Handwritten Mathematical Symbols Classification Using WEKA

Sakshi, Shivani Gautam, Chetan Sharma, and Vinay Kukreja

Abstract Machine learning tools have been extensively used for the prediction and classification of mathematical symbols, formulas, and expressions. Although the recognition and classification in handwritten text and scripts have reached a point of commensurate maturity, yet the recognition work related to mathematical symbols and expressions has remained a stimulating and challenging task throughout. So, in this work, we have used Weka, a machine learning tool, for the classification of handwritten mathematical symbols. The current literature witnesses a limited amount of research works for classification for handwritten mathematical text using this tool. We have endeavored to explore the potential classification rate of handwritten symbols while analyzing the performance by comparing the results obtained by several clustering, classification, regression, and other machine learning algorithms. The comparative analysis of 15 such algorithms has been performed, and the dataset used for the experiment incorporates selective handwritten math symbols. The experimental results output accuracy of 72.9215% using the Decision Table algorithm.

Keywords Handwritten · Symbols · Characters · Machine learning · Decision table · J48 · Bayes algorithm · Classification · Weka

Sakshi · V. Kukreja
Institute of Engineering and Technology, Chitkara University, Punjab, India
e-mail: sakshi@chitkara.edu.in

V. Kukreja
e-mail: vinay.kukreja@chitkara.edu.in

S. Gautam
School of Computer Applications, Chitkara University, Himachal Pradesh, India
e-mail: shivani.gautam@chitkarauniversity.edu.in

C. Sharma (✉)
Chitkara University, Himachal Pradesh, India
e-mail: chetan.sharma@chitkarauniversity.edu.in

A. Choudhary et al. (eds.), *Applications of Artificial Intelligence and Machine Learning*,
Lecture Notes in Electrical Engineering 778,
https://doi.org/10.1007/978-981-16-3067-5_4

1 Introduction

The emerging trend noticed in the continuous evolution of the tremendous scientific community has witnessed the generation and usage of a massive amount of scientific documents. It is a remarkable and prodigious achievement, as the scientific and research zones are consistently and constantly generating supplementary knowledge, which is represented in a different form. And this knowledge and developed information need to be represented in the form of scientific and mathematical notations. Nowadays, it is crucial to have this information digitalized, to make feasible searching, and to access a set of relevant documents. Being an important part of the scientific and engineering literature, the classification and recognition of the mathematical expressions have become an exciting and stimulating research area of the pattern recognition with unlimited real-world implications [1]. However, it is for analyzing or accessing research works or be it for information retrieval for other scientific and educational purposes. Also, the recognition of images in the field of artificial intelligence is prevalent among researchers for many years, as this can be used in various areas. Machine learning tools have gained commensurable popularity in the last decade. Thus, today individuals or organizations are using the machine-dependent application in every sector to perform their tasks. Thus, the interest of this research work is inclined towards implementing and exploring the potential of machine learning tools like WEKA. Moreover, recognizing and classifying a handwritten math symbol is an arduous classification problem, requiring real-time identification for all the symbols containing an input as well as the complex 2D relationships between symbols and subexpressions [2]. So, the researchers need to access math information recurrently while working with computational systems. Mathematical character or symbols recognition is yet challenging, and the emerging field of research [3].

In the educational area, the recognition of math symbols and characters is incorporated with a marking system to evaluate the marks or scores of mathematical questions and exercises automatically. However, learning notations like LATEX and MathML, or using graphic editors is an essential requirement for introducing math notation into an electronic device. The process of recognition of math symbols and characters aims at building recognition systems and models that can automatically understand mathematical content provided by humans in the form of printed or handwritten characters or symbols. Optical character recognition mainly focused on the machine printed output, where the number of font styles can be used to write any character or symbol. In that inconsistency between the character font, style and attributes are small, whereas when a character is written impersonal, their variability is relatively high. This variability and distortion make recognition very challenging [4]. The writing style of any individual varies from person to person, and this variety of writing style of community creates distortion and variation in the dataset [5]. Identification of character or symbol from where refined features can be extracted from the data is one of the primary task to make recognition rate more accurate, and to locate such region from the data; various sampling techniques are used in the

field of pattern recognition [6]. So, it becomes more critical to extract stable and reliable features to enhance system performance. In future, character and symbol recognition in the field of mathematics might serve as a foundation stone to start the paperless strategy by digitizing and processing the saved hard copy documents. Handwritten data is vague by nature as they don't contain the sharp and perfectly straight lines so, the goal of recognition is to extract the essential information from any raw image data [7]. Tokas and Bhadu [8] Illustrate structural, statistical, and global transformation classification methods of feature extraction techniques. The analytical approach is used to select the data, and it uses the information related to the statistical distribution of pixels in the image. Neves et al. [9] Conduct study on the NIST SD19 dataset using SVM based offline handwritten digit recognition system and concluded that SVM outperforms in their experiment. Perwej and Chaturvedi [10] Convert the handwritten dataset into electronic data and used the NN approach to make machines capable of recognizing the dataset.

2 Literature Review and Related Work

Study is conducted on single character recognition of math symbols by the use of Support vector machines. They focus on improving the classification of InftyReader, which is optical character recognition (OCR) software used to recognize text, scientific figures, and math symbols. SVM is used to improve the classification of InftyReader. InftyReader confuses in the classification of pairs letter, so the author firstly compares the performance of SVM kernels and features of pair letters. Then they illustrate the multi-class classification with SVM, utilizing the ranking of alternatives within InftyReader's confusion clusters. The proposed technique decreases its misrecognition rate by 41% [11].

Author considered machine printed and handwritten document images from three Indic scripts (Bangla, Devnagri, and Roman) for their study. They applied the OCR technique on printed and handwritten document images. The author has taken 277 document images from both the methods in three mentioned scripts. They used a Multilayer perceptron classifier with 5-fold cross-validation, an average accuracy rate of 98.75% for Bangla, 100% for Devnagari, and 100% for Roman scripts are obtained. When they combined all three scripts, the average accuracy rate of 98.9% is obtained [12].

Author proposed solution to locate the mathematical formula in any PDF document using machine learning and heuristic rule methods. They recommended four new features in their study for preprocessing and post-processing techniques. LibSVM-R-D, LibSVM-R, LibSVM-P, Logistic regression, MLP, J48, Random Forest, BayesNet, PART, Bagging-RF, AdaBoost-RF learning algorithms are taken by the author to experiment on Ground-truth dataset which is now publically available. The author concludes that they increased overall accuracy through the proposed system by 11.52 and 10.65% compared to the previous studies [13].

Study was conducted on Devnagri script, which is widely used in many languages. The author has taken 60 handwritten Devanagari symbols from different writers; out of 60 characters, 50 are letters, and 10 are digits. 60 sample of each character has been taken so in total; 3600 samples are taken for feature extraction. The author performed classification through Multilayer perceptron, K-Nearest Neighbour, Naive Bayes classifier, and Classification tree on the selected dataset in WEKA. They compare the performance of the chosen classifier and found that a multilayer perceptron performs well among all classifiers and achieves a 98.9% accuracy rate from the multilayer perceptron classifier [14].

Study is done to discover the software tool which is capable of identifying the character or digits. The different writing skills of an individual make this field challenging. The author used the MNIST image dataset to perform classification. Various steps are followed to achieve rating on the considered dataset, and firstly dataset is chosen, then the input image is converted to a grayscale image. Finally, all images are converted to binary format. WEKA tool is selected to experiment, and as WEKA accepts CSV and ARFF format, so all processed images are converted to a comma-separated file for training and testing purposes. They used Random forest, Decision tree, and Hoeffding Trees machine learning algorithms to perform classification on the selected dataset. On the base of different parameters, the author results in that the Hoeffding tree is the best classification technique for considered datasets out of all classification techniques [15].

Author conducted this study to recognize the handwritten digit, which is a significant problem in the field of pattern recognition. Researchers are working in this field to develop an efficient algorithm for identifying the handwritten numbers as input from the user through digital devices. They used a collection of 3689 digit datasets, which is making available by the Austrian Research Institute for Artificial Intelligence, Austria. Out of 3689, 1893 samples are taken to train the system, and 1796 samples are taken for testing on the train system. J48, Multilayer Perceptron, Random forest, SVM, Bayes classifiers, Random Tree machine learning algorithms were considered by the author to conduct their study to recognize the digits using WEKA. The author found a 90.37% accuracy rate in Multilayer Perceptron, out of all considered machine learning algorithms. So, it results in that Multilayer Perceptron algorithms perform significantly well to recognize the digits [4].

Author illustrates the approach for offline recognition of handwritten mathematical symbols. The study included symbol recognition for over more than 300 classes. The objective of designing the classifier to recognize these 300 symbols. Firstly they describe the issues related to segmentation using SLIC and study experiment results shows different accuracy rate for different algorithms. They achieved 84% for the kNN classifier, 57% for HOG, 53% for LBP. The author modified 87 classes using the LeNet and gained a 90% accuracy rate. SqueezeNet is used to pre-trained the 101 classes and result in a 90% accuracy rate [16].

Study is conducted on handwritten offline Urdu character recognition using different machine learning techniques. They created their dataset with 9600 instances from the various native writers. The author used edge histogram descriptor, ColourLayout, and Binary Pyramid to extract the feature from the considered dataset.

They applied different machine learning algorithms like MLP, SVM, SMO, and simple logistic using the WEKA 3.8 tool an achieved 98.60% of accuracy through SVM [17].

3 Proposed Work

- **Data Collection**: In any recognition system, the first step is to collect the data, and data can be obtained in any form. Data sets are created by taking handwritten documents from different users. Further, these documents are scanned through digital types of equipment and develop a scanned image for extracting feature purposes.
- **Registration**: Images collected are mostly in the RGB scale, so after receiving the data, these images have to be converted into a grayscale format using proper threshold values to avoid the loss of information and after that these images are converted to binary format so that feature extraction can be done efficiently.
- **Preprocessing**: Preprocessing is an essential factor to be considered when we take any data for recognition as we know any data which is raw by nature contain some noise factor, vague and inconsistent data that is required to remove to achieve better performance. Preprocessing is also used to enhance the signal of the binary image data. Images are preprocessed in different matrix values like 5×7, 14×10, 32×32, and many more to get better recognition.
- **Feature Extraction**: In feature extraction processed image is represented in feature vector and the main goal through this is to extract a set of feature which helps the system to maximize the recognition rate. In handwritten documents, this is very challenging to get a useful feature due to the high degree of variability. It can be solved by dividing the processed image is into the $N \times M$ zone to obtain local characteristics rather than global characteristics. The feature can be extracted through statistical, structural, and global transformations methods.
- **Classification**: Once the features are extracted from the images, then we can classify them through different classification techniques like Hoeffding tree, random tree, Bayes Classifier, J48, Neural Networks, Random Forest, Support Vector Machines (SVM), etc. No classification method is considered to be the best classification method as the use of classifiers depends upon factors like training dataset, test dataset, number of features, and many more (Fig. 1).

4 Implementation

Dataset to conduct this study is downloaded from https://www.kaggle.com/guru001/hasyv2, which is publically available for the experiment. Dataset contain 369 symbols with 168,236 png images for symbols, each 32px $\times$ 32px from the different writers. We had taken 35 mathematical symbols shown in Fig. 2 out of 369 symbols with 3650

Fig. 1 Stages in character recognition system [14]

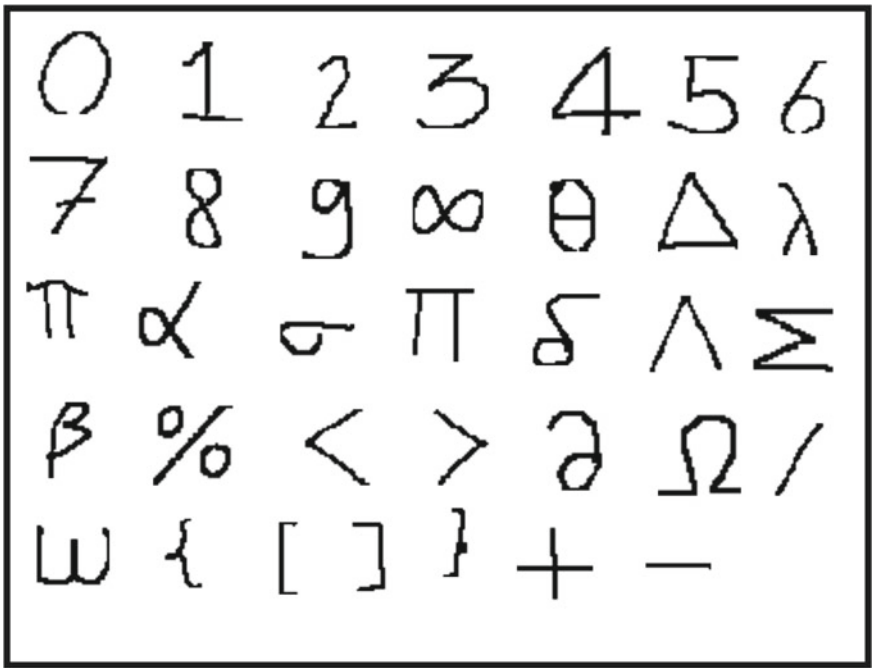

Fig. 2 Dataset handwritten mathematical symbols

png images each of 32px × 32px to perform our experiment on different classification methods. Firstly, in WEKA, we preprocessed dataset images included in the .csv file using a simple color histogram filter and change other attributes to nominal values for classification.

WEKA 3.6 machine learning tool written in Java and developed at the University of Waikato is used to conduct our experiment as this tool provides us with different classifiers to examine the performance. WEKA is used to evaluate different data mining tasks like pre-processing, classification, regression and many more. WEKA accepts .csv and .arff file format and the chosen dataset have already created the required data in the mentioned format. To determine and inspect the performance of different classification methods mentioned in Tables 1 and 2 comparison has

Table 1 Experiment results based on correctly classified instances and model time

Algorithms	Correctly classified instances (%)	Incorrectly classified instances (%)	Time taken to build the model (s)	Kappa statistic
J48	72.36	27.64	1	0.454
Hoeffding tree	67.45	32.55	6.71	0
Decision stump	67.45	32.55	0.16	0
Random tree	72.68	27.32	0.84	0.461
REPTree	67.45	32.55	0.09	0
Bayesnet	72.76	27.24	0.44	0.46
Naïve Bayes	72.68	27.32	0.09	0.462
Multinomial Naive Bayes	67.45	32.55	0.01	0
Decision table	72.93	27.07	15.05	0.456
Jrip	68.90	31.10	8.39	0.083
One R	6.69	93.31	0.08	0.0009
PART	71.80	28.20	3.08	0.444
Zero R	67.45	32.55	0	0
Input map classifier	67.45	32.55	0.01	0
Kstar	72.76	27.24	18.93	0.449

been performed. The algorithms named Naïve Bayes, Naïve Bayes Mutlinomial are Bayesian classifiers which belongs to the family of simple "probabilistic classifiers" based on Bayes' theorem and Decision Stump, Hoeffding Tree, Hoeffding Option Tree, Hoeffding Adaptive Tree are the Decision tree classifiers, which is popular supervised machine learning classification algorithm. Authors use the same method or procedure as per the WEKA tool suggestions and dataset is considered as instances and features in the data. To understand the experiment results we divided the results into two subparts for easier analysis and evaluation. First part of the results are shown in Table 1 which contain the correctly, incorrectly classified instances, time taken to build model, and kappa statistic. In the second part Table 2 contain the different errors during the simulation in WEKA. We run the different classifiers on the considered dataset in WEKA, and their results are shown in Tables 1 and 2.

5 Conclusion and Future Scope

In this paper, we have demonstrated several machine learning-based algorithms for identifying the classification rate of the handwritten mathematical symbols for determining and comparing the classification rate of these considered algorithms. More

Table 2 Experiment results based on different errors

Algorithms	Mean absolute error	Root mean squared error	Relative absolute error (%)	Root relative squared error (%)
J48	0.01	0.05	0.60	0.85
Hoeffding tree	0.01	0.06	1.00	1.00
Decision stump	0.01	0.06	0.91	0.98
Random tree	0.01	0.05	0.61	0.85
REPTree	0.01	0.06	0.96	1.00
Bayesnet	0.01	0.05	0.72	0.84
Naïve Bayes	0.01	0.05	0.69	0.84
Multinomial naive bayes	0.01	0.06	1.00	1.00
Decision Table	0.01	0.07	1.24	1.02
Jrip	0.01	0.06	0.92	0.98
One R	0.01	0.12	1.63	1.86
PART	0.01	0.06	0.60	0.88
Zero R	0.01	0.06	1.00	1.00
Input map classifier	0.01	0.06	1.00	1.00
Kstar	0.01	0.05	0.63	0.81

importantly, it becomes crucial to analyze the performance based on several metrics and compare the results meticulously. This paper compared 15 classifiers used for the recognition of different handwritten mathematical symbols. All considered algorithms performed well on the considered dataset except the one R algorithm. The accuracy rate for each algorithm is mentioned in Table 1, and we conclude that the Decision table presents exceptionally well with an accuracy rate of 72.9251% out of all algorithms. We propose to extend this work in the future by using different preprocessing methods and considering an extended and modified dataset with other exclusive handwritten mathematical symbols and diverse machine learning and deep learning algorithms.

References

1. Afshan N, Afshar Alam M, Ali Mehdi S (2017) An analysis of mathematical expression recognition techniques. Int J Adv Res Comput Sci 8(5):2021–2026
2. MacLean S, Labahn G (2015) A Bayesian model for recognizing handwritten mathematical expressions. Pattern Recognit 48(8):2433–2445
3. Liu C-L, Nakashima K, Sako H, Fujisawa H (2003) Handwritten digit recognition: benchmarking of state-of-the-art techniques. Pattern Recognit 36(10):2271–2285

4. Shamim SM, Miah MBA, Angona Sarker MR, Al Jobair A (2018) Handwritten digit recognition using machine learning algorithms. Glob J Comput Sci Technol 18(1)
5. Plamondon R, Srihari SN (2000) Online and off-line handwriting recognition: a comprehensive survey. IEEE Trans Pattern Anal Mach Intell 22(1):63–84
6. Das N, Sarkar R, Basu S, Kundu M, Nasipuri M, Basu DK (2012) A genetic algorithm based region sampling for selection of local features in handwritten digit recognition application. Appl Soft Comput 12(5):1592–1606
7. AlKhateeb JH, Pauplin O, Ren J, Jiang J (2011) Performance of hidden Markov model and dynamic Bayesian network classifiers on handwritten Arabic word recognition. knowl Syst 24(5):680–688
8. Tokas R, Bhadu A (2012) A comparative analysis of feature extraction techniques for handwritten character recognition. Int J Adv Technol Eng Res 2(4):215–219
9. Neves RFP, Lopes Filho ANG, Mello CAB, Zanchettin C (2011) A SVM based off-line handwritten digit recognizer. In: IEEE international conference on systems, man, and cybernetics, pp 510–515
10. Perwej Y, Chaturvedi A (2012) Machine recognition of hand written characters using neural networks. arXiv Prepr.arXiv1205.3964
11. Malon C, Uchida S, Suzuki M (2008) Mathematical symbol recognition with support vector machines. Pattern Recognit Lett 29(9):1326–1332
12. Obaidullah SM, Das N, Roy K (2014) An approach to distinguish machine printed and handwritten text from document images for indic script. Int J Appl Eng Res 9(20):4670–4675
13. Lin X, Gao L, Tang Z, Baker J, Sorge V (2014) Mathematical formula identification and performance evaluation in PDF documents. Int J Doc Anal Recognit 17(3):239–255
14. Shelke SV, Chandwadkar DM (2016) Handwritten Devnagri character recognition. GRD J Glob Res Dev J Eng 1(5):67–70
15. Lavanya K, Bajaj S, Tank P, Jain S (2017) Handwritten digit recognition using hoeffding tree, decision tree and random forests—a comparative approach. In: International conference on computational intelligence in data science (ICCIDS), pp 1–6
16. Nazemi A, Tavakolian N, Fitzpatrick D, Suen CY et al (2019) Offline handwritten mathematical symbol recognition utilising deep learning. arXiv Prepr.arXiv1910.07395
17. Jameel M, Shuja M, Mittal S (2020) Improved handwritten offline urdu characters recognition system using machine learning techniques. Int J Adv Sci Technol 29(6):4865–4978