WisdomModel: convert data into wisdom

Israa Mahmood and Hasanen Abdullah
Department of Computer Science, University of Technology, Baghdad, Iraq

Received 12 June 2021
Revised 13 July 2021
7 August 2021
Accepted 9 August 2021

Abstract

Purpose – Traditional classification algorithms always have an incorrect prediction. As the misclassification rate increases, the usefulness of the learning model decreases. This paper presents the development of a wisdom framework that reduces the error rate to less than 3% without human intervention.

Design/methodology/approach – The proposed WisdomModel consists of four stages: build a classifier, isolate the misclassified instances, construct an automated knowledge base for the misclassified instances and rectify incorrect prediction. This approach will identify misclassified instances by comparing them against the knowledge base. If an instance is close to a rule in the knowledge base by a certain threshold, then this instance is considered misclassified.

Findings – The authors have evaluated the WisdomModel using different measures such as accuracy, recall, precision, f-measure, receiver operating characteristics (ROC) curve, area under the curve (AUC) and error rate with various data sets to prove its ability to generalize without human involvement. The results of the proposed model minimize the number of misclassified instances by at least 70% and increase the accuracy of the model minimally by 7%.

Originality/value – This research focuses on defining wisdom in practical applications. Despite of the development in information system, there is still no framework or algorithm that can be used to extract wisdom from data. This research will build a general wisdom framework that can be used in any domain to reach wisdom.

Keywords Artificial neural network, Wisdom, DIKW, Knowledge base

Paper type Research paper

1. Introduction

The misclassification rate is a fundamental issue in many domains, where wrong predictions are costly. In real life, a high error rate in applications such as churn prediction [1], fraud detection [2], flood forecasting [3] and medical diagnosis [4] can lead to a costly outcome. Classification algorithms are often used to convert the information into knowledge that can be used by specialists to facilitate their work. This knowledge is represented as models. A model with high-performance metrics means that the classification algorithm was successfully able to extract knowledge and distinguish between different patterns of the data [5]. However, the extracted knowledge may not always provide correct predictions.

In some domains, models that have incorrect predictions even if the model accuracy is high (e.g. >80%) may not satisfy specialists due to the great impact of misclassification. Nevertheless, eliminating incorrect predictions is not possible in traditional classification algorithms unless a domain specialist can examine all the data to identify misclassified samples, which would make the model pointless.

Applied Computing and Informatics
Vol. 21 No. 1/2, 2025
pp. 131-140
Emerald Publishing Limited
e-ISSN: 2210-8327
p-ISSN: 2634-1964
DOI 10.1108/ACI-06-2021-0155

The main problem is that traditional classification algorithms will always have incorrect predictions. Eliminating misclassification error means the use of human intervention to cross-check the result of the classifier and identify the misclassified instances, which eliminates the need for machine learning algorithms. The goal of this study is to build a *WisdomModel* that achieves a minimum error rate (less than 3%) without human intervention. Our approach identifies misclassified instances and corrects them. In contrast to classical classification algorithms that have overfitting issues when the accuracy is too high, the *WisdomModel* is unlikely to face such an issue. Moreover, the *WisdomModel* is designed for binary classification.

In this research, we develop a *WisdomModel* that uses the knowledge gained from traditional classification algorithms to make a better judgment. The model builds a knowledge base (KB) for misclassified instances of the trained data. The KB is used by the *WisdomModel* to find data with incorrect predictions.

The main contributions of this study are illustrated below:

(1) *WisdomModel* is a general framework that converts the knowledge gained from machine learning models into wisdom, which can be used to make a good decision regardless of the data domain and without human intervention.

(2) *WisdomModel* minimizes the error rate to less than 3%.

(3) *WisdomModel* can be applied to formerly developed models.

This research is organized as follows: Section 2 presents the related work. Section 3 describes the architecture of the *WisdomModel*. Section 4 presents the experiment and results of the proposed system, and Section 5 provides the conclusion and the direction of future work.

2. Related work

In recent years, there has been a trend toward machine learning algorithms in different fields such as medical diagnosis [6, 7], earthquake prediction [8], churn prediction [9], voice recognition [10], scientific research [11, 12] and risk estimation [13]. The purpose of using these algorithms is to avoid human interference and minimize the error rate. Many studies have aimed to eliminate the error rate by refusing to predict doubtful instances [14]. Instead, these methods deferred the decision of these suspicious instances to specialists who have to examine the data manually.

Alpaydin [15] introduced several ideas regarding the doubt region in classification problems. Alpaydin defined the doubt region by the data that is covered by the general hypothesis and not included in the specific hypothesis. The general hypothesis contains positive instances and includes as many features without covering negative instances. The specific hypothesis contains only positive instances and includes as small feature extent as possible.

Amin *et al.* [16] developed a prudent churn prediction model that will generate an alert to the decision-makers whenever there is a new case that is not covered by the KB system. The developed approach uses ripple down rule (RDR) classifier to build the KB.

Tran and Aygun [17] proposed a WisdomNet model that adds an extra neuron to the output layer of the neural network. The additional neuron is used to recognize the rejected data. The trustable learning model refuses to decide for suspicious instances. The WisdomNet model minimizes the error rate to 0% after deferring more than 10% of the data to the specialists.

Although these models achieve zero misclassification error, the rejected rate may reach more than 10%. As the rejection rate increases, the cost associated with scanning the data increases. When dealing with big-data applications, a 10% rejection rate means millions of instances that need to be scanned manually by the experts. Such a process is unfeasible and abolishes the need for classification algorithms. Our approach eliminates the need for human experts and focuses on achieving a minimum error rate.

In the literature, there is very little research regarding converting data into wisdom in practical application. However, there are many discussions about the definitions of wisdom in information systems. Jankowski and Skowron [18] describe wisdom as the capability to recognize critical issues and solve them using the cumulated knowledge and experience to achieve its target. Wognin *et al.* [19] define wisdom as the capability to utilize knowledge to act correctly in a certain context to achieve its goal. Liew [20] lists wisdom dimensions: perspicacity, the quick use of information, judgment, learning from mistakes, sagacity and the ability to reason. Similarly, Ermine [21] divides the concept of wisdom into two types: individual wisdom, which means the ability to use skills and knowledge to enhance performance, and organizational wisdom that can be defined as the capability to perform actions. Van Meter [22] states several characteristics of wisdom such as wisdom does not mean excellence or infallibility, it can be constructed by grouping information, knowledge and experience, and it is used to improve the environment.

The *WisdomModel* presented in this paper is a general framework that can be used in any field to convert data into wisdom. The proposed model aims to minimize the error rate without human intervention by building an automated KB for misclassified instances and using it to reduce the error rate. Wisdom as it relates to machine learning is defined as the ability to reduce the misclassification rate without increasing the cost.

3. Architecture of WisdomModel

In this section, the architecture of data, information, knowledge and wisdom (DIKW) pyramid is discussed in Section 3.1. Datasets used in this research are presented in Section 3.2. The architecture of the *WisdomModel* is illustrated in Section 3.3.

3.1 DIKW model

The DIKW hierarchy is very popular in information and knowledge systems. DIKW is often represented as a pyramid, with data as the foundation and wisdom at the top of the pyramid [23]. This representation means that wisdom cannot be reached without processing the lowest elements of the pyramid. In this paper, our definitions for each element in the DIKW pyramid will be discussed.

(1) Data is the untreated material that is gathered by sensors or humans.

(2) Information is the interpretation of data that makes it easier to present and evaluate.

(3) Knowledge is the ability to distinguish between various patterns of information.

(4) Wisdom is the capability to use knowledge, experience and fine judgment to achieve the required goals

3.2 Datasets

In this study, we tested our approach on multiple datasets in different areas such as telecommunication, financial services and the medical field. The description of the datasets is listed below:

(1) Churn prediction dataset that belongs to a telecom operator.

(2) Credit card fraud detection dataset can be found in Kaggle [24]. However, we noticed that the dataset is imbalanced, and we used the python SMOTE technique to balance it.

(3) Lung cancer prediction dataset can be found in Kaggle [25].

(4) Diabetes dataset can be found in Kaggle [26].

We used 80% of the data for training and 20% for testing. The number of instances used in training and testing is illustrated in Table 1.

3.3 WisdomModel

The *WisdomModel* is a general structure that is used to convert data into wisdom. Its name is inspired by Benjamin Franklin's quote, "The doorstep to the temple of wisdom is a knowledge of our own ignorance." Intelligent systems should be able to identify the ignorance regions of the data and deal with them in a wise way [27]. The ignorance region of the data is when the classifier fails to predict data correctly. The proposed *WisdomModel* is used to find the weakness of the classifier by recognizing and analyzing the misclassified instances and correcting them. The model consists of four stages: (1) build a classifier using the training dataset, (2) isolate misclassified instances, (3) construct an automated KB and (4) rectify incorrect predictions. Our approach is designed to work on binary classification issues for structured datasets.

3.3.1 Build a classifier using the training dataset. Consider building a Multilayer Perceptron (MLP) deep learning model D with M hidden layers to solve binary classification issues. A deep learning model can be represented as

$$D = D\,(nn_0,\, nn_1,\, nn_2,\, nn_3,\, \ldots,\, nn_M,\, 1)$$

where nn_0 represents the number of neurons at the first hidden layer, nn_1 represents the number of neurons at the second hidden layer and so on. We will train the deep learning model D using the training dataset. The architectures of the deep learning models used in churn prediction, credit card fraud detection, lung cancer prediction and diabetes detection are expressed in D1, D2, D3 and D4, respectively.

Table 1.
Number of instances used in training and testing the *WisdomModel*

Dataset name	Number of attributes	Training dataset size	Testing dataset size
Telecom churn prediction	666	2,108,606	527,151
Credit card fraud detection	29	30,005	12,669
Lung cancer prediction	23	534	134
Diabetes detection	8	1600	400

D1 = D1(64, 64, 64, 64, 64, 64, 64, 64, 128, 128, 256, 1)

D2 = D2(3, 4, 2, 4, 5, 6, 1)

D3 = D3(2, 3, 6, 1)

D4 = D4(6, 4, 2, 1)

3.3.2 Isolate misclassified instances. The trained deep learning model *D* is used to classify the training dataset. We obtain the model prediction and compare it against the actual output to identify the incorrect prediction. The misclassified instances are isolated in a new dataset called *R*.

3.3.3 Construct an automated knowledge base. The *R* dataset is used to build the KB of misclassified instances. Choosing to build a KB for incorrect predictions instead of correct predictions should make the *WisdomModel* faster. We developed an algorithm to generate the KB in an automated manner without human intervention. The algorithm obtains rules that describe the general characteristics of misclassified data. The steps of building the KB are illustrated in Algorithm 1.

Algorithm 1: Automated Knowledge Base Generation algorithm

Input:
R: dataset of misclassified instances with Y label
T: threshold value for knowledge base
Output: class knowledge base, non-class knowledge base
Split R according to Y class (e.g. churn and non-churn) into different datasets
For each class-dataset do
While (criteria not satisfied)
For each attribute in class-dataset
Get the unique values of the feature.
If the weight of the value >= T
Add feature to the common-feature-list
End for
$FC_{dataset}$ =Generate new dataset with common-feature-list
Prune $FC_{dataset}$
Generate Rules from $FC_{dataset}$
Update T
End While
Save Rules as knowledge base for class label
End For
Return class knowledge base, non-class knowledge base

3.3.4 Rectify incorrect prediction. The *WisdomModel* will use the knowledge and experience that was gained from the previous stages to provide good judgment. The KB and the deep learning model prediction are used in this stage to identify incorrect predictions in the test dataset and rectify them. Our approach will first use the *D* model to get the predictions of test dataset and then identify misclassified instances by comparing them against the KB. If an instance is close to a rule in the KB by a certain threshold, then this instance is considered misclassified. The process is summarized in Algorithm 2.

Algorithm 2: Rectify incorrect prediction algorithm

```
Input:
    T_dataset: Test dataset
    C_KB: class knowledge base (e.g. churn knowledge base)
    NC_KB: non-class knowledge base (e.g. non-churn knowledge base)
    D: deep learning model
    T_distance: distance threshold
Output: WisdomModel Prediction
 Classify T_dataset using D model
 For each instance
  If model prediction = 0 then
     C_Flag = check existence in C_KB
     If C_Flag = True then
        Set WisdomModel prediction to 1
     Else
        Check distance to C_KB
        If distance < T_distance then
           Set WisdomModel prediction to 1
        Else
           Set WisdomModel prediction to 0
  Else if model prediction = 1 then
     NC_Flag = check existence in NC_KB
     If NC_Flag = True then
        Set WisdomModel prediction to 0
     Else
        Check distance to NC_KB
        If distance < T_distance then
           Set WisdomModel prediction to 0
        Else
           Set WisdomModel prediction to 1
End For
Return WisdomModel prediction
```

4. Results and discussion

The performance of the *WisdomModel* is evaluated against various datasets using diverse measures. Different architectures of the neural network were used to demonstrate the ability of the *WisdomModel* to generalize. We developed our model in Python3. The performance measures used in this paper are provided in Section 4.1. The results of the *WisdomModel* are discussed in Section 4.2.

4.1 Evaluation measures

Various measures are used to evaluate the performance of traditional classification algorithms such as accuracy, precision, recall, F-measure, receiver operating characteristic (ROC) curve and area under the ROC curve (AUC) [28, 29]. These measures are derived from the confusion matrix that is used to depict the performance of the classifier. The confusion matrix consists of four items: true positive (TP), true negative (TN), false positive (FP) and false negative (FN) [30, 31]. The definition of each measure is illustrated in Table 2.

4.2 Analysis and discussion of WisdomModel

We first tested the performance of the *WisdomModel* against a big data telecom dataset that consists of millions of records. The deep learning model D1 was used to train the classifier and was able to achieve a 90.2% accuracy rate on the test dataset, although a 10% error rate is considered acceptable in many domains. But when dealing with big data such as our churn dataset, the 10% means that more than 260,000 subscribers were misclassified. This very large number of misclassified instances will highly affect the performance and the revenue of the company. Table 3 shows the results of the *WisdomModel* and deep learning model. The

Measure	Definition	Mathematical definition
Accuracy	Measures the rate of correctly classified data	$\frac{TP+TN}{TP+TN+FP+FN}$
Precision	Measures the number of TP compared to the predicted positive classes (TP + FP)	$\frac{TP}{TP+FP}$
Recall	Measures the number of TP compared to the actual positive classes (TP + FN)	$\frac{TP}{TP+FN}$
F-measure	Combines precision and recall measures into one measure in a consistent mean	$2 \times \frac{precision \times recall}{precision+recall}$
ROC CURVE	A great tool to depict the performance of the classifier. It shows the trade-off between true positive rate (TPR) and false positive rate (FPR)	–
AUC	Measures the ability of the classifier to separate between different classes in the dataset	–
Error rate	Measures the number of misclassified instances in the dataset	$\frac{FP+FN}{TP+TN+FP+FN}$

Table 2. Evaluation measures definition

Measures	Telecom churn prediction dataset		Credit card fraud detection dataset	
	Deep learning	WisdomModel	Deep learning	WisdomModel
Accuracy %	90.22	97.00	84.16	97.26
Precision %	90.86	97.19	86.16	97.29
Recall %	89.35	96.72	83.94	97.29
F-measure %	89.87	96.90	83.88	97.26

Table 3. Performance measures of deep learning model versus *WisdomModel* for different datasets

ROC curve of the *WisdomModel* is closer to the top left corner than the deep learning model, which is shown in Figure 1. We also measured the ability of the *WisdomModel* to separate between classes using an AUC measure. The *WisdomModel* surpassed the deep learning model with 0.97 AUC, whereas deep learning attained 0.89. Moreover, the error rate was reduced to 3%, and the number of misclassified instances was reduced by 70%.

We also applied the framework to other datasets to demonstrate the ability of the *WisdomModel* to generalize. We used a deep learning model D2 to predict credit card fraud.

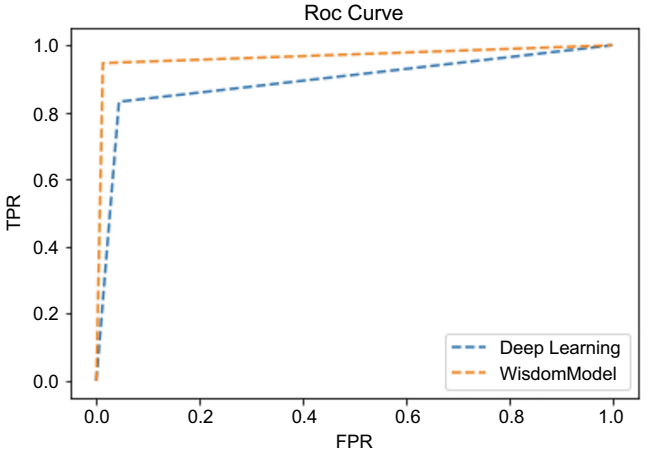

Figure 1. ROC curve of *WisdomModel* versus deep learning model for the telecom churn prediction dataset

It achieved an 84.16% accuracy rate. After applying the *WisdomModel*, the accuracy increased by 13.1% and attained 97.26%. The performance measures for the deep learning model and the *WisdomModel* for the credit card fraud detection dataset are illustrated in Table 3.

Furthermore, *WisdomModel* obtained 0.97 AUC while the deep learning model achieved 0.89 AUC. The error rate was reduced from 15.84% to 2.74%, and the number of misclassified instances was reduced by 82%.

Later on, we applied the model to two medical datasets. The results of the lung cancer prediction dataset showed that it is possible to reduce the error rate to 0% using the *WisdomModel* framework, as depicted in Figure 2(a).

Additionally, the results of the diabetes dataset revealed that the *WisdomModel* could increase the accuracy by 20% and attained a 98.51% accuracy rate compared to the deep learning model D4, which achieved a 78.33% accuracy rate, as presented in Figure 2(b). The error rate was reduced from 21.67% to 1.49%. Additionally, the *WisdomModel* attained 0.982 AUC compared to the deep learning model, which obtained 0.841.

5. Conclusion

Classical machine learning algorithms are always negatively affected by the incorrect predictions. Often error rate means that there are certain patterns in the data that the model is unable to classify. The benefit of applying learning models decreases when the misclassification rate increases. This research aims to develop a general framework that can convert data into wisdom in practical application. The *WisdomModel* uses the knowledge and experience gathered from the deep learning model and KB to identify misclassified instances and rectify them without human intervention. The proposed model was able to reduce the error rate to less than 3% in different domains. It is also possible to reach a zero error rate if the model continues to update the KB. Our approach can be applied to any classification algorithm and is not limited to neural networks. However, the *WisdomModel* cannot work on unstructured datasets such as images and videos. The future work is related to developing a wisdom model that can work on multinomial classification issues.

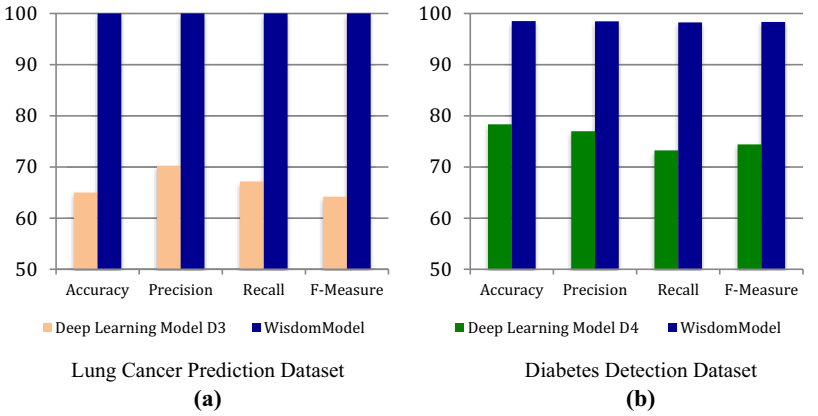

Figure 2.
Performance measures of deep learning model versus *WisdomModel*

References

1. Idris A, Iftikhar A, Ur Rehman Z, Intelligent churn prediction for telecom using GP-AdaBoost learning and PSO undersampling, Cluster Comput. 2019; 22: 7241-55. doi: 10.1007/s10586-017-1154-3.
2. Goy G, Gezer C, Gungor VC, Credit card fraud detection with machine learning methods, UBMK 2019 - 4th international conference on computer science and application engineering: 2019. 350-54. doi: 10.1109/UBMK.2019.8906995.
3. Mitra P, Ray R, Chatterjee R, Basu R, Saha P, Raha S, Barman R, Patra S, Biswas SS, Saha S, Flood forecasting using Internet of things and artificial neural networks, 7th IEEE annual information technology, electronics and mobile communication conference IEEE IEMCON 2016: 2016. doi: 10.1109/IEMCON.2016.7746363.
4. Giger ML, Machine learning in medical imaging, J Am Coll Radiol. 2018; 15: 12-520. doi: 10.1016/j.jacr.2017.12.028.
5. Syeda Farha Shazmeen SFS. Performance evaluation of different data mining classification algorithm and predictive analysis, IOSR J Comput Eng. 2013; 10: 1-6. doi: 10.9790/0661-1060106.
6. Salehi AW, Baglat P, Gupta G, Alzheimer's disease diagnosis using deep learning techniques, Int J Eng Adv Technol. 2020; 9: 874-80. doi: 10.35940/ijeat.c5345.029320.
7. Ojha U, Goel S, A study on prediction of breast cancer recurrence using data mining techniques, Proceedings of the 7th international conference confluence 2017 on cloud computing, data science and engineering: 2017. 527-30. doi: 10.1109/CONFLUENCE.2017.7943207.
8. Asim KM, Martínez-Álvarez F, Basit A, Iqbal T, Earthquake magnitude prediction in Hindukush region using machine learning techniques, Nat Hazards. 2017; 85: 471-86. doi: 10.1007/s11069-016-2579-3.
9. Ahmad AK, Jafar A, Aljoumaa K, Customer churn prediction in telecom using machine learning in big data platform, J Big Data. 2019; 6. doi: 10.1186/s40537-019-0191-6.
10. Eljawad L, Aljamaeen R, Alsmadi MK, Al-Marashdeh I, Abouelmagd H, Alsmadi S, Haddad F, Alkhasawneh RA, Alzughoul M, Alazzam M.B, Arabic voice recognition using fuzzy logic and neural network, Int J Appl Eng Res. 2019; 14: 651-62.
11. Jiao C, Xu Z, Bian Q, Forsberg E, Tan Q, Peng X, He S, Machine learning classification of origins and varieties of Tetrastigma hemsleyanum using a dual-mode microscopic hyperspectral imager, Spec Acta A Mol Bio Spect. 2021; 261: 120054. doi: 10.1016/j.saa.2021.120054.
12. Xu Z, Jiang Y, Ji J, Forsberg E, Li Y, He S, Classification, identification, and growth stage estimation of microalgae based on transmission hyperspectral microscopic imaging and machine learning, Opt Exp. 2020; 28: 30686-700. doi: 10.1364/OE.406036.
13. Kruppa J, Ziegler A, König IR, Risk estimation and risk prediction using machine-learning methods, Hum Genet. 2012; 131: 1639-54. doi: 10.1007/s00439-012-1194-y.
14. Tran TX, Pusey ML, Aygun RS, Else-tree classifier for minimizing misclassification of biological data, 2018 IEEE international conference on bioinformatics and biomedicine: 2018. 2301-8. doi: 10.1109/BIBM.2018.8621322.
15. Alpaydin E, Introduction to machine learning, MIT Press, 2014.
16. Amin A, Rahim F, Ramzan M, Anwar S, Prudent based approach for customer churn prediction, Commun Comput Inf Sci. 2015; 521: 320-32. doi: 10.1007/978-3-319-18422-7_29.
17. Tran TX, R.S. Aygun, WisdomNet: trustable machine learning toward error-free classification, Neural Comput Appl. 2021; 33: 2719-34, doi: 10.1007/s00521-020-05147-4.
18. Jankowski A, Skowron A, A wistech paradigm for intelligent systems BT - transactions on rough sets VI: commemorating the life and work of zdzisław pawlak, Part I, Peters JF, Skowron A, Düntsch I, Grzymała-Busse J, Orłowska E, Polkowski L (Eds), Berlin, Heidelberg; Springer Berlin Heidelberg: 2007. 94-132. doi: 10.1007/978-3-540-71200-8_7.
19. Wognin R, Henri F, Marino O, Data, information, knowledge, wisdom: a revised model for agents-based knowledge management systems BT - the next generation of distance education:

unconstrained learning, in: Moller L, Huett JB (Eds), Boston, MA; Springer US: 2012. 181-9. doi: 10.1007/978-1-4614-1785-9_12.

20. Liew A, DIKIW: data, information, knowledge, intelligence, wisdom and their interrelationships, Bus Manag Dyn. 2013; 2: 49-62.
21. Ermine JL, A knowledge value chain for knowledge management, J Knowl Commun Manag. 2013; 3: 85. doi: 10.5958/j.2277-7946.3.2.008.
22. Van Meter HJ, Revising the DIKW pyramid and the real relationship between data, information, knowledge and wisdom, law, Technol Humans. 2020; 2: 69-80. doi: 10.5204/lthj.1470.
23. Ackoff RL, From data to wisdom, J Appl Syst Anal. 1989; 16: 3-9.
24. Maji A, Kaggle credit card fraud detection. n.d. Available from: https://www.kaggle.com/adhyanmaji31/credit-card-fraud-detection (accessed February 10 2021).
25. Maharana D, Kaggle lung cancer dataset. n.d. Available from: https://www.kaggle.com/divyanimaharana/lung-cancer-dataset (accessed May 1 2021).
26. Ukani V, Diabetes dataset. n.d. Available from: https://www.kaggle.com/vikasukani/diabetes-data-set (accessed June 1 2021).
27. Kim TW, Mejia S, From artificial intelligence to artificial wisdom: what socrates teaches us, Computer (Long Beach Calif). 2019; 52: 70-4. doi: 10.1109/MC.2019.2929723.
28. Fatourechi M, Ward RK, Mason SG, Huggins J, Schlögl A, Birch GE, Comparison of evaluation metrics in classification applications with imbalanced datasets, 2008 seventh international conference on machine learning and applications; IEEE: 2008. 777-82.
29. Sigdel M, Aygün RS, Pacc-a discriminative and accuracy correlated measure for assessment of classification results, International Workshop on Machine Learning and Data Mining in Pattern Recognition; Springer: 2013. 281-95.
30. Han J, Kamber M, Pei J, Data mining concepts and techniques third edition, Morgan Kaufmann Ser. Data Manag Syst. 2011; 5: 83-124.
31. Sahli H, An introduction to machine learning, TORUS 1 – towar. An open resour. Using Serv. 2020: 61-74. doi: 10.1002/9781119720492.ch7.

Corresponding author
Israa Mahmood can be contacted at: asraa_nafaa@yahoo.com