Decision Support System for Plant Disease Identification

Sachin Prabhu Thandapani[1,2], Subikshaa Senthilkumar[1,2],
and S. Shanmuga Priya[1,2](✉)

[1] Department of Computer Science and Engineering,
Amrita School of Engineering, Coimbatore, India
{cb.en.u4cse14441,
cb.en.u4cse14451}@cb.students.amrita.edu,
ss_priya@cb.amrita.edu
[2] Amrita Vishwa Vidyapeetham, Coimbatore, India

Abstract. The significance of agriculture in India and the amount of damage to the sector due to plant diseases, calls for a system which can identify plant diseases accurately. Improper identification of diseases and taking wrong measures to prevent the disease will be cost inefficient and time consuming. There are highly accurate existing techniques to identify plant diseases but they are specific to a particular crop. In this project, a generic system is developed to identify plant diseases accurately based on textual description of plant diseases. Dataset containing description of diseases is created using the concept of web scraping. The dataset is preprocessed where, keywords are extracted, categorized and grouped to obtain the list of symptoms from the disease description. Based on the symptoms provided by the user to the system, plant disease is identified and the output is the identified disease along with preventive measures.

Keywords: Decision support system · Plant disease identification
Grouping keywords · Rice disease

1 Introduction

Agricultural sector plays a vital role in contributing towards country's health and economy. Major variation in the climatic conditions which is not expected for a seasonal plant, has led to the arrival of pest and diseases. One of the major challenge faced by the farmers in the current scenario is that the proper care at the earlier stage will not affect the quantity and quality of the plant. Naked eye observation is common and an easy method to identify the diseases. But, these observations require continuous monitoring and high expertise in that field thereby failing to produce accurate results. Hence, identification of plant diseases is important as well as difficult. The proposed system focuses on addressing this issue. The advantage of using automatic detection technique is the significant reduction in cost as well as achieving high level of accuracy.

A. K. Luhach et al. (Eds.): ICAICR 2018, CCIS 956, pp. 217–229, 2019.
https://doi.org/10.1007/978-981-13-3143-5_19

We intend to create a system that helps farmer identify plant diseases based on textual description and obtain relevant measures for treatment. Various symptoms of a plant disease is obtained from the user. These symptoms are compared with the existing dataset containing plant diseases and their corresponding symptoms. Identified disease is provided to the user along with various preventive measures and methods for treatment.

The project uses web scraping technique to create dataset containing textual descriptions of plant diseases. Keywords are extracted from this dataset based on NLP techniques and various features. To increase the frequency of keywords in the document, textual description of plant diseases are scrapped from multiple websites. Extracted keywords are classified based on various predefined categories. Keywords are then grouped using distance and frequency measures. List of symptoms for each disease are generated and a novel approach is used to identify the plant disease or possible plant diseases. Users are provided with preventive measures scrapped from websites and based on user's requirements, they are connected to a website elaborating the symptoms of the disease.

The paper is organized as follows. Section 2 gives the reason for why the system has been developed. Section 3 examines various existing techniques and the limitations of them. Section 4 discusses the architecture of the system. Section 5 contains background information required to achieve the goals of this work and an overview of the system. In Sect. 6 screenshots of the system describing its functions are provided. Section 7 discusses of future work and Sect. 8 concludes the work.

2 Motivation

Identifying plant diseases is very crucial as it causes great deal of damage to the crops and thus affecting the agricultural sector. Identifying the plant diseases at the right time accurately is essential because, failing to identify the plant disease at the right time will lead to damaging crops beyond repair. Also wrongly identifying the plant disease will cause improper treatment of the disease which is both waste of crops and money. There are many existing tools to identify the plant's disease using image processing techniques [1, 2] or developing a knowledge base for rule based approach [3–5]. These systems have their limitations and thus requires development of a system without requirement for a camera or need to develop a knowledge base specific to certain plant disease. Hence, a technique is devised in this work that can identify the plant's disease accurately with just textual descriptions of the disease.

3 Related Works

Plant disease identification is a task of finding the right plant disease using the given symptoms. Earlier plant disease identification or classification techniques are mostly image based where the user is asked to provide the image of the affected plant to the system.

In the work by Singh and Misra [1], plant leaf diseases are identified using image segmentation and soft computing techniques. Plant disease is identified using image processing technique. The image of the plant is taken, quality improved by increasing the contrast and preprocessing techniques are applied to segment the affected parts. Green colored pixels are masked, threshold values are set and features are identified. But increasing the quality of the image alters the image, thus providing inaccurate results. Also, using only the green colored pixels is a drawback since other major symptoms contributing to the plant's disease are ignored. Johannes et al. [2] proposed a system for diagnosing plant disease using mobile capture devices, applied on a wheat use case. This method uses image processing techniques to identify the plant disease. Images of plant during early stages of disease is taken. Leaf is clipped from the image, segmented and analyzed. The features obtained from the process is then compared with the disease detection inference model, processed by meta-classifier and the result is provided. Only leaf of the plant is observed for identifying the disease. When the stem or any other part is affected by the disease, this technique fails. In these two works, there is a requirement of an additional hardware, a camera to capture the image of the plant. Hence, even if these techniques provide accurate results, they are not cost effective. Our tool does not require an additional hardware as it requires only textual descriptions and provides accurate results based on them.

Khan et al. [3] have developed a web based expert system- Dr. Wheat, for Diagnosis of Diseases and Pests in Pakistani Wheat. A database for wheat diseases is created, a survey was conducted to identify the problems in wheat. Finally, rule based technique is used to identify the wheat plant's disease. Shikar et al. [4] have created an expert system for diagnosis of diseases of rice plants. This is a rule based approach with a knowledge base of simple if-then rules. People with experience in the domain reviewed the rules. These rules are the basis of the internal logic of the inference engine. Forward and backward chaining techniques are used in this technique. In the techniques mentioned, the rule-based approach was used and this requires the creation of knowledge base specific to each disease. This requires a lot of human effort and thus leading to the possibility of human errors. In our proposed work, we have introduced a novel approach to solve this issue thus avoiding human errors entirely. Our method requires a database containing plant disease description which is automated using web scraping techniques, therefore requires less human effort. In our work, we extract keywords from the created database.

Matsuo and Ishizuka [6] extracted keywords from a single document using word co-occurrence statistical information and uses frequency to identify the keywords and the probability of them occurring. A co-occurrence matrix is constructed for the same. $\chi 2$ values are calculated and the terms with the highest $\chi 2$ value are then selected as the keyword. Terms that are not frequent but relevant are not identified by this technique and thus becomes a major drawback. Claude [7] performed single document keyphrase extraction using sentence clustering and latent dirichlet allocation. This technique uses the brown corpus to identify the potential words. A matrix is created and co-occurrence is computed. Dimensionality reduction and data clustering are then performed. Latent Dirichlet algorithm is then used to identify the keywords. However, this approach requires a corpus of similar documents, which is not always readily available and it is a major drawback. There are many existing techniques for keyphrase extraction [8–12] but most of them require the keyphrases to have a high frequency. This problem is solved during the creation of the database by scraping content from multiple websites thus increasing the frequency of the keywords. The above mentioned techniques for keyword extraction are successful but are less accurate. Thus the work by Balasubramanian et al. [13] is adopted in this system and it is explained in detail in later sections of the paper.

4 System Architecture

The scope of this project is to meet the needs of a farmer with symptoms of plant diseases but is clueless about the plant's disease and needs help with identification of the same to treat the disease accordingly. In this work, a system is developed to get symptoms from researchers and provide them with the respective plant disease and method of treatment. The working of the system is as follows:

Since there is no existing dataset with textual descriptions for plant diseases to work on, a dataset containing symptoms of plant diseases is created using the method of web scraping. For every disease, keyword extraction is performed by using NLP techniques and various text-related features. These keywords are categorized and a novel approach is used to group keywords into symptoms. Symptoms of plant diseases are obtained from the user, compared with the grouped keywords and identified plant disease or possible plant diseases are provided to the user. Also, the system provides the user with an option to visit a website containing symptoms and various techniques to prevent and treat the disease based on their requirements.

In order to meet the goals of this project, a system is developed. The architecture of the system is shown in Fig. 1. Major components of the system include –Web scraper, Keyword Extractor, Keyword Classifier, Disease Classifier, Database and User Interface. The working of the system and the functionalities of each component is explained in detail subsequently.

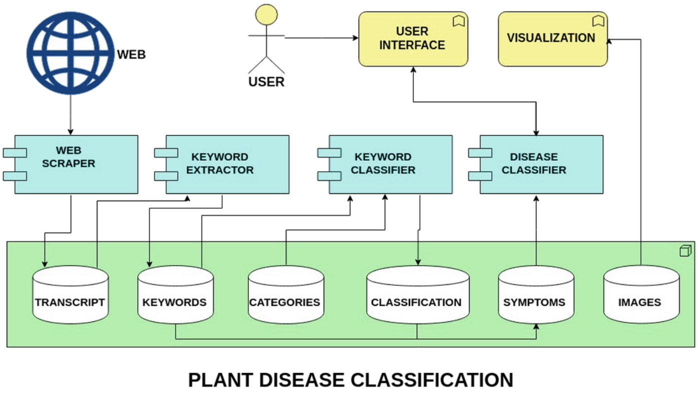


Fig. 1. System architecture

1. *Web Scraper*: Extracts relevant information related to plant disease description from the websites using web scraping techniques. Uses HTMLParser and BeautifulSoup packages to meet the needs of the work. Creates a dataset containing Description of plant diseases from multiple websites.
2. *Keyword Extractor*: Using the dataset created by web scraper, applying the techniques mention in Sect. 3, keywords are extracted from documents and stored in the database.
3. *Keyword Classifier*: This module uses certain predefined characteristics with keywords associated with them and uses the keywords generated by keyword extractor to categorize the keywords based on characteristics.
4. *Disease Identifier*: Major component of the system which groups keywords and generates scores for them. Interacts with the user interface to obtain symptoms from the user and processes them to obtain the required result.
5. *Database*: Stores all the results from intermediate steps involved in identifying the plant's disease. Information stored in the database include:
    i. Description of plant disease
    ii. Keywords and categories
    iii. Categorized and grouped keywords
    iv. Various scores generate
6. *User Interface*: Serves as a medium for interaction between the user and the system. Obtains symptoms from the user and displays the results.

5 Methodology

In this section, the background information required to meet the goals of the work and the working of the system is explained in detail.

5.1 Dataset Creation (Web Scraping)

Web scraping is a software technique used to extract information from websites. There are many python libraries such as BeautifulSoup, Scrapy, lxml, etc. for web scraping. In this project, to extract the HTML content from websites, the following libraries have been used:

(i) HTML Parser
(ii) BeautifulSoup

Many websites offer information related to plant diseases but not all of these websites are well organized. Some of the websites provide little information for only a few diseases and others have repetitive information. Considering these problems, four websites have been identified which contain enough information about the majority of the diseases. The websites are as follows: TNAU Agritech Portal [14], Plantwise [15] and KnowledgeBank [16]. URLs of all diseases from the list of websites taken and categorized based on diseases. Using the URLs imported, the goal is to extract symptoms of plant diseases from the website. It is essential to understand the way in which HTML documents from different websites are structured. The HTML content, in different web pages, were structured in a way specific to that particular domain. Hence a pattern common to all the diseases belonging to the same website is identified to extract the information related to the symptoms of the disease. Tags associated with the symptoms are analyzed and also regular expression matching is performed since some web pages are poorly structured. Using these identified patterns, we retrieve the symptoms of the diseases. For every plant disease, symptoms of the disease from multiple websites are exported to a document and dataset is created.

5.2 Keyword Extraction and Categorization

Web Keyword extraction helps in identifying words that relate best to the subject of the document. Various approaches to keyword extraction include rule-based linguistic approaches, statistical approaches, machine learning approaches and domain-specific approaches. For this project, we use a supervised machine learning approach, defined from previous works on keyword extraction [1]. This method uses multiple features of keywords such as dispersion, C-value, and TF-IDF.

(i) Dispersion is the measure of the spread of keywords in the document. This helps us to identify the words that have high spread and high frequency. Dispersion of a keyword, with occurrences I_a, length of occurrences |I_a| and variance as var(I_a) is mathematically defined as

$$dispersion(a) = \frac{|I_a|}{var(I_a)} \tag{1}$$

(ii) C-value identifies the significance of a term to a document. It combines both linguistic approach and statistical approach, thereby, improving accuracy. It is calculated as

$$C - value(a) = \begin{cases} \log_2|a| * freq(a), & if\,a\,not\,nested \\ \log_2|a| \; * \; \left\{\frac{1}{T_a} * \sum b\,in\,T_a \; * \; freq(b)\right\}, & otherwise \end{cases} \tag{2}$$

(iii) TF-IDF, frequency-inverse document frequency, is a common measure to identify the importance of a term to the document. It is a frequency based measure and thus higher occurrence of the keyword, higher is its TF-IDF value.

For every document containing plant disease descriptions, stop words are removed, stemming is performed and keywords are extracted. Now the extracted keywords are categorized into characteristics specific to plant diseases such as part affected, visible symptom, shape and size of the symptom, weather condition, etc. To do so, a dataset containing these characteristics and all the keywords related to these categories was created. Each keyword from the extracted keyword list is compared with different characteristics and if it belongs to a particular category it is tagged with that category.

Categorizing the keyword involves a lot of search operations. In order to improve the efficiency of search operations and to reduce the search complexity, trie data structure is used. Multiple trie data structures corresponding to multiple characteristics are created and the keywords specific to a particular characteristic are added to the respective trie. Searching for keywords is performed by prefix matching which reduces the search complexity to O (log(n)). Keywords along with their categories are obtained from this process. Table 1 displays various keywords, their feature scores and corresponding categories.

Table 1. Keyword feature scores and categories

Keyword	tfidf	Dispersion	c-score	Category
Panicle	0.002323	0.5000	0.0	Part
Powdery	0.014785	0.8571	0.0	Symptom
Brown	0.006902	0.8000	1.0	Color
Long	0.001221	0.0000	4.0	Size

5.3 Grouping of Keywords

From the extracted keywords, it is essential to combine these keywords in order to identify the symptoms explicit to the disease. For this purpose, keywords are grouped in pairs of two based on a novel approach. Considering keywords Key1and Key2, the algorithm for grouping the keywords is as follows:

(i) Key1 and Key2 are compared and various positions of the keywords in the document are obtained.

(ii) Positions of Key1 and Key2 are subtracted to find the distance between occurrences between the key pair. Only the keywords occurring close to each other, within 50 characters distance limit and their distance values in list DistVal are taken into consideration. Also, the keyword pairs belonging to same categories are excluded since there are no two same characteristics will together describe a symptom.

(iii) Length of DistVal is decided to be the *frequency* of occurrence of the keywords together in the document. Mean and Standard Deviation of values in the list are also considered to make sure that the distances between Key1 and Key2 are similar throughout the document.

(iv) Key1 and Key2 pair is given a score that will help determine the closeness and frequency of occurrence of keywords in the document. The score for the group is determined as follows:

$$combscore(key1, key2) = \frac{closeness(key1, key2)}{frequency(key1, key2)} \tag{3}$$

$$closeness(key1, key2) = (mean(DistList) * standard\,deviation(DistList)) \tag{4}$$

(v) Keywords with lower combscore are given higher priority because lower mean value suggests that the keywords occur close to each other; lower standard deviation implies that they are close to each other throughout the document; higher frequency shows that they occur together more frequently in the document.

(vi) The combscore of all keyword pairs specific to a document are then normalized with respect to other scores in the document to minimize the impact of uncertainties in multiple documents.

Thus keywords are grouped based on their closeness and frequency of occurrence together in the document and combination scores are generated. Table 2 shows the sample of keyword pairs and their generated scores.

Table 2. Keywords combination score

Keyword 1	Keyword 2	Combscore
Leaf	Fungal	270.998
Leaf	Sheath	11.322
Powdery	Whitish	1.031
Panicle	Brown	61.967
Leaf	Lesion	103.681

5.4 User Input and Identification of Plant Disease

In order to identify the plant disease, symptoms of the plant disease for which the user needs identification if obtained from them. The user is given an option to enter a group of symptoms in the order "part affected, symptom, the color of the part affected or the symptom, size of the symptom" for more accurate results. The input for this kind of symptom will be like "leaf, spot, yellow, large". The user is also provided with an option to enter unique factors individually.

Using the symptoms provided by the user, plant disease/possible plant diseases are identified using the following steps

(i) Since the keywords are grouped into pairs, the symptoms provided the by the user are also grouped into pairs of two. Hence from the list of symptoms provided by the user as a group of 4 elements, keywords are converted into groups of two and one. This does not apply to the unique factors as they are entered individually.

(ii) Two scores are generated for the list of symptoms entered by the user.
   i. Hitscore
   ii. Keyscore

(iii) For every disease, using the symptoms provided by the user, hitscore is determined as the ratio of the number of keywords in symptom present in the document to the total number of keywords in the symptom. For the keywords *Ks* entered as a symptom by the user:

$$Hitscore(doc) = \frac{N(Ks)\, in\, doc}{N(Ks)} \tag{5}$$

(iv) For every disease, keyscore is determined as the sum of all inverted combscore of keyword groups generated from symptoms list entered by the user. For the keyword group *Kg* entered as symptoms by the user:

$$Keyscore(doc) = \sum (1 - combscore(Kg)) \tag{6}$$

(v) A weighted sum of these scores is calculated and keyword combination scores are given higher weight.

$$Finalscore(doc) = 0.25 * Hitscore(doc) + 0.75 * Keyscore(doc) \tag{7}$$

Table 3 contains the hitscore, combscore and finalscore generated for various diseases. Based on the generated finalscore for each document, results are provided to the user based on conditions explained in the following subsection.

Table 3. Final score for different diseases

Disease name	Hit score	Combination score	Final score
Brown spot	0.89	5.50	3.96
False smut	0.11	0.0	0.04
Blast	0.78	7.02	4.94

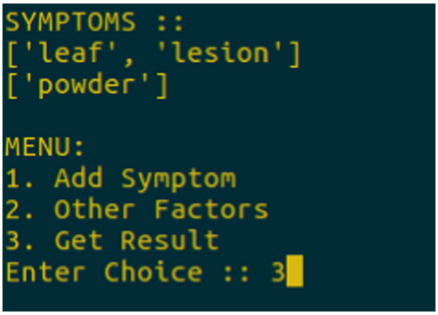
5.5 Output

Identified plant diseases are provided to the user based on their input and their requirements. When the user provides less number of symptoms, the system cannot pinpoint the plant disease. Thereby, instead of providing the user with one plant disease, he is provided with a set of possible plant diseases. The diseases provided are chosen based on the criteria where Hitscore == 1. The user is also notified that the symptoms provided are not sufficient and given an option to include more symptoms.

When the user provides enough symptoms, the plant disease containing the highest generated finalscore is provided to the user as the “Identified Plant Disease”. The user is also provided with a link to a website containing details of the plant’s disease, its symptoms, preventive measure and methods for treatment of the identified disease in an elaborate manner.

6 Results

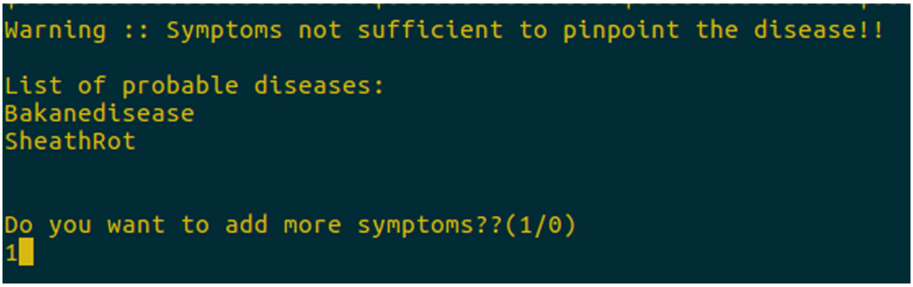
This section contains a set of images representing various kinds of inputs and different types of outputs provided to the user, step-by-step. Figures 2 and 3 display the first set of symptoms entered by the user and the corresponding result generated with the list of possible diseases. Figure 4 shows the additional symptom entered and Fig. 5 displays the result generated. Then the user is redirected to a website containing the detailed description of the disease and method to prevent and treat them.



```
SYMPTOMS ::
['leaf', 'lesion']
['powder']

MENU:
1. Add Symptom
2. Other Factors
3. Get Result
Enter Choice :: 3
```

Fig. 2. First set of symptoms

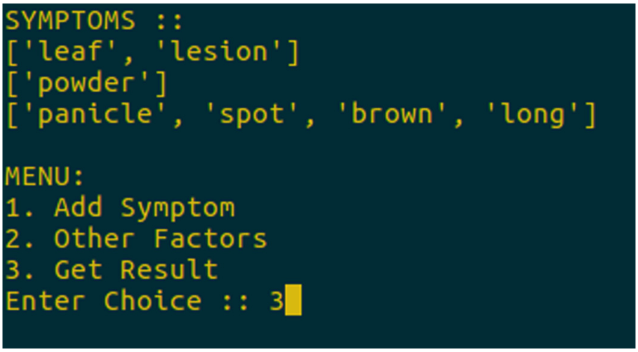


```
Warning :: Symptoms not sufficient to pinpoint the disease!!

List of probable diseases:
Bakanedisease
SheathRot


Do you want to add more symptoms??(1/0)
1
```

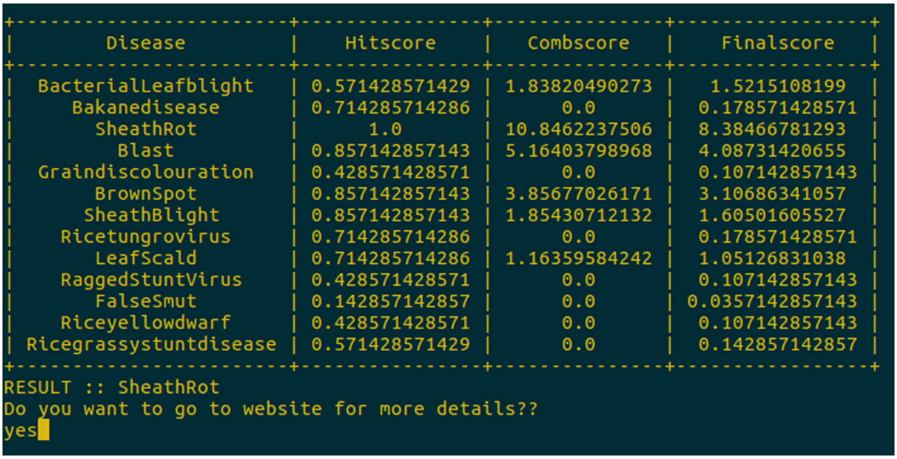
Fig. 3. List of possible diseases



```
SYMPTOMS ::
['leaf', 'lesion']
['powder']
['panicle', 'spot', 'brown', 'long']

MENU:
1. Add Symptom
2. Other Factors
3. Get Result
Enter Choice :: 3
```

Fig. 4. Second set of symptoms



```
+------------------------+----------------+---------------+-----------------+
|        Disease         |    Hitscore    |   Combscore   |   Finalscore    |
+------------------------+----------------+---------------+-----------------+
|  BacterialLeafblight   | 0.571428571429 | 1.83820490273 |   1.5215108199  |
|      Bakanedisease     | 0.714285714286 |      0.0      |  0.178571428571 |
|        SheathRot       |      1.0       | 10.8462237506 |  8.38466781293  |
|         Blast          | 0.857142857143 | 5.16403798968 |  4.08731420655  |
|  Graindiscolouration   | 0.428571428571 |      0.0      |  0.107142857143 |
|        BrownSpot       | 0.857142857143 | 3.85677026171 |  3.10686341057  |
|       SheathBlight     | 0.857142857143 | 1.85430712132 |  1.60501605527  |
|     Ricetungrovirus    | 0.714285714286 |      0.0      |  0.178571428571 |
|        LeafScald       | 0.714285714286 | 1.16359584242 |  1.05126831038  |
|     RaggedStuntVirus   | 0.428571428571 |      0.0      |  0.107142857143 |
|        FalseSmut       | 0.142857142857 |      0.0      | 0.0357142857143 |
|     Riceyellowdwarf    | 0.428571428571 |      0.0      |  0.107142857143 |
| Ricegrassystuntdisease | 0.571428571429 |      0.0      |  0.142857142857 |
+------------------------+----------------+---------------+-----------------+
RESULT :: SheathRot
Do you want to go to website for more details??
yes
```

Fig. 5. Result with generated scores

7 Future Work and Limitations

The system can further be developed to identify the diseases of various other crops. In this project, rice diseases was taken into consideration. Similarly, this system can be expanded to identify the plant diseases of agricultural and horticultural plants. This can be done by scraping the textual description of agricultural and horticultural plants and then performing the algorithm used in this project. In the proposed system, user has to specify symptoms in English. The system can be developed wherein users can specify symptoms in any language in order to identify the plant disease. For further development of the system, an application can be created where users can select from the options in drop down menu rather than typing the symptoms.

The system identifies the right disease when the correct symptoms are provided by the user. In case the user provides the wrong symptom, system will not be able to produce accurate results. In this system, dataset has been created only for all the diseases of rice plant. Using this dataset, the diseases are identified. The proposed system can further be extended to identify disease of all the plants by creating datasets.

8 Conclusion

A technique that identifies the plant disease based on the textual descriptions of symptoms provided by the user is developed. A dataset containing plant disease descriptions is developed. The system extracts keywords using supervised machine learning algorithm, categorizes and groups them. Finally based on user input scores are generated for every disease and the list of possible diseases or the identified disease is provided to the user as output. Also user is allowed to visit a website containing more details on the disease. This technique does not require any additional hardware to achieve its goal. The system is flexible as it provides user with different options to enter the symptoms. This developed technique is generic and can be applied for all plant diseases.

References

1. Singh, V., Misra, A.K.: Detection of plant leaf diseases using image segmentation and soft computing techniques. Inf. Process. Agric. **4**(1), 41–49 (2017)
2. Johannes, A., et al.: Automatic plant disease diagnosis using mobile capture devices, applied on a wheat use case. Comput. Electron. Agric. **138**, 200–209 (2017)
3. Khan, F.S., et al.: Dr. Wheat: a web-based expert system for diagnosis of diseases and pests in Pakistani wheat. In: Proceedings of the World Congress on Engineering, vol. 1, pp. 2–4 (2008)
4. Sarma, S.K., Singh, K.R., Singh, A.: An expert system for diagnosis of diseases in rice plant. Int. J. Artif. Intell. **1**(1), 26–31 (2010)
5. Shanmuga Priya, S., Abinaya, M.: Feature selection using random forest technique for the prediction of pest attack in cotton crops. Int. J. Pure Appl. Math. **118**, 2899–2903
6. Matsuo, Y., Ishizuka, M.: Keyword extraction from a single document using word co-occurrence statistical information. Int. J. Artif. Intell. Tools **13**(01), 157–169 (2004)
7. Pasquier, C.: Task 5: single document keyphrase extraction using sentence clustering and latent Dirichlet allocation. In: Proceedings of the 5th International Workshop on Semantic Evaluation. Association for Computational Linguistics, pp. 154–157 (2010)
8. El-Beltagy, S.R.: KP-miner: a simple system for effective keyphrase extraction. In: 2006 Innovations in Information Technology. IEEE, pp. 1–5 (2006)
9. Wan, X. Xiao, J.: Single document keyphrase extraction using neighborhood knowledge. In: AAAI, vol. 8, pp. 855–860 (2008)
10. Emu, I.H., et al.: An efficient approach for keyphrase extraction from english document. Int. J. Intell. Syst. Appl. **9**(12), 59 (2017)
11. D'Avanzo, E., Magnini, B.: A keyphrase-based approach to summarization: the lake system at duc-2005. In: Proceedings of DUC (2005)
12. Hulth, A.: Improved automatic keyword extraction given more linguistic knowledge. In: Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 216–223 (2003)

13. Balagopalan, A., et al.: Automatic keyphrase extraction and segmentation of video lectures. In: 2012 IEEE International Conference on Technology Enhanced Education (ICTEE). IEEE, pp. 1–10 (2012)
14. TNAU Agritech Portal. http://agritech.tnau.ac.in
15. CABI Plantwise. http://www.plantwise.org
16. IRRI Rice Knowledge Bank. http://www.knowledgebank.irri.org