

DEVELOPMENT OF A DECISION SUPPORT SYSTEM FOR LEGAL CASE ANALYSIS AND DECISION MAKING USING RECURRENT NEURAL NETWORK

Sobowale, A. A., Omodunbi, B. A., Abdul-Hameed, T. A., Sobowale, P. O., Amuzat, B. J.

Abstract Over the last few decades, there have been substantial developments in a variety of domains, including computer science, artificial intelligence, and machine learning, which has accelerated the evolution of intelligent systems such as decision support systems with applications in fields such as healthcare, finance, manufacturing, transportation and legal systems. Decision Support Systems (DSS) in legal systems also known as Judicial Decision Support Systems (JDSS) are designed to assist judges and other legal professionals in making legal decisions. JDSS can be used in various areas of law, such as criminal law, civil law, and family law. JDSS can be helpful in various ways, such as reducing the time and effort required to make legal decisions, increasing consistency and fairness in legal decisions, and providing judges and legal professionals with access to relevant legal information and expertise. However, JDSS also have limitations and challenges, such as the need for accurate and up-to-date legal data and the potential for bias in the data or algorithms used in the system. This research work developed a system to assist legal practitioners and it is also designed to enhance the efficiency, transparency, and accessibility of the judicial process. The performance of the developed system was evaluated using some metrics and the following results were obtained: accuracy of 80%, 87% precision, 87% sensitivity/recall and an F1-score of 87%.

Keywords: Legal system, machine learning, information system, Decision Support system

I. INTRODUCTION

An information system (IS) is a set of interconnected components that gather, process, store, and disseminate data and information in an organized and efficient manner to support decision making, coordination, control, analysis, and visualization of organizational activities and processes. Information systems can be classified according to their scope, purpose, architecture, and technology. They can range from simple manual systems, such as a file cabinet, to complex computerized systems, such as an enterprise resource planning (ERP) system.

Information systems are essential for organizations to compete and survive in today's rapidly changing business environment. They provide a means for organizations to improve their operations, increase their efficiency, reduce costs, and enhance their customer service. Moreover, information systems can facilitate innovation, collaboration, and knowledge management within organizations, which can lead to competitive advantage and value creation [1].

Judicial Decision Support System (JDSS) is a type of Decision Support System (DSS) that are specifically designed to assist judges and other legal professionals in making decisions in the context of the court of law. JDSS can assist with a range of tasks, such as case management, legal research, and decision-making [2]. JDSS uses

Sobowale, A. A., Omodunbi, B. A., Sobowale, P. O.,
Amuzat, B. J.

*(Department of Computer Engineering, Federal
University, Oye-Ekiti, Nigeria)*

Abdul-Hameed, T. A.

*(Department of Electrical and Electronic
Engineering, Federal Polytechnic, Ayede, Nigeria)*

Corresponding Author: Omodunbi B.A.

Email: bolaji.omodunbi@fuoye.edu.ng

Phone Number: 08033836990

artificial intelligence (AI) and machine learning (ML) techniques including natural language processing, semantic analysis, machine learning algorithms, and decision trees to analyze legal data, including case law, statutes, regulations, and other legal documents which then provides relevant insights and help legal professionals make decisions based on legal precedent and other relevant factors. Some JDSS also incorporate predictive analytics, which can help judges and lawyers forecast the outcome of a legal case based on similar cases in the past [3].

There are various judicial decision support systems in developing and developed countries across the globe, some of these judicial decision support systems are reviewed in this section expounding their strengths and weaknesses and their relevance to this research work.

AfricanLII is a JDSS that provides free access to legal information from African countries, including case law, legislation, and other legal materials. However, one of its limitations is that it relies on the availability and quality of legal information provided by individual countries, which can be inconsistent and incomplete [4]. Furthermore, [5] proposed a model used by federal judges in the United States to determine the appropriate sentence for criminal defendants. The Guidelines are designed to provide consistency and fairness in sentencing by considering factors such as the nature and severity of the offense, the offender's criminal history, and other relevant factors. The limitations to this work are the inflexibility as it do not allow judges to take into account individual circumstances, such as the defendant's rehabilitation efforts or mitigating factors that may justify a sentence that deviates from the Guidelines and for exacerbating existing racial

disparities in the criminal justice system, as they tend to result in harsher sentences for minorities than for whites convicted of similar offenses.

CARS is a JDSS used by legal practitioners in Singapore to search for and analyze case law. However, CARS has been criticized for its limited coverage of cases and for not providing sufficient context or analysis to assist legal practitioners in making informed decisions [6]. The Patent Register is a JDSS used by patent examiners in Europe to search for and analyze patent applications. However, the Patent Register has been criticized for its complex search interface and for not providing sufficient analysis or context to assist examiners in making informed decisions [7]. This is however not in line with the focus of this research. LawPavilion is a JDSS that provides access to electronic versions of judicial opinions, legislation, and other legal materials in Nigeria. However, its limitations include incomplete coverage of cases, particularly from lower courts, and a lack of standardization in the reporting of cases[8].

LexisNexis is a global JDSS that provides access to legal information, including case law and legislation, from Nigeria and other countries. However, its limitations include the high cost of access, which can limit its use by small law firms and individual legal practitioners[9]. HUDOC is a JDSS used by judges and legal practitioners in Europe to search for and analyze cases related to human rights. However, HUDOC has been criticized for its limited search functionality and for not providing sufficient context or analysis to assist judges in making informed decisions [10]. CanLII Connects is a JDSS used by legal practitioners in Canada to share and discuss legal cases. However, CanLII Connects has been

criticized for its lack of quality control and for the potential for inaccurate or misleading information to be shared [11]. ELRS is a JDSS that provides access to electronic versions of judicial opinions in Kenya. However, its limitations include incomplete coverage of cases and a lack of uniformity in the reporting of cases, which can make it difficult for legal practitioners to find relevant case law [12].

From all the aforementioned studies, majority of the studies had limited access to case facts and also search functionality limitation due to little amount of legal cases inserted in their database. This is a major gap this study aimed to fill by increasing the reach to case facts, providing a larger and wider range of legal cases thereby improving the quality of legal information in the database and accuracy of legal verdicts.

II. MATERIALS AND METHODS

The architectural framework and structure developed for the JDSS is shown in Figure 1a and Figure 1b. The architectural framework consists of the following components: data sets, artificial neural network and python programming language. The framework shows how the components are functionally related. The function of each of the components is discussed in this section.

A. Data Sets

Data sets are an essential component of training and testing Recurrent Neural Networks (RNNs). RNN require large amounts of data to learn from, it is essential to split the data set into training, validation, and testing sets, and to balance the data set when necessary. Data set augmentation can be useful to increase the size of the training data set. The quality of data used to train a Recurrent Neural Network (RNN) has a significant impact on the accuracy and

effectiveness of the model. Here are some data collection and preparation methods in RNN:

- i. **Data Collection:** The first step in data collection is to define the problem that the RNN will be used to solve. This involves identifying the relevant input variables and the expected output. The next step is to collect data that represents the problem domain. This can involve gathering data from various sources, such as databases, surveys, or sensors. It is important to ensure that the data is representative of the problem domain and is relevant to the problem being solved.
- ii. **Data Cleaning:** Once the data has been collected, it must be cleaned to remove errors, inconsistencies, and missing data. This can involve identifying and removing outliers, correcting errors in the data, and imputing missing data. This step is important to ensure that the data is of high quality and can be used to train the RNN effectively.
- iii. **Data Transformation:** After cleaning the data, it may be necessary to transform the data into a format that is suitable for input into the RNN. This can involve scaling the data, encoding categorical variables, and normalizing the data. This step ensures that the data is in a format that can be processed by the RNN and is suitable for training the model.
- iv. **Data Splitting:** The next step is to split the data into training, validation, and testing sets. The training set is used to train the RNN, the validation set is used to tune the hyperparameters of the RNN, and the testing set is used to evaluate the performance of the RNN. The splitting ratio is typically 60-80% for training,

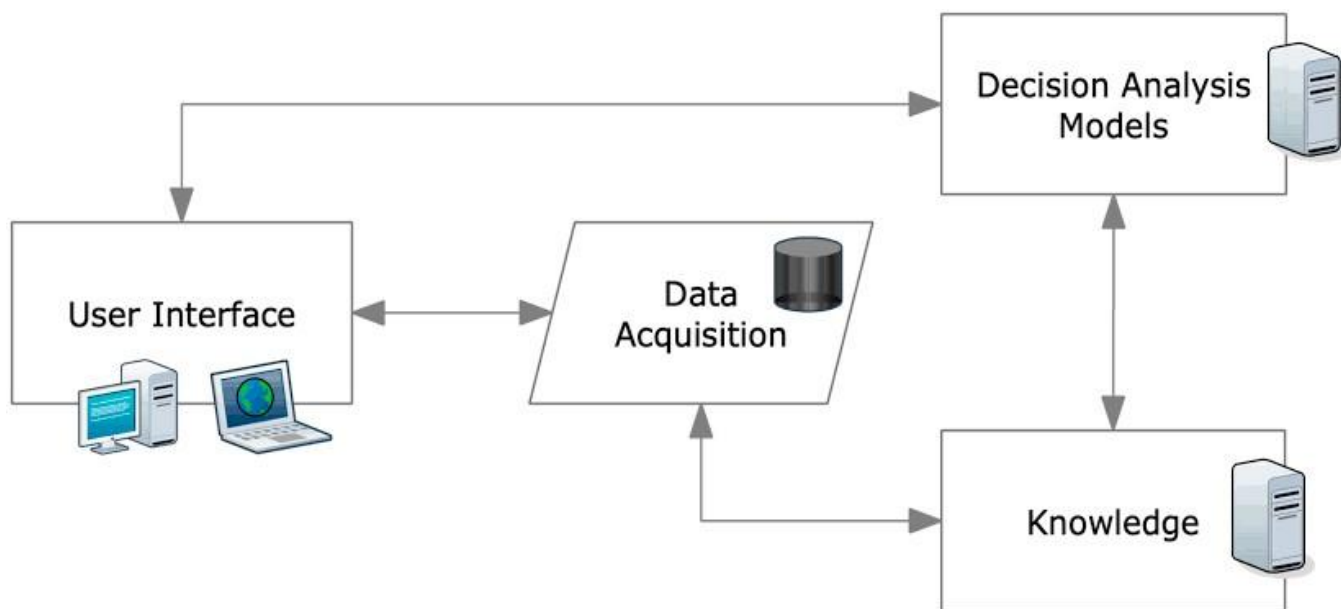


Figure 1a: The JDSS architecture framework

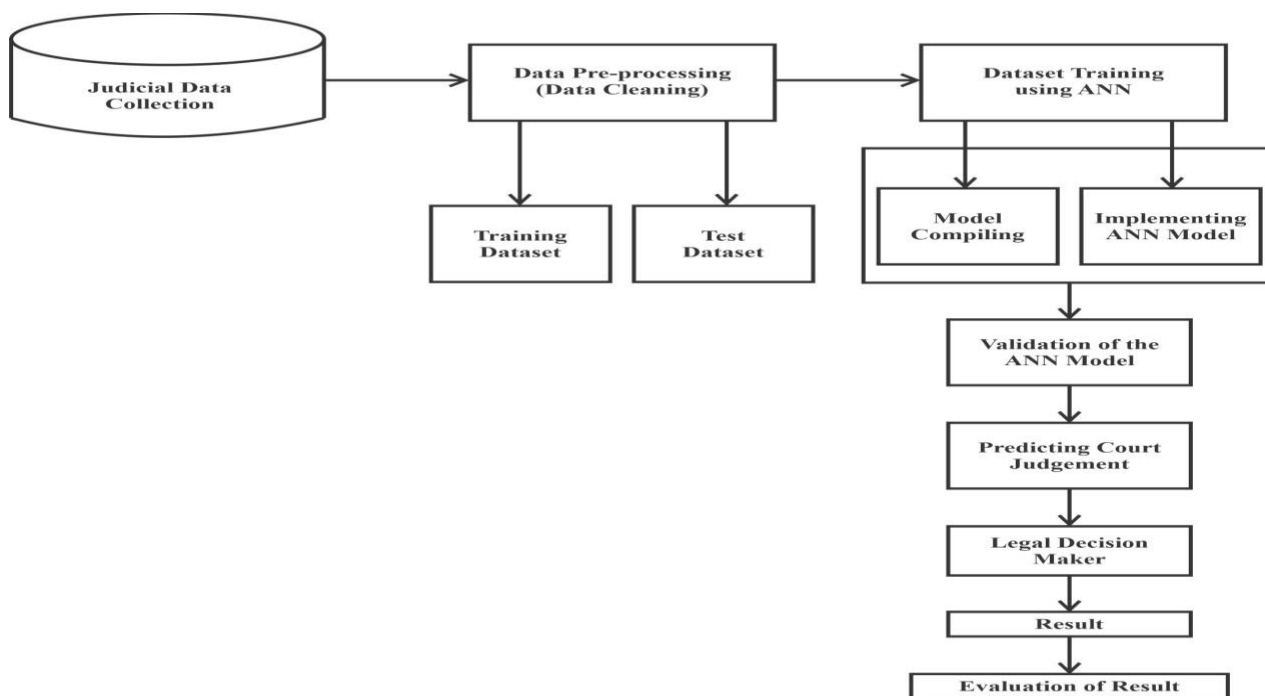


Figure 1b: Block diagram of the artificial neural network

1020% for validation, and 10-20% for testing.

- v. **Data Augmentation:** In some cases, it may be necessary to increase the size of the training data set to improve the performance of the RNN. This can be achieved through data augmentation techniques, such as flipping, rotating, or cropping images, or adding noise to the data. This step ensures that the RNN has sufficient data to learn from and can generalize well to new data.

B. Feature Extraction

Feature extraction is the process of selecting and extracting relevant features from raw input data in order to create a more compact representation of the data that can be used for analysis or further processing. In the context of Recurrent Neural Networks (RNNs), feature extraction is often used to reduce the dimensionality of the input data, which can help improve the efficiency and accuracy of the network.

C. Training the Datasets

The training data set is typically divided into two parts: the training set and the validation set. The training set is used to train the RNN by adjusting the weights and biases based on the error between the predicted output and the actual output for each input sample. The validation set is used to monitor the performance of the RNN during training and to avoid over-fitting. Over-fitting occurs when the RNN becomes too specialized to the training data set and does not generalize well to new, unseen data. The quality of the training data set is critical for the accuracy of the RNN. The data set should be representative of the problem domain and free from errors, inconsistencies, and outliers. The data set may also need to be

preprocessed by normalization, scaling, or encoding categorical variables to ensure that the RNN can learn from the data.

D. Implementation of the Developed DSS Framework using RNN Model

RNN is not a standalone decision-making system, but rather a component of a larger decision system. The system architecture is designed to support the integration of the RNN model with other decision-making components, such as rule-based systems, and to enable the deployment of the system on different platforms, such as web-based or desktop-based applications. The software components of the RNN are developed using Python programming language and then integrated it with the JDSS. This involves connecting the output of the RNN to the input of the JDSS, and vice versa. The JDSS is also developed with security and privacy in mind. Legal case data is often sensitive and confidential, and it is important to ensure that the system protects the privacy of the data and that it is only accessible by authorized users.

Finally, the integration of the RNN model with the DSS should be evaluated to ensure that it meets the requirements of the legal domain and provides accurate and effective analysis and decision-making capabilities. Evaluation may involve testing the system with real case data, evaluating its performance and accuracy, and soliciting user feedback to identify areas for improvement.

III. RESULTS AND DISCUSSION

This model was trained with lot of various court cases with different types of capital offences such as murder, manslaughter, treason, rape which had totally different scenarios and facts. The model was then put into actual working

```
# Training the dataset
model.fit(X_train, y_train, epochs=100, batch_size=100)

Epoch 1/100
9/9 [=====] - 2s 197ms/step - loss: 0.2829 - acc
uracy: 0.4789
Epoch 2/100
9/9 [=====] - 2s 180ms/step - loss: 0.2829 - acc
uracy: 0.4789
Epoch 3/100
9/9 [=====] - 2s 192ms/step - loss: 0.2829 - acc
uracy: 0.4789
Epoch 4/100
9/9 [=====] - 2s 226ms/step - loss: 0.2829 - acc
uracy: 0.4789
Epoch 5/100
9/9 [=====] - 2s 187ms/step - loss: 0.2829 - acc
uracy: 0.4789
Epoch 6/100
9/9 [=====] - 2s 185ms/step - loss: 0.2829 - acc
uracy: 0.4789
Epoch 7/100
9/9 [=====] - 2s 185ms/step - loss: 0.2829 - acc
uracy: 0.4789
```

conditions thereby tested with these data and the experiment shows that the model is well trained and also provides accurate verdicts.

These are the actual results from the datasets being used to train the prediction model. Exploratory Data Analysis (EDA) was put into use as it aids in the provision of data-driven insights which are used to check the dataset for imbalances and validity of the data prior to the creation of a decision model.

Over a thousand unique crime data spread over the between the years of 1970 and 2020 were accumulated and put into testing and training of the decision model (as shown in figure 2b). After sorting the datasets, there were some categories under which the data were classified, such as; 51.4% and 48.6% of the court cases were decided via Jury trial and Bench trial

respectively (as shown in figure 2c). Also, places the first party to have a 47.9% winning rate and second party got a 53.1% winning rate (as shown in figure 2d). The court cases dispositions were correctly distributed as sentenced, convicted, acquitted and appeal accordingly.

The confusion matrix of the RNN classifier classified 1746 court cases correctly as true positive while 245 court cases was misclassified as false positive. The classifier also classified 260 court cases correctly as true negative and 249 court cases classified as false negative (as shown in figure 2a). The performance metrics were evaluated and generated to have an accuracy of 80%, 87% precision, 87% sensitivity/recall and an F1-score of 87%.

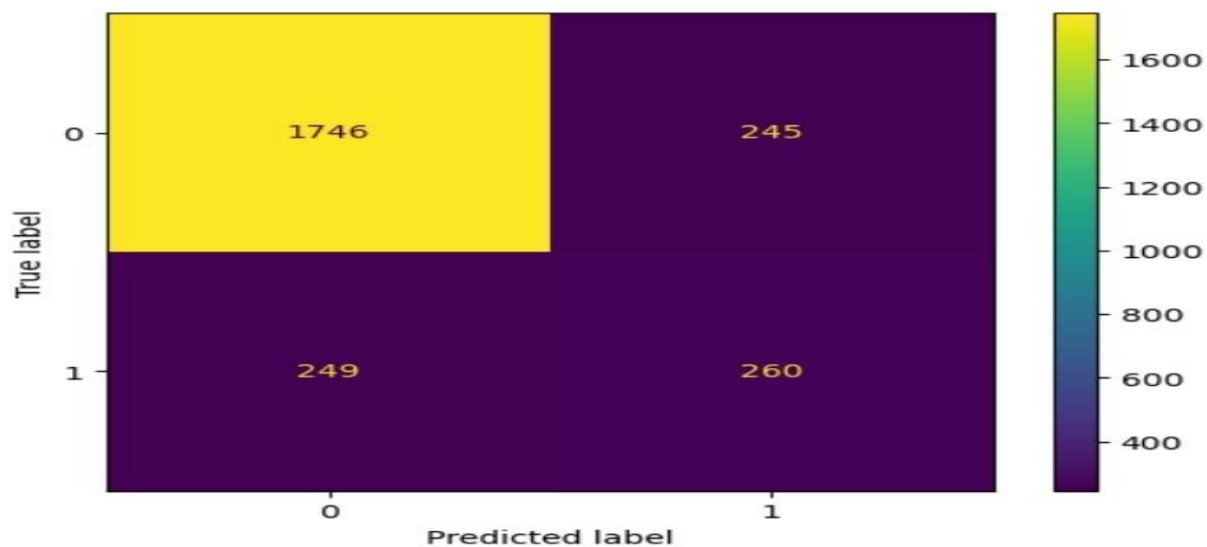


Figure 2a: Confusion matrix plot

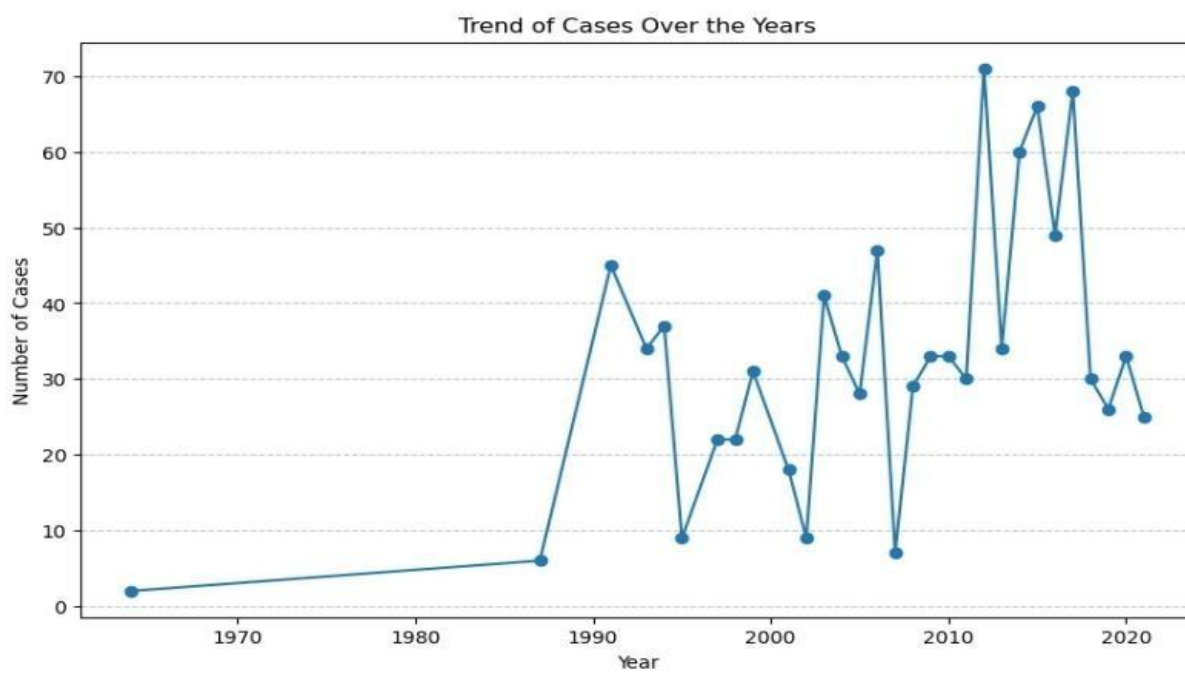


Figure 2b: Representation of crime cases

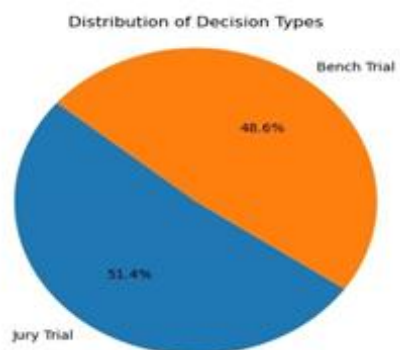


Figure 2c: Decision Types

Distribution of First Party Winners

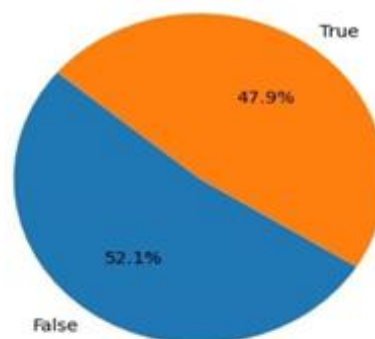


Figure 2d: Distribution of winners

A screenshot of the JDSS mode input form. The form contains the following fields:

- First party ***: Usman
- Second party ***: The State
- Year ***: Sep 12, 2023
- type of crime ***: murder
- facts ***: Kidnapping: Abduction of a political figure
- majority_vote ***: Guilty

A screenshot of the JDSS mode output form. The form displays the following information:

- type of crime**: murder
- facts ***: Kidnapping: Abduction of a political figure
- majority_vote ***: Guilty
- minority_vote ***: Guilty
- disposition ***: convicted
- COURT CASE DECISION**
- BENCH TRIAL**
- These are the reasons behind the decision:::** murder Kidnapping: Abduction of a political figure Guilty Guilty convicted

Figure 2e: Pictorial representation of the developed JDSS mode

IV. CONCLUSION

The paper presents the design and implementation of a Decision Support System for Legal Practitioners using the Recurrent Neural Network. The developed system provides the legal sector with benefits that improves the legal processes, resulting in faster and better decisions being made and knowledge sharing being performed more easily. Decision support systems process mass data, run complex statistical, mathematical models, data analyses and support decisions which leads to the provision of decision options for legal practitioners. Although, it should be kept in mind that these systems are not designed to eliminate “bad” decision but is a great tool for facilitating decision-making process thereby drastically reduce the rate of wrong judgement and time spent in analysing. It is not a one-stop-shop application that can replace human judgement as the effectiveness relies mostly on human competencies and on how responsibly it is used.

References

- [1] Laudon, K. C., & Laudon, J. P. "Management information systems: managing the digital firm (16th ed.)." *Pearson Education*. 2021
- [2] Kamal, M. M., Sarwar, M. S., & Bhatti, U. A.. "Exploring the factors that affect the adoption of decision support systems in legal systems: A case study of Pakistan." *Journal of Decision Systems*, vol 30 no 1, 2021, pp.1-25.
- [3] Green, P. F., & Hull, J. J. "Decision support systems in legal systems: A systematic review." *Journal of Decision Systems*, Vol 28 No 2, 2019, pp.125-157.
- [4] Maluwa, S. "A Novel Approach to Decision Support Systems in Healthcare" *International Journal of Healthcare Information Systems and Informatics*, Vol 13 No 4, 2017, pp. 56-73.
- [5] Moulton, R. . "Optimizing Decision-Making Processes: A Comprehensive Review of Decision Support Systems" *Journal of Information Technology Management*, Vol 37 No 3, 2017, pp.45-62.
- [6] Lee, K. C. & Kim, J. "Decision-making models and frameworks: A review article." *International Journal of Information Management*, 51, 2017, 102073.
- [7] Svensson, A.. "Integrating Artificial Intelligence with Decision Support Systems: A Case Study in Financial Forecasting." *Journal of Applied Information Systems*, Vol. 12 No 2, 2019, pp. 87-104.
- [8] Alao, O. "Enhancing Business Decision-Making: A Study on the Implementation of Decision Support Systems" *International Journal of Business Information Systems*, Vol 38 No 4, 2019, pp. 112-128.
- [9] Ogwuma, C. & Egede. "Integration of Machine Learning Techniques in Decision Support Systems for Healthcare Management." *Journal of Health Informatics and Decision Support Systems*, Vol 15 No 2, 2020, pp. 75-89.
- [10] Kurki, E.. "Enhancing Decision-Making with Dynamic Decision Support Systems." *Journal of*

- Advanced Information Technology, Vol 7 No. 2*, 2019, pp.87-102.
- [11] Fisher, J. "Improving Decision-Making Processes: An Evaluation of Decision Support Systems in Manufacturing." *Journal of Operations Management, Vol 32 No 3*, 2017, pp 112-128.
- [12] Akhavan, P., Yazdani, M., & Zolfaghari, S. (2021). "Investigating the factors influencing the adoption of decision support systems in organizations: A systematic review. *Journal of Decision Systems*, 30(1), 1-28. "