Contents lists available at Science-Gate

International Journal of Advanced and Applied Sciences

Journal homepage: http://www.science-gate.com/IJAAS.html

Optimizing semantic error detection through weighted federated machine learning: A comprehensive approach

Naila Samar Naz [1], Sagheer Abbas [1], Muhammad Adnan Khan [2, 3, 4, *], Zahid Hassan [1], Mazhar Bukhari [5], Taher M. Ghazal [6, 7]

[1]*School of Computer Science, National College of Business Administration and Economics, Lahore, Pakistan*
[2]*School of Computing, Skyline University College, Sharjah, UAE*
[3]*Department of Software, Faculty of Artificial Intelligence and Software, Gachon University, Seongnam, South Korea*
[4]*Riphah School of Computing and Innovation, Faculty of Computing, Riphah International University, Lahore, Pakistan*
[5]*Department of Computer Sciences, The Institute of Management Sciences, Lahore, Pakistan*
[6]*Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM), Bangi, Malaysia*
[7]*Applied Science Research Center, Applied Science Private University, Amman, Jordan*

A R T I C L E I N F O

Article history:
Received 30 August 2023
Received in revised form
18 December 2023
Accepted 9 January 2024

Keywords:
Artificial neural network
Semantic error detection
Federated learning
Natural language processing
SED-WFML

A B S T R A C T

Recently, the improvement of network technology and the spread of digital documents have made the technology for automatically correcting English texts very important. In English language processing, finding and fixing mistakes in the meaning of words is a very interesting and important job. It is also important to fix wrong data in cleaning data. Usually, systems that find errors need the user to set up rules or statistical information. To build a good system for finding mistakes in meaning, it must be able to spot errors and odd details. Many things can make the meaning of a sentence unclear. Therefore, this study suggests using a system that finds semantic errors with the help of weighted federated machine learning (SED-WFML). This system also connects to the web ontology's classes and features that are important for the area of knowledge in natural language processing (NLP) text documents. This helps identify correct and incorrect sentences in the document, which can be used for many purposes like checking documents automatically, translating, and more. During its training and checking stages, the new model identified correct and incorrect sentences with an accuracy of 95.6% and 94.8%, respectively, which is better than earlier methods.

1. Introduction

Semantic errors (SE) result in phonological correct words that make no sense or are nonsensical in context (McKinnon et al., 2018). Typically, writers make such mistakes due to ignorance or typos on the keyboard. An inexperienced writer may mix up the desired term with another with a similar spelling or sound. Once the writer's grasp of word meanings is shaky, he/she may select a word with a meaning that appears applicable but is actually erroneous (McCarthy et al., 2017). The following sentence provides two examples of spelling errors (SE) caused by the writer's misunderstanding of the word "piece": "Can I have a peace/member (piece) of pizza?"

In the first example, the writer incorrectly spells "piece" as "peace" because the two words sound alike. In the second example, the writer uses "member" instead, thinking it means the same as the intended word. In the case of "pizza," these mistakes are phonologically and syntactically correct but conceptually wrong. Spelling errors due to writing mistakes often involve using a word that looks similar to the correct one (for example, adding an extra letter or replacing one letter with another). For instance: "Ice cream and cookies are the best deserts (desserts)."

In this case, deleting a letter results in the word "desert," which disrupts the text's cohesiveness. Currently, all writing editors include error detection (Ed) tools. All writing editors concentrate on phonological problems, and the correction recommendations are comparable to the suspect

* Corresponding Author.
Email Address: muhammad.adnan@skylineuniversity.ac.ae (M. A. Khan)
https://doi.org/10.21833/ijaas.2024.01.018
Corresponding author's ORCID profile:
https://orcid.org/0000-0003-4854-9935

word in letters/sounds. Syntactic problems are not always evident due to syntactic analyzer inadequacies, and current syntax repair tools are rare/poor. SE is completely undetected. In fact, dealing with such problems is more challenging. It necessitates knowledge beyond morphology and syntax. Natural language processing researchers are still working on detecting and fixing semantic mistakes. These are mistakes that result in words that are lexically correct but semantically erroneous (Zribi, 2023). The proposed model focuses on the detection of SE in English texts. Using the framework of the word to be checked, a method that combines several layers of Marvin Minsk's thinking base model is presented.

The leading edge of technology, Natural Language Processing (NLP), is altering how we interact with and comprehend human language. The identification and correction of semantic flaws in text is a crucial and frequently difficult component of the large field of natural language processing (NLP). Semantic errors happen when the meaning a piece of writing conveys does not match the intended message, which can cause misinterpretations, misunderstandings, and perhaps even unintended consequences. This branch of NLP focuses on spotting and fixing these imperceptible but significant mistakes that can significantly improve the caliber and clarity of written communication (Kamal and Himel, 2023).

In recent years, WFML has become increasingly popular due to its ability to learn from limited, secure data while preserving privacy. Instead of relying on outdated methods like integrating data from different databases, FL permits the creation of an inclusive model on a simple server while still maintaining data confidentiality. By collaborating with multiple enterprises, WFML enables the construction of a master model utilizing trained data from assorted derivations without directly sending data. Many Machine Learning (ML) algorithms have been advanced using this approach to accurately predict SE in the text (Tabassum et al., 2021).

WFML works successfully despite the fact that it does not share data because it is a smart ML algorithm. WFML performs effectively despite the fact that no data is given since it is a formidable ML technique (Khan et al., 2020). WFML uses ML models to increase data secrecy and safety, most notably to safeguard the FL process and data. The implementation of federated learning protects data privacy across different locations. To solve an ML problem with WFML, a collection of productions or hypothetical organizations work together under the supervision of a centralized server. Data is kept private during the training process. Unlike prior distributed learning approaches, which kept protected data on a sole server, contemporary distributed learning uses a standardized worldwide architecture that any organization can utilize (Ahmad et al., 2019; Alhaidari et al., 2021). The data is then used by each organization to create its own model. Each censer then transmits the data to the server utilizing the model's gradient of inaccuracy.

The central server assembles all opinions from participants and, in addition, adjusts the inclusive model based on established standards. In other words, centers that report bad or unexpected outcomes may go unnoticed. This technique is used till the inclusive model is learned in a solo round of FL. Fig. 1 displays the entire FL design but with substantial variations (Abad et al., 2021).

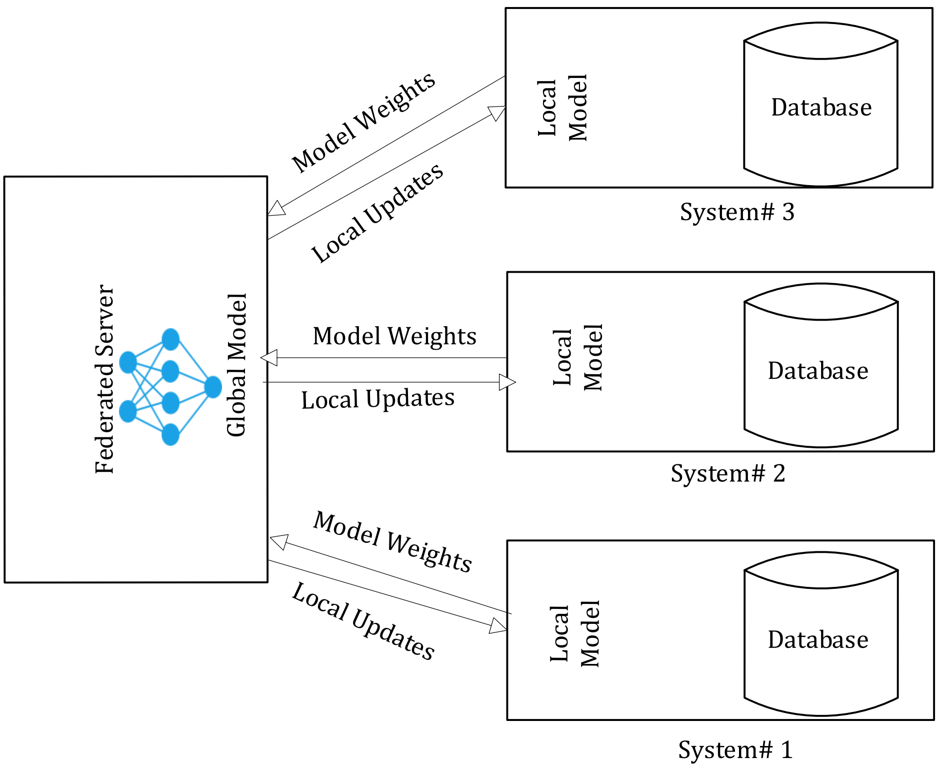

Fig. 1: Flow chart of the FML (Ali et al., 2023)

To enable quicker data dispensation and more bandwidth, in addition to data control, edge computing entails placing computer processing capabilities near the source of facts, which is recurrently an IoT device/sensor. By performing data processing at the network's edge (Fu et al., 2023), edge computing diminishes the need for huge amounts of statistics to travel amid the servers, cloud services, and endpoints or edge locations for processing. Its goal is to make intelligent computing easier and to connect data sources and devices, especially in contemporary applications similar to data science and artificial intelligence. Edge computing's major goal is to bring data sources/devices nearer composed, resulting in increased application and device performance and efficiency (Ali et al., 2023; Oueida et al., 2018).

This study explores the application of federated machine learning to achieve a particular objective, introducing a system called Semantic Error Detection driven by Weighted Federated Machine Learning (SEDWFML). The effectiveness of the proposed model is evaluated using various statistical measures, including accuracy, miss rate, recall, selectivity, and precision.

2. Literature review

Many studies have been conducted on philological and syntactic errors, but fewer studies have been conducted on SE, which are extra problematic to fix. Zribi (2023) introduced an innovative approach to address semantic errors within Arabic documents. This research focused on a methodology referred to as "Easy" meta-embedding, which was designed to identify and rectify semantic inaccuracies in text written in the Arabic language. In contrast, specific details of the methodology were not provided, and they hold significant potential for natural language processing and text analysis applications. It also indicated an attempt to improve the quality and accuracy of Arabic textual content by spontaneously detecting and modifying semantic errors, thus contributing to improved document comprehension and overall linguistic superiority in the Arabic language area.

Semantically Enhanced Classification of Real-world Tasks (SECRET) was a system developed by Akmandor et al. (2020) that combined the benefits of supervised ML in addition to NLP approaches into a single system. SECRET classified data by mixing semantic statistics in the docket with the existing data. SECRET produces up to 14.0 % accuracy in addition to 13.1 % F1 score gains when matched to regular supervised learning, according to experimental results. Furthermore, as compared to ensemble approaches, SECRET improves accuracy by 12.7 percent and F1 score by 13.3 percent. This alludes to a new research avenue in supervised classification that incorporates semantic data. In Deng and Liu (2018), text mining tools were used to conduct a retrospective observational feasibility study based on ontologies. From June 2006 through June 2016, a corpus of social media material containing 13,757,900 Caring Bridge diary entries.

According to Chen et al. (2023), the researchers have proposed the critical task of error detection in the context of text-to-SQL semantic analysis. While specifically addressed, a significant challenge in natural language processing involved converting natural language text into structured SQL queries for database interactions, and this process can be error-prone. By developing techniques or algorithms for error detection in this domain, the research aimed to enhance the accuracy and reliability of such conversions. This work has the potential to improve the performance of natural language interfaces to databases, making it easier for users to interact with databases using everyday language and ultimately advancing the field of text-to-SQL semantic parsing.

Yang and Huang (2023) have conducted a systematic review of existing studies in the field of Natural Language Processing (NLP) as it relates to aviation safety. The study investigated the applications and implications of NLP techniques in the aviation domain. While specifically exploring how NLP can be harnessed to improve aviation safety through the analysis of textual data, such as incident reports, maintenance logs, and communication transcripts. This systematic review shed light on the current state of research in this area and delivered insights into potential future directions. This review was valuable for advancing the utilization of NLP to enhance aviation safety measures and decision-making processes within the aviation industry.

Fu et al. (2023) have developed an innovative approach using Mogrifier Long Short-Term Memory (LSTM) architecture and semantic representation to enhance the accuracy and effectiveness of identifying anomalies within log data. This proposed research signified an effort to improve the capabilities of anomaly detection within log data, which was critical for maintaining the reliability and security of various systems and services. The utilization of semantic representation suggested a focus on contextual understanding, potentially enabling the system to identify more subtle and complex anomalies. This research also embraced the promise of contributing to the development of more robust and sophisticated anomaly detection techniques within the field of services computing (Fu et al., 2023).

This research presents a study that explores the use of federated learning for improving audio-semantic communication. The authors proposed a new framework for audio semantic communication based on federated learning and evaluated its performance using a dataset of audio signals. The results show that the suggested framework outstrips traditional methods in terms of precision and efficiency, demonstrating the potential of FL for enhancing audio-semantic communication (Tong et al., 2021).

In a study by Li et al. (2020), the team applied federated learning to the analysis of functional magnetic resonance imaging (fMRI) data from different locations. This approach allowed for the analysis of fMRI data from various sites while ensuring the privacy of the individuals whose data was used. Additionally, they implemented domain adaptation methods to mitigate the discrepancies in fMRI data collected from different locations. The method they introduced significantly enhanced the precision of fMRI data analysis across various sites and upheld the privacy of the participants. This research was conducted using the ABIDE dataset, a common resource in the realm of fMRI analysis.

This research presents a study that explores the use of a dictionary of literal paronyms for correcting semantic errors in natural language texts. The authors may have proposed a method to identify and correct semantic errors by utilizing a dictionary of words similar in sound or spelling but with different meanings. The viability of the suggested approach was likely evaluated on a corpus of natural language texts, and the results might substantiate the success of the suggested strategy in correcting semantic errors. The study may provide a useful contribution to the field of NLP and text analysis.

In contrast to prior research, the propound model provides the purpose of information extraction in this study. The proposed Model internments the categorization of data based on its importance. Artificial Neural Network (ANN) is used to manage vast amounts of data and extract information quickly. In machine learning, the ANN approach to the classification of a huge dataset is particularly effective. It then plans a standard for retrieving information based on the major traits notorious for handling well-known issues.

3. Limitation of previous work and our contribution

Table 1 illustrates the comparative analysis between the present study and prior research in the same domain regarding Prediction, Decision-making, Techniques, and limitations. Previous work (Li et al., 2020; Akmandor et al., 2020; Yang and Huang, 2023; Fu et al., 2023) have limits such as privacy issues, limited Semantic detection, research scope, and limited data availability,

Main Contribution of Our Proposed System:

- We applied a weighted federated learning machine to the dataset.
- We Proposed an ANN-based system, SED-WEML.
- The proposed system improves the accuracy of the system, which is 95.60%.
- We achieved the optimum results by using the proposed system SED-WFML.

Table 1: Limitation of the previously published works

Reference	Technique	Concern	Prediction	Decision making	Weighted federated learning	Limitation
Li et al. (2020)	Privacy-preserving federated learning	Multi-site fMRI analysis	Yes	No	No	Privacy constraints
Akmandor et al. (2020)	Semantically enhanced classification	Real-world task classification	Yes	No	No	Limited semantic coverage
Yang and Huang (2023)	Natural language processing (NLP)	Aviation safety research	Yes	No	No	Lack of research scop
Fu et al. (2023)	Log anomaly detection	Semantic representation	Yes	No	No	Limited data availability
Proposed model	Weighted federated learning	Semantic error detection	Yes	Yes	Yes	Transparencymay be integrated using explainable AI

4. Materials and methods

FL is a method of training ML models with sensitive data that is dispersed across different locations or organizations. In this approach, a common, inclusive model is trained on a central server, but the data remains on the participating organizations' servers. During the training process, the server receives feedback from each organization in the form of error gradients, which are used to update the global model. No actual data is exchanged, preserving data privacy and localization. This process is repeated until the model has reached the desired level of accuracy, with each cycle forming a federated learning cycle.

The SED-WFML model shown in Fig. 2 is designed to identify mistakes in meaning using different sources of data like sensors and devices that detect text. During the training stage, two types of ANN methods, Levenberg-Marquardt (LM) and Bayesian Regularization (BR), are used, one at a time. It also checks how well they work by looking at their accuracy, precision, sensitivity, and ability to distinguish between different cases. If the results are not good enough, the models are trained again. If the results are good, the models are combined to create a unified model. This unified model's performance is then checked. If it meets the required standards, it is saved in the cloud. If not, it needs to be trained again.

The aim of using different methods on the client side is to gather data and decide on the best weights to use on the server side. The data is split into 70% for training and 30% for validation, and both steps are carried out at the same time. The learned models then send the weights to the server, where FML is applied to SED. FML offers better accuracy and enhanced data security, making it a useful choice.

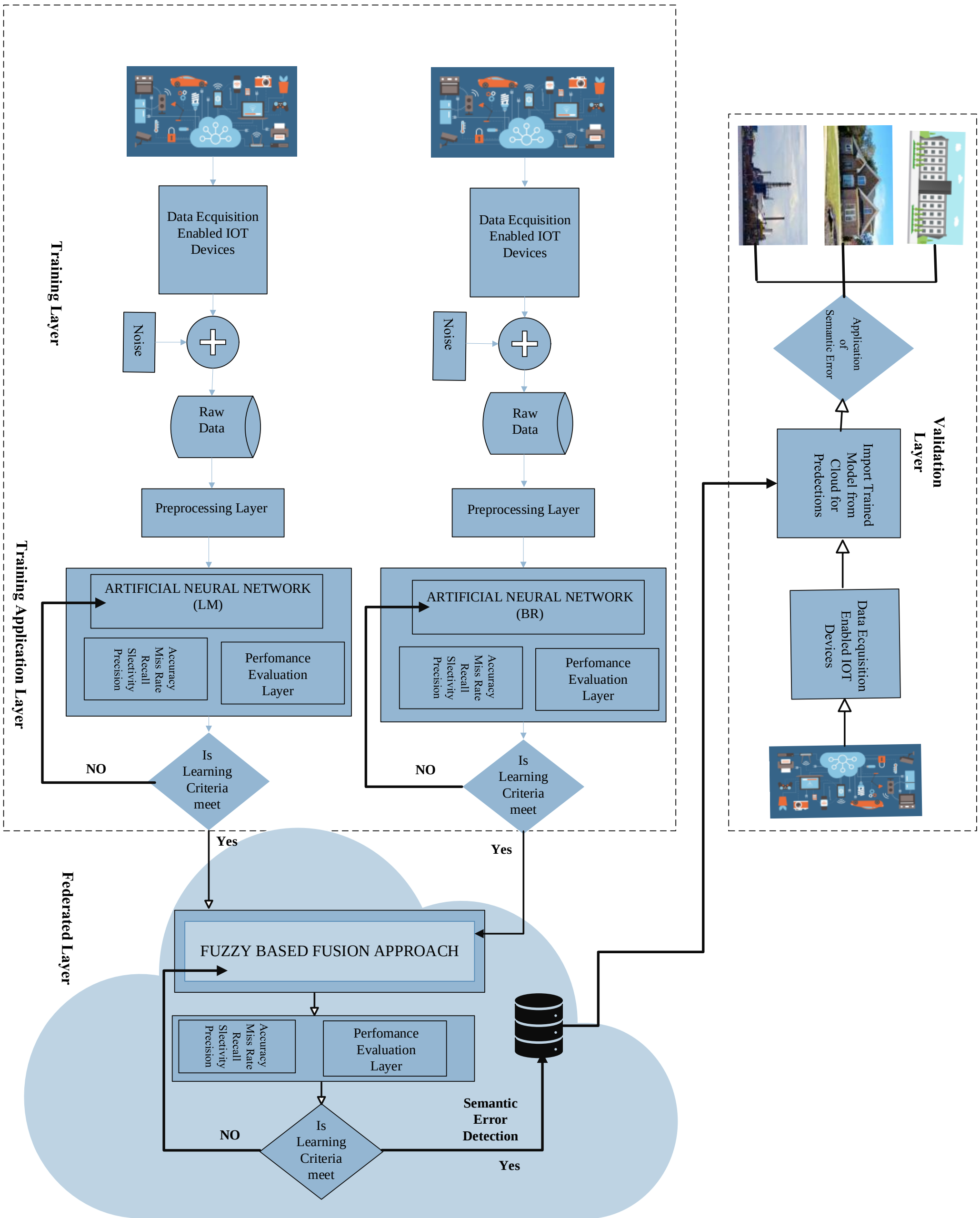

Fig. 2: Proposed model for semantic error detection

The input features are denoted by $[s_1, s_2, s_3, \ldots s_3]$, where t, f, and k represent the starting point of each layer element. c_1 and c_2 represent the bias created for each layer. Weights between the input in addition hidden layers are signified by $a_{f,t}$, and weights among the hidden plus output layers are denoted by $b_{f,n}$.

The entire quantity of variables in every input, hidden, and output layer is given by n, k, and g, representing the length, width, and height of each layer, respectively. Eq. 1 (Bibi et al., 2021) is used to determine the result at every single neuron of the hidden layer, where $w^{cli}{}_f$ pertains to the result of the i^{th} client cl_i of the f^{th} hidden neuron.

$$w^{cli}{}_f = \frac{1}{1+e^{-(b_1+\sum_{t=1}^{n}(a^{cli}{}_{t,f}*s_t))}} \quad \text{Where } 1 \le f \le k. \tag{1}$$

Correspondingly, $x^{cli}{}_n$ denotes the results at the output stage at the nth neuron in Eq. 2 (Ali et al., 2023).

$$x^{cli}{}_n = \frac{1}{1+e^{-(b_2+\sum^{k}{}_{n=1}(b^{cli}{}_{f,n}*w^{cli}{}_f)}} \quad \text{where } 1 \le n \le g \tag{2}$$

$$F^{cli} = \frac{1}{2}\sum{}_n(\beta^{cli}{}_n - x^{cli}{}_n)2. \tag{3}$$

In Eq. 3 (Khan et al., 2020), F^{cli} represents the i^{th} client error, while $\beta^{cli}{}_n$ and $x^{cli}{}_n$ signify the expected and anticipated outputs, respectively.

Eq. 4 (Ali et al., 2023; Rehman et al., 2020) expresses the weight variation for the output layer as:

$$\Delta A \propto -\frac{\partial F^{cli}}{\partial A^{cli}} \tag{4}$$

$$\Delta B \propto -\frac{\partial F^{cli}}{\partial B^{cli}} \tag{5}$$

$$\Delta b^{cli}_{f,n} \propto -\frac{\partial F^{cli}}{\partial b^{cli}_{f,n}} \quad (6)$$

$$\Delta a^{cli}_{f,t} \propto -\frac{\partial F^{cli}}{\partial a^{cli}_{t,f}}. \quad (7)$$

The above equation can be expressed using the chain rule method as:

$$\Delta b^{cli}_{f,n} = -\zeta \frac{\partial F^{cli}}{\partial x^{cli}_{n}} \times \frac{\partial x^{cli}_{n}}{\partial b^{cli}_{f,n}} \quad (8)$$

where, ζ denotes the constant. By substituting the variables in Eq. 5, the change of weight is calculated as Eq. 6.

$$\Delta b^{cli}_{f,n} = \zeta(\beta^{cli}_{n} - x^{cli}_{n}) \times x^{cli}_{n}(1 - x^{cli}_{n}) w^{cli}_{f} \quad (9)$$

$$\Delta b^{cli}_{f,n} = \zeta \lambda^{cli}_{n} w^{cli}_{f} \quad (10)$$

where,

$$\lambda^{cli}_{n} = (\beta^{cli}_{n} - x^{cli}_{n}) \times x^{cli}_{n}(1 - x^{cli}_{n}). \quad (11)$$

Updating input and hidden weights and employing the chain rule.

$$\Delta a^{cli}_{f,t} \propto -\left[\sum_n \frac{\partial F^{cli}}{\partial x^{cli}_{n}} \times \frac{\partial x^{cli}_{n}}{\partial w^{cli}_{f}}\right] \times \frac{\partial w^{cli}_{f}}{\partial a^{cli}_{t,f}} \Delta a^{cli}_{f,t} = -\left[\sum_n \frac{\partial F^{cli}}{\partial x^{cli}_{n}} \times \frac{\partial x^{cli}_{n}}{\partial w^{cli}_{f}}\right] \times \frac{\partial w^{cli}_{f}}{\partial a^{cli}_{t,f}} \quad (12)$$

$$\Delta a^{cli}_{t,f} = \zeta \sum_n (\beta^{cli}_{n} - x^{cli}_{n}) \times x^{cli}_{n}(1 - x^{cli}_{n}) \times (b^{cli}_{f,n}) \times w^{cli}_{f}(1 - w^{cli}_{f}) \times s_t \quad (13)$$

$$\Delta a^{cli}_{t,f} = \zeta\left(\sum_n \lambda^{cli}_{n} * b^{cli}_{f,n}\right) \times w^{cli}_{f}(1 - w^{cli}_{f}) \times s_t. \quad (14)$$

It can be represented as follows after being simplified:

$$\Delta a^{cli}_{t,f} = \zeta a^{cli}_{f} \times s_t \quad (15)$$

where,

$$a^{cli}_{f} = \zeta\left(\sum_n \lambda^{cli}_{n} * b^{cli}_{f,n}\right) \times w^{cli}_{f}(1 - w^{cli}_{f}) \quad (16)$$

$$b^{cli}_{f,n}(d+1) = b^{cli}_{f,n}(d) + \mathcal{K}\Delta b^{cli}_{f,n}. \quad (17)$$

Eq. 17 utilizes changing of weights in output and hidden layers (Gutierrez et al., 2017). Eq. 18 is applied in adjusting weights in input as well as hidden layers (Ali et al., 2023; Rehman et al., 2020).

$$a^{cli}_{t,f}(d+1) = a^{cli}_{t,f}(d) + \mathcal{K}\Delta a^{cli}_{t,f} \quad (18)$$

4.1. Proposed i^{th} client ML algorithm

Table 2 displays the pseudo-code of the proposed MLA that is implemented at the i^{th} client.

4.2. Transfer weights

The weights are subsequently transmitted to the FL server. To ensure the security of the system, these weights can be encoded prior to transmission.

4.3. FL server

Every client is sending its optimized weight $(A^{cli}_{SE1}, B^{cli}_{SE0})$ to the server federation. In this study, every client is trained using the proposed federated machine learning techniques, and the upgrade weights for the LM and BR algorithms are provided in Eqs. 19 to 20, respectively (Siddiqui et al., 2022).

Table 2: Proposed i^{th} client ML pseudo code

Client Training Algorithm $(d, A^{cli}_{SE1}, B^{cli}_{SE0})$
1. Starting
2. Local data is divided into compact groups of size Cs
3. Starting with Input and hidden layer weights $((A^{cli}_{SE1}, B^{cli}_{SE0}))$, $F^{cli} = 0$, epochs d = 0
4. for each compact group (Cs)
i. Using feed-forward phase to
a. Compute w^{cli}_{f} using Eq. 1
b. Compute estimated output (x^{cli}_{n}) using Eq. 2
ii. Compute the Error values (F^{cli} using Eq. 3)
iii. Weights updating phase
a. Compute $\Delta b^{cli}_{f,n}$ using Eq. 10
b. Compute $\Delta a^{cli}_{f,t}$ using Eq. 15
c. Adjust the weights between hidden and output layers $b^{cli}_{f,n}$ (d + 1) with Eq. 17
d. Adjusting weights in input and hidden layers $a^{cli}_{f,t}$ (d + 1) with Eq. 18
if terminating norms do not complete, then
Repeat step 4
else, repeat step 5
5. Returning the optimal weights $(A^{cli}_{SE1}, B^{cli}_{SE0})$ to Federated Server
End

$$A^{cl1}_{\mathrm{SE1}\,(ANN)} = \begin{pmatrix} a^1_{11} & \cdots & a^1_{1c_n} \\ \vdots & \ddots & \vdots \\ a^1_{r_{m1}} & \cdots & a^1_{r_{mcn}} \end{pmatrix}_{d1*d2} \quad (19)$$

$$A^{cl2}_{\mathrm{SE1}\,(SVM)} = \begin{pmatrix} a^2_{11} & \cdots & a^2_{1c_n} \\ \vdots & \ddots & \vdots \\ a^2_{r_{m1}} & \cdots & a^2_{r_{mcn}} \end{pmatrix}_{d3*d4} \quad (20)$$

The integrated optimized weights for the server federation, from the input and hidden layer, can be expressed using Eq. 21, where A^{n}_{SE1} (FS) indicates the weighted average of all nearby trained clients.

$$A^{n}_{\mathrm{SE1}\,(FS)} = 2A^{cl1}_{\mathrm{SE1}\,(LM)} + A^{cl2}_{\mathrm{SE1}\,(BR)} \quad (21)$$

The aggregation process encounters an issue due to the additive property of matrices, as the matrices being added must have consistent dimensions. As evident from Eq. 21, the locally proficient matrices cannot be added directly as they have different proportions. To address this issue, the proportions of all relevant matrices need to be standardized. This is achieved by concatenating a zero matrix with each matrix where necessary. To determine the appropriate size of the zero matrices, Eq. 22 is used to calculate the maximum number of rows across all locally proficient clients (Ali et al., 2023).

$$\mathrm{Max}_{\mathrm{r-SE1}} = maxf_0(\mathrm{d1, d3}) \quad (22)$$

Similarly, we employ Eq. 22 to determine the maximum number of columns from all locally trained clients (Eq. 23).

$$\mathrm{Max}_{\mathrm{c-SE1}} = maxf_0(\mathrm{d2, d4}) \quad (23)$$

The following is the process for embedding a zero matrix with each ideal weight matrix: The symbols ZM_{LM}, and ZM_{BR} in Eqs. 24-25 indicate the zero matrices for the LM and BR algorithms correspondingly. This will yield a matrix of 0, which will be mixed with the ideal weight matrix (Yang and Huang, 2023).

Each locally trained model weight will be horizontally concatenated with these zero matrices.

$$ZM_{SE1-LM} = zeros(Max_{r-SE1}, Max_{c-SE1} - d2) \quad (24)$$
$$ZM_{SE1-BR} = zeros(Max_{r-SE1}, Max_{c-SE1} - d4) \quad (25)$$

The horizontal concatenation is given in Eqs. 26–27:

$$A_{SE1-LM} = horcat(ZM_{LM}, aSE1(LM)) \quad (26)$$
$$A_{SE1-LM} = horcat(ZM_{BR}, aSE1(BR)). \quad (27)$$

In Eqs. 26-27, $A_{SE1-ANN}$ and $A_{SE1-SVM}$ have similar dimensions; therefore, now, these matrices can be aggregated collectively. To attain the server federation or global model, we utilize Eq. 28.

$A_{SE1-ANN}$ and $A_{SE1-SVM}$ have the same dimensions in Eqs. 26-27, so these matrices can now be aggregated to each other. Eq. 28 is used to gain the federated server otherwise global model.

$$A_{SE1-FS} = 2A_{SE1-LM} + A_{SE1-BR}. \quad (28)$$

A_{SE1-FS} in Eq. 28 reflects the optimal federated weights between the insertion and invisible layers. Based on their performance, the locally taught clients are assigned different scaling factors.

4.4. Optimal weights of invisible output layer

The optimal weights of the invisible layer to the output layer for LM a, like the insertion layer to the invisible layer, may be described using Eqs. 29-30.

$$B^{cli}_{SE0\ (FS)} = 2B^{cl1}_{SE0\ (LM)} + B^{cl2}_{SE0\ (BR)} \quad (31)$$

$$B^{cl1}_{SE0\ (LM)} = \begin{pmatrix} b^1_{11} & \cdots & b^1_{1c_n} \\ \vdots & \ddots & \vdots \\ b^1_{r_{m1}} & \cdots & b^1_{r_{mcn}} \end{pmatrix}_{d5*d6} \quad (29)$$

$$B^{cl2}_{SE0\ (BR)} = \begin{pmatrix} b^2_{11} & \cdots & b^2_{1c_n} \\ \vdots & \ddots & \vdots \\ b^2_{r_{m1}} & \cdots & b^2_{r_{mcn}} \end{pmatrix}_{d7*d8} \quad (30)$$

$$B^{cli}_{SE0\ (FS)} = 2B^{cl1}_{SE0\ (LM)} + B^{cl2}_{SE0\ (BR)}. \quad (31)$$

The federated weights can be gotten utilizing Eq. 31; however, this fusion suffers from dimension inconsistency. To verify that the dimensions of all client weight matrices are constant, the same approach used for embedding the 0- matrix by means of each ideal weight matrix will be applied.

$$Max_{r-SE0} = max(d5, d7) \quad (32)$$
$$Max_{c-SE0} = max(d6, d8) \quad (33)$$
$$ZM_{SE0-LM} = zeros(Max_{r-SE0}, Max_{c-SE0} - d2) \quad (34)$$
$$ZM_{SE0-BR} = zeros(Max_{r-SE0}, Max_{c-SE0} - d4) \quad (35)$$
$$B_{SE0-LM} = horcat(ZM_{LM}, b_{SE0}(LM)) \quad (36)$$
$$B_{SE0-BR} = horcat(ZM_{BR}, b_{SE0}(BR)) \quad (37)$$
$$B_{SE0-FS} = 2B_{SE0-LM} + B_{SE0-BR}. \quad (38)$$

In Eq. 38, B_{SE0-FS} the optimal federated weights of the invisible layer towards the output layer are represented. Based on their performance, the locally explained clients are assigned different scaling factors.

4.5. Pseudo code for the proposed WFML algorithm

Table 3 displays the server-side pseudo-code for the proposed WFML Algorithm.

Table 3: Proposed WFML algorithm pseudo code

1. Starting
2. Load weights (A_{SE1-FS}, B_{SE0-FS})
3. For each cycle Do
for each client Do
$[A^{cli}_{SE1}, B^{cli}_{SE0}]$ = Client ($d, A^{cli}_{SE1}, B^{cli}_{SE0}$)
End
End
4. Calculate B_{SE0-FS} using Eq. 38
5. Calculate A_{SE1-FS} using Eq. 28
6. Unknown data sample estimation
a. for I = No. of Samples
i. Calculate $w^{FS}{}_f = \frac{1}{1+e^{-(b_1+\sum_{t=1}^{n}(a^{cli}{}_{t,f} * s_t))}}$ Where $1 \le f \le k$
ii. Calculate $x^{FS}{}_n = \frac{1}{1+e^{-(b_2+\sum_{n=1}^{k}(b^{cli}{}_{f,n} * w^{cli}{}_f)}}$ where $1 \le n \le g$
iii. Calculate error $F^{FS} = \frac{1}{2}\sum_{n=1}^{g}(\beta^{FS}{}_n - x^{FS}{}_n)^2$
7. Stop

5. Results

In order to evaluate the performance of the proposed SED-WFML model, a MATLAB simulation was conducted using a dataset that contained 70% of the samples for training and 30% for validation. To assess the effectiveness of the model, various statistical parameters were considered, including accuracy, misclassification rate (MCR), selectivity, recall, precision, false positive rate, false omission rate (FOR), false discovery rate (FDR), and F1 score (Khan et al., 2020).

$$Accuraccy = \frac{\frac{o_{si}}{l_{si}} + \frac{o_{sk}}{l_{sk}}}{\frac{o_{si}}{l_{si}} + \frac{\sum_{j=1}^{n}(o_{sj,j\neq i}}{l_{sj}} + \frac{o_{sk}}{l_{sk}} + \frac{\sum_{l=1}^{n}(o_{sl,l\neq k}}{l_{sk}}}, where\ i/j/k/l = 1,2,3,\ldots,n \quad (39)$$

$$Miss\ rate = \frac{\frac{\sum_{l=1}^{n}(o_{sl,l\neq k}}{l_{sk}}}{\frac{\sum_{l=1}^{n}(o_{sl,l\neq k}}{l_{sk}} + \frac{o_{si}}{l_{si}}}, where\ i/k/l = 1,2,3,\ldots,n \quad (40)$$

$$\text{True positive rate/Recall} = \frac{\frac{o_{si}}{l_{si}}}{\frac{o_{si}}{l_{si}} + \frac{\sum_{l=1}^{n}(o_{sl,l\neq k}}{l_{sk}}}, \text{where } i/k/l = 1,2,3,\ldots,n \quad (41)$$

$$True\ negative\ rate/Selectivity = \frac{\frac{o_{sk}}{l_{sk}}}{\frac{o_{sk}}{l_{sk}} + \frac{\sum_{j=1}^{n}(o_{sj,j\neq i}}{l_{sj}}}, where\ j/k = 1,2,3,\ldots,n \quad (42)$$

$$Precision = \frac{\frac{o_{si}}{l_{si}}}{\frac{o_{si}}{l_{si}} + \frac{\sum_{j=1}^{n}(o_{sj,j\neq i}}{l_{sj}}}, where\ i/j = 1,2,3,\ldots,n \quad (43)$$

$$\text{False Ommission Rate} = \frac{\frac{\sum_{l=1}^{n}(o_{sl,l\neq k}}{l_{sk}}}{\frac{\sum_{l=1}^{n}(o_{sl,l\neq k}}{l_{sk}} + \frac{o_{sk}}{l_{sk}}}, \text{where } k/l = 1,2,3,\ldots,n \quad (44)$$

$$\text{False Discovery Rate} = \frac{\frac{\sum_{j=1}^{n} (o_{sj,j\neq i}}{l_{sk}}}{\frac{o_{si}}{l_{si}} + \frac{\sum_{j=1}^{n} (o_{sj,j\neq i}}{l_{sj}}}, \text{where } i/j = 1,2,3,\ldots,n \quad (45)$$

$$F_1 \text{ Score} = 2 \times \text{Precision} \times \frac{\text{Recall}}{\text{Precision}+\text{Recall}}, \text{where } \frac{i}{j} = 1,2,3,\ldots,n \quad (46)$$

(Hu et al., 2021)

Table 4 presents a Semantic error detection model that uses machine learning techniques during the training phase, specifically ANN-LM. The training dataset consisted of 139 samples, with 78 positive samples and 61 negative samples. During the evaluation, the model predicted 74 samples as positive, indicating the presence of SE, while incorrectly predicting four samples as negative, indicating no SE. For the negative samples, which indicate no SE, the model predicted 59 samples correctly as negative, but two samples were imprecisely predicted as positive, representing the presence of SE.

Table 4: Training of proposed semantic error model using the ML technique (ANN-LM)

Proposed model training			
	Total samples (139)	Output	
Input	Expected output	Predicted positive	Predicted negative
	78 Positive	TP 74	FN 4
	61 Negative	FP 2	TN 59

Table 5 displays a Semantic error detection model that employs machine learning techniques, specifically ANN-LM, during the validation phase. The validation dataset included 58 samples, with 28 positive samples and 30 negative samples. The evaluation results show that the model accurately predicted 26 positive samples, indicating the presence of Semantic Error, but incorrectly predicted 02 samples as negative, indicating no SE. For the negative samples, which represent no traffic routing, the model predicted 29 samples correctly as negative, indicating no SE, but imprecisely predicted 01 samples as positive, demonstrating the occurrence of SE.

Table 5: Validation of the proposed semantic error model using the ML technique (ANN-LM)

Proposed model validation			
	Total samples (58)	Output	
Input	Expected output	Predicted positive	Predicted negative
	30 Positive	TP 28	FN 2
	28 Negative	FP 1	TN 27

Table 6 presents a Semantic error detection model that uses machine learning techniques during the training phase, specifically ANN. The training dataset consisted of 139 samples, with 78 positive samples and 61 negative samples. During the evaluation, the model predicted 73 samples as positive, indicating the presence of SE, while incorrectly predicting five samples as negative, indicating no SE. For the negative samples, which indicate no SE, the model predicted 57 samples correctly as negative, but four samples were inaccurately predicted as positive, indicating the presence (Ali et al., 2023) of SE.

Table 6: Training of proposed semantic error model using the ML technique (ANN-BR)

Proposed model training			
	Total samples (139)	Output	
Input	Expected output	Predicted positive	Predicted negative
	78 Positive	TP 71	FN 7
	61 Negative	FP 4	TN 57

Table 7 displays a Semantic error detection model that employs ML techniques, specifically BR, during the validation phase. The validation dataset included 58 samples, with 28 positive samples and 30 negative samples. The evaluation results show that the model accurately predicted 26 positive samples, indicating the presence of SE, but incorrectly predicted 02 samples as negative, indicating no SE. For the negative samples, which represent no traffic routing, the model predicted 28 samples correctly as negative, indicating no SE, but inaccurately predicted 02 samples as positive, indicating the presence of SE.

Table 7: Validation of the proposed semantic error model using the ML technique (ANN-BR)

Proposed model validation			
	Total samples (58)	Output	
Input	Expected output	Predicted positive	Predicted negative
	26 Positive	TP 24	FN 2
	32 Negative	FP 4	TN 28

Table 8 and Fig. 3 present the performance of the proposed Semantic Error Detection model using machine learning techniques in both the training and validation phases, using both the LM and BR approaches. The results display that the proposed model using the ANN-LM approach achieved an accuracy of 95.6%, 93.5%, 93.4%, 4.4%, and 94.08% in the training phase, and 94.8%, 92.8%, 93.3%, 5.2%, and 92.8% in the validation phase, in terms of sensitivity (True Positive Rate), specificity (True Negative Rate), miss rate (False Negative Rate), and precision (Positive Predictive Value), respectively. Furthermore, the proposed model achieved a False Positive Rate (FPR) of 6.557, 14.609, 14.593, and 91.9%, and a Likelihood Positive Ratio (LR+ve), Likelihood Negative Ratio (LR-ve), and Negative Predictive Value (NPV) of 6.665, 13.071, 13.140, and 93.3% in the training and validation phases, respectively.

This specifies that the proposed model applying the BR approach delivers 92.09%, 96.1%, 96.7%, 7.91%, and 97.49% within the training and provides 89.6%, 92.81%, 96.67%, 10.4%, and 96.3% within the validation, in terms of accuracy, TPR expressed as sensitivity, TNR expressed as specificity, FNR expressed as miss rate, and PPV expressed as precision, respectively. In addition, more statistical measures of the proposed model are providing 3.278, 25.448, 25.448, and 95.2% within the training, and 3.333, 13.071, 13.606, and 93.5% within the validation in terms of fall-out, Likelihood LR+ve, Likelihood LR-ve, and NPV, respectively.

Table 8: Performance assessment of the proposed semantic error detection model using ML techniques in training and validation using various statistical methods

		Accuracy (%)	Sensitivity TPR (%)	Specificity TNR (%)	Miss-Rate FNR (%)	Fall-out FPR (%)	LR +ve	LR -ve	PPV (Precision) (%)	NPV (%)
ANN-LM	Training	95.6	93.5	93.4	4.4	6.557	14.609	14.593	94.8	91.9
	Validation	94.8	92.8	93.3	5.2	6.665	13.071	13.140	92.8	93.3
ANN-BR	Training	92.09	96.1	96.7	7.91	3.278	25.448	25.448	97.49	95.2
	Validation	89.6	92.81	96.67	10.4	3.333	13.071	13.606	96.3	93.5

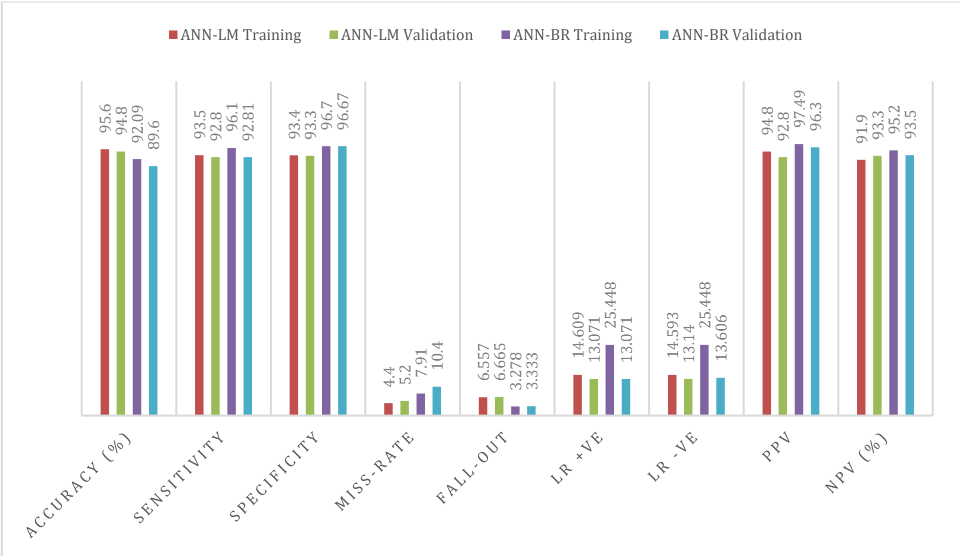

Fig. 3: Statistical parameters of proposed model SED-WFL

As demonstrated in Fig. 4, the model presented in this study achieved a remarkable overall accuracy of 95.60% with a minimal misclassification rate (MCR) of only 4.40%. However, future studies could look into using different machine learning techniques to improve the model's performance. On the server side, for example, support vector machine (SVM), deep extreme ML, and particle swarm optimization can be used to increase accuracy.

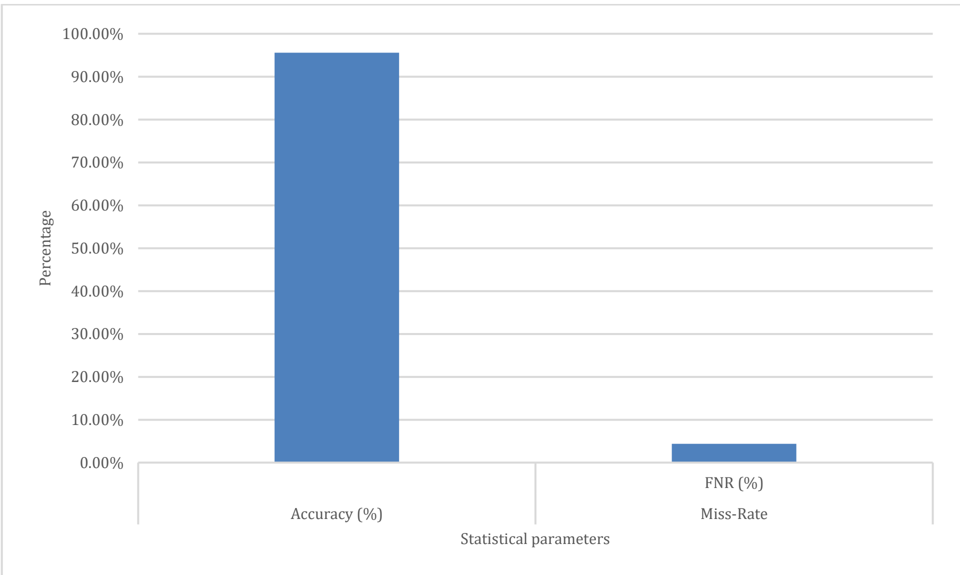

Fig. 4: The total accuracy and MCR of the proposed SEDWFML model for semantic error detection

Table 9 compares the current study to earlier research on the same topic. The results show that the suggested SED-WFML model's accuracy is compatible with prior research findings. Table 9 gives a brief introduction to several error detection models, their corresponding approaches, and the claimed accuracies, enabling them to evaluate and assess the merits of various error detection strategies.

Table 9: Comparison of the previous studies with the proposed model.

Authors	Error detection model	Techniques	Accuracy
Gutierrez et al. (2017)	Semantic error detection	Information extraction	94.7%
Jianbin et al. (2021)	Grammar error detection	Machine learning	89%
Lanzhi et al. (2022)	Automatic error detection	Deep neural network	92.5%
Proposed model	Semantic error detection	Weighted federated machine learning	95.60%

6. Conclusions

The increasing need for automatic error detection technology for English text, specifically in detecting semantic errors, which is a challenging and essential task in English language processing. The study proposes a Semantic Error detection System empowered with Weighted Federated ML (SED-WFML), which overcomes the limitations of traditional error detection systems that require user-provided input configurations such as rules or statistical parameters. The SED-WFML model utilizes a web ontology for a knowledge domain relevant to the natural language text document, creating a link between the classes and characteristics of the ontology and the document's semantic correctness. The proposed model's matching algorithm detects semantically correct and erroneous sentences in a document, enabling a range of applications such as automated document verification and translation. The accuracy of the suggested model during the training and validation phases is high using ANN-LM and ANN-BR techniques, with a detection accuracy of 95.6%, 94.8%, 92.08%, and 89.6%, respectively, surpassing previous techniques. The overall accuracy of the Semantic Error Detection System empowered with Weighted Federated Machine Learning (SED-WFML) is 95.6%, and the Miss Classification rate is 4.40%. The results show that the Semantic Error Detection System, which is empowered with Weighted Federated Machine Learning (SED-WFML), is better than the previous studies, which overcomes the limitations of traditional error detection systems. The model's accuracy during the training and validation phases is promising, highlighting its potential for various natural language processing applications.

Compliance with ethical standards

Conflict of interest

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

Abad G, Picek S, Ramírez-Durán VJ, and Urbieta A (2021). On the security and privacy in federated learning. ArXiv Preprint ArXiv:2112.05423. https://doi.org/10.48550/arXiv.2112.05423

Ahmad G, Khan MA, Abbas S, Athar A, Khan BS, and Aslam MS (2019). Automated diagnosis of hepatitis b using multilayer Mamdani fuzzy inference system. Journal of Healthcare Engineering, 2019: 6361318. https://doi.org/10.1155/2019/6361318 **PMid:30867895 PMCid:PMC6379845**

Akmandor AO, Ortiz J, Manotas I, Ko B, and Jha NK (2020). SECRET: Semantically enhanced classification of real-world tasks. IEEE Transactions on Computers, 70(3): 440-456. https://doi.org/10.1109/TC.2020.2989642

Alhaidari F, Almotiri SH, Al Ghamdi MA, Khan MA, Rehman A, Abbas S, and Rahman AU (2021). Intelligent software-defined network for cognitive routing optimization using deep extreme learning machine approach. Computers, Materials and Continua, 67(1): 1269-1285. https://doi.org/10.32604/cmc.2021.013303

Ali L, Alnawayseh SE, Salahat M, Ghazal TM, Tomh MA, and Mago B (2023). AI-based intelligent model to predict epidemics using machine learning technique. Intelligent Automation and Soft Computing, 36(1): 1095-1104. https://doi.org/10.32604/iasc.2023.031335

Bibi R, Saeed Y, Zeb A, Ghazal TM, Rahman T, Said RA, and Khan MA (2021). Edge AI-based automated detection and classification of road anomalies in VANET using deep learning. Computational Intelligence and Neuroscience, 2021: 6262194. https://doi.org/10.1155/2021/6262194 **PMid:34630550 PMCid:PMC8494564**

Chen S, Chen Z, Sun H, and Su Y (2023). Error detection for text-to-SQL semantic parsing. ArXiv Preprint ArXiv:2305.13683. https://doi.org/10.48550/arXiv.2305.13683

Deng L and Liu Y (2018). A joint introduction to natural language processing and to deep learning. In: Deng L and Liu Y (Eds.), Deep learning in natural language processing: 1-22. Springer, Singapore, Singapore. https://doi.org/10.1007/978-981-10-5209-5

Fu Y, Liang K, and Xu J (2023). MLog: Mogrifier LSTM-based Log anomaly detection approach using semantic representation. IEEE Transactions on Services Computing, 16(5): 3537-3549. https://doi.org/10.1109/TSC.2023.3289488

Gutierrez F, Dou D, de Silva N, and Fickas S (2017). Online reasoning for semantic error detection in text. Journal on Data Semantics, 6: 139-153. https://doi.org/10.1007/s13740-017-0079-6

Hu Y, Ge H, Wang H, and Wang D (2021). Spectral efficiency of network-assisted full-duplex for cell-free massive MIMO system under pilot contamination. IEEE Access, 9: 110826-110841. https://doi.org/10.1109/ACCESS.2021.3100491

Kamal M and Himel AS (2023). Redefining modern marketing: An analysis of AI and NLP's influence on consumer engagement, strategy, and beyond. Eigenpub Review of Science and Technology, 7(1): 203-223.

Khan F, Khan MA, Abbas S, Athar A, Siddiqui SY, Khan AH, and Hussain M (2020). Cloud-based breast cancer prediction empowered with soft computing approaches. Journal of Healthcare Engineering, 2020: 8017496. https://doi.org/10.1155/2020/8017496 **PMid:32509260 PMCid:PMC7254089**

Li X, Gu Y, Dvornek N, Staib LH, Ventola P, and Duncan JS (2020). Multi-site fMRI analysis using privacy-preserving federated learning and domain adaptation: ABIDE results. Medical Image Analysis, 65: 101765. https://doi.org/10.1016/j.media.2020.101765 **PMid:32679533 PMCid:PMC7569477**

McCarthy LM, Kalinyak-Fliszar M, Kohen F, and Martin N (2017). Effects of semantic context on access to words of low imageability in deep-phonological dysphasia: A treatment case study. Aphasiology, 31(5): 542-562. https://doi.org/10.1080/02687038.2016.1208803 **PMid:28659653 PMCid:PMC5484078**

McKinnon ET, Fridriksson J, Basilakos A, Hickok G, Hillis AE, Spampinato MV, Gleichgerrcht E, Rorden C, Jensen JH, Helpern JA, and Bonilha L (2018). Types of naming errors in chronic post-stroke aphasia are dissociated by dual stream axonal loss. Scientific Reports, 8(1): 14352. https://doi.org/10.1038/s41598-018-32457-4 **PMid:30254222 PMCid:PMC6156587**

Oueida S, Kotb Y, Aloqaily M, Jararweh Y, and Baker T (2018). An edge computing based smart healthcare framework for resource management. Sensors, 18(12): 4307. https://doi.org/10.3390/s18124307 **PMid:30563267 PMCid:PMC6308405**

Rehman A, Athar A, Khan MA, Abbas S, Fatima A, and Saeed A (2020). Modelling, simulation, and optimization of diabetes type II prediction using deep extreme learning machine. Journal of Ambient Intelligence and Smart Environments, 12(2): 125-138. https://doi.org/10.3233/AIS-200554

Siddiqui SY, Khan MA, Abbas S, and Khan F (2022). Smart occupancy detection for road traffic parking using deep extreme learning machine. Journal of King Saud University-Computer and Information Sciences, 34(3): 727-733. https://doi.org/10.1016/j.jksuci.2020.01.016

Tabassum N, Ditta A, Alyas T, Abbas S, Alquhayz H, Mian NA, and Khan MA (2021). Prediction of cloud ranking in a hyperconverged cloud ecosystem using machine learning. Computers, Materials and Continua, 67(3): 3129-3141. https://doi.org/10.32604/cmc.2021.014729

Tong H, Yang Z, Wang S, Hu Y, Semiari O, Saad W, and Yin C (2021). Federated learning for audio semantic communication. Frontiers in Communications and Networks, 2: 734402. https://doi.org/10.3389/frcmn.2021.734402

Yang C and Huang C (2023). Natural language processing (NLP) in aviation safety: Systematic review of research and outlook into the future. Aerospace, 10(7): 600. https://doi.org/10.3390/aerospace10070600

Zribi CBO (2023). “Easy” meta-embedding for detecting and correcting semantic errors in Arabic documents. Multimedia Tools and Applications, 82: 21161–21175. https://doi.org/10.1007/s11042-023-14553-4