International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Internet Service Classification Using Swarm-Intelligent K-Nearest Neighbour Algorithm

Harshita Kaushik[1], Shambhu Bhardwaj[2], Sovers Singh Bisht[3], Dr. Kalaiarasan C.[4]

Submitted:20/04/2023 **Revised:**11/06/2023 **Accepted:**22/06/2023

Abstract*:* Effective ways for identifying and organizing this data are now essential due to the Internet's explosive growth and the growing amount of data produced every day. We provide a novel Dragonfly Optimised K-Nearest Neighbour (DF-KNN) algorithm for classifying internet data in this work. The DF-KNN method combines the KNN classifier and DF, a swarm-intelligent optimization algorithm drawn from dragonfly swarms as its primary source of inspiration. By using the DF to find the KNN algorithm's ideal parameter values, categorization accuracy, and efficiency are improved. We ran tests on an actual internet database to measure how well the suggested strategy performed. We contrasted the DF-KNN algorithm's classification outcomes with those of more established methods. The results of the experiments show that the DF-KNN technique performs superior than the conventional KNN algorithm and classifies internet data with more accuracy and efficiency.

Keywords*: Internet data, classification, Dragonfly Optimised K-Nearest Neighbour (DF KNN), swarm-intelligent*

1. Introduction

Internet connection becomes a vital component of the infrastructure. The majority of infrastructure system applications can be accessed over an internet connection. The main goal of IoT is to create a sophisticated community era that consistently anticipates client needs and operates from the point of view [1]. A computer theory known as the Internet of Things (IoT) imagines a time when commonplace physical things will be linked to the Internet and will be able to know about other devices. Information identification, intelligent management, and location tracking are the fundamental purposes of the Internet of Things. From a business perspective, they fit into the general service method, the vertical application method, and the field's common platform model. Big data is becoming significant in the sciences, governments, and businesses as a consequence of this technological revolution [2]. Big Data is a data set that, with present technology, is challenging to collect, store, filter, exchange, examine, and display. Big Data is being applied in healthcare to anticipate epidemics, diagnose diseases, promote quality of life, and prevent avoidable deaths. Big data can assist patients in making the best selection in a timely manner. Network-connected IoT devices exchange information with one another, building up a huge database. To increase the effectiveness of IoT operations, modern methods like data mining and machine learning are integrated with conventional techniques. Forecasting truth mining is the inverse of "data mining," as defined. According to the definition of "data mining," data is quickly and comprehensively examined, producing a vast amount of statistics. Machine learning techniques were developed, which primarily tapped into network behaviors and traffic flow statistics [3].

Numerous trained machine learning algorithms are used for traffic categorization, such as Naive Bayes (NB), SVMs, Bayesian networks, k-nearest neighbor (k-NN), C4.5 decision trees, and neural networks. Internet traffic can be classified using machine learning, which is utilized in a variety of industries, including assistive robotics, recommender systems, and the development and execution of UAVs. While machine learning can be helpful for classifying Internet traffic when packet encryption is used, ISPs or mobile carriers must gather application packets and user data for the training stage. The current study explores this issue and suggests a new machine-learning technique for traffic classification based on the statistical characteristics of packets. The concept is to extract statistical features from packets that are traveling over the network and cluster them based on commonalities [4]. The unconstrained learning algorithm K-means is used to do this. Traffic flow categorization based on the characteristics of the flow is a non-intrusive methodology used in the last few years. In this technique, statistical features of traffic flows are derived and compared to

Assistant Professor, Department of Computer Science & Engineering, Vivekananda Global University, Jaipur, India, Email Id: harshita.kaushik@vgu.ac.in
Associate Professor, College of Computing Science and Information Technology, Teerthanker Mahaveer University, Moradabad, Uttar Pradesh, India, Email id: shambhu.bharadwaj@gmail.com
Assistant Professor & Dy. HoD, Department of Data Science (DS), Noida Institute of Engineering and Technology, Greater Noida, Uttar Pradesh, India, Email id: soverssingh@niet.co.in
Associate Dean, Department of Computer Science and Engineering, Presidency University, Bangalore, India, Email Id- kalaiarasan@presidencyuniversity.in

previously-learned models to determine the generating application. Examining the communications that a host deals in and comparing them to behavior signatures associated with particular application servers are a method for identifying the applications that cause network traffic. This method of traffic classification relies heavily on the topological position of the observer and functions well when the observer can observe both sides of the flow under inspection [5].

We present a new Dragonfly Optimised K-Nearest Neighbour (DF-KNN) algorithm is the combination of the KNN classifier with DF, a swarm-intelligent optimization method that takes concepts mostly from dragonfly swarms for categorizing internet data. We tested the suggested technique on a real online database to see how well it worked.

The remainder of this paper is arranged as follows: Part 2-related work, part 3- metrology, Part 4- results, and Part 5-conclusion.

2. Related Works

Internet connection becomes a vital component of the infrastructure. The majority of infrastructure system applications can be accessed over an internet connection. The main goal of IoT is to create a sophisticated community era that consistently anticipates client needs and operates from the point of view [1]. A computer theory known as the Internet of Things (IoT) imagines a time when commonplace physical things will be linked to the Internet and will be able to know about other devices. Information identification, intelligent management, and location tracking are the fundamental purposes of the Internet of Things. From a business perspective, they fit into the general service method, the vertical application method, and the field's common platform model. Big data is becoming significant in the sciences, governments, and businesses as a consequence of this technological revolution [2]. Big Data is a data set that, with present technology, is challenging to collect, store, filter, exchange, examine, and display. Big Data is being applied in healthcare to anticipate epidemics, diagnose diseases, promote quality of life, and prevent avoidable deaths. Big data can assist patients in making the best selection in a timely manner. Network-connected IoT devices exchange information with one another, building up a huge database. To increase the effectiveness of IoT operations, modern methods like data mining and machine learning are integrated with conventional techniques. Forecasting truth mining is the inverse of "data mining," as defined. According to the definition of "data mining," data is quickly and comprehensively examined, producing a vast amount of statistics. Machine learning techniques were developed, which primarily tapped into network behaviors and traffic flow statistics [3].

Numerous trained machine learning algorithms are used for traffic categorization, such as Naive Bayes (NB), SVMs, Bayesian networks, k-nearest neighbor (k-NN), C4.5 decision trees, and neural networks. Internet traffic can be classified using machine learning, which is utilized in a variety of industries, including assistive robotics, recommender systems, and the development and execution of UAVs. While machine learning can be helpful for classifying Internet traffic when packet encryption is used, ISPs or mobile carriers must gather application packets and user data for the training stage. The current study explores this issue and suggests a new machine-learning technique for traffic classification based on the statistical characteristics of packets. The concept is to extract statistical features from packets that are traveling over the network and cluster them based on commonalities [4]. The unconstrained learning algorithm K-means is used to do this. Traffic flow categorization based on the characteristics of the flow is a non-intrusive methodology used in the last few years. In this technique, statistical features of traffic flows are derived and compared to previously-learned models to determine the generating application. Examining the communications that a host deals in and comparing them to behavior signatures associated with particular application servers are a method for identifying the applications that cause network traffic. This method of traffic classification relies heavily on the topological position of the observer and functions well when the observer can observe both sides of the flow under inspection [5].

We present a new Dragonfly Optimised K-Nearest Neighbour (DF-KNN) algorithm is the combination of the KNN classifier with DF, a swarm-intelligent optimization method that takes concepts mostly from dragonfly swarms for categorizing internet data. We tested the suggested technique on a real online database to see how well it worked.

The remainder of this paper is arranged as follows: Part 2-related work, part 3- metrology, Part 4- results, and Part 5-conclusion.

3. Method

3.1 Dataset

We built up a small experimental network with about 150 hosts to generate the simulated traffic, in contrast to the typical dataset acquisition, which is simply labeled by the payload identity or by port characteristics. Allow each host to simultaneously run the relevant programs (such as HTTP, SMTP, POP3, FTP, and P2P). We collect the traces throughout various time periods and add them to the database. Due to the predetermined programs that operate on the host, it is simple to identify and categorize the traffic flow according to the IP address. Since each application's behavior must be realistic, the gathered data set closely resembles actual Internet traffic. Initially, we sample 1500 instances employing 5-fold cross-validation.

Data pre-processing: Min-max normalization is a normalization approach that creates a balance of value comparisons between data prior to and after the operation by executing linear transformations on the original information. The following formula can be used in this manner.

$$Y_{new} = \frac{Y - \min(Y)}{\max(Y) - \min(Y)} \qquad (1)$$

Y_{new} = New value obtained from the outcomes of normalization

Y = Old value

$\max(Y)$ = The highest value in the dataset

$\min(Y)$ = The lowest value in the dataset

3.2 Dragonfly (DF) Algorithm

The DA is inspired by dragonflies' distinct and better swarming activity. The activities of DF swarms include both travel and hunting. Assume that there are M dragonflies in the world. Eq.(2) specifies the location of N DF.

$$U_i = (u_i^1, u_i^v, \ldots, u_i^N) \qquad (2)$$

While $j = 1,2,3, \ldots., M$and M represents the count of search agentsu_i^v and indicates the location of the i^{th} DF in the v^{th}searchable dimension.

According to the initial location data, the fitness function, which is randomly generated between the largest and smallest bounds of parameters, is estimated. For the variables D (separation weight), v (cohesion), p (alignment), a (food), and l (opponent factors), each DF has a unique starting value. Components for upgrading the velocity and location of dragonflies are computed using Equations (3) to (5).

$$D_i = -\sum_{j=1}^{N} u - u_i \qquad (3)$$

$$P_i = \frac{\sum_{j=1}^{N} W_i}{N} \qquad (4)$$

$$E_i = \frac{j=1}{N} - u \qquad (5)$$

$W_i and\ U_i$ represents the person's velocity and location, respectively. N represents a group of nearby persons and $W_i$represents the individual's current location. Equations (6) and (7) show how to measure A_i (Attraction towards a food source) and L_i (Distraction from Opponents).

$$L_i = U^+ - U \qquad (6)$$

$$L_i = U^- + U \qquad (7)$$

While U represents the player's current position, U^- represents the enemy source and U^+represents the food source. We utilize the Euclidean distance between all N dragonflies to calculate how far away they are. Equation (8) is used to calculate distance, indicated by q_i.

$$q_{ji} = \sqrt{\sum_{j=1}^{v}(u_i, f - u_i, f)^2} \qquad (8)$$

The DF's position will be modified using Equation (9) which is similar to the PSO location formulation. This will be done using Equation (10), which is comparable to the PSO velocity formulation.

$$\Delta U_{g+1} = (dD_i + pP_i + vV_i + aA_i + lA_i) + y\Delta U_g \qquad (9)$$

$$U_{g+1} = U_g + \Delta U_{g+1} \qquad (10)$$

The DA location will be adjusted in the surrounding area by applying the Levy Flight equation, which may be found in Equation (11). When there is no DF radius, this will occur. This makes dragonfly behavior even more arbitrary and unpredictable while simultaneously enhancing their ability to search globally.

$$U_{g+1} = U_g + levy\ (v) U_g \qquad (11)$$

The revised velocities and positions are then used to compute the fitness function.

3.3 K-NN Algorithm

The The K-nearest neighbors algorithm, often known as the k-NN algorithm, divides data using learning data (train datasets) obtained from the k-nearest neighbors. Where k represents the count of closest neighbors. K-nearest neighbors classify data generated from learning in multidimensional space. This area is separated into sections corresponding to the learning data standards. Each learning data point c is displayed in multidimensional space. The newly categorized information is then projected onto a multidimensional space containing c learning data points. The categorization procedure begins by locating the closest c point to the new c (nearest neighbor). The Euclidean distance formula, which can be determined using the Equation that follows, is a usual method for locating the nearest neighbor. Algorithm 1 shows the algorithm for K-NN.

$$dist\ (b, o) = \sqrt{(b_1 - o_1)^2 + (b_2 - o_2)^2 + \cdots + (b_m - o_m)^2}$$
(12)

Algorithm 1: Algorithm for K-NN

Step 1: Load the dataset having the labels and relevant features for the labeled data points.

Step 2: Normalise the characteristics if necessary to make sure they have an equal scale. This step is especially essential when working with characteristics that have varying units or ranges.

Step 3: Choose a value for k, which defines the number of neighbors considered for categorization or prediction.

Step 4: Choose a suitable distance metric to assess the similarity of data points. Euclidean distance, Manhattan distance, and cosine similarity are all common distance measurements.

Step 5: Separate learning and examining portions of the dataset. The training set will be used to train the model, and the test set will be used to evaluate its effectiveness.

Step 6: Apply the provided distance metric to every data point in the set of exercises to determine the distances to all other data points. Based on the estimated distances, select the k nearest neighbors.

Step 7: For every value in the test set, determine the distances to every data point in the training set. Find the dominant class (for classification) or average value (for regression) among the k nearest neighbors. As the intended label, assign this class or value to the test data point.

Step 8: Compare the predicted labels to the actual labels in the data set to evaluate the model's performance. For categorization tasks, standard evaluation criteria include accuracy, precision, recall, and F1 score. For regression tasks, measurements like mean squared error (MSE) or mean absolute error (MAE) are typically used.

Step 9: The test with different values of k and the distance measure to find the ideal combination of the two.

Step 10: Once you're fulfilled with the performance of the model, you may use it to make predictions on previously unseen data points.

3.4 Hybrid dragonfly optimized KNN (DF-KNN)

Create a collection of dragonflies at random in the search area, with each dragonfly standing for a possible response (feature vector) to the classification of the ISP's problem. Utilize each dragonfly's individual positions as feature vectors in the KNN algorithm to determine each one's fitness for ISP categorization. To prepare and evaluate the KNN classifier, use a labeled dataset with historical information about ISPs and their respective classes (such as "Good," "Average," and "Poor"). The fitness can be assessed using any appropriate statistic, such as classification accuracy, precision, recall, F1-score, etc. In order to maximize fitness (classification operations), we will organize the dragonflies according to their fitness values. The place of the dragonfly with the greatest fitness should be updated as the global best position. For each dragonfly, modify the local ideal spot. Choose the k-nearest neighbors for each dragonfly based on the Euclidean distance among their locations. Modify the current dragonfly by calculating the average location of the chosen neighbors. Determine the separation between each dragonfly's current location and its ideal position. To make sure that the upgraded locations of the dragonflies stay within the specified search area, use boundary limitations. The final requirement is satisfied (for example, convergence or the highest number of repetitions). The best dragonfly should be chosen as the final selection, and its location should be used as the characteristic vector for categorizing ISPs.

4. Result

The existing approaches like Cost-Sensitive Back Propagation Neural Network (CSBPNN), Deep Neural Network (DNN). Parameters like “accuracy, recall, precision, and f1- measure”. Where “true positives (S_1), false positives (T_1), true negatives (S_2), and false negatives (T_2)”.

When referring to data analysis or forecasts, accuracy is the measure of correctness or precision in relation to a standard or expected value as in Eq.13. Figure 1 Shows the accuracy comparison for existing CSBPNN and DNN is 80.5% and 78.3%, and our proposed approach DF-KNN has 97.6%. It shows that our proposed method is more accurate than the existing methods.

$$Accuracy = \frac{S_2+S_1}{S_2+S_1+T_2+T_1}$$
(13)

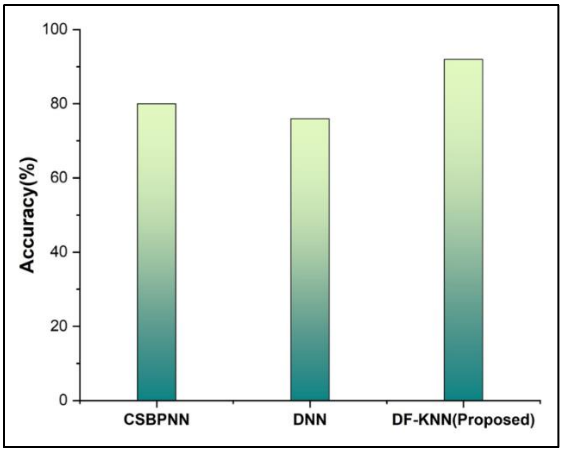

Fig 1: Accuracy

Precision is the degree of exactness or accuracy in a task or evaluation. It measures the level of accuracy and reliability of the obtained results as in Eq.14. Figure 2 depicts a precision comparison of the existing and proposed approaches. The existing approaches CSBPNN and DNN have 80.2% and 89.7%, and our proposed method DF-KNN has 98.4%. It indicates that our proposed technique has more precision.

$$Precision = \frac{S_1}{S_1+T_1} \qquad (14)$$

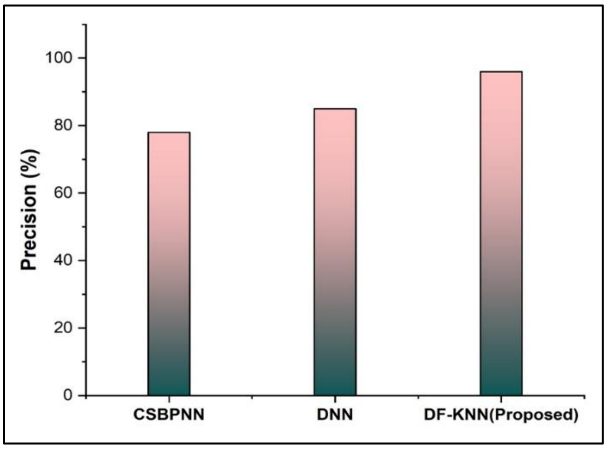

Fig 2: Precision

Recall is a measurement of accuracy or the capacity to locate specific data or instances in a dataset or memory. It measures how effective retrieval is at a given task. This provides the percentage that is used in the calculation of recall as in Eq.15. Figure 3 displays the performance of recall for CSBPNN and DNN in 89.5%, 78.3% compared with the proposed approach DF-KNN is 99.1%.

$$Recall = \frac{S_1}{S_1+T_2} \qquad (15)$$

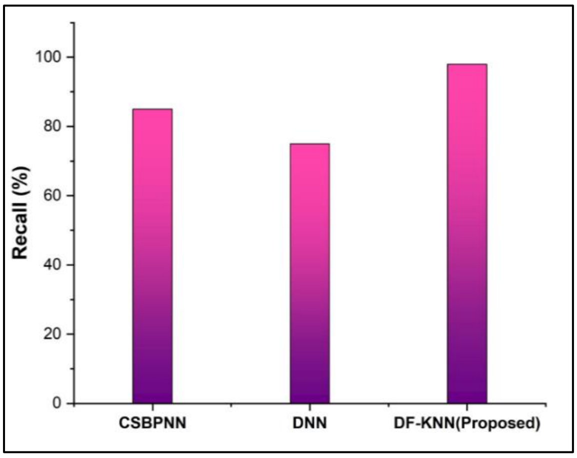

Fig 3: Recall

The mean percentage of recall and precision is the F1-measure as in Eq.16. The f1-measure comparison for existing and proposed approaches is shown in Figure 4. The existing approaches CSBPNN and DNN have values of 75.4% and 89.8%, and the proposed technique DF-KNN has a value of 97.5%.

$$F1 - measure = \frac{2\times(Presicion\times Recall)}{Presicion+Recall} \qquad (16)$$

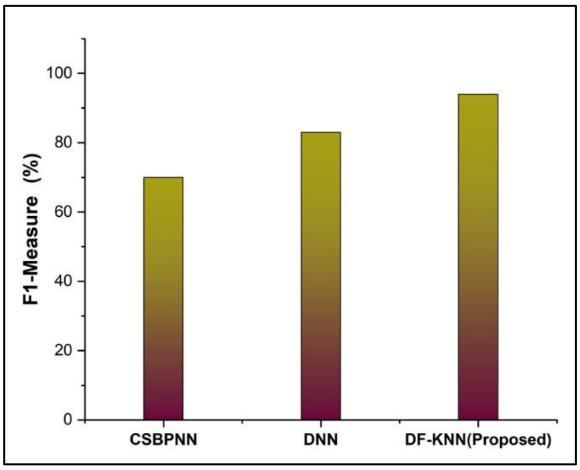

Fig 4: F1-measure

5. Conclusion

In this study, we proposed a novel way to enhance wind speed prediction. It is a hybrid of the Backpropagation Neural Network (BPNN) and the Satin Bowerbird optimization (SBO) algorithm. Since the BPNN is capable of modeling nonlinear interactions, it is ideally suited for simulating variations in wind speeds. The findings demonstrated performance of our proposed approach obtained from various parameters in terms of AE, AE, MSE, and MAPE is 0.15, 0.33, 0.21, 21.93, and 97%. Utilizing more recent versions of the SBO could potentially lead to the discovery of a solution that is more efficient. The utilization of additional comparative algorithms and the application of this idea can be used to investigate further for potential future works..

References

[1] Purohit, L. and Kumar, S., 2019. Web services in the internet of things and smart cities: A case study on classification techniques. IEEE Consumer Electronics Magazine, 8(2), pp.39-43.

[2] Ahmadi, H., Arji, G., Shahmoradi, L., Safdari, R., Nilashi, M. and Alizadeh, M., 2019. The application of internet of things in healthcare: a systematic literature review and classification. Universal Access in the Information Society, 18, pp.837-869.

[3] Guo, J., Chen, R. and Tsai, J.J., 2017. A survey of trust computation models for service management in internet of things systems. Computer Communications, 97, pp.1-14.

[4] Paukstadt, U., Strobel, G. and Eicker, S., 2019. Understanding services in the era of the internet of things: a smart service taxonomy.

[5] Shao, S., Tunc, C., Al-Shawi, A. and Hariri, S., 2019, November. One-class classification with deep autoencoder neural networks for author verification in

internet relay chat. In 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA) (pp. 1-8). IEEE.

[6] Umair, M.B., Iqbal, Z., Bilal, M., Almohamad, T.A., Nebhen, J. and Mehmood, R.M.,

2021. An efficient internet traffic classification system using deep learning for IoT. arXiv preprint arXiv:2107.12193.

[7] Sang, J., Pang, S., Zha, Y. and Yang, F., 2019. Design and analysis of a general vector space model for data classification in Internet of Things. EURASIP Journal on Wireless Communications and Networking, 2019, pp.1-10.

[8] Huang, J., Zhu, L., Liang, Q., Fan, B. and Li, S., 2018. Efficient classification of distribution-based data for Internet of Things. IEEE Access, 6, pp.69279-69287.

[9] Le, T.T.H., Oktian, Y.E. and Kim, H., 2022. XGBoost for imbalanced multiclass classification-based industrial internet of things intrusion detection systems. Sustainability, 14(14), p.8707.

[10] Roy, S., Shapira, T. and Shavitt, Y., 2022. Fast and lean encrypted Internet traffic classification. Computer Communications, 186, pp.166-173.

[11] Wahla, A.H., Chen, L., Wang, Y., Chen, R. and Wu, F., 2019. Automatic wireless signal classification in multimedia Internet of Things: An adaptive boosting enabled approach. IEEE Access, 7, pp.160334-160344.

[12] Ducange, P., Mannarà, G., Marcelloni, F., Pecori, R. and Vecchio, M., 2017, July. A novel approach for internet traffic classification based on multi-objective evolutionary fuzzy classifiers. In 2017 IEEE international conference on fuzzy systems (FUZZ-IEEE) (pp. 1-6). IEEE.

[13] Manju, N., Harish, B.S. and Prajwal, V., 2019. Ensemble feature selection and classification of internet traffic using XGBoost classifier. International Journal of Computer Network and Information Security, 11(7), p.37.

[14] Mattei, E., Dalton, C., Draganov, A., Marin, B., Tinston, M., Harrison, G., Smarrelli, B. and Harlacher, M., 2019, November. Feature learning for enhanced security in the Internet of Things. In 2019 IEEE Global Conference on Signal and Information Processing (GlobalSIP) (pp. 1-5). IEEE.

[15] Biswas, S.K., Devi, D. and Chakraborty, M., 2018. A hybrid case based reasoning model for classification in internet of things (iot) environment. Journal of Organizational and End User Computing (JOEUC), 30(4), pp.104-122.

[16] Thanagaraju, V. ., & Nagarajan, K. K. . (2023). A Detailed Analysis of Air Pollution Monitoring System and Prediction Using Machine Learning Methods. International Journal on Recent and Innovation Trends in Computing and Communication, 11(2s), 51–58. https://doi.org/10.17762/ijritcc.v11i2s.6028

[17] Sharma, M. K. (2021). An Automated Ensemble-Based Classification Model for The Early Diagnosis of The Cancer Using a Machine Learning Approach. Machine Learning Applications in Engineering Education and Management, 1(1), 01–06. Retrieved from http://yashikajournals.com/index.php/mlaeem/article/view/1