the explainability issue (Chaddad et al., 2023; Gaur et al., 2021).

To combat the issue of explainability in DL, the AI academic community has been actively investigating mathematical approaches, such as LIME (local interpretable model-agnostic explanations) (Ribeiro et al., 2016) and SHAP (SHapley Additive exPlanations) (Lundberg & Lee, 2017), to improve the DL model explainability (Yang et al., 2021). However, explainability techniques based on SHAP and LIME are not suitable for non-experts (or end users). Additionally, Yang et al.'s (Yang et al., 2021) study confirms the same: the explainability techniques based on SHAP and LIME provide an explanation from the computer scientists' perspective, not the user's, thereby widening the gap in user-centred explainability techniques.

Agriculture is one of the sectors dominated by non technological experts (or non-experts), such as farmers. Therefore, any technological solution aimed at the agricultural sector, e.g. for disease detection and classification, must be centred on farmers, as they are the end users, in order to be effective. The use of SHAP and LIME based explanations is therefore ineffective. For example, say there are the three input features (or independent variables) A, B, and C and the dependent variable (or target label) Y. Using SHAP, information about, for example, the impact of A, B, and C on predicting Y but not the explanation of Y can be obtained. These SHAP-derived explanations are helpful for understanding model choices, but they are not very useful for farmers (or non-experts), for whom knowing about the disease would be most helpful. Fig. 1 shows the explainability using SHAP for cassava (Fig. 1(a)). Fig. 1(b) (leftmost image) displays the explanations generated by the SHAP that illustrate the importance of the features upon which the predictions were based. As can be observed from Fig. 1(b), SHAP-based explanations are not useful for users, i.e. farmers. Existing studies, particularly those based on DL in agriculture (see Section 2), are mostly concentrated on performance improvement and therefore lack user-level explainability. This therefore constitutes the first motivation for this work. However, the importance of the performance improvement made by existing works (Section 2) cannot be understated. This is because having just explainability would not solve the problem, as the identification of a potential disease must also be correct, which is the other focus of this work. Therefore, both explainability and prediction accuracy are equally important.

The second motivation is the limited contextual information in existing studies and the exclusion of the domain knowledge, such as in DL (or DL-based studies) (Chhetri, Kurteva, et al., 2022; Holzinger & Müller, 2021). Domain knowledge is knowledge acquired by domain experts over time and is a valuable source of information unavailable in datasets like images. Moreover, images have limited contextual information (Gaur et al., 2022). For example, images lack information about the relationship between plant types, the area they are grown in, and the environmental conditions, such as soil moisture, of that region and their impact on plant health. Both domain knowledge and contextual information help in the improvement of performance, support explainability, and ensure the safety of AI by preventing, for example, hallucinations (Gaur et al., 2022). Semantic technologies, namely ontologies and knowledge graphs (KGs), on the other hand, can incorporate domain expert knowledge, provide reasoning capability, and also enables context awareness to support explainability (Chhetri, Kurteva, et al., 2022; Sharma et al., 2019). Studies have also been conducted using semantic technology for disease classification such as by (Jearanaiwongkul et al., 2018) and (Lacasta et al., 2018). However, such approaches based on semantic technology are limited in terms of their capabilities, such as their ability to learn complex patterns like statistical approaches like DL. Therefore, there is a need to synergise DL-based approaches with semantic technology-based approaches, particularly in the domain of agriculture, so that the benefits of both worlds can be obtained, which are lacking in current work. In the Dagstuhl Seminars report (Benedikt et al., 2020), Claudia d'Amato made a similar point about combining symbol-based methods, e.g. methods based on KGs, and numerical methods like ML.

Therefore, to address the limitations discussed above, this paper presents the research to improve the classification of plant diseases by combining semantic technologies and DL. The main objectives of this research are as follows: (i) to generate user-level (or user-comprehensible natural language) explanations about diseases; and (ii) to improve the prediction accuracy of disease classification by incorporating domain knowledge and contextual information, such as

effects of soil moisture and relative humidity on plant health, using semantic technology and combining it with DL. This study on plant disease classification focuses on cassava plants. This is because of the importance of cassava plants. Cassava, which is mostly grown in Africa, is the fourth most important staple crop and plays a significant role in the diets of over a billion people around the world (Ajayi & Olutumise, 2018). The following contributions are made in response to this study's objectives:

1. A generic approach is proposed to combine semantic technology and DL in order to improve prediction accuracy and generate user-comprehensible natural language explanations using cassava disease classification as a case study.
2. The reusable cassava disease ontology is developed to enable the incorporation of domain-specific cassava disease knowledge.
3. The proposed approach is designed and implemented as deployable software to facilitate reusability and evaluated the proposed method for both performance and explainability.

The remainder of the paper is organised as follows. Section 2 presents the related works. Section 3 details the proposed approach and Section 4 provides details about the experiment, including the dataset, system information, evaluation metrics used, and the implementation. Section 5 discusses the results and finally, Section 6 provides the conclusion.

2. Related work

This section provides an overview of the related work. Section 2.1 provides a brief overview of the explainable AI techniques. Section 2.2 provides an overview of the ML-based studies, whereas Section 2.3 provides an overview of the semantic technology-based studies. In the review, the studies (Chan et al., 2022; Zhang et al., 2020) that focus on the explainability of DL, for example, using SHAP have been excluded as this study is focused on user-level explanations for non-experts.

Additionally, because the focus of this work is on plant diseases, the review of the work only examines related works in this area with particular focus on cassava plant.

2.1. Explainable AI techniques

The majority of the work on explainability focuses on using techniques like SHAP and LIME (Machlev et al., 2022). (Kuzlu et al., 2020) and (Mitrentsis & Lens, 2022) use the techniques SHAP and LIME in their work for explainability. Unlike the cases of SHAP and LIME, Toubeau et al.'s (Toubeau et al., 2022) explainability is based on the attention mechanism. However, similar to the case of the SHAP and LIME-based explanations, the explanations of Toubeau et al.'s (Toubeau et al., 2022) work are also focused on understanding the impact of input variables on the prediction outcome. Unique to previous works, Bahani et al.'s (Bahani et al., 2020) work uses the knowledge base, which contains the explanations of the target labels, to provide explainability for the predictions made, a work inline to the proposed approach of this study. The mapping is done via fuzzy logic. In addition, new model-agnostic approaches similar to SHAP and LIME have evolved recently for graph neural networks (GNNs) (Jiménez-Luna et al., 2020) and their variants. Examples of such explainability techniques include GNNExplainer (Ying et al., 2019) and CF-GNNExplainer (Lucic et al., 2022), where CF represents the counterfactual. Recently, (Sammani et al., 2022) introduced a GPT (Generative Pre-Trained Transformer)-based language model that can simultaneously make predictions and generate natural language explanations, similar to the research in this study. Two limitations exist: the first is the requirement for large training data, and the second is that the accuracy of the generated explanations may not be accurate due to the hallucinations (Gaur et al., 2022). Further, Sammani et al.'s (Sammani et al., 2022) approach lacks the benefit that comes with the use of the KGs, which is the ability ot incorporate additional contextual information. Yang et al.'s (Yang et al., 2023) work focuses on generating knowledge aware explanations for natural language inference using KGs, for which they propose a generative model that makes use of the KGs. In particular, the KGs are used to address the following problems: (i) lack of conformance to the common sense of existing models; and (ii) their lack of informativeness. However, the focus of this work is on language tasks, and their application to a multimodal scenario such as in this study remains unexplored. Moreover, Yang et al.'s (Yang et al., 2023) work demonstrates the additional benefits that can be realised by utilising KGs (or semantic technology). Similarly, other works, such as the one by (Amador-Domínguez et al., 2023) focus on explainability to understand the predictions made by ML models. Their work focuses on generating explanations of KG embedding predictions.

In conclusion, the majority of works on explainability are not geared towards non-experts and are primarily concerned with comprehending predictions and the implications of input variables on final predictions. Some works, such as the one by (Sammani et al., 2022), focus on generating natural language explanations. However, their work lacks the benefits that can be obtained through the use of KGs, and there are limitations such as the need for a large amount of training data. Therefore, there is still a need for further research on explainability approaches that can be both reliable and useful to non-experts, the issue that the proposed work addresses.

2.2. ML-based studies

(Emmanuel et al., 2023) conducted research on the classification of cassava diseases utilising the pretrained models VGG16 and MobileNet V2. Emmanuel et al. demonstrated the utility of their proposed hybrid kernel methods by attaining a 90.1% accuracy rate. The hybrid kernel methods combine the quadratic kernel with the squared exponential kernel. Similarly, (Kumar et al., 2023) conducted a study on cassava disease detection. Their work focuses on improving accuracy by ensembling different computer vision models: EfficientNet, SEResNeXt, ViT, DeIT and MobileNetV3. With their result of 90.75% accuracy, (Kumar et al., 2023) have demonstrated the effectiveness of their proposed approach. (Ravi et al., 2022) conducted a study using attention-based models on cassava disease classification. Similar to the case of (Kumar et al., 2023), (Ravi et al., 2022) demonstrated the advantages of ensembling by achieving the best accuracy of 87.08%, where they combined the penultimate layer features of A_EfficientNetB4, A_EfficientNetB5, and A_EfficientNetB6. The combined final model is called A_L_EfficientNet. In a similar effort to improve the prediction accuracy of cassava plant disease, (Ahishakiye et al., 2023), proposed ensemble model combining Generalised Learning Vector Quantisation (GLVQ), Generalised Matrix LVQ (GMLVQ), and Local Generalised Matrix LVQ (LGMLVQ). Their work achieves an accuracy of 82% (100% with overfitting) using the ensemble model. This work also builds on these findings about the advantage of ensemble models, which in the case of this study combine semantic technology with DL, and addresses the limitations of explainability that have not been addressed by these studies. (Paiva-Peredo, 2023) similarly conducted a study on the classification of cassava disease using pretrained models such as VGG16, RASNET50, and MobileNetV2. A total of 12 different models were examined for cassava disease classification. Paiva-Peredo achieved the best accuracy of 74.77% with DenseNet169.

Similar to other studies, (Chen et al., 2022) conducted studies on cassava disease classification using pretrained models like EfficientNet. However, unlike other works, (Chen et al., 2022) proposed a cross-entropy loss and demonstrated the robustness of cross-entropy loss in noisy datasets, achieving an accuracy of 89.3%. This work, particularly the DL, makes use of the cross-entropy loss, takes advantage of the findings of (Chen et al., 2022), and makes further improvements both in terms of accuracy and explainability. Unlike previous studies, (Anitha &

Saranya, 2022) demonstrated the benefits of the data augmentation for cassava disease achieving an accuracy 90%. Their work makes use of convolutional neural network (CNN). A similar benefit of data augmentation has been demonstrated by (Riaz et al., 2022) for cassava disease classification. Their work uses the pretrained DL model EfficiennetB3 and achieves an accuracy of 83.03%. As with the case of (Chen et al., 2022), this work also takes advantage of these findings about the data augmentation.

(Too et al., 2019) conduct a comparative study of the fine-tuning of DL models: VGG 16, Inception V4, ResNet with 50, 101, and 152 layers, and DenseNets with 121 layers, for identifying plant diseases based on images of leaves. According to their analysis of the plantVillage dataset, the DenseNets model outperforms other models, achieving an accuracy of up to 99.75%. (Atila et al., 2021) also conducted a comparative study and discovered that EfficientNet B5 and B4 were the most effective models on the plantVillage dataset, even outperforming Too et al.'s (Too et al., 2019) discovery of DenseNets.

(Chen et al., 2021) and (Ferentinos, 2018) investigated the plant disease. Similar to (Too et al., 2019), their research focuses on using plant leaf images for disease detection (or classification) and employs computer vision based on DL. The (Ferentinos, 2018) study uses an open database containing images of 25 different plants, and (Chen et al., 2021) use the dataset from the Fujian Institute of Subtropical Botany in Xiamen, China. (Chen et al., 2021) introduced location-wise soft attention to pre-trained MobileNet-V2, improving disease identification. On the other hand, (Ferentinos, 2018) evaluated the specific convolutional neural network (CNN) models, such as AlexNetOWTBn, GoogLeNet and VGG. (Ferentinos, 2018) made an important discovery that CNN models trained on images of laboratory conditions perform significantly worse in the real-world, dropping a success rate (i.e. accuracy of detection) as low as 33%.

Similarly, (Abbas et al., 2021; Ashwinkumar et al., 2022; Bedi & Gole, 2021; Nagasubramanian et al., 2019; Roy & Bhaduri, 2021; Sahu & Sinha, 2022) and (Shah et al., 2022) conducted studies on plant disease detection and classification using images (of plant leaves) and CNN models such as VGG-FCN-VD16, VGG-FCN-S, DenseNet121, and ResNet50. The study of (Shah et al., 2022), similarly to (Too et al., 2019), uses the plantVillage dataset. However, unlike (Too et al., 2019), which focuses on comparative studies, the work of (Shah et al., 2022) concentrates on interpretability via visualisation. The study of (Ashwinkumar et al., 2022), on the other hand, proposed an automated model for detecting and classifying plant leaf diseases and used MobileNet and emperor penguin optimiser algorithm. The proposed model was evaluated by conducting a simulated experiment. The study of (Abbas et al., 2021) focuses on tomato disease classification and uses DenseNet121 and plantVillage datasets. However, unlike other studies using the plantVillage dataset, the study of (Abbas et al., 2021), in addition to DenseNet121, also employs conditional generative adversarial networks to generate synthetic images to complement the lack of data. (Roy & Bhaduri, 2021) conducted a multi-class plant disease classification using an improved version of the YOLOV4 (Bochkovskiy et al., 2020) algorithm, while (Sahu & Sinha, 2022) showed an improvement in disease classification using transfer learning with models such as VGG-16, Inception V3, and ResNet50. (Bedi & Gole, 2021), however, proposed a hybrid approach based on CNN and a convolutional autoencoder (CAE) network, where CAE is used to reduce the dimensionality of the input leaf images and CNN to classify the disease based on the image.

This study addresses the three major limitations of previously discussed studies. They are: (i) not including domain knowledge; (ii) only using the limited contextual information of images; and (iii) the lack of user-level explainability along with the performance improvement.

2.3. Semantic-based studies

(Jearanaiwongkul et al., 2018) propose a semantics-based system (i.e. system architecture) for identifying rice diseases. In addition, a rice disease ontology (Detras et al., 2016) was developed by reusing the rice ontology, plant protection ontology (Halabi, 2009), and plant disease ontology (American Phytopathological Society, 2016). The authors further show how the developed ontology can be used given a farmer's observation. However, the presented system architecture is yet to be implemented. Similarly, (Lacasta et al., 2018) present their work on an agricultural recommendation system based on SPARQL (SPARQL Protocol and Resource Description Framework Query Language) queries for crop protection to help with the identification of pests and selection of suitable treatments. To facilitate the recommendation system, the authors developed the "Pests in Crops and their Treatments" ontology. (Rodríguez-García et al., 2021) present their research on integrated pest management (IPM), a decision support system for crop pest identification and disease recognition. The purpose of their work is to reduce the virulence of the pests and also to manage the disease. Their work incorporates the CropPestO ontology (Rodríguez-García & García-Sánchez, 2020) and natural language processing (NLP) into a symptom analyser to provide a diagnosis and treatment based on the symptoms provided. Due to the fact that the ontology is populated with information from the official Spanish guide, it is only helpful for Spanish-speaking users. (Lagos-Ortiz et al., 2017), similar to (Rodríguez-García et al., 2021), present a decision support system to assist farmers in making effective decisions regarding the diagnosis of plant disease. Similar to other semantics-based studies, their work is dependent on ontology, specifically the phytopathology ontology. The phytopathology ontology is developed by reusing the plant disease ontology (American Phytopathological Society, 2016).

Similar to the case of ML-based studies discussed in Section 2.2, this work also takes advantage of existing semantic-based studies, such as (Jearanaiwongkul et al., 2018). In particular, this study addresses the following major limitations of semantic-based approaches: (i) their inability to learn complex patterns like DL; and (ii) improving prediction accuracy.

3. Materials and methods

This section describes the proposed approach. However, prior to discussing the proposed approach, the ontology modelling will be discussed in Section 3.1. This is because ontology is one of the core components of the proposed approach and ontology also supports the generation of user-comprehensible natural language explanations about disease. Section 3.2 provides details about the proposed approach that combines the semantic technology and DL.

3.1. Ontology modelling

This section describes the ontology utilised in this research. Section 3.1.1 describes the ontology requirements (i.e. the ontology's scope). Section 3.1.2 describes the ontology's development, which follows the ontology development guidelines from (Noy & McGuinness, 2001).

3.1.1. Ontology requirements

The first step to developing any ontology is to define the scope of the ontology, such as what it covers (or should model). The scope of the ontology in this research is as follows:

- The ontology should cover three different categories (or domains), namely, sensor observations, cassava plant disease, and cassava plant environments.
- The ontology should be able to classify the cassava disease based on the sensor observations information in an understandable manner.

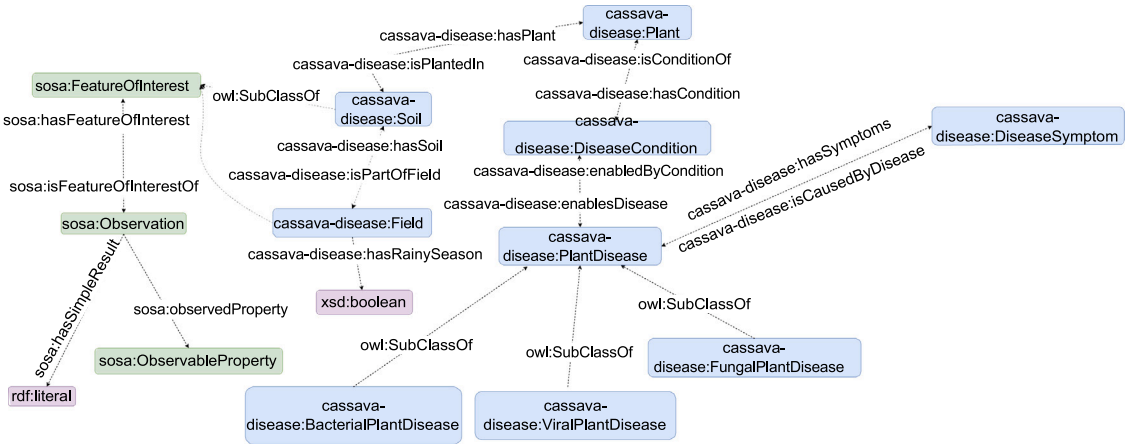

Fig. 2. Domain ontology.

Following the scope, the ontology should be able to answer the competency questions (CQs), which were derived based on the research question. The CQs are as follows:

1. What are all the possible diseases that cassava plants could have?
2. What is the possible disease that cassava plants can have the following symptoms of A, B, ...?
3. How is the cassava plant's environment and the sensor observation related to the cassava plant?
4. What are the symptoms of a cassava plant with disease X?
5. What diseases are caused by viruses (or bacteria, or fungi)?

3.1.2. Ontology development

Fig. 2 shows the ontology, which was developed based on the CQs outlined in Section 3.1.1. The ontology makes use of the SOSA (Sensor, Observation, Sample, and Actuator) ontology (Janowicz et al., 2019), whose concepts are denoted by the prefix *sosa* in Fig. 2. Numerous ontologies exist for plant diseases, such as the plant stress ontology.[1] In addition, there is a cassava ontology[2] that models information such as cassava characteristics. The existing ontologies are modelled as exhaustive knowledge sources, which differs from the use case of this study and necessitates the creation of a new ontology for cassava disease. In order to develop the ontology, the disease hierarchy from (Jearanaiwongkul et al., 2018) study is adopted, in which they modelled the rice disease. The ontology used in this study can be easily expanded to include information from other ontologies, including the cassava ontology and the plant stress ontology. The modelled cassava disease in the ontology of this study is marked by the prefix $cassava-disease$, as can be seen in Fig. 2.

The $cassava-disease:PlantDisease$ models the information about the cassava disease with its subclass $cassava-disease:BacterialPlant-Disease$, $cassava-disease:ViralPlantDisease$ and $cassava-disease:FungalPlantDisease$ in a similar fashion as (Jearanaiwongkul et al., 2018). The cassava disease symptoms are modelled using class $cassava-disease:DiseaseSymptom$. To model the information about the field, the classes $cassava-disease:Soil$ and $cassava-disease:Field$ were used. The classes $cassava-disease:Soil$ and $cassava-disease:Field$ are also connected to the sensor ontology for observing specific properties such as humidity. Moreover, the classes $cassava-disease:Soil$ and $cassava-disease:Field$ are also connected to the class $cassava-disease:Plant$. This is because the conditions of the field will have an impact on the plant's disease and are connected via object properties $cassava-disease:isPlantedIn$ and $cassava-disease:hasPlant$. The concept $cassava-disease:DiseaseCondition$ is introduced to have more fine-grained information about the cassava disease and represents the conditions that are favourable for the occurrence of the disease. The $cassava-disease:DiseaseCondition$ class is linked to a disease with the property *enablesDisease*, and its inverse *enabledByCondition*, which allows to infer the disease given certain disease conditions.

3.2. Proposed approach

This section describes the proposed approach in detail. Fig. 3 illustrates the proposed method, which combines DL and semantic technology to improve cassava disease prediction and enable explainability. As shown in Fig. 3, the proposed method consists of three major elements: the vision model, the semantic model, and the decision engine. This is based on a microservices strategy where each component of the software is built separately based on its functionality. Moreover, the proposed approach is also implemented as deployable software following the microservices strategy (see Section 4.3). The high-level overview of the proposed approach, which is discussed in detail in the subsequent sections, is summarised as follows:

- First, the cassava image is taken and passed through the vision model, which performs the image classification using the vision transformer, from which the prediction result is obtained in terms of the prediction probability.
- Second, the sensor information, such as temperature and soil moisture, is passed through the semantic model, which also performs the classification but using semantic technology, i.e. domain ontology and reasoning using Semantic Web Rule Language (SWRL) rules. Similar to the case of the vision model, the output is obtained in terms of prediction probability. In addition to the prediction probability, the symptoms information and explanations are also obtained as outputs.
- Finally, the prediction results from the vision model and semantic model are combined (or ensembled) using weighted majority voting. In the proposed approach, the component that performs this task is referred to as the decision engine. After combining the predictions, the explanations are generated by fetching the information from the domain ontology.

3.2.1. Vision model

The vision model performs the image classification task, making disease predictions based on cassava image data. This research utilises the vision transformer within the vision model. However, the experiment

[1] https://wiki.plantontology.org/index.php/Plant_Stress_Ontology
[2] https://cropontology.org/ontology/CO_334

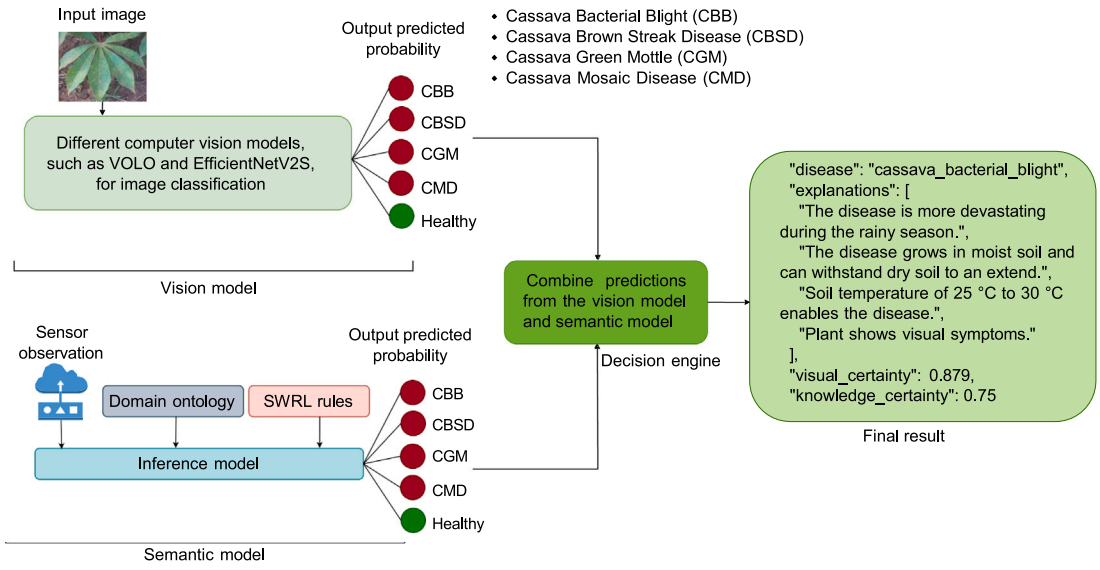

Fig. 3. Proposed approach.

with other pre-trained models such as RESNEXT50_32X4[3] and EfficientNetV2S (Tan & Le, 2021b) were also performed. In contrast to CNN, the vision transformer is the most recent advancement in computer vision and is utilised by the vision model. The vision transformer, via the attention mechanism, enables the modelling of long-range dependencies and provides greater flexibility for modelling visual content (Yuan et al., 2021). The vision model in this study employs the state-of-the-art vision transformer, vision outlooker (VOLO), from (Yuan et al., 2021), the first model of its kind to achieve an accuracy of greater than 87% on the ImageNet[4]-1K benchmark dataset without additional training data. VOLO improves by introducing a lightweight attention mechanism that can represent fine-level information based on aggregation and extrapolation and reduces the complexity of expensive dot products (i.e. $Softmax(Q^T K/\sqrt{d})$, where d is a dimension) by linear projection.

The vision model receives the image as input and generates a prediction regarding the plant's health, whether it is healthy or infected with a particular disease. The output of the prediction is the predicted probability, which indicates the likelihood that a particular disease will occur.

3.2.2. Semantic model

The semantic model, as its name suggests, is based on semantic technology and is the second component of the proposed methodology. The semantic model performs SWRL (Semantic Web Rule Language) (Horrocks et al., 2004) reasoning over the input sensor observation, such as soil temperature, soil moisture, and relative humidity, using the domain ontology (see Section 3.1). The SWRL reasoning can be defined as a reasoning that corresponds to finding new assertions $\hat{A} \ni A \in H$ based on the set of SWRL rules R of the form $R = B \rightarrow H$ applied on the domain ontology O of the form $O = (T, A)$, where T refers to the terms (also called the vocabulary) that capture the particular domain and is the union of the classes (C) and properties (P). The $A \ni A_C \cup A_P$ in ontology denotes assertions made regarding classes and properties. The B in the SWRL rules represent the body axioms, also called as antecedent and H, which is also referred to as consequent, is the head axiom. Fig. 4 shows the snippet of the SWRL rule used in this study, where $hasCondition(?plant, CBBSoilTemperature) \in H$ and the remaining belongs to B. The SWRL rule in Fig. 4 is use for

```
Plant(?plant),
isPlantedIn(?plant, ?soil),
isFeatureOfInterestOf(?soil, ?obs),
observedProperty(?obs, SoilTemperature),
hasSimpleResult(?obs, ?result),
greaterThanOrEqual(?result, 25),
lessThanOrEqual(?result, 30),
-> hasCondition(?plant, CBBSoilTemperature)
```

Fig. 4. A snippet of the CBB disease SWRL rule based on observations of soil temperature that is utilised by the semantic model for disease classification.

predicting CBB (cassava bacterial blight) disease based on the soil temperature (or sensor observation). The inference model component of the semantic model performs the reasoning. Like the vision model, the semantic model yields the final output in terms of probability. In contrast to the vision model, the semantic model also generates disease explanations based on the reasoning result (or prediction) using the domain ontology. Since no sensor dataset is available, the simulated sensor data (see Section 4.1) are used for the semantic model. Fig. 5 shows semantic model prediction.

Following is a summary of the overall steps involved in the semantic model:

- First, inference model component of the semantic model obtains the sensor observation, such as temperature and soil moisture.
- The inference model then populates (or annotates) the domain ontology with the received sensor observation (i.e. creates KG instance).
- Lastly, the corresponding SWRL based on sensor observations is retrieved, and reasoning is performed for disease prediction.

3.2.3. Decision engine

The decision engine combines (or ensembles) the predictions from the vision model and the semantic model to produce the final prediction. The steps involved in combining the results of the semantic model and vision model are depicted in Algorithm 1. As shown in Algorithm 1, the predictions represented by the probabilities from the vision model ($predictionVisionmodel$) and semantic model ($predictionSemanticmodel$) are obtained. The predictions from the vision model and the semantic model are then ensembled ($finalPrediction$). The weighted majority

[3] Dhttps://pytorch.org/vision/main/models/generated/torchvision.models.resnext50_32x4d.html

[4] https://www.image-net.org

```
{

  "cassava_mosaic_disease": {
    "probability": 0.6666666666666666,
    "symptoms": "The disease usually causes the following visual symptoms:
        mosaic, mottling, twisted leaflet, reduction in size of leaves and plants.",
    "explanations": [
      "The disease able to adapt to different soil moisture levels.",
      "Soil temperature of 20 °C to 32 °C enables the disease."
    ]
    }

}
```


Fig. 5. A snippet of prediction from the semantic model after performing SWRL reasoning over the sensor observation.

voting (Raschka & Mirjalili, 2017), as indicated by Eq. (1), is used for ensembling the prediction results. The reason for considering weighted majority voting is that it allows for fine-grained control over the prediction results, as a greater weight can be assigned to the classifier on which one wishes to rely. This is beneficial in certain situations, like those with poor visibility. For instance, during periods of poor visibility, such as the winter or rainy season, it may be desirable to rely on sensor observations that are not affected by inclement weather, which in the case of this study is the semantic model. This can be accomplished by assigning a greater weight to the semantic model's predictions. i represents the classifier and p_{ci} represents the predicted probability. w_i in Eq. (1) is the weight of the classifier. In this study, there are two different classifiers: the semantic model and the vision model, and therefore, the value of i in Eq. (1) is 2. The CBSD (cassava brown streak disease), CMD (cassava mosaic disease), CBB (cassava bacterial blight), CGM (cassava green mite) and Healthy in Eq. (1) represent different classes (or target variable).

For example, let $p_{vision\_model} = [0.21, 0.05, 0.05, 0.19, 0.5]$[5] be the prediction (i.e. as predicted probability) from the vision model and $p_{semantic\_model} = [0.49, 0.3, 0.4, 0.5, 0.19]$ be the prediction from the semantic model. With only a vision model, the prediction would be $healthy$ and the prediction only based on a semantic model would be the CMD. Now, if the weight $w_{vision\_model} = 0.3$ is assigned to the vision model and $w_{semantic\_model} = 0.8$ to the semantic model, the resulting prediction would be $Prediction_{decision\_engine} = [0.414, 0.232, 0.305, 0.415, 0.275]$ and the disease (or resulting target label) would be CMD. However, if the weight is changed as $w_{semantic\_model} = w_{vision\_model} = 0.5$, the final prediction would be $Prediction_{decision\_engine} = [0.35, 0.175, 0.225, 0.345, 0.345]$ and the disease would be disease CBB. This weighted majority voting approach used by the decision engine, therefore, allows fine-grained control over the predictions. At the same time, if the weights differ by a large margin, say 0.8 (or 80%), that can have a negative impact on the prediction. Therefore, it is recommended not to have a large weight difference unless one wishes to rely more on one model (or classifier). Moreover it is recommended to utilise the suitable weights and finding the suitable weights necessitates experimentation as the weights differs from use case to use case similar to other ML hyperparameters.

Following the combination of the predictions from the vision model and the semantic model, the final prediction is obtained. The final prediction about the disease is then used for generating the user-comprehensible natural language explanations. The explanations are generated by retrieving disease information from the domain ontology based on the final results of the prediction. The final prediction results, along with the user-comprehensible explanations and the confidence score from the individual vision and semantic models, are returned in a JSON (JavaScript Object Notation) format as shown in Fig. 6.

$$Prediction_{decision\_engine} = argmax_c \sum_{i=1}^{n} w_i p_{ci} \quad c \epsilon \{CBB, CBSD, CGM, CMD, Healthy\} \quad i = \{Vision \quad model, Semantic \quad model\} \tag{1}$$

Algorithm 1: Algorithm to combine the prediction results from the vision model and semantic model

Input: Disease predictions from the semantic model and vision model
Output: Disease prediction with user-level explanation about disease

```
predictionVisionmodel ← Vision model predicted probability score;
predictionSemanticmodel ← Semantic model predicted probability score;
finalPrediction ← combine_predictions(predictionVisionmodel,
  predictionSemanticmodel);
explanation ← get_explanation_about_disease(finalPrediction);
finalResult← combine_results_for_user(finalPrediction, explanation,
  predictionVisionmodel, predictionSemanticmodel);
return finalResult;
```

4. Experiment

This section provides details about the experiment. Section 4.1 provides information about the datasets and Section 4.2 provides information about the libraries and system used for the implementation and to conduct experiment. Section 4.3 details the implementation of the proposed approach as deployable software. In a similar fashion, Section 4.4 describes the evaluation metrics, and Section 4.5 describes training and testing.

4.1. Datasets

This research utilises the cassava image dataset made available by the Makerere University AI Lab[6] for training the convolutional neural networks (or image classification). The Makerere University AI Lab provided two distinct datasets for a Kaggle competition, one in 2019[7] and one in 2020.[8] (Mwebaze et al., 2019) In this study the combined dataset from (2019 and 2020), which is available at (Gohil, 2021) is utilised. The combined dataset consists of 27053 image samples in total. The dataset, along with the healthy images, contains the four most common cassava diseases: CBSD, CMD, CBB and CGM. The details about these diseases are available in the study by (Mwebaze et al., 2019). Fig. 7 shows the distribution of four prevalent cassava diseases and the healthy images in this study's dataset. From Fig. 7, it can observed that the dataset contains only a small number of healthy samples, which is 2893, out of the total dataset. Moreover, it can also be observed that the occurrence of the CMD disease is more prevalent,

[5] The probabilities of occurrence are as follows: CBB, CBSD, CGM, CMD, and Healthy.

[6] https://air.ug/

[7] https://www.kaggle.com/c/cassava-disease

[8] https://www.kaggle.com/c/cassava-leaf-disease-classification

$$Prediction_{decision\_engine} = argmax_{c} \sum_{i=1}^{n} wi p_{ci}$$
$$c \epsilon \{CBB, CBSD, CGM, CMD, Healthy\} \quad (1)$$
$$i = \{Vision \quad model, Semantic \quad model\}$$


```
{
  "disease": "cassava_bacterial_blight",
  "explanations": [
    "The disease is more devastating during the rainy season.",
    "The disease grows in moist soil and can withstand dry soil to an extend.",
    "Soil temperature of 25 °C to 30 °C enables the disease.",
    "Plant shows visual symptoms."
  ],
  "visual_certainty": 0.879,
  "knowledge_certainty": 0.75
}
```


Fig. 6. The final prediction result after combining the predictions from the vision model and semantic model along with the user-level explanations about the disease and the prediction confidence of the vision model (visual_certainty) and semantic model (knowledge_certainty).

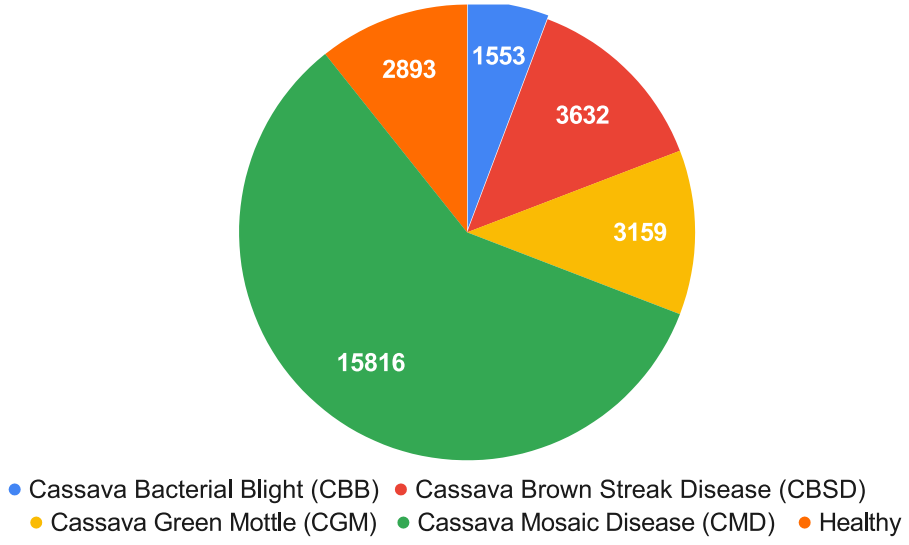

Fig. 7. Distribution of cassava image data based on target class labels.

which is why CMD occupies a large number of samples in the dataset, followed by CBSD, CGM, and CBB.

The dataset was collected via crowd sourcing from approximately 200 farmers whose farms were located in various regions of Uganda (Mwebaze et al., 2019); as a result, their quality varies, making it challenging to perform computer vision tasks. Fig. 8 depicts the visualisation of the images extracted from the dataset for each disease and the healthy sample. The 2020 dataset contains additional complexities, such as multiple diseases associated with each plant, in comparison to the 2019 dataset (Mwebaze et al., 2019). Therefore, the complexity of the combined dataset is anticipated to be greater.

The semantic model (or for semantics-based disease classification) requires sensor data, such as soil information for cassava plants, but no such dataset exists. Because of this, the simulated data were generated. This approach to the generation and use of simulated data is inspired by other areas of computer science like cloud computing, ML, and predictive maintenance research, where, in the absence of real-world data, simulated data is generated and used (Anzolin et al., 2021; Fakhfakh et al., 2017; Greff et al., 2022; Kannammal et al., 2023; Rawat et al., 2021). For example, in a study by Chhetri et al. in the absence of target labels in the dataset, the target labels were generated manually following the failure characteristics of the cloud computing environment (Chhetri, Dehury, et al., 2022), which is a situation analogous to the generation of cassava disease sensor data in this study. The dataset was created using information obtained from experts about the cassava plant. The simulated sensor data was generated with a uniform distribution range for each of the five classes, simulating the favourable disease condition. On the basis of available expert knowledge about cassava plants, the distribution's limits were chosen manually so that the probability p of a certain condition occurring is $50 < p <= 65$. Moreover, Table 1 shows the simulated sensor data for each of the cassava diseases. The minimum and maximum values (i.e. indicated by Min and Max) in Table 1 indicate the limits of the uniform distribution, while the $probability$ indicates the likelihood that the input will fall within the specified disease. The lower and upper boundaries (i.e. $Lower$ and $Upper$ in Table 1) indicate the rule boundaries used in the SWRL reasoning for semantics-based classification. In the case of "healthy", however, there is no probability value. This is because, for other diseases, the same value of sensor observation can cause different diseases. For example, if the soil moisture value is 0.5, then it could lead to both CBB and CBSD. This is not the case in the case of "healthy" and is the reason for not having the probability value. The effects of moisture, humidity, PH (Potential of Hydrogen), and temperature on crops (i.e. including cassava) have been extensively

Fig. 8. Images of cassava corresponding to the five classes in the dataset, including healthy cassava leaves and four prevalent diseases.

Table 1
The rules that are used for generating sensor data. Moisture and relative humidity are measured in percentage, indicated by the range 0–1, and temperature in degrees Celsius (0–100). The PH is measured in units between 0–14.

Disease	Sensor	Lower	Upper	Min	Max	Probability
CBB	Soil moisture	0.3	1	0.1	1	77.78%
	Soil PH	6.5	7.2	6.3	7.5	58.33%
	Soil temperature	25	30	23	31	62.50%
CBSD	Relative humidity	0.7	0.85	0.65	0.88	65.22%
	Soil moisture	0.1	1	0	1	90.00%
	Soil temperature	10	32	5	33	78.57%
CGM	Soil moisture	0.7	1	0.55	1	66.67%
	Relative humidity	0.7	1	0.6	1	75.00%
	Soil temperature	27	40	24	40	81.25%
CMD	Temperature	30	50	24	50	76.92%
	Soil moisture	0.3	1	0.1	1	77.78%
	Soil temperature	20	32	18	34	75.00%
	Relative humidity	0.8	1	0.7	1	66.67%
Healthy	Soil moisture	–	–	0.2	0.8	–
	Soil moisture	–	–	0.2	0.8	–
	Soil temperature	–	–	5	40	–
	Relative humidity	–	–	0.2	0.8	–
	temperature	–	–	10	50	–
	Soil PH	–	–	3	10	–

Table 2
Overall configuration of the LEO4.

Node type	Nodes	Cores/Nodes	Memory/Nodes	GPUs
Standard	44	28 × Broadwell	64 GB	–
Big memory	4	28 × Broadwell	512 GB	–
Fat memory	1	80 × Skylake	3000 GB	–
GPU	1	28 × Skylake	384 GB	4 × Nvidia Tesla V100

Table 3
Specification of Nvidia Tesla V100 GPU used in LEO4.

Performance	Double precision	7.8 TFlop/s (Teraflops per second)
	Single precision	15.7 TFlop/s
	Tensor	31.3 TFlop/s
		125 TFlop/s
Interconnect	NVLINK	300 GB/s
Memory	Capacity	32 GB
	Bandwidth	900 GB/s

Table 4
A list of the libraries and software used to implement the proposed method.

Libraries/Software	Version
Python	3.10
Docker	20.10
Owlready2	0.37
FastAPI	0.75.1
Requests	2.27.1
BentoML	1.0.0a7
PyTorch	1.12.0
timm	0.6.5

studied, therefore, the readers are recommended to studies (Luampon & Charmongkolpradit, 2019; Seena Radhakrishnan et al., 2022) for additional information on their implications.

4.2. System setup

This section describes the software and libraries utilised in the implementation of the proposed system. In addition, this section includes details about the systems used to conduct the experiment.

Table 4 shows the list of the libraries and software that were utilised to implement the proposed work. The implementation is performed in the Python,[9] programming language. Docker[10] FastAPI,[11] and Requests[12] are used to modularise the implementation. Owlready2[13] is used to deal with the semantic model, namely, ontologies and KGs, and to perform reasoning. PyTorch[14] is used for the implementation of the vision model and BentoML[15] is used for model serving purposes. The library timm[16] is used for model implementations and ImageNet weights.

The supercomputer LEO4[17] is used to perform the experiment. LEO4 is a high performance compute cluster at the University of Innsbruck. Table 2 shows the overall configuration of the LEO4. LEO4 consists of a total of 50 nodes totalling 1452 cores and a total memory of 8.4 terabytes. LEO4 is powered by either Broadwell or Skylake Intel Xeon processors. Similarly, the LEO4 GPU (graphics processing unit) node utilised in the experiment consists of 4 Nvidia Tesla V100[18] GPUs with a total memory of 384 GB (gigabytes) per node. Table 3 shows the detail specification of the LEO4 GPU, Nvidia Tesla V100[19] providing details on performance, memory and interconnectivity. In particular, GPU used in LEO4 has NVLink[20] connectivity for high-speed data transfer. Moreover, the LEO4 uses a high-performance, low latency 100 Gb/s (gigabits per second) infiniband interconnect for MPI (Message Passing Interface) communications in order to communicate between nodes and GPFS (General Parallel File System) file system.

4.3. Implementation

This section describes the implementation details of the proposed approach as deployable software to facilitate the reusability and accessibility of the proposed work in real-world deployment scenarios. The implementation follows the microservices strategy (or architecture). This is because of the scalability that microservices offer. The microservices allow scaling of the individual components independently. With the microservices architecture, for instance, compute-intensive system

[9] https://www.python.org/
[10] https://www.docker.com/
[11] https://fastapi.tiangolo.com/
[12] https://requests.readthedocs.io/en/latest/
[13] https://owlready2.readthedocs.io/en/v0.37/
[14] https://pytorch.org/
[15] https://docs.bentoml.org/en/latest/
[16] https://timm.fast.ai/
[17] https://www.uibk.ac.at/zid/systeme/hpc-systeme/leo4/
[18] https://www.nvidia.com/en-us/data-center/v100/
[19] https://www.uibk.ac.at/zid/systeme/hpc-systeme/common/software/leo-gpu.html
[20] https://www.nvidia.com/en-us/data-center/nvlink/

components can be deployed on high-performance servers or cloud systems. The source code for the implementation and other resources, such as ontology, are available at (Chhetri, Hohenegger, et al., 2022).

For the implementation of the vision model service, BentoML is used, which is an open-source model serving framework at a production scale and provides easy deployment of the ML models (see footnote 15). The implementation of BentoML for model serving includes the following steps: (i) saving the trained model weight with BentoML; (ii) defining the service configuration, such as defining the API (Application Programming Interface) service for prediction using the @svc.api decorator; and (ii) performing input image preprocessing, such as resizing. The runner is then invoked, which translates the API definition into an HTTP (Hypertext Transfer Protocol) endpoint `/predict` for making a prediction.

The semantic model (or semantic classifier) REST (Representational State Transfer) Service implementation makes use of libraries such as FastAPI and Owlready2 and the domain ontology. The semantic model consists of the three endpoints. `/soils/{soil_id}/observations` is used for sensor observations of soil data, such as soil temperature. The second endpoint, `/fields/{field_id}/observations`, is used for observations in the field, such as whether it rained. The endpoint `/plants/{plant_id}/disease-vector` is used to retrieve the specific disease information about the particular plant based on its unique identifier (ID). The ID in this study is initialised as a number between 1–6. Using the domain ontology and SWRL reasoning, the disease information is retrieved. Pellet Reasoner available in Owlready2 is used to perform the reasoning. In addition, in order to prevent any unnecessary inconsistencies in predictions, the previous observations are deleted. The domain ontology, which is saved as OWL file, is loaded at service startup and stored in an internal triple store.

The decision engine, which is implemented as another service component, consists of the realisation of the decision engine steps outlined in Section 3.2. The decision engine takes the image and plant information as an input and calls REST endpoints `/predict` and `/plants/{plant_id}/disease-vector` from the image classifier service component and semantic models service component to obtain the predictions. The obtained predictions are then combined to produce the final prediction along with their explanations. The relevant information is retrieved from the domain ontology and the explanations are generated based on the obtained predictions.

4.4. Evaluation metrics

In ML, the evaluation metrics help to understand the performance of the model (or how well the model is likely to perform) in an untested scenario. In this study, the evaluation of the proposed approach is made in terms of accuracy and prediction latency to make the inference.

4.4.1. Accuracy, precision, recall and F1 score

The accuracy measures the overall correct predictions made by the model. Precision, also known as the positive predicted value, measures the model's exactness, i.e. the proportion of positives correctly identified, whereas recall measures the actual positives correctly identified. Recall is also known as sensitivity or the rate of true positives. The F1 score represents the harmonic mean of the precision and recall.

Accuracy, precision, recall and F1 score can be calculated using Eqs. (2), (3), (4) and (5) respectively. In Eqs. (2), (3), and (4), the TP stands for the true positive value and the TN for the true negative. In the same way, FP in represents a false positive, and FN is a false negative. True positives and true negatives are values that the model accurately predicts for both the positive and negative class labels, in this case, cassava disease and healthy. The false positive, on the other hand, is the prediction that is identified by the model as positive while in reality, it is not. The false negative, however, is similar to the false positive but for the negative class.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1 \quad Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

4.4.2. Prediction latency

The prediction latency quantifies the time required to make a prediction. The prediction latency is measured by calculating the difference between the invocation and response time using Eq. (6). The t_i in Eq. (6) represents the prediction invocation time and t_r is the response time.

$$Latency_{prediction} = t_i - t_r \tag{6}$$

4.5. Training and testing

This section describes the training and testing of the different pre-trained models, RESNEXT50 (Xie et al., 2017), EfficientNetV2S (Tan & Le, 2021a) and VOLO, that were experimented with. These models, RESNEXT50, EfficientNetV2S, and VOLO, were chosen based on their superior performance, which has been shown in earlier studies such as (Chen et al., 2022) and (Ravi et al., 2022). In addition, this section describes the testing of the proposed method that combines the predictions from the vision model and the semantic model.

The training and testing of the different pre-trained models, VOLO, EfficientNetV2S and RESNEXT50 was performed using the cassava image dataset (see Section 4.1). 80% of the dataset was used for training, and the remaining 20% was used for testing. Fig. 9 illustrates the distribution of the test data, which demonstrates that, similar to the original dataset, the test data is unbalanced. Transfer learning was utilised for the model's training using Pytorch. Similarly, the library timm was used to load the model and pretrained ImageNet weights. The hyperparameters used for different pre-trained models for training are specified in Table 5. The values of the hyperparameters in this study were determined based on experimentation and the authors' prior experience with ML research (Chhetri, Dehury, et al., 2022; Chhetri, Kurteva, et al., 2022). The training was performed over 10 epochs with a batch size of 32 for RESNEXT50_32X4D and 16 for EfficientNetV2S and VOLO. The other significant criterion is the learning rate, which determines the success of the learning process (how well the algorithm learns). The Adam optimiser[21] with a learning rate of 10^{-4} and weight decay of 10^{-6} is employed. The Adam optimiser is an adaptive learning rate optimiser capable of handling dynamic situations, such as a loss that is either increasing or decreasing. Moreover, the learning rate scheduler is also used with a cosine annealing schedule with warm restarts (Loshchilov & Hutter, 2016). As a loss function, the Taylor Cross Entropy Loss (Feng et al., 2021) with label smoothing (Müller et al., 2019) is used, which has been demonstrated to be robust against the noisy dataset (Chen et al., 2022). A value of 0.05 is used for label smoothing. Additionally, following the findings of the previous works, the different data augmentation techniques are used, details of which are available in Table 5. In the case of the semantic models, however, no training is required as the semantic models are based on SWRL rules and the rules are defined manually.

In the case of the testing, however, different levels of testing were performed. The different tests that were performed are presented below.

[21] https://pytorch.org/docs/stable/generated/torch.optim.Adam.html

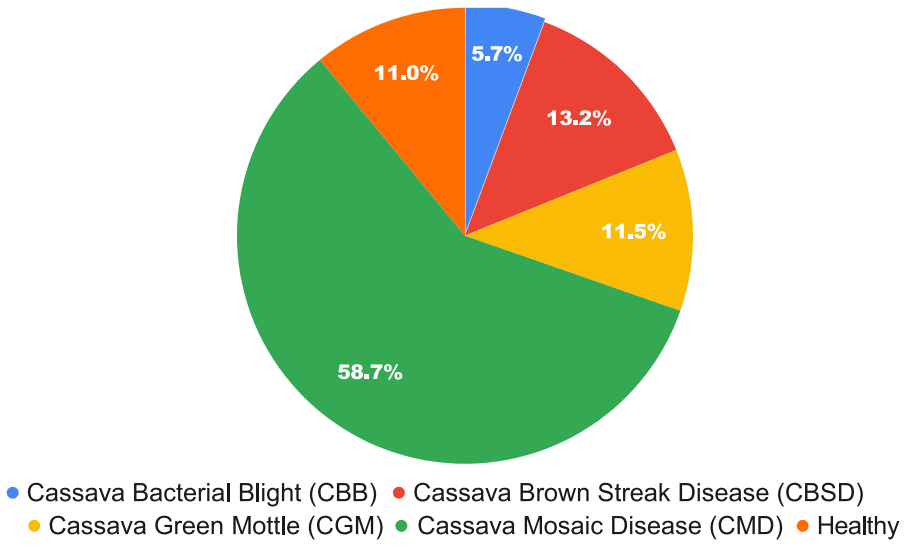

Fig. 9. Class-based distribution of test cassava image data.

Table 5
Hyperparameters of various CNN-based vision models and a vision transformer, where LR is the learning rate, p is the probability, std is the standard deviation, T_0 represents the number of iterations for the first restart, T_mult is a factor increase for T_i after restart and eta_min is the minimum LR.

Model	Batch size	Hyperparameters			
		LR	Loss	LR scheduling	Data augmentation
RESNEXT50_32X4D	32	1.00E−04	Taylor cross entropy loss with label smoothing	Cosine annealing warm restarts scheduling with T_0 = 10, T_mult = 1 and eta_min = 1e-6	RandomResizedCrop HorizontalFlip (p = 0.5) Transpose (p = 0.5) VerticalFlip (p = 0.5) ShiftScaleRotate (p = 0.5) Normalise(mean = [0.485, 0.456, 0.406], std = [0.229, 0.224, 0.225],)
EfficientNetV2S (256 × 256)	16				
EfficientNetV2S (384 × 384)	16				
VOLO-D1 (224 × 224)	16				
VOLO-D2 (224 × 224)	16				
VOLO-D1 (384 × 384)	16				
VOLO-D2 (384 × 384)	16				

1. The first test are conducted on the various pre-trained computer vision models that are being examined. 20% of the test cassava dataset was used for this purpose.
2. The second evaluation concerned the proposed method, which combines a semantic model and a vision model to improve the accuracy of disease predictions and provide user-level disease explanations. As different computer vision models are experimented with, the one with the highest performance is selected (evaluated in Step 1) to combine with the semantic model. The explainability was evaluated qualitatively using three identified dimensions: (i) correctness, i.e. if the generated explanations are correct; (ii) usefulness, i.e. whether the generated explanations are helpful to the users; and (iii) comprehension, i.e. whether the generated explanations are understandable to the users. The accuracy of the explanations was manually examined by inspecting the explanations that were generated. Regarding the remaining two dimensions, comprehension and usefulness, an online survey[22] was conducted. The survey asked participants if they were able to understand the SHAP-based explanations as well as the explanations provided by the proposed approach. The survey also asked about the usefulness of these explanations, both based on SHAP and the proposed method in terms of understanding the disease. In addition to questions about explainability, the survey also asked questions related to demographics, age groups, and education. The details of how the survey is conducted and the evaluations will be presented in Section 5
3. The third test is about latency. As this study implements the proposed approach into a fully deployable system (see Section 4.3), the prediction latency is also evaluated.

The results of the tests conducted will be discussed in Section 5.

5. Results and discussion

This section provides the experimental results from the experiment in this study. Moreover, this study also provides a comparison of the experimental results with the state-of-the-art study.

As discussed in Section 4.5, the first evaluation is about pre-trained models. Table 6 shows the accuracy of the vision transformer, VOLO, and also other CNN models, such as EfficientNetV2S and RESNEXT50, that were experimented with. In the result of the vision model, only the accuracy score of the models experimented with except for the best model, for which the precision, recall, and f1 score were also

[22] The survey questions are available in GitHub where the code is stored.

Table 6
A performance comparison of various CNN-based vision models and a vision transformer.

Model	Image size	Accuracy (%)
RESNEXT50_32X4D	256 × 256	0.8721
EfficientNetV2S	256 × 256	0.8785
EfficientNetV2S	384 × 384	0.8840
VOLO-D1	224 × 224	0.8256
VOLO-D2	224 × 224	0.8764
VOLO-D1	384 × 384	0.8914
VOLO-D2	**384 × 384**	**0.8964**

recorded. The reason for only recording accuracy, despite the fact that sometimes accuracy can be deceptive, is because of the test data distribution (see Section 4.5), which is unbalanced but not in such a manner that it can lead to deceptive results. This is also consistent with other studies (Chen et al., 2022; Paiva-Peredo, 2023) that are used for comparison with the work of this study, which also only records accuracy. From Table 6, it can be observed that the VOLO (or vision transformer) clearly outperforms the corresponding CNN models. The other interesting observation that can be made is that the VOLO is also the model with the lowest performance. The VOLO-D1 with an image size of 224 is the model with the lowest performance, while the VOLO-D2 with an image size of 384 is the model with the highest performance. Moreover, the precision, recall, and F1 scores for the best performing model, VOLO-D2, are 0.828, 0.807, and 0.818, respectively, indicating the robustness of the model. However, the accuracy of the VOLO in this study is consistent with the original study, in which VOLO-D1 was the lowest performing model. The second best-performing model is EfficientNetV2S, with an image size of 384. This is similar to the VOLO where both VOLO-D1 and VOLO-D2 with an image size of 384 are performing better. This is because as the size gets smaller, the features get lost and have an impact on the performance as observed. The EfficientNetV2S and RESNEXT50, with an image size of 256, have a comparable performance with a small variation of 0.0063%. The best performing VOLO model makes an improvement of 0.012% compared to the best performing EfficientNetV2S model and an improvement of 0.024% compared to that of RESNEXT50.

Fig. 10 likewise shows the confusion matrix of the VOLO (best performing) model and Table 7 shows the individual class-level performance evaluation of VOLO. As can be observed from the confusion matrix and also from the precision and recall, there is still high misclassification. For example, the higher accuracy can be seen in the case of CBB but a huge drop in precision and recall. This is because of the class imbalance. A similar observation has been made by (Chen et al., 2022) and (Paiva-Peredo, 2023) in their study, observing misclassification due to a large class imbalance. In the case of CMD, however, no such drop is observed, all precision, recall, and accuracy are higher, greater than or equal to 94%. However, upon close observation of the precision and recall, there is no significant difference between them, indicating no case of overfitting or underfitting. Moreover, similar conclusions can be drawn from the F1 score. The reason for this is that the dataset that is used in this study contains noisy data. The images were taken by farmers and have different lighting conditions, for example, and therefore, have an impact on the performance. Moreover, studies such as (Ferentinos, 2018) have demonstrated that the accuracy of computer vision models in real-world datasets can drop as low as 33 percent.

As discussed in Section 4.5, the second evaluation focuses on the proposed method, which is to combine the vision model with the semantic model. For this, the best performing vision model is taken, which in the case of this study is VOLO-D2 with an image size of 384 × 384 (or VOLO-D2@384x384). Table 8 shows the accuracy after combining the vision model with the semantic model. Moreover, Table 8 also shows the weight that is used for weighted majority voting. As can be clearly observed that by integrating the semantic model, an improvement in the accuracy has been made. The enhancement is

Table 7
Individual class-level performance evaluation of the VOLO-D2 model with an image size of 384 × 384, i.e. the best performing vision model.

Class name	Precision	Recall	Accuracy	F1-score
CBB	0.699	0.652	0.964	0.675
CBSD	0.864	0.853	0.963	0.859
CGM	0.864	0.801	0.963	0.831
CMD	0.949	0.975	0.955	0.962
Healthy	0.768	0.754	0.948	0.761

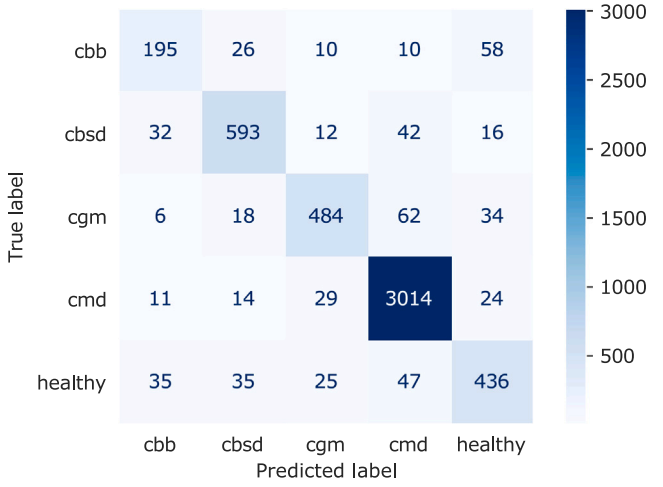

Fig. 10. Confusion matrix of the VOLO-D2 model with an image size of 384x384, i.e. the best performing vision model.

0.0086%. While the accuracy improvement is modest, it is still superior to the state-of-the-art.

In addition to the evaluation of the performance, explainability is also evaluated. A qualitative evaluation was performed using the three dimensions: (i) correctness; (ii) usefulness; and (iii) comprehension to determine whether the generated natural language explanations about disease (see Section 3.2.3) were helpful and informative enough to convey the disease information to users. For correctness, the generated explanations were checked by manually inspecting the explanations by the authors and comparing the explanations against the information present in the ontology, as the explanations were based on the information present in the ontology. The generated explanations were found to be correct (100%) and aligned with the predictions. The second dimension is usefulness, where the generated explanations were evaluated to see whether they were informative enough. The comprehension dimension is the third. This evaluation examines the generated natural language explanations to determine if they are sufficiently comprehensive and written in a natural language that avoids technical jargon that is incomprehensible to non-expert users. Regarding the evaluation of the second and third dimensions, i.e. usefulness and comprehension, an online survey was conducted following the study by (Ball, 2019). Fig. 11 depicts the steps taken to assess the second and third dimensions.

First, the designed survey was shared among the participants to gather their opinions. The participants were invited via private channels and their peer group. The following criteria were used for inviting the participants: (i) they had to be over 18 years old; and (ii) they had to have a valid email address and access to a computer or mobile phone in order to be able to take the survey. In addition, the farming experience, which is a major focus, is also considered but not a hard requirement. This is to evaluate the benefits of the explanations of the proposed approach in other domains, as the proposed approach is generalisable. While inviting participants from other domains, the education level was taken into account. The final invited participants had

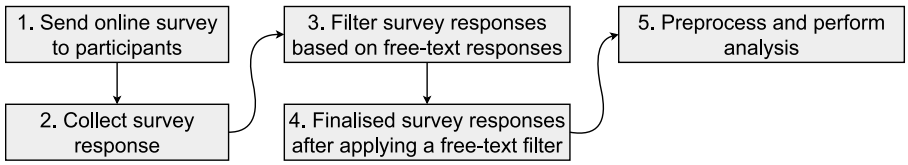

Fig. 11. The evaluation process for an online survey.

backgrounds in computer science, healthcare, social science, agriculture, finance, education, and engineering. A total of 22 responses were collected. Since the survey was conducted online, random responses are possible. To avoid random responses, the collected survey data was subsequently filtered based on the free-text responses provided in the survey. Following data filtering, two responses were eliminated from the survey. The responses were removed when they were highly inconsistent, no corresponding explanations were provided, or random answers were given in the free-text option. After preprocessing, the remaining 20 responses were used for additional analysis and to draw conclusions. The responses were anonymised, cleaned,[23] i.e. removing free-text responses as they were not consistent and organised as part of the preprocessing. Furthermore, as a part of the preprocessing, adjustments to the responses,[24] were also made for the question that asked the participants to rate the usefulness of the explanations (measured on a scale between 1 and 5). This is because of the observed anomaly. For example, the participants responded that they did not find the explanations useful and also did not understand the explanations, but despite this, some participants rated the explanations as useful (measured on a scale between 1 and 5). Corrections to such responses were made by adjusting the given rating to 1, which means "not at all useful". The analysis is performed using Tableau[25] a data visualisation and analysis tool.

The participants were from different age groups and geographical distributions. 50% of the participants were in the age group of 25–34, 25% were in the 18–14 age group, 10% were in the 35–44 age group, 10% were in the 45–54 age group, and the remaining 5% were in the age group of 65 and above. Similarly, the 55% of the participants were from Nigeria, 15% were from Ghana, and the remaining 10% each came from Austria, the Netherlands, and the United States of America. In terms of educational qualifications, a majority of the participants (50%) have either a bachelor's degree completed or are pursuing one. Similarly, 30% of respondents are either studying for or have completed a PhD, followed by those with a master's degree (completed or studying for), who constitute 15% of the participants. 5 percent of the participants are at the diploma level. The occupational backgrounds, which are shown in Fig. 12, of the participants vary widely. However, despite the varying occupational backgrounds, the majority of the respondents are either still involved in or were involved in farming in the past. Sixty-five percent of respondents indicated that they are currently engaged in farming or have been in the past.

In terms of AI familiarity, the majority of survey respondents, 90%, were familiar with AI. However, as can be observed in Fig. 13, the majority of them were not experts, only around 5% had a good understanding of AI. The remainder had either heard of AI or had a moderate understanding of it. Following the occupational data, a conclusion can be made that the majority of respondents are non-expert AI users, which makes the survey results more useful for the evaluation, as the explainability is geared towards regular users and not AI specialists.

[23] The cleaned version of the survey data is available in the GitHub repository where the code is present.

[24] Only the responses related to SHAP explanations (or the breakdown of SHAP explanations' usefulness) were modified. Adjustments were made to four responses.

[25] https://www.tableau.com/

The original input image, the corresponding prediction (top 1), and the SHAP explanation values were displayed in order to assess the explainability of the SHAP. The SHAP explainability image from Section 1 is used. Then, a yes-or-no question is asked to determine whether the survey participants found the explanation helpful for understanding the disease, i.e. the predicted disease. However, prior to posing a question, the information regarding the meaning of SHAP values is provided. After analysing the survey data, an insightful observation can be made. Fig. 14 compares the SHAP-based explanations and the explanations based on the proposed approach in terms of usefulness and comprehension. As can be observed from Fig. 14(a), 65% of the respondents find the SHAP explanations (see Fig. 1) useful, while 35% did not find the SHAP explanations helpful. However, in terms of comprehension (see Fig. 14(b)), 65% of respondents were not able to understand the disease following SHAP-based explanations. The remaining 35% of survey participants who were able to understand the SHAP-based explanations could be attributed to those with higher education qualifications and someone with a background in computer science or someone who has been working in a related field. Moreover, Fig. 15 shows the fine-grained analysis of the SHAP-based explanations. As can be observed from Fig. 15, 25% of the participants find the SHAP-based explanations very useful, 20% find them slightly useful, 15% find them moderately useful, and 5% find them extremely useful. Moreover, following the occupational statistics (see Fig. 12), it can said that the SHAP-based explanations are not very useful, even for highly qualified people such as researchers. Overall, a significant proportion of participants find the SHAP-based explanations to be little to moderately useful.

Similarly, when it comes to the explainability of the proposed method for this study, as shown in Fig. 14(c), 95% of participants find the explanations helpful (or useful), with 5% being the exception. Regarding the comprehension, 85% of the participants (see Fig. 14(d)) were able to understand the explanations of the proposed approach, while 15% were not able to understand. This indicates that there is still room for improvement in the proposed approach, which can be one of the future research directions. Similar feedback was received in the open-text response, where participants mentioned that there is a need to add examples of symptoms in explanations, suggested adding images along with textual explanations, and also suggested providing explanations on how to prevent the illness. A similar response to not being able to see the images is also provided in the case of the usefulness of proposed method explainability where it was marked with response "no", i.e. participants did not find useful. Table 10 in Appendix provides the free-text survey responses from the participants for both SHAP and the proposed approach. Additionally, in comparison with the comprehension of SHAP-based explainability, the proposed approach's comprehension is 50% higher, demonstrating the advantages of this study's explainability approach.

Furthermore, Fig. 16 further depicts a fine-grained evaluation of the usefulness (or helpfulness) of the explanations generated by the proposed method. As can be observed from Fig. 16, with the exception of 5% of participants, all of them find the explanations generated by the proposed approach useful. Overall, 10% find the explanations of the

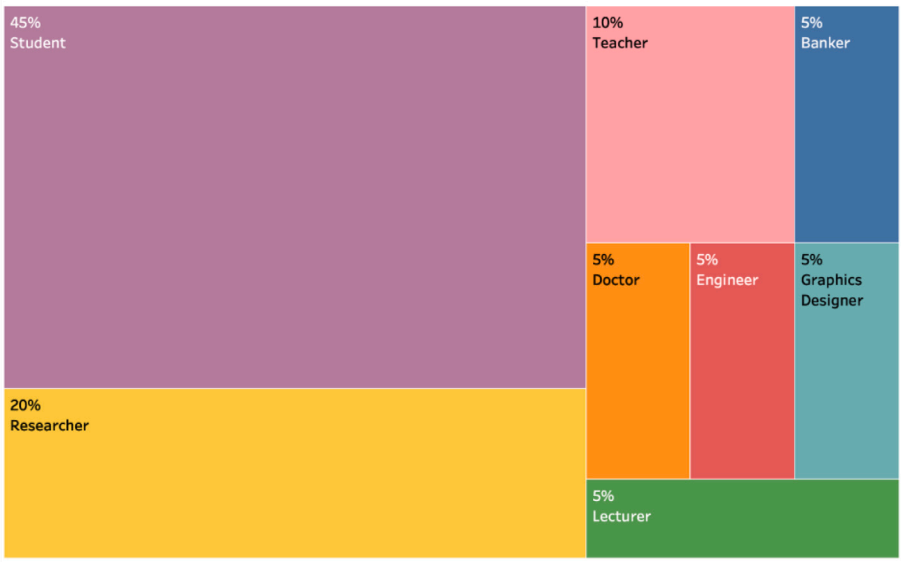

Fig. 12. Occupation-based distribution of survey respondents.

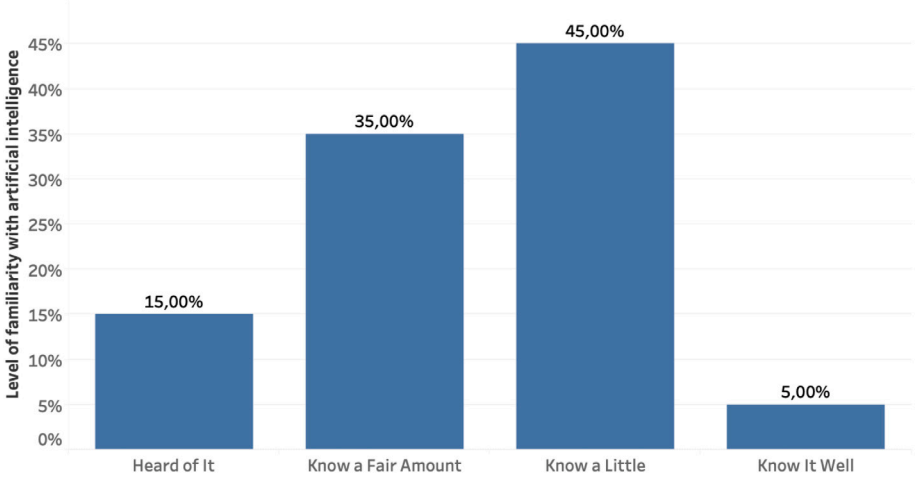

Fig. 13. The familiarity of the participants with AI, which is measured on a scale of 1–5. The respective options are: (1) never heard of it, (2) heard of it, (3) know a little, (4) know a fair amount, and (5) know it well.

proposed approach extremely useful, 60% find them very useful, 20% find them moderately useful, and 5% find them slightly useful. Upon comparing the usefulness of the proposed approach with the SHAP-based explanations, it clearly shows the benefit of the explainability based on the proposed approach and validates the claim about the benefits of the proposed explainability approach.

In summary, the following conclusions can be drawn from the evaluation of the survey response: (i) the visualisation of SHAP-based explanations is still useful even if they are not comprehensible, (ii) the proposed approach's explainability is very effective both in terms of usefulness and comprehension compared to explanations based on SHAP (or LIME), which are the popular and becoming a de-facto standard in ML explainability (Scapin et al., 2022), (iii) the fact that participants from diverse fields were able to understand and find the explanations useful demonstrates the very high value of the proposed explainability approach, even in other domains, and (iv) despite being highly effective, there is still room for improvement in the explainability of the proposed approach, which is to show where in the plant the diseases are and add more explanations to the cause and remedy of the diseases.

Table 9 shows the comparison of the proposed approach results with the state-of-the-art studies in terms of the accuracy. The study's relevance (i.e. the study utilises the same dataset as in this study) is the primary criterion for its selection. The second criterion is based on its recency (2022 and 2023). In addition, the best performance is selected from state-of-the-art studies when selecting performance for

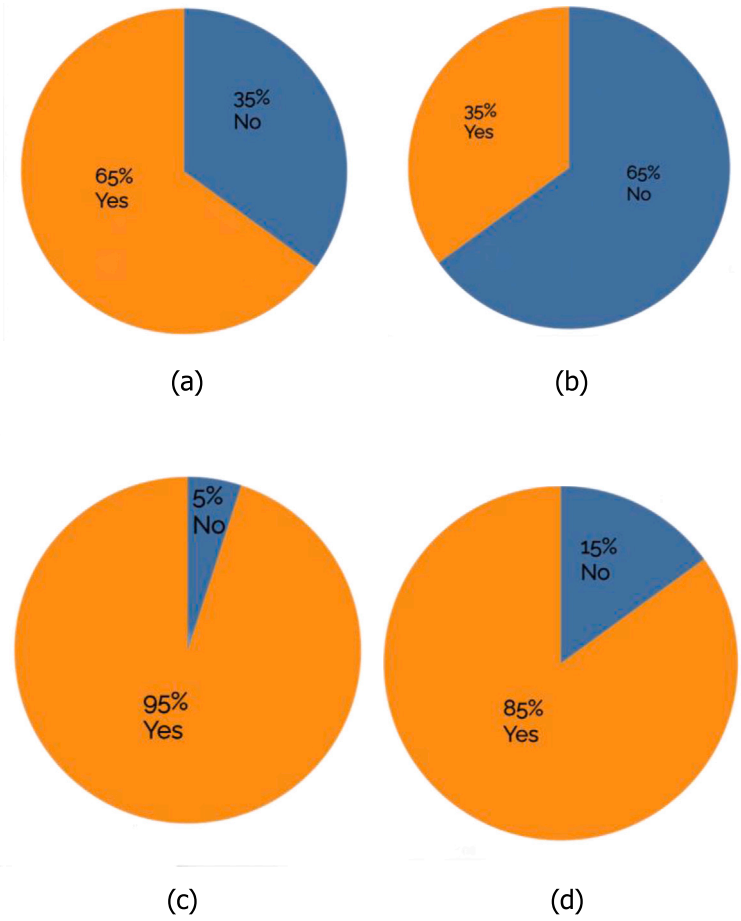

Fig. 14. Comparison of the usefulness and comprehension of the SHAP-based explanations (top) and the proposed approach (bottom). **(a)** Distributions of the participants responses according to the usefulness of the explanations generated using SHAP. **(b)** Participant distributions based on their comprehension of SHAP-based explanations. **(c)** Distributions of the participants responses based on the usefulness of the explanations generated using the proposed approach. **(d)** Participant distributions based on their comprehension of the proposed approach explanations.

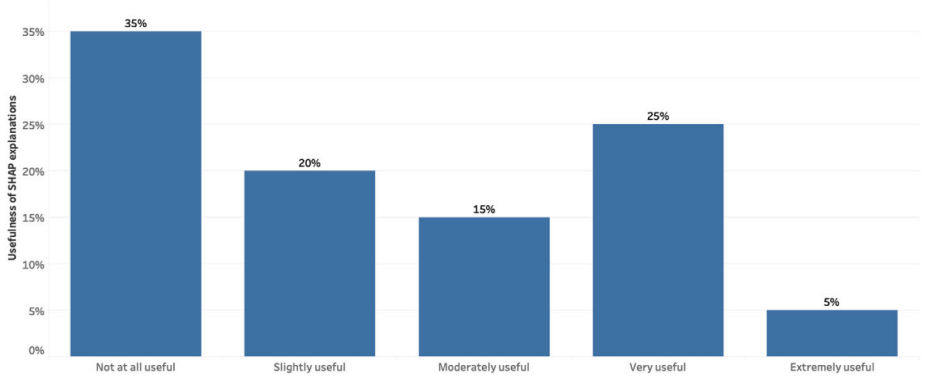

Fig. 15. The level of usefulness of the SHAP explanations that are measured on a scale of 1–5. The respective options are: (1) not at all useful, (2) slightly useful, (3) moderately useful, (4) very useful, and (5) extremely useful.

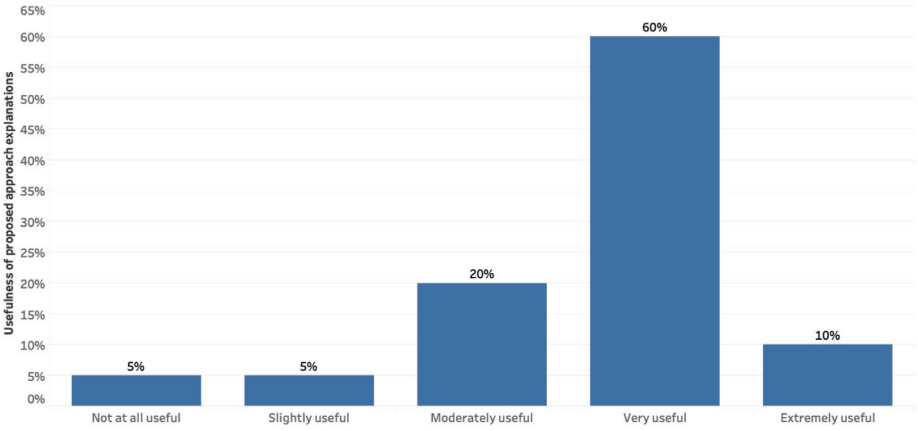

Fig. 16. The level of usefulness of the explanations that are measured on a scale of 1–5 and that are based on the proposed approach of this study. The respective options are: (1) not at all useful, (2) slightly useful, (3) moderately useful, (4) very useful, and (5) extremely useful.

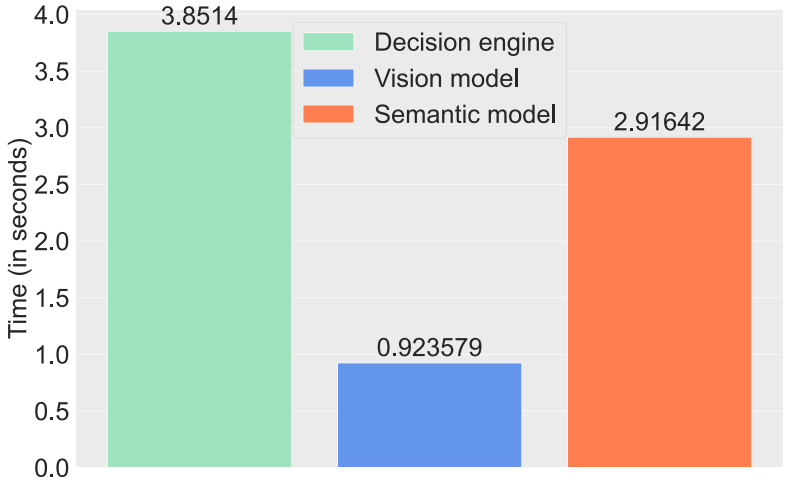

Fig. 17. The average prediction time (or prediction latency) measured for 5268 samples for vision models, semantic models, and the combination of predictions from vision and semantic models (i.e. decision engine), i.e. a proposed approach.

comparison. Table 9 clearly demonstrates that the proposed method outperforms state-of-the-art methods, with the exception of Kumar et al.'s (Kumar et al., 2023) work, which outperforms the result of this study by 0.0025. However, there are two important differences to take into account: (i) Kumar et al.'s (Kumar et al., 2023) work does not provide explainability, and (ii) this study uses a large, noisy cassava image dataset. This is because this study uses the combined datasets released by the Makerere University AI Lab in 2019 and 2020 (see Section 4.1), while Kumar et al.'s (Kumar et al., 2023) work only uses the dataset from 2019. A similar situation is present in other studies. Therefore, conclusion can be made that the proposed method is robust.

The third evaluation is on latency, as discussed in Section 4.5. Fig. 17 illustrates the average inference time for of the proposed approach. The semantic model requires an average of 2.91 s, while the vision model requires only 0.91 s. The vision model utilises GPUs, whereas the semantic model does not. This is one reason for the significant difference in inference time between the vision model and semantic model. In addition, the combined average inference time is 3.85 s, which is acceptable given the nature of the task (i.e. no hard real-time requirements). Therefore, with this, a conclusion can be drawn that the proposed approach is also suitable for real-world deployment.

Table 8
The accuracy after combining the predictions from the vision model and the semantic model, i.e. the proposed approach.

Vision model weight	Semantic model weight	Combined accuracy
0.5	0.5	**0.905**

Table 9
A comparison of the proposed method to the state-of-the-art studies in terms of accuracy and user-level explainability.

Study	Model/Methods	Accuracy	User-level explanation
(Emmanuel et al., 2023)	MobileNet V2	0.901	No
(Ahishakiye et al., 2023)	Ensemble of GLVQ, GMLVQ and LGMLVQ	0.82	No
(Kumar et al., 2023)	Ensemble of EfficientNet, SEResNeXt, ViT, DeIT and MobileNetV3	0.9075	No
(Paiva-Peredo, 2023)	DenseNet169	0.7477	No
(Chen et al., 2022)	Smooth-Taylor CE	0.893	No
(Ravi et al., 2022)	A_L_EfficientNet	0.8708	No
(Riaz et al., 2022)	EfficiennetB3	0.8303	No
(Anitha & Saranya, 2022)	CNN with data augmentation	0.90	No
The proposed approach	**Fusion of DL and semantic technology**	**0.905**	**Yes**

6. Conclusion

This study presents novel work on combining semantic technology and DL with a focus on cassava disease. This research demonstrated how semantic technology and DL can be combined to address the following limitations: (i) the lack of domain knowledge and contextual information in DL (see Section 3), and (ii) the lack of user-level explainability of DL (see Section 3.2.3). Moreover, by combining semantic technology and DL, as demonstrated by the results and comparison with the state-of-the-art works (see Section 5), this study also addressed the limitations of semantic technology, which are their limited abilities to learn complex patterns like DL. Furthermore, through the latency evaluation, this study demonstrated the suitability of the proposed approach for real-world scenarios.

From the performance evaluation and comparison with state-of-the-art studies, this study demonstrated the benefits of the proposed method and its robustness. However, the true benefit lies in the generalisability of the proposed approach, which can be used to solve other problems in the agricultural (and also other) domain, its ability to incorporate domain knowledge and generate, generate user-level explainability (see Section 3.2.3) and combine the symbolic approach based on semantic technology with a non-symbolic approach, also referred to as a numeric approach (or neuro-symbolic approach). This is particularly important when building AI (or AI systems) in a societal context, as the majority of the users are non-experts, and it is essential that they do not blindly trust predictions from the AI system. Moreover, this work enables the ability to include knowledge and rich contextual information that is not available in data, e.g. images, that helps in improving prediction, as demonstrated by the result (see Table 8), and build trustworthy AI systems.

However, if the domain knowledge is inaccurate or highly biased, it can act as a bottleneck, decreasing rather than increasing the accuracy of the prediction. Therefore, the information (or domain knowledge) should be incorporated with extreme caution, and multiple domain experts should be involved to eliminate (or reduce) bias. In the case of strict real-time requirements, the semantic model's performance can be a bottleneck, and this is therefore regarded as a limitation of this study that requires future improvement.

As potential future work, three directions are identified. The first is to make additional performance enhancements and incorporate more domain knowledge. The second objective is to apply the proposed work to domains, such as healthcare, that require AI systems that are explainable, have good performance, and can also make use of domain knowledge. The third research direction is to improve the explainability further following the user response, as discussed in Section 5.

CRediT authorship contribution statement

Tek Raj Chhetri: Conceptualization, Methodology, Investigation, Validation, Data curation, Writing – original draft, Writing – review & editing, Visualisation, Software, Formal analysis. **Armin Hohenegger:** Investigation, Software, Validation, Formal analysis, Data curation, Writing – review & editing, Visualisation. **Anna Fensel:** Writing – review & editing, Funding acquisition. **Mariam Aramide Kasali:** Writing – review & editing, Investigation. **Asiru Afeez Adekunle:** Writing – review & editing, Investigation.

Declaration of competing interest

The authors declare no conflict of interest.

Data availability

The data and code are made available online details of which is provided in the manuscript.

Survey Data: Towards Improving Prediction Accuracy and User-Level Explainability Using Deep Learning and Knowledge Graphs: A Study on Cassava Disease (Original data) (GitHub)

Acknowledgements

This research is partially supported by Interreg Österreich-Bayern 2014–2020 programme project KI-Net: Bausteine für KI-basierte Optimierungen in der industriellen Fertigung (grant agreement : AB 292). We would like thank to the High-Performance Computing Centre (HPC) at the University of Innsbruck for providing the LEO HPC infrastructure for our experiment. We also want to express our gratitude to Michael Fink, a member of the HPC staff, for his assistance with HPC administrative tasks and for accelerating the procedure to reduce delay. In addition, we would like to thank everyone who participated in our survey and gave us permission to analyse and utilise their responses in our research.

Appendix

See Table 10.

Table 10
The responses from the participants that provides reasons why they selected a particular option. The responses presented in this table remove responses that include text such as "I don't understand" or incomplete answer.

SHAP-based explanations	Explanations based on proposed approach
Free-text response providing the reasons why participants selected the option "yes" for usability and comprehension for the yes-or-no question.	
1. The colour in the prediction and explanation is same. 2. The spots show the affected area. 3. It affects chlorophyll production and leads to the reduction in crop yield.	1. It gave a written explanation of the disease. 2. It is clear what the disease is, the explanation is given and the percentage certainty are given as well. 3. The written text helps. 4. Learnt that the disease occurs mostly in moist soil. 5. Because the text tell the exact thing and it can be understood. 6. The generated explanation gave a view of how the disease thrives and this will aid in how to combat it. 7. Form the generated explanation, I could decipher that cassava mosaic is a high casualty disease that affects cassava plants. 8. There is a proper breakdown on the casual environment for this particular disease. 9. It explains the predisposing factors that could promote the disease growth. 10. I learnt that the disease could be devastating during the rainy season.
Free-text response providing the reasons why participants selected the option "no" for usability and comprehension for the yes-or-no question.	
1. Prediction is not clear to me. 2. The explanation is just a series of dots on a canvas. However, from the text above the prediction image, I am able to tell what the disease is. 3. Not clear how the cassava disease affects plants and where it shows up on the plant.	The explanation is not visible.
Free-text response providing the reasons why participants gave particular scores for usefulness ratings that are measured on a scale between 1 and 5.	
1. I did not understand How the prediction model is made and how the explanation helps or adds to diagnose and explain additional information about the disease. 2. Not clear if the photo shows a distribution of the disease on the plant as in location (leaves or roots). 3. Because the colour is based on the observations. Some people may perceive the colour in different way. 4. I responded NO because I am not an avid user of AI and the generated images could not be easily understood. 5. It requires a high level of knowledge to understand. 6. It will help farmers to identify the disease faster. 7. It explains how wide the disease has spread.	1. Since the prediction and explanation are correct, it is helpful to know when the disease is worse, what soil condition and temperature. 2. It is straight forward. 3. The explanation could be improved in terms of for instance cause, how it can be fixed and perhaps, other diseases that may look the same even to a human observer 4. Should add examples of the symptoms in the text. 5. Having more explanation with some graphics as well would be helpful. 6. I gave the above rating because seeing the explanation above is an Avenue to know how to move with the precautions necessary to combat the disease 7. Easily understood. 8. It gave detailed information about the disease 9. I often see the plants on the field 10. Not technical to understand.

References

Abbas, A., Jain, S., Gour, M., & Vankudothu, S. (2021). Tomato plant disease detection using transfer learning with C-GAN synthetic images. *Computers and Electronics in Agriculture*, *187*, Article 106279. http://dx.doi.org/10.1016/j.compag.2021.106279.

Ahishakiye, E., Mwangi, W., Murithi, P., Wario, R., Kanobe, F., & Danison, T. (2023). An ensemble model based on learning vector quantization algorithms for early detection of cassava diseases using spectral data. In P. Ndayizigamiye, H. Twinomurinzi, B. Kalema, K. Bwalya, & M. Bembe (Eds.), *Digital-for-development: Enabling transformation, inclusion and sustainability through ICTs* (pp. 320–328). Cham: Springer Nature Switzerland.

Ajayi, C. O., & Olutumise, A. I. (2018). Determinants of food security and technical efficiency of cassava farmers in Ondo State, Nigeria. *International Food and Agribusiness Management Review*, *21*(7), 915–928. http://dx.doi.org/10.22434/ifamr2016.0151.

Amador-Domínguez, E., Serrano, E., & Manrique, D. (2023). GEnI: A framework for the generation of explanations and insights of knowledge graph embedding predictions. *Neurocomputing*, *521*, 199–212. http://dx.doi.org/10.1016/j.neucom.2022.12.010.

American Phytopathological Society (2016). Plant disease ontology. https://github.com/Planteome/plant-disease-ontology. (Online accessed 18 July 2022).

Anitha, J., & Saranya, N. (2022). Cassava leaf disease identification and detection using deep learning approach. *International Journal of Computers Communications & Control*, *17*(2), http://dx.doi.org/10.15837/ijccc.2022.2.4356.

Anzolin, A., Toppi, J., Petti, M., Cincotti, F., & Astolfi, L. (2021). SEED-g: Simulated EEG data generator for testing connectivity algorithms. *Sensors*, *21*(11), 3632. http://dx.doi.org/10.3390/s21113632.

Ashwinkumar, S., Rajagopal, S., Manimaran, V., & Jegajothi, B. (2022). Automated plant leaf disease detection and classification using optimal MobileNet based convolutional neural networks. *Materials Today: Proceedings*, *51*, 480–487. http://dx.doi.org/10.1016/j.matpr.2021.05.584, CMAE'21.

Atila, Ü., Uçar, M., Akyol, K., & Uçar, E. (2021). Plant leaf disease classification using EfficientNet deep learning model. *Ecological Informatics*, *61*, Article 101182. http://dx.doi.org/10.1016/j.ecoinf.2020.101182.

Ayoub Shaikh, T., Rasool, T., & Rasheed Lone, F. (2022). Towards leveraging the role of machine learning and artificial intelligence in precision agriculture and smart farming. *Computers and Electronics in Agriculture*, *198*, Article 107119. http://dx.doi.org/10.1016/j.compag.2022.107119.

Bahani, K., Ali-Ou-Salah, H., Moujabbir, M., Oukarfi, B., & Ramdani, M. (2020). A novel interpretable model for solar radiation prediction based on adaptive fuzzy clustering and linguistic hedges. In *Proceedings of the 13th international conference on intelligent systems: Theories and applications*. New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3419604.3419807.

Ball, H. L. (2019). Conducting online surveys. *Journal of Human Lactation*, *35*(3), 413–417. http://dx.doi.org/10.1177/0890334419848734, PMID: 31084575.

Bedi, P., & Gole, P. (2021). Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network. *Artificial Intelligence in Agriculture*, *5*, 90–101. http://dx.doi.org/10.1016/j.aiia.2021.05.002.

Benedikt, M., Kersting, K., Kolaitis, P. G., & Neider, D. (2020). Logic and learning (Dagstuhl Seminar 19361). *Dagstuhl Reports*, *9*(9), 1–22. http://dx.doi.org/10.4230/DagRep.9.9.1.

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.

Chaddad, A., Peng, J., Xu, J., & Bouridane, A. (2023). Survey of explainable AI techniques in healthcare. *Sensors*, *23*(2), 634. http://dx.doi.org/10.3390/s23020634.

Chan, M.-C., Pai, K.-C., Su, S.-A., Wang, M.-S., Wu, C.-L., & Chao, W.-C. (2022). Explainable machine learning to predict long-term mortality in critically ill ventilated patients: a retrospective study in central Taiwan. *BMC Medical Informatics and Decision Making*, *22*(1), 75. http://dx.doi.org/10.1186/s12911-022-01817-6.

Chen, Y., Xu, K., Zhou, P., Ban, X., & He, D. (2022). Improved cross entropy loss for noisy labels in vision leaf disease classification. *IET Image Processing*, *16*(6), 1511–1519. http://dx.doi.org/10.1049/ipr2.12402.

Chen, J., Zhang, D., Suzauddola, M., & Zeb, A. (2021). Identifying crop diseases using attention embedded MobileNet-V2 model. *Applied Soft Computing*, *113*, Article 107901. http://dx.doi.org/10.1016/j.asoc.2021.107901.

Chhetri, T. R., Dehury, C. K., Lind, A., Srirama, S. N., & Fensel, A. (2022). A combined system metrics approach to cloud service reliability using artificial intelligence. *Big Data and Cognitive Computing*, *6*(1), 26. http://dx.doi.org/10.3390/bdcc6010026.

Chhetri, T. R., Hohenegger, A., Fensel, A., Aramide, K. M., & Adekunle, A. A. (2022). Code: Towards an explainable artificial intelligence using deep learning and knowledge graphs: A study on cassava disease. https://github.com/Research-Tek/xai-cassava-agriculture. (Last accessed 28 March 2023).

Chhetri, T. R., Kurteva, A., Adigun, J. G., & Fensel, A. (2022). Knowledge graph based hard drive failure prediction. *Sensors*, *22*(3), 985. http://dx.doi.org/10.3390/s22030985.

Detras, J., Borja, F. N., McNally, K., Mauleon, R., William, J. M. P., Ruaraidh, E., Hamilton, S., & Grenier, C. (2016). Rice ontology. https://cropontology.org/term/CO_320:ROOT. (Online accessed 19 July 2022).

Emmanuel, A., Mwangi, R. W., Murithi, P., Fredrick, K., & Danison, T. (2023). Classification of cassava leaf diseases using deep Gaussian transfer learning model. *Engineering Reports*, Article e12651. http://dx.doi.org/10.1002/eng2.12651.

Fakhfakh, F., Kacem, H. H., & Kacem, A. H. (2017). Simulation tools for cloud computing: A survey and comparative study. In *2017 IEEE/ACIS 16th international conference on computer and information science* (pp. 221–226). http://dx.doi.org/10.1109/ICIS.2017.7959997.

Feng, L., Shu, S., Lin, Z., Lv, F., Li, L., & An, B. (2021). Can cross entropy loss be robust to label noise? In *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence* (pp. 2206–2212).

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture*, *145*, 311–318. http://dx.doi.org/10.1016/j.compag.2018.01.009.

Food and Agriculture Organization of the United Nations (2021). Climate change fans spread of pests and threatens plants and crops, new FAO study. URL: https://www.fao.org/news/story/en/item/1402920/icode/. (Last accessed 31 March 2023).

Gaur, M., Faldu, K., & Sheth, A. (2021). Semantics of the black-box: Can knowledge graphs help make deep learning systems more interpretable and explainable? *IEEE Internet Computing*, *25*(1), 51–59. http://dx.doi.org/10.1109/MIC.2020.3031769.

Gaur, M., Gunaratna, K., Bhatt, S., & Sheth, A. (2022). Knowledge-infused learning: A sweet spot in neuro-symbolic AI. *IEEE Internet Computing*, *26*(4), 5–11. http://dx.doi.org/10.1109/MIC.2022.3179759.

Gohil, S. (2021). Dataset: Cassava plant disease Merged 2019–2020. https://www.kaggle.com/datasets/srg9000/cassava-plant-disease-merged-20192020. (Last accessed 27 August 2022).

Greff, K., Belletti, F., Beyer, L., Doersch, C., Du, Y., Duckworth, D., Fleet, D. J., Gnanapragasam, D., Golemo, F., Herrmann, C., Kipf, T., Kundu, A., Lagun, D., Laradji, I., Liu, H.-T. D., Meyer, H., Miao, Y., Nowrouzezahrai, D., Oztireli, C., ... Tagliasacchi, A. (2022). Kubric: A scalable dataset generator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 3749–3761).

Halabi, A. (2009). Plant protection ontology. https://sites.google.com/site/ppontology/home. (Online accessed 19 July 2022).

Holzinger, A., & Müller, H. (2021). Toward human–AI interfaces to support explainability and causability in medical AI. *Computer*, *54*(10), 78–86. http://dx.doi.org/10.1109/MC.2021.3092610.

Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosof, B., & Dean, M. (2004). SWRL: A semantic web rule language combining OWL and RuleML. *W3C Member Submission*, *21*(79), 1–31.

Janowicz, K., Haller, A., Cox, S. J., Le Phuoc, D., & Lefrançois, M. (2019). SOSA: A lightweight ontology for sensors, observations, samples, and actuators. *Journal of Web Semantics*, *56*, 1–10. http://dx.doi.org/10.1016/j.websem.2018.06.003.

Jearanaiwongkul, W., Anutariya, C., & Andres, F. (2018). An ontology-based approach to plant disease identification system. In *IAIT 2018*, *Proceedings of the 10th international conference on advances in information technology*. New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3291280.3291786.

Jiménez-Luna, J., Grisoni, F., & Schneider, G. (2020). Drug discovery with explainable artificial intelligence. *Nature Machine Intelligence*, *2*(10), 573–584. http://dx.doi.org/10.1038/s42256-020-00236-4.

Kannammal, A., Guhanesvar, M., & Venketesz, R. R. (2023). Predictive maintenance for remote field IoT devices—A deep learning and cloud-based approach. In S. Shakya, G. Papakostas, & K. A. Kamel (Eds.), *Mobile computing and sustainable informatics* (pp. 567–585). Singapore: Springer Nature Singapore.

Kumar, H., Velu, S., Lokesh, A., Suman, K., & Chebrolu, S. (2023). Cassava leaf disease detection using ensembling of EfficientNet, SEResNeXt, ViT, DeIT and MobileNetV3 models. In R. P. Yadav, S. J. Nanda, P. S. Rana, & M.-H. Lim (Eds.), *Proceedings of the international conference on paradigms of computing, communication and data sciences* (pp. 183–193). Singapore: Springer Nature Singapore.

Kuzlu, M., Cali, U., Sharma, V., & Güler, Ö. (2020). Gaining insight into solar photovoltaic power generation forecasting utilizing explainable artificial intelligence tools. *IEEE Access*, *8*, 187814–187823. http://dx.doi.org/10.1109/ACCESS.2020.3031477.

Lacasta, J., Lopez-Pellicer, F. J., Espejo-García, B., Nogueras-Iso, J., & Zarazaga-Soria, F. J. (2018). Agricultural recommendation system for crop protection. *Computers and Electronics in Agriculture*, *152*, 82–89. http://dx.doi.org/10.1016/j.compag.2018.06.049.

Lagos-Ortiz, K., Medina-Moreira, J., Paredes-Valverde, M. A., Espinoza-Morán, W., & Valencia-García, R. (2017). An ontology-based decision support system for the diagnosis of plant diseases. *Journal of Information Technology Research (JITR)*, *10*(4), 42–55. http://dx.doi.org/10.4018/JITR.2017100103.

Loshchilov, I., & Hutter, F. (2016). Sgdr: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983.

Luampon, R., & Charmongkolpradit, S. (2019). Temperature and relative humidity effect on equilibrium moisture content of cassava pulp. *Research in Agricultural Engineering*, *65*(1), 13–19.

Lucic, A., Ter Hoeve, M. A., Tolomei, G., De Rijke, M., & Silvestri, F. (2022). CF-GNNExplainer: Counterfactual explanations for graph neural networks. In G. Camps-Valls, F. J. R. Ruiz, & I. Valera (Eds.), *Proceedings of machine learning research*: *vol. 151*, *Proceedings of the 25th international conference on artificial intelligence and statistics* (pp. 4499–4511). PMLR, URL: https://proceedings.mlr.press/v151/lucic22a.html.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems, vol. 30*. Curran Associates, Inc., URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf.

Machlev, R., Heistrene, L., Perl, M., Levy, K., Belikov, J., Mannor, S., & Levron, Y. (2022). Explainable artificial intelligence (XAI) techniques for energy and power systems: Review, challenges and opportunities. *Energy and AI*, *9*, Article 100169. http://dx.doi.org/10.1016/j.egyai.2022.100169.

Mitrentsis, G., & Lens, H. (2022). An interpretable probabilistic model for short-term solar power forecasting using natural gradient boosting. *Applied Energy*, *309*, Article 118473. http://dx.doi.org/10.1016/j.apenergy.2021.118473.

Müller, R., Kornblith, S., & Hinton, G. E. (2019). When does label smoothing help? In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems, vol. 32*. Curran Associates, Inc.

Mwebaze, E., Gebru, T., Frome, A., Nsumba, S., & Tusubira, J. (2019). iCassava 2019 fine-grained visual categorization challenge. http://dx.doi.org/10.48550/ARXIV.1908.02900, arXiv.

Nagasubramanian, K., Jones, S., Singh, A. K., Sarkar, S., Singh, A., & Ganapathysubramanian, B. (2019). Plant disease identification using explainable 3D deep learning on hyperspectral images. *Plant Methods*, *15*(1), 98. http://dx.doi.org/10.1186/s13007-019-0479-8.

Noy, N., & McGuinness, D. L. (2001). *Ontology development 101: A guide to creating your first ontology*: *Technical report KSL-01-05 and SMI-2001-0880*, Stanford Knowledge Systems Laboratory and Stanford Medical Informatics.

Paiva-Peredo, E. (2023). Deep learning for the classification of cassava leaf diseases in unbalanced field data set. In I. Woungang, S. K. Dhurandher, K. K. Pattanaik, A. Verma, & P. Verma (Eds.), *Advanced network technologies and intelligent computing* (pp. 101–114). Cham: Springer Nature Switzerland.

Raschka, S., & Mirjalili, V. (2017). *Python machine learning*. Packt Publishing Ltd.

Ravi, V., Acharya, V., & Pham, T. D. (2022). Attention deep learning-based large-scale learning classifier for cassava leaf disease classification. *Expert Systems*, *39*(2), Article e12862. http://dx.doi.org/10.1111/exsy.12862.

Rawat, A., Sushil, R., Agarwal, A., & Sikander, A. (2021). A new approach for VM failure prediction using stochastic model in cloud. *IETE Journal of Research*, *67*(2), 165–172. http://dx.doi.org/10.1080/03772063.2018.1537814.

Riaz, S. M., Ahsan, M., & Akram, M. U. (2022). Diagnosis of cassava leaf diseases and classification using deep learning techniques. In *2022 16th International conference on open source systems and technologies* (pp. 1–8). http://dx.doi.org/10.1109/ICOSST57195.2022.10016854.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135–1144). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/2939672.2939778.

Rodríguez-García, M. Á., & García-Sánchez, F. (2020). CropPestO: An ontology model for identifying and managing plant pests and diseases. In R. Valencia-García, G. Alcaraz-Marmol, J. Del Cioppo-Morstadt, N. Vera-Lucio, & M. Bucaram-Leverone (Eds.), *Technologies and innovation* (pp. 18–29). Cham: Springer International Publishing.

Rodríguez-García, M. Á., García-Sánchez, F., & Valencia-García, R. (2021). Knowledge-based system for crop pests and diseases recognition. *Electronics*, *10*(8), http://dx.doi.org/10.3390/electronics10080905.

Roy, A. M., & Bhaduri, J. (2021). A deep learning enabled multi-class plant disease detection model based on computer vision. *AI*, *2*(3), 413–428. http://dx.doi.org/10.3390/ai2030026.

Sahu, P., & Sinha, V. K. (2022). Plant disease detection using transfer learning with DL model. In S. Shakya, K. Ntalianis, & K. A. Kamel (Eds.), *Mobile computing and sustainable informatics* (pp. 169–180). Singapore: Springer Nature Singapore.

Sammani, F., Mukherjee, T., & Deligiannis, N. (2022). NLX-gpt: A model for natural language explanations in vision and vision-language tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 8322–8332).

Scapin, D., Cisotto, G., Gindullina, E., & Badia, L. (2022). Shapley value as an aid to biomedical machine learning: a heart disease dataset analysis. In *2022 22nd IEEE international symposium on cluster, cloud and internet computing* (pp. 933–939). http://dx.doi.org/10.1109/CCGrid54584.2022.00113.

Seena Radhakrishnan, A., Suja, G., & Sreekumar, J. (2022). How sustainable is organic management in cassava? Evidences from yield, soil quality, energetics and economics in the humid tropics of South India. *Scientia Horticulturae*, *293*, Article 110723. http://dx.doi.org/10.1016/j.scienta.2021.110723.

Shah, D., Trivedi, V., Sheth, V., Shah, A., & Chauhan, U. (2022). ResTS: Residual deep interpretable architecture for plant disease detection. *Information Processing in Agriculture*, *9*(2), 212–223. http://dx.doi.org/10.1016/j.inpa.2021.06.001.

Sharma, R., Kamble, S. S., Gunasekaran, A., Kumar, V., & Kumar, A. (2020). A systematic literature review on machine learning applications for sustainable agriculture supply chain performance. *Computers & Operations Research*, *119*, Article 104926.

Sharma, S., Santra, B., Jana, A., Tokala, S., Ganguly, N., & Goyal, P. (2019). Incorporating domain knowledge into medical NLI using knowledge graphs. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing* (pp. 6092–6097). Hong Kong, China: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/D19-1631.

Tan, M., & Le, Q. (2021a). EfficientNetV2: Smaller models and faster training. In M. Meila, & T. Zhang (Eds.), *Proceedings of machine learning research*: *vol. 139*, *Proceedings of the 38th international conference on machine learning* (pp. 10096–10106). PMLR, URL: https://proceedings.mlr.press/v139/tan21a.html.

Tan, M., & Le, Q. V. (2021b). EfficientNetV2: Smaller models and faster training. arXiv:2104.00298.

Too, E. C., Yujian, L., Njuki, S., & Yingchun, L. (2019). A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, *161*, 272–279. http://dx.doi.org/10.1016/j.compag.2018.03.032, BigData and DSS in Agriculture.

Toubeau, J.-F., Bottieau, J., Wang, Y., & Vallée, F. (2022). Interpretable probabilistic forecasting of imbalances in renewable-dominated electricity systems. *IEEE Transactions on Sustainable Energy*, *13*(2), 1267–1277. http://dx.doi.org/10.1109/TSTE.2021.3092137.

UN (2021). The sustainable development goals report 2021. https://unstats.un.org/sdgs/report/2021/The-Sustainable-Development-Goals-Report-2021.pdf. (Online accessed 16 July 2022).

Xie, S., Girshick, R., Dollar, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Yang, L., Wang, H., & Deleris, L. A. (2021). What does it mean to explain? A user-centered study on AI explainability. In H. Degen, & S. Ntoa (Eds.), *Artificial intelligence in HCI* (pp. 107–121). Cham: Springer International Publishing.

Yang, Z., Xu, Y., Hu, J., & Dong, S. (2023). Generating knowledge aware explanation for natural language inference. *Information Processing & Management*, *60*(2), Article 103245. http://dx.doi.org/10.1016/j.ipm.2022.103245.

Ying, Z., Bourgeois, D., You, J., Zitnik, M., & Leskovec, J. (2019). GNNExplainer: Generating explanations for graph neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems, vol. 32*. Curran Associates, Inc., URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/d80b7040b773199015de6d3b4293c8ff-Paper.pdf.

Yuan, L., Hou, Q., Jiang, Z., Feng, J., & Yan, S. (2021). Volo: Vision outlooker for visual recognition. arXiv preprint arXiv:2106.13112.

Zhang, K., Xu, P., & Zhang, J. (2020). Explainable AI in deep reinforcement learning models: A SHAP method applied in power system emergency control. In *2020 IEEE 4th conference on energy internet and energy system integration* (pp. 711–716). http://dx.doi.org/10.1109/EI250167.2020.9347147.