ISSN 3027-6365 (Online)

วารสารวิทยาศาสตร์และเทคโนโลยี Homepage : *http://li01.tci-thaijo.org/index.php/tstj*
ปีที่ 33 ฉบับที่ 1 (มกราคม-กุมภาพันธ์ 2568) หน้า [106-119]

การจำแนกประเภทของเห็ดระหว่างเห็ดมีพิษหรือไม่มีพิษ โดยใช้เทคนิคการเรียนรู้ของเครื่อง

Mushroom Classification between Poisonous and Edible Mushrooms Using Machine Learning Techniques

ภิรมย์พร เถี่ยนมิตรภาพ[1*], วราภรณ์ วิยานนท์[2]

[1]สาขาวิทยาการข้อมูล คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กรุงเทพมหานคร 10110

[2]ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กรุงเทพมหานคร 10110

Phiromporn Tianmitrapap[1*], Waraporn Viyanon[2]

[1]Program of Data Science, Faculty of Science, Srinakharinwirot University, Bangkok 10110

[2]Department of Computer Science, Faculty of Science, Srinakharinwirot University, Bangkok 10110

Received 16 April 2023; Received in revised 25 January 2024; Accepted 4 June 2024

บทคัดย่อ

วิธีการจำแนกเห็ดพิษแบบดั้งเดิมอาจให้ผลไม่แม่นยำและใช้เวลานาน โดยวิธีที่คนนิยมใช้คือพิจารณาจากลักษณะทางกายภาพของเห็ด ซึ่งมีหลายลักษณะ ทำให้เสียเวลาในการพิจารณาได้ ดังนั้นงานวิจัยนี้จึงศึกษาการใช้เทคนิคการเรียนรู้ของเครื่องเพื่อจำแนกเห็ดออกเป็นสองประเภท คือ เห็ดรับประทานได้และเห็ดพิษ จากชุดข้อมูลสาธารณะของเห็ดครีบ จำนวน 8,124 รายการ ในตระกูล Agaricus และ Leiota จำนวน 23 สายพันธุ์ นำมาใช้ในการเรียนรู้ด้วย 5 แบบจำลอง ได้แก่ Logistic regression SVM decision tree random forest และ XGBoost ร่วมกับการใช้เทคนิคการคัดเลือกและดึงข้อมูลสำคัญในการจำแนกชนิดเห็ด ผลลัพธ์พบว่าแบบจำลอง Decision tree ร่วมกับเทคนิค RFE โดยใช้ชุดข้อมูล 60% และพิจารณาเพียง 3 คุณสมบัติ ได้แก่ กลิ่น ขนาดครีบเห็ด และสีรอยพิมพ์สปอร์ ให้ผลลัพธ์ดีที่สุด โดยมีค่า F1 score ถึง 99.32% และนำไปสร้างต้นแบบเว็บแอปพลิเคชันสำหรับการใช้งานต่อไป

คำสำคัญ: การจำแนกประเภทเห็ด; เทคนิคการเรียนรู้ของเครื่อง; ต้นไม้ตัดสินใจ; กระบวนการลบลักษณะตามลำดับ; เห็ดพิษ

***ผู้รับผิดชอบบทความ:** phiromporn.tia@g.swu.ac.th

doi: 10.14456/tstj.2025.8

Abstract

Traditional methods of identifying poisonous mushrooms can be both inaccurate and time-consuming. The most common approach involves visually inspecting the mushroom's physical characteristics, which can be labor-intensive and challenging. This study investigated the use of machine learning techniques to classify mushrooms into two categories: edible and poisonous. A public dataset of 8,124 gilled mushrooms from 23 species within the Agaricus and Leiota genera was used to train five machine learning models: logistic regression, support vector machines (SVMs), decision trees, random forests, and XGBoost. Feature selection and extraction technique were applied to identify the most important attributes for classifying mushroom species. The results indicated that the decision tree model, coupled with recursive feature elimination (RFE), achieved the best performance when using 60% of the data and focusing on three features: odor, gill size, and spore print color. This model produced an F1 score of 99.32%. Based on these finding, a prototype web application was developed for potential future use.

Keywords: Mushroom classification; Machine learning; Decision tree; Recursive feature elimination; Poisonous mushroom

1. บทนำ

เห็ดเป็นสิ่งมีชีวิตที่อยู่ในอาณาจักรเห็ดรา (Kingdom fungi) ซึ่งเห็ดจัดเป็นราชั้นสูง ที่อยู่ใน 2 ไฟลัม คือ Ascomycota และ Basidiomycota โดยเห็ดที่คนรู้จักทั่วไปและพบเห็นได้บ่อยจะเป็นเห็ดในไฟลัม Basidiomycota ที่มีชื่อทางวิชาการว่า “เห็ดครีบ (Gilled mushroom หรือ agaric mushroom)”[1] ซึ่งเห็ดตามธรรมชาติมีทั้งเห็ดที่รับประทานได้และเห็ดที่มีพิษ สำหรับเห็ดที่รับประทานได้นั้นให้คุณค่าทางโภชนาการ ส่วนเห็ดพิษคือเห็ดที่สร้างสารพิษชีวภาพทำให้คนเจ็บป่วยและถึงแก่ชีวิตได้ ดังนั้นการจำแนกประเภทของเห็ดระหว่างเห็ดมีพิษและเห็ดที่รับประทานได้จึงมีความสำคัญ

การจำแนกเห็ดพิษในประเทศไทยนั้นมีหลากหลายวิธี ได้แก่ การทดสอบเห็ดพิษด้วยภูมิปัญญาชาวบ้าน แต่พบว่าไม่ครอบคลุมเห็ดพิษทุกชนิด ทำให้เกิดผลลบลวงได้ และโดยทั่วไปมักมีการจำแนกเห็ดพิษด้วยลักษณะสัณฐานวิทยาที่สามารถมองด้วยตาเปล่าได้ แต่ยังมีข้อจำกัดคือเห็ดพิษบางชนิดมีลักษณะภายนอกคล้ายกับเห็ดที่รับประทานได้ จึงเกิดความเข้าใจผิดได้ [2] นอกจากนี้ยังมีการใช้วิธีตรวจชนิดสารพิษในห้องปฏิบัติการทางเคมี แต่จำเป็นต้องใช้สารพิษมาตรฐานในการเปรียบเทียบและใช้ระยะเวลานาน อีกทั้งยังมีวิธีที่สามารถตรวจชนิดของเห็ดอย่างจำเพาะเจาะจงได้โดยใช้การตรวจทางพันธุกรรม แต่ต้องใช้เครื่องมือและอุปกรณ์ซึ่งอาจไม่สะดวกในการใช้งาน [3]

ดังนั้นหากนำเทคนิคการเรียนรู้ของเครื่องมาประยุกต์ใช้สำหรับการจำแนกประเภทของเห็ด จะช่วยลดระยะเวลาและประหยัดค่าใช้จ่ายในการตรวจได้ ซึ่งพบงานวิจัยที่มีการใช้เทคนิคการเรียนรู้ของเครื่องในการจำแนกประเภทเห็ด ดังนี้ งานกลุ่มที่ 1: ไม่ได้ทำความสะอาดข้อมูลแต่มีการคัดเลือกหรือสกัดฟีเจอร์ เช่น งานของ Shuhaida Ismail และคณะ [4] ใช้เทคนิค PCA โดยไม่ได้ระบุจำนวนคอมโพแนนท์ และใช้แบบจำลอง Decision tree ซึ่งได้ค่า F1 score = 100% โดยฟีเจอร์ “กลิ่นของเห็ด” เป็นฟีเจอร์ที่มีความสำคัญที่สุด, งานของ S.K. Verma และคณะ [5] มีการสกัดฟีเจอร์แต่ไม่ระบุเทคนิค และมีการแบ่งชุดข้อมูลเป็น 5 อัตราส่วน คือ 40:60 50:50 60:40 70:30 และ 80:20 พบว่าแบบจำลอง ANFIS ที่อัตราส่วน 80:20 มีประสิทธิภาพดีที่สุด (Accuracy = 99.87%) ส่วนงานกลุ่มที่ 2: มีการทำความสะอาดข้อมูลและคัดเลือกหรือสกัดฟีเจอร์ เช่น งานของ Nadya Chitayae และคณะ [6] มีการลบแถวข้อมูลที่มีค่าว่างจำนวน 2,480 แถว (ในคอลัมน์ชื่อ “stalk_root”) และเลือกฟีเจอร์จำนวน 8 ฟีเจอร์ไปใช้ พบว่าแบบจำลอง Decision tree ให้ Cross validation score มากที่สุด (91.93%), งานของ V. Vanitha และคณะ [7] ได้ลบฟีเจอร์ “stalk-root” ซึ่งมีค่าว่างจำนวนมาก และคัดเลือกฟีเจอร์ด้วยวิธี Wrapper และ filter ได้ฟีเจอร์จำนวน 2 ฟีเจอร์คือ “odor” (กลิ่นของเห็ด) และ “spore_print_color” (สีรอยพิมพ์สปอร์) พบว่าแบบจำลอง Decision tree ให้ F1 score = 99%

จากการที่คนทั่วไปนิยมใช้ลักษณะทางสัณฐานวิทยาที่สามารถตรวจสอบได้จากภายนอก แต่ลักษณะภายนอกของเห็ดมีหลายคุณลักษณะซึ่งหากพิจารณาให้ครบทุกส่วน อาจทำให้เสียเวลาในการพิจารณาได้ ดังนั้นผู้วิจัยจึงได้สร้างเทคนิคการเรียนรู้ของเครื่องที่มีประสิทธิภาพดี เพื่อจำแนกประเภทของเห็ด โดยใช้ข้อมูลคุณลักษณะของเห็ด ประกอบการใช้เทคนิคการคัดเลือกฟีเจอร์ เพื่อค้นหาฟีเจอร์ที่สำคัญและจำเป็นต่อการจำแนกประเภทของเห็ด

2. วิธีดำเนินการวิจัย

งานวิจัยนี้ประกอบด้วยขั้นตอนการดำเนินงานดังรูปที่ 1 โดยอธิบายรายละเอียดในหัวข้อ 2.1-2.6

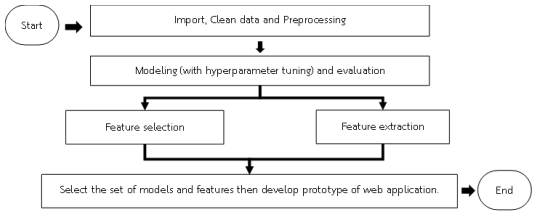

Figure 1 Experimental design

2.1 การนำเข้าชุดข้อมูล ทำความสะอาดข้อมูล และเตรียมข้อมูลเพื่อเข้าสู่แบบจำลอง

งานวิจัยนี้ใช้ชุดข้อมูลสาธารณะของเห็ดครีบจำนวน 23 สายพันธุ์ในตระกูล Agaricus และ Lepiota จำนวน 8,124 แถว ที่มีการระบุว่าเป็นเห็ดพิษหรือเห็ดที่รับประทานได้ โดยได้จากเว็บไซต์ kaggle ซึ่งเป็นข้อมูลที่ Jeff Schlimmer ได้บริจาคให้กับ UCI Machine Learning Repository โดยดึงข้อมูลจากหนังสือ The audubon society field guide to North American Mushrooms (1981) [8] โดยชุดข้อมูลประกอบด้วยตัวแปร 23 คอลัมน์ ได้แก่ สีหมวกเห็ด (cap_color), ลักษณะพื้นผิวบนหมวกเห็ด (cap_surface), รูปทรงหมวกเห็ด (cap_shape), สีครีบเห็ด (gill_color), ความแนบชิดของครีบกับก้านดอก (gill_attachment), ระยะห่างของครีบเห็ดแต่ละครีบ (gill_spacing), ขนาดครีบเห็ด (gill_size), รูปทรงก้านดอก (stalk_shape), ลักษณะรากที่ปลายก้านดอก (stalk_root), สีและลักษณะของพื้นผิวก้านดอกบริเวณเหนือวงแหวน และบริเวณด้านล่างของวงแหวน (stalk_color_above_ring stalk_color_below_ring stalk_surface_above_ring และ stalk_surface_below_ring) ประเภทเยื่อหุ้มดอกเห็ดที่พบ (veil_type), สีของเยื่อที่หุ้มดอกเห็ด (veil_color), จำนวนวงแหวนของเห็ด (ring_number), รูปร่างของวงแหวน (ring_type), รอยช้ำบนเห็ด (Bruise), กลิ่นของเห็ด (Odor), สีรอยพิมพ์สปอร์ (spore_print_color), การเกาะกลุ่มกันของเห็ด (Population) บริเวณที่พบเจอเห็ดเติบโต (habitat) และประเภทของเห็ด (class_ep) ดังตารางที่ 1

เมื่อสำรวจข้อมูลพบว่าตัวแปร “VTY” (ประเภทของเยื่อที่หุ้มดอกเห็ด) มีเพียงค่าเดียวคือ “Partial” และพบว่าตัวแปร “SRO” (ลักษณะรากที่อยู่ปลายก้านดอก) มีค่าว่างมากกว่า 30% ของทั้งหมด ดังนั้นจึงลบ 2 ตัวแปรนี้ทิ้งไป และแทนค่าภายในตัวแปร “RNU” (จำนวนวงแหวนของเห็ด) ให้เป็นตัวเลขโดยตรง คือ จาก “none” เป็น 0 จาก “one” เป็น 1 และจาก “two” เป็น 2

จากนั้นกำหนดให้ “CEP” (ประเภทของเห็ด) เป็นลาเบล และอีก 20 ตัวแปรที่เหลือเป็นฟีเจอร์ที่ใช้ในการสร้างแบบจำลอง พร้อมกับการแปลงค่าให้เป็นตัวเลขโดยสำหรับฟีเจอร์ใช้เทคนิค Ordinal encoder ส่วนลาเบลใช้ Label encoder

จากนั้นแบ่งชุดข้อมูลเป็นชุดข้อมูลสำหรับสร้างแบบจำลอง (Training set) และชุดข้อมูลสำหรับการทดสอบ (Test set) ทั้งหมด 3 อัตราส่วน คือ 50:50 (Training size = 50%) 60:40 (Training size = 60%) และ 70:30 (Training size = 70%) โดยกำหนดให้ในแต่ละอัตราส่วนมีเห็ดทั้งสองประเภทในจำนวนเท่าๆกัน แล้วปรับช่วงขอบเขตของฟีเจอร์ ด้วยเทคนิค “MinMax-scaler” ก่อนนำไปสร้างแบบจำลอง

Table 1 Mushroom dataset

Variables (Abbreviation)	Possibilities
cap_color (CCO)	brown, buff, cinnamon, gray, green, pink, purple, red, white, yellow
cap_surface (CSU)	fibrous, grooves, scaly, smooth
cap_shape (CSH)	bell, conical, convex, flat, knobbed, sunken
gill_color (GCO)	black, brown, buff, chocolate, gray, green, orange, pink, purple, red, white, yellow
gill_attachment (GAT)	attached, descending, free, notched
gill_spacing (GSP)	close, crowded, distant
gill_size (GSI)	broad, narrow
stalk_shape (SSH)	enlarging, tapering
stalk_root (SRO)	bulbous, club, cup, equal, rhizomorphs, rooted, missing
stalk_color_above_ring (SCA)	brown, buff, cinnamon, gray, orange, pink, red, white, yellow
stalk_color_below_ring (SCB)	
stalk_surface_above_ring (SSA)	fibrous, scaly, silky, smooth
stalk_surface_below_ring (SSB)	
veil_type (VTY)	partial, universal
veil_color (VCO)	brown, orange, white, yellow
ring_number (RNU)	none, one, two
ring_type (RTY)	cobwebby, evanescent, flaring, large, none, pendant, sheathing, zone
bruise (BRU)	bruises, no
odor (ODO)	almond, anise, creosote, fishy, foul, musty, none, pungent, spicy
spore_print_color (SPC)	black, brown, buff, chocolate, green, orange, purple, white, yellow
population (POP)	abundant, clustered, numerous, scattered, several, solitary
habitat (HAB)	grasses, leaves, meadows, paths, urban, waste, woods
class_ep (CEP)	edible, poisonous

2.2 การสร้างแบบจำลองเทคนิคการเรียนรู้ของเครื่องและปรับจูนหาค่า hyperparameters

งานวิจัยนี้ศึกษาทั้งหมด 5 แบบจำลอง โดยปรับจูน Hyperparameters ด้วย RandomizedSearchCV โดยแต่ละแบบจำลองมีหลักการดังนี้

2.2.1 การถดถอยลอจิสติก (Logistic regression) หาค่าความน่าจะเป็น โดยใช้สมการ Sigmoid function (สมการที่ 1)

$$\sigma(X) = \frac{1}{1+e^{-x}} \quad (1)$$

2.2.2 ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine; SVM) สร้างเส้นแบ่งระหว่าง class (เส้น Hyperplane) ของข้อมูลบน Feature space เพื่อแบ่งข้อมูลออกเป็นกลุ่ม โดยเป็นเส้นที่มีขอบเขตห่างจาก Support vector ที่เหมาะสม [9]

2.2.3 ต้นไม้ตัดสินใจ (Decision tree) มีการสร้างลำดับชั้นของการตัดสินใจขึ้นมาจากข้อมูลตัวแปรต้น โดยเริ่มจาก เงื่อนไขแรกของการตัดสินใจ (Root node) แล้วแตกกิ่งการตัดสินใจไปที่เงื่อนไขถัดไป (Decision node) จนได้ผลลัพธ์สุดท้ายออกมา (Leaf node) ซึ่งแบบจำลองนี้จะแตกกิ่งจนได้ผลลัพธ์ที่ถูกต้องที่สุด [10]

2.2.4 แรนดอมฟอเรสท์ (Random Forest) เป็น Ensemble model ของ Decision tree คือนำ Decision tree หลายอัน มาทำนายผล ซึ่งใช้เทคนิค Bootstrap aggregating โดยแต่ละ Decision tree นั้นมีการสุ่มฟีเจอร์และตัวอย่างที่ไม่เหมือนกัน ผลลัพธ์สุดท้ายของการทำนายนั้น ได้จากผลโหวตที่มากที่สุดของ Decision tree แต่ละต้น [11]

2.2.5 เอ็กซ์ตรีมเกรเดียนต์บูสติ้ง (Extreme Gradient Boosting; XGBoost) เป็น Ensemble model ของ Decision tree ด้วยเทคนิค Boosting คือ นำ Decision tree หลายอันมาเชื่อมกันเป็นลำดับ โดยแต่ละ Decision tree เรียนรู้จาก Error ของ Decision tree ก่อนหน้า และจะหยุดเรียนรู้เมื่อไม่เหลือรูปแบบของ Error จาก Decision tree ก่อนหน้าให้เรียนรู้อีก [12]

ในขั้นตอนนี้จะใช้ฟีเจอร์ทั้งหมด 20 ฟีเจอร์ในการสร้างแบบจำลอง และนำ Hyperparameters ที่ได้ไปใช้ในขั้นตอนที่ 2.3 และ 2.4 และตรวจสอบประสิทธิภาพของแบบจำลองเบื้องต้น

2.3 การคัดเลือกฟีเจอร์ (Feature selection)

งานวิจัยนี้ใช้ 2 เทคนิค คือ Recursive Feature Elimination (RFE) และ Chi-square โดย RFE เป็นวิธีคัดเลือกฟีเจอร์โดยคำนวณจากการเรียนรู้และทดสอบ ในการสร้างแบบจำลองที่เลือกไว้ และ Chi-square เป็นวิธีคัดเลือกฟีเจอร์ก่อนนำเข้าสู่แบบจำลองเพื่อสร้างการเรียนรู้ โดยวิธีนี้เหมาะสมกับข้อมูลที่มีฟีเจอร์และลาเบลที่มีลักษณะเป็น Category ซึ่งหากคะแนน Chi-square มากแปลว่ามีความสำคัญมาก [13]

โดยทั้งสองเทคนิค ได้ศึกษาประสิทธิภาพกับชุดข้อมูลสำหรับทดสอบ เมื่อใช้จำนวนฟีเจอร์เท่ากับ 3 5 7 และ 9 ฟีเจอร์

2.4 การสกัดฟีเจอร์ (Feature extraction)

งานวิจัยนี้ใช้เทคนิคการวิเคราะห์องค์ประกอบหลัก (Principal component analysis: PCA) ซึ่งเป็นการแปลงฟีเจอร์เดิมให้เป็นฟีเจอร์ใหม่ที่มีมิติลดลง [14] และศึกษาประสิทธิภาพกับชุดข้อมูลสำหรับทดสอบเมื่อใช้จำนวนคอมโพแนนท์เท่ากับ 3 5 7 และ 9 คอมโพแนนท์ พร้อมกับการหาองค์ประกอบของฟีเจอร์ในแต่ละคอมโพแนนท์ (Component loading)

2.5 การประเมินประสิทธิภาพของแบบจำลอง

งานวิจัยนี้ประเมินประสิทธิภาพของแบบจำลองเป็นค่า F1 score โดยหากค่า F1 score มีค่าสูงเข้าใกล้ 1 แปลว่าแบบจำลองมีประสิทธิภาพดี [15]

2.6 การพัฒนาต้นแบบเว็บแอปพลิเคชัน

งานวิจัยนี้ได้พิจารณาแบบจำลองและฟีเจอร์จากผลลัพธ์ที่ได้จากขั้นตอนที่ 2.3 เป็นหลัก โดยพิจารณาจากปัจจัยต่างๆ ได้แก่ ประสิทธิภาพ เทคนิคและจำนวน

ฟีเจอร์ที่ใช้เหมาะสม โดยผู้วิจัยสร้างต้นแบบหน้าเว็บแอปพลิเคชัน ด้วยเครื่องมือ “Streamlit” ซึ่งเหมาะกับการสร้างหน้าเว็บแอปพลิเคชันสำหรับเทคนิคการเรียนรู้ของเครื่อง [16]

3. ผลการวิจัยและวิจารณ์

3.1 ผลลัพธ์ของการสร้างแบบจำลองโดยใช้ 20 ฟีเจอร์

การสร้างแบบจำลองจำแนกประเภทเห็ดโดยใช้ฟีเจอร์ทั้ง 20 ฟีเจอร์พบว่าแบบจำลอง Logistic regression ให้ประสิทธิภาพที่ต่ำที่สุด เนื่องจาก Logistic regression เป็น Linear model ซึ่งมีความสามารถในการประมวลผลข้อมูลที่มีความซับซ้อนมาก ได้ไม่ดีเมื่อเทียบกับแบบจำลองอื่นในงานวิจัยนี้ ส่วนแบบจำลอง Decision tree ให้ค่า F1 score เท่ากับ 100% ยกเว้นที่ใช้ Training size = 60% ที่มีค่า F1 score เท่ากับ 99.69% เนื่องจาก Hyperparameters ที่แบบจำลอง Decision tree นั้นปรับค่าได้นั้นมีความแตกต่างกันในแต่ละ Training size ดังตารางที่ 2 และอาจเกิดจากความแตกต่างของการแบ่งชุดข้อมูลที่เกิดขึ้นในแต่ละ Training size ทำให้แบบจำลอง Decision tree ที่ใช้ Training size ที่แตกต่างกันเรียนรู้ได้ต่างกัน ส่วนอีก 3 แบบจำลอง คือ SVM Random Forest และ XGBoost ให้ค่า F1 score เท่ากับ 100% ดังรูปที่ 3

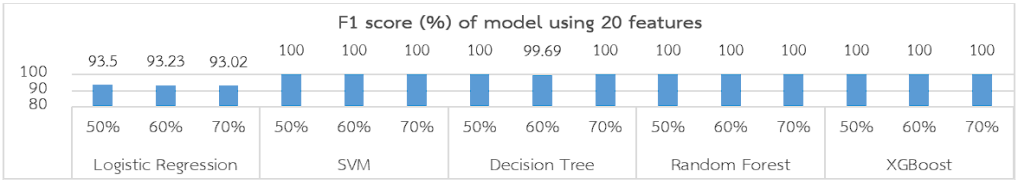

Figure 2 Performance metric (F1 score) of model using 20 features

Table 2 Parameters of each model

Models	Training sizes	Parameters
Logistic Regression	All	solver = ‘newton-cg’, penalty = ‘none’, C = 22.456734693877554
SVM	All	kernel= ‘poly’, gamma = 5.0, degree=3.6666666666666665, C=73.6868420526315
Decision tree	50%	min_samples_split = 15, max_leaf_nodes = 46, max_features = ‘sqrt’, max_depth = 30
	60%, 70%	min_samples_split = 9, max_leaf_nodes = 35, max_features = ‘log2’, max_depth = 21
Random Forest	50%	n_estimators= 300, min_samples_split = 71, max_leaf_nodes = 57, max_features = ‘log2’, max_depth =52
	60%, 70%	n_estimators= 300, min_samples_split = 52, max_leaf_nodes = 43, max_features = ‘log2’, max_depth =85
XGBoost	50%, 60%	n_estimators=300, subsample = 0.5, min_child_weight= 1, max_depth= 3, learning_rate= 0.22749999999999998, colsample_bytree= 1
	70%	n_estimators= 300, subsample = 1, min_child_weight= 10, max_depth= 4, learning_rate= 0.3, colsample_bytree= 1

3.2 ผลลัพธ์จากการคัดเลือกฟีเจอร์

ผลโดยรวมของทั้งสองเทคนิคพบว่าแบบจำลอง Logistic regression มีประสิทธิภาพน้อยกว่าแบบจำลองอื่น เนื่องจาก Logistic regression เป็น Linear model ซึ่งมีความสามารถในการประมวลผลข้อมูลที่มีความซับซ้อนมากๆ ได้ไม่ดี ส่วนผลลัพธ์ของเทคนิค RFE นั้นพบว่า แบบจำลองส่วนใหญ่มีประสิทธิภาพที่ดี ส่วนแบบจำลอง XGBoost ที่ Training size เท่ากับ 70% และใช้จำนวนฟีเจอร์เท่ากับ 3 ให้ประสิทธิภาพที่ต่ำที่สุดในการศึกษาทั้งหมด ซึ่งพบว่าฟีเจอร์ที่ได้นั้นค่อนข้างแตกต่างจากแบบจำลองอื่นที่ใช้จำนวนฟีเจอร์เท่ากัน (เท่ากับ 3) คือ ไม่พบว่าเลือกฟีเจอร์ชื่อ กลิ่นของเห็ด (ODO) หรือสีรอยพิมพ์สปอร์ (SPC) มาใช้ในการประมวลผล โดยสามารถแสดงข้อมูลฟีเจอร์ได้ในตารางที่ 3 แต่เทคนิค RFE นี้มีข้อจำกัดกับแบบจำลองที่ไม่มีฟังก์ชั่นการหา Feature importance หรือค่า Coefficient อย่าง SVM ที่ Kernel เป็น “Polynomial” นอกจากนี้พบว่าแบบจำลอง Decision tree และ Random forest มีประสิทธิภาพใกล้เคียงกัน โดยเฉพาะที่จำนวนฟีเจอร์เท่ากับ 3 พบว่ามีประสิทธิภาพเท่ากันในทุก Training size

เมื่อเปรียบเทียบผลลัพธ์ระหว่างเทคนิค Chi-square กับเทคนิค RFE นั้นพบว่าที่จำนวนฟีเจอร์น้อยๆ การใช้เทคนิค RFE มีประสิทธิภาพดีกว่า เนื่องจาก RFE เป็นเทคนิคที่คำนวณจากการเรียนรู้ในการสร้างแบบจำลอง ในขณะที่ Chi-square เป็นเทคนิคที่คัดเลือกฟีเจอร์ก่อนเข้าสู่แบบจำลอง ซึ่งจากกราฟในรูปที่ 4 เห็นได้ว่าที่ฟีเจอร์เท่ากับ 3 ของทุกๆ Training size นั้น การใช้เทคนิค RFE ทำให้ F1 score ของแบบจำลองส่วนใหญ่มีค่ามากกว่า 99% ในขณะที่การใช้เทคนิค Chi-square นั้น แบบจำลองต่างๆ มีค่า F1 score น้อยกว่า 90% เนื่องจากฟีเจอร์ที่ถูกคัดเลือกในแต่ละเทคนิคนั้นมีความแตกต่างกัน ดังแสดงในตารางที่ 3-4

Table 3 Feature names of feature selection (RFE)

No. of features	Models	Training sizes	Feature names (as Abbreviation)
3	Logistic regression	All	VCO, RNU, SPC
	XGBoost	50%, 60%	
		70%	SSH, VCO, RNU
	Random Forest	All	ODO, GSI, SPC
	Decision tree	50%, 60%	
		70%	ODO, SPC, POP
5	Logistic regression	All	BRU, GSP, VCO, RNU, SPC
	XGBoost	50%, 60%	ODO, GSI, VCO, RNU,SPC
		70%	GSP, SSH, VCO, RNU, SPC
	Random Forest	All	ODO, GSI, RTY, SPC, POP
	Decision tree	50%	ODO, SPC, SCB, POP, HAB
		60%	CCO, ODO, GSI, SPC, POP
		70%	ODO, GSP, GSI, SCA, HAB

Table 3 Feature names of feature selection (RFE) (Continue)

No. of features	Models	Training sizes	Feature names (as Abbreviation)
7	Logistic regression	All	BRU, GSP, SSA, VCO, RTY, RNU, SPC
	XGBoost	50%	ODO, GSP, GSI, VCO, RNU, SPC, POP
		60%	ODO, GSP, GSI, SSA, VCO, RNU, SPC
		70%	ODO, GSP, GSI, SSH, VCO, RNU, SPC
	Random Forest	All	BRU, ODO, GSP, GSI, RTY, SPC, POP
	Decision tree	50%	BRU, ODO, GSI, SSA, SSB, RTY, SPC
		60%	BRU, ODO, GSI, SSB, SCB, SPC, POP
		70%	ODO, GSP, GSI, SCA, RTY, SPC, POP
9	Logistic regression	All	CSU, BRU, GSP, GSI, SSA, VCO, RTY, RNU, SPC
	XGBoost	50%, 60%	ODO, GSP, POP, SSA, VCO, SSB, GSI, RNU, SPC
		70%	CSU, ODO, GSI, GSP, SSH, VCO, RNU, SPC, POP
	Random Forest	50%, 60%	BRU, ODO, GSP, GSI, RTY, SSA, SSB, SPC, POP
		70%	BRU, ODO, GSP, GSI, RTY, SSB, HAB, SPC, POP
	Decision tree	50%	BRU, GSP, GSI, SSH, HAB, SSB, SCB, SPC, POP
		60%	ODO, SSH, HAB, SSB, SCA, RTY, RNU, SPC, POP
		70%	BRU, ODO, GSP, GSI, SSH, HAB, RTY, SPC, SCA

Table 4 Feature names of feature selection (Chi-square)

No. of features	Models	Training sizes	Feature names (as Abbreviation)
3	All	All	GSI, BRU, GSP
5	All	All	GSI, BRU, GSP, SPC, RTY
7	All	All	GSI, BRU, GSP, SPC, RTY, GCO, CSU
9	All	All	GSI, BRU, GSP, SPC, RTY, GCO, CSU, POP, SCA

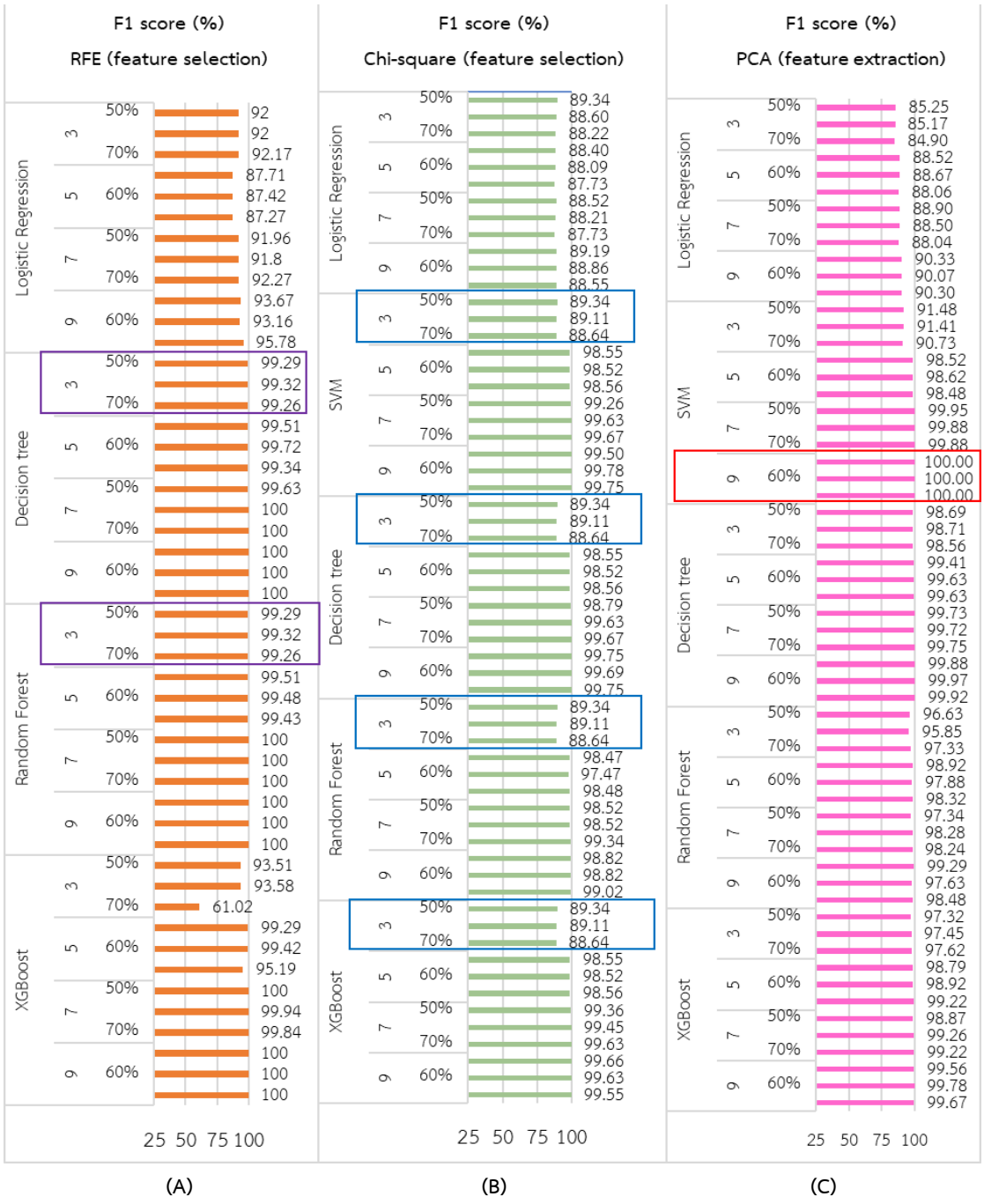

Figure 4 Performance metric of model: (A) Feature selection (RFE), (B) Feature selection (chi-square) and (C) Feature extraction (PCA)

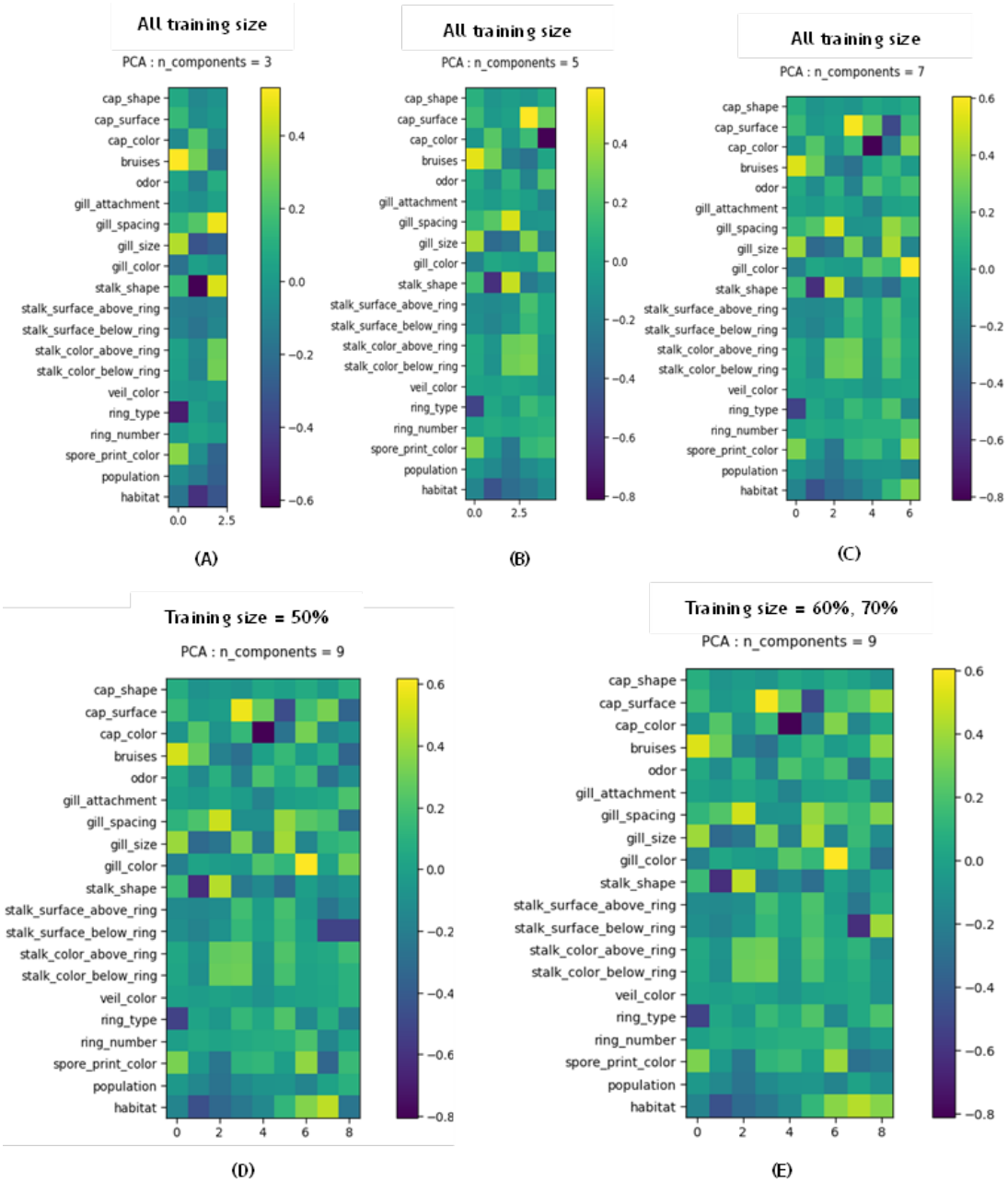

Figure 5 Component Loading of PCA technique in each n_components and training size: (A) n_components = 3 of all training size, (B) n_components = 5 of all training size, (C) n_components = 7 of all training size, (D) n_components = 9 of training size = 50% and (E) n_components = 9 of training size = 60%, 70%

3.3 ผลลัพธ์จากการสกัดฟีเจอร์

แบบจำลอง SVM ที่ใช้จำนวนคอมโพแนนท์เท่ากับ 9 ให้ค่า F1 score มากที่สุด คือ 100% เนื่องจาก SVM ที่มี kernel เป็น Polynomial สามารถประมวลผลในการใช้ฟีเจอร์ที่มีมิติสูงๆ ได้ดีและพบว่า Logistic regression ให้ประสิทธิภาพที่ค่อนข้างน้อยกว่าแบบจำลองอื่น ดังรูปที่ 4 และสามารถแสดงองค์ประกอบของฟีเจอร์ในแต่ละคอมโพแนนท์ ได้ตามรูปที่ 5 โดยในแต่ละองค์ประกอบของฟีเจอร์ในแต่ละคอมโพแนนท์ ฟีเจอร์ที่มีค่าเป็นบวก (สีเหลือง) แปลว่าเป็นฟีเจอร์ที่สำคัญมาก โดยให้ผลเชิงบวกต่อคอมโพแนนท์นั้นๆ มาก และฟีเจอร์ที่ให้ค่าเป็นลบ (สีน้ำเงินเข้ม) แปลว่าเป็นฟีเจอร์ที่สำคัญมากโดยให้ผลเชิงลบต่อคอมโพแนนท์นั้นๆ มาก เช่น ฟีเจอร์ “BRU” นั้นให้ผลเชิงบวกในคอมโพแนนท์ที่ 1 ของทุกการใช้ n_components ส่วนฟีเจอร์ “SSH” ให้ผลเชิงลบในคอมโพแนนท์ที่ 1 ของทุกการใช้ n_components นอกจากนี้ในคอมโพแนนท์อื่นๆ ยังพบฟีเจอร์ที่ให้ผลเชิงบวกได้แก่ “CSU” “GCO” “GSP” และ “SSH” เป็นต้น และพบฟีเจอร์ที่ให้ผลเชิงลบ ได้แก่ “CCO” “SSH” “CSU” และ “SSB” เป็นต้น

3.4 การพัฒนาต้นแบบเว็บแอปพลิเคชัน

จากผลการศึกษาของการคัดเลือกฟีเจอร์ เมื่อพิจารณาจากทั้งประสิทธิภาพของแบบจำลอง, เทคนิคการคัดเลือก/การสกัดฟีเจอร์ Training size ที่ใช้ และจำนวนฟีเจอร์ พบว่าแบบจำลอง Decision tree ที่ใช้เทคนิค RFE Training size เท่ากับ 60% และใช้จำนวนฟีเจอร์เท่ากับ 3 ฟีเจอร์ ซึ่งสามารถแสดงชื่อฟีเจอร์คือ กลิ่นของเห็ด (ODO), ขนาดครีบเห็ด (GSI) และสีรอยพิมพ์สปอร์ (SPC) มีความเหมาะสมที่สุด (F1 score = 99.32%) โดยสามารถแสดงต้นแบบเว็บแอปพลิเคชันได้ตาม Figure 6

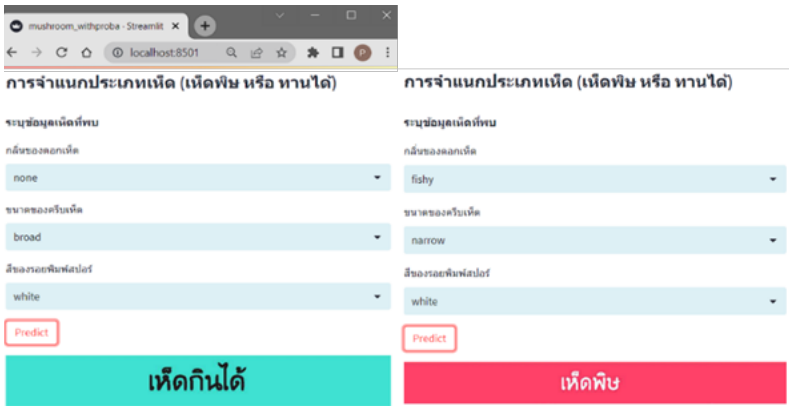

Figure 6 Example of prototype of web application

3.5 วิจารณ์ผลการวิจัย

จากผลการศึกษาการจำแนกประเภทของเห็ดข้างต้น พบว่าการศึกษาที่ให้ประสิทธิภาพที่เหมาะสมที่สุด คือ แบบจำลอง Decision tree ร่วมกับการคัดเลือกฟีเจอร์ด้วยเทคนิค RFE ที่ใช้ Training size เท่ากับ 60% และใช้ฟีเจอร์จำนวน 3 ฟีเจอร์คือ กลิ่นของเห็ด (ODO) ขนาดครีบเห็ด (GSI) และสีรอยพิมพ์สปอร์ (SPC) ซึ่งให้ค่า F1 score = 99.32% ซึ่งใกล้เคียงและสอดคล้องกับงานวิจัยของ Vanitha และคณะ [7] ที่มี F1 score = 99% แต่ใช้ฟีเจอร์เพียงแค่ 2 ฟีเจอร์ คือ “odor” (กลิ่นของเห็ด) และ “spore_print_color” (สีรอยพิมพ์สปอร์) ซึ่งอาจจะเกิดจากการความแตกต่างของการปรับจูน Hyperparameters และการทำความสะอาดข้อมูลที่แตกต่างกัน

4. สรุป

จากผลการศึกษาการจำแนกประเภทของเห็ดระหว่างเห็ดพิษและเห็ดที่รับประทานได้ โดยใช้ชุดข้อมูลสาธารณะของเห็ดครีบจากหนังสือข้อมูลเห็ดในฝั่งอเมริกาเหนือ ปี 1981 [8] ค้นพบว่าการใช้เทคนิคการคัดเลือกฟีเจอร์ เพื่อค้นหาฟีเจอร์ที่สำคัญ จะช่วยลดการใช้ฟีเจอร์ที่ไม่จำเป็นต่อการจำแนกประเภทของเห็ด โดยที่แบบจำลองยังคงมีประสิทธิภาพดี อีกทั้งสามารถสรุปได้ว่า เทคนิคการเรียนรู้ของเครื่องสามารถจำแนกประเภทของเห็ด ได้เทียบเท่ามนุษย์เป็นอย่างน้อย

อย่างไรก็ตามการวิจัยนี้ใช้ข้อมูลของเห็ดจากแหล่งข้อมูลดั้งเดิมคือเป็นเห็ดที่พบในฝั่งอเมริกาเหนือ ซึ่งอาจแตกต่างกับเห็ดภายในประเทศไทย ดังนั้นการศึกษาครั้งถัดไปจึงควรมีการเก็บข้อมูลของเห็ดที่พบภายในประเทศไทยเพิ่มเติม และนำมาเปรียบเทียบผลลัพธ์ และควรมีการศึกษาแบบจำลองอื่นเพิ่มเติม ได้แก่ K-nearest neighbors Naïve Bayes Voting-Classifier หรือ AutoSklearnClassifier (แบบจำลองที่มีการปรับจูนหาแบบจำลองและ hyperparameters ที่เหมาะสมอย่างอัตโนมัติ) หรือนอกจากนี้อาจนำรูปภาพมาใช้ในการประมวลผลภาพ (Image processing) เพื่อใช้ในการจำแนกประเภทเห็ด ร่วมกับการใช้แบบจำลองโครงข่ายประสาทเทียม (Neural network) หรือเทคนิคการเรียนรู้เชิงลึก (Deep learning techniques)

5. กิตติกรรมประกาศ

ขอขอบคุณคณาจารย์ทุกท่านในภาควิชาวิทยาการข้อมูล มหาวิทยาลัยศรีนครินทรวิโรฒ ที่แนะนำความรู้และแนวทางในการทำงานวิจัยเรื่องนี้ และขอขอบคุณบัณฑิตวิทยาลัย มหาวิทยาลัยศรีนครินทรวิโรฒ ที่สนับสนุนการเผยแพร่การวิจัยนี้

6. References

[1] Sakonrak, B., Duangkae, K., Ayawong, J., Pongpanich, K. and Khemaman, W., 2006, Gilled mushrooms: kaeng krachan forest complex chiang dao wildlife sanctuary and phu khieo wildlife sanctuary, Forest conservation research division forest and plant conservation research office department of national parks, Wildlife and plant conservation, Bangkok, 240 p. (in Thai)

[2] Boonpratuang, T., Choeyklin, R., Promkiam-on, P. and Tiyaphan, P., 2012, The identification of poisonous mushrooms in Thailand during 2008-2012, Thai mushroom 2555, pp. 6-13. (in Thai)

[3] Nooron, N., Parnmen, S., Sikaphan, S., Leudang, S., Uttawichai, C. and Polputpisatkul, D., 2020, The situation of mushrooms food poisoning in Thailand: Symptoms and common species list, Thai J Toxicol. 35(2): 58–69.

[4] Islam, M.D., 2015, Cultivation techniques of edible mushrooms: Agaricus bisporus, Pleurotus spp., Lentinula edodes and Volvariella volvocea, Available source: https://www.researchgate.net/publication/ 275179411_Cultivation_Techniques_of_Edible_Mushrooms_Agaricus_bisporus_Pleurotus_spp_Lentinula_edodes_and_Volvariella_volvocea, Sep 22, 2022.

[5] Verma, S. and Dutta, M., 2018, Mushroom classification using ANN and ANFIS algorithm, (IOSRJEN), 8(1): 26-32.

[6] Chitayae, N. and Sunyoto, A., 2020, Performance comparison of mushroom types classification using k-nearest neighbor method and decision tree method, In 2020 3rd international conference on information and communications technology (ICOIACT), pp. 308-313. IEEE.

[7] Vanitha, V., Ahil, M.N. and Rajathi N., 2020, Classification of mushrooms to detect their edibility based on key attributes, Biosci. Biotech. Res. Asia., 13(11): 37-41.

[8] Schlimmer, J., Mushroom data set, machine learning repository, Available source: https://archive.ics.uci.edu/ml/datasets/Mushroom, Sep 22, 2022.

[9] Agarwal, D., 2021, Introduction to SVM (Support vector machine) along with python code, Data science blogathon, Available source: https://www.analyticsvidhya.com/blog/2021/04/insight-into-svm-support-vector-machine-along-with-code/, Oct 12, 2022.

[10] Saini, A., 2021, Decision tree algorithm – A complete guide, Data science blogathon, Available source: https://www.analyticsvidhya.com/blog/2021/08/decision-tree-algorithm/, Oct 12, 2022

[11] What is a random forest?, TIBCO software inc, Available source: https://www.tibco.com/reference-center/what-is-a-random-forest, Oct 12, 2022.

[12] Mariajesusbigml. Introduction to boosted trees, Available source: https://blog.bigml.com/2017/03/14/ introduction-to-boosted-trees/, Oct 12, 2022.

[13] Brownlee, J., How to choose a feature selection method for machine learning, Machine learning mastery, Available source: https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/, Oct 12, 2022

[14] Hira, Z.M. and Gillies, D.F., 2015, A review of feature selection and feature extraction methods applied on microarray data, Adv. Bioinform., 2015: 1-13.

[15] Sagrawal, S.K., Metrics to evaluate your classification model to take the right decisions, Analytics vidhya, Available source: https://www.analyticsvidhya.com/blog/2021/07/metrics-to-evaluate-your-classification-model-to-take-the-right-decisions/, Oct 12, 2022.

[16] Streamlit documentation, Streamlit inc, Available source: https://docs.streamlit.io/, Feb 15, 2023.

พิมพ์ที่: โรงพิมพ์มหาวิทยาลัยธรรมศาสตร์, พ.ศ. 2568
โทรศัพท์ 0-2564-3104 ถึง 6
โทรสาร 0-2564-3119
http://www.thammasatprintinghouse.com