Int. j. inf. tecnol. (October 2021) 13(5):1993–1999
https://doi.org/10.1007/s41870-021-00723-w

ORIGINAL RESEARCH

Handwritten Marathi numeral recognition using stacked ensemble neural network

Deepak T. Mane[1] · Rushikesh Tapdiya[2] · Swati V. Shinde[3]

Received: 9 August 2020 / Accepted: 8 June 2021 / Published online: 17 July 2021

Abstract Pattern Recognition is the method of mapping the inputs to their respective target classes based on features of data. In this paper a stacked ensemble meta-learning approach for customized convolutional neural network is proposed for Marathi handwritten numeral recognition. Stacked ensemble merges the pre-trained base pipe lines to create a multi-head meta-learning classifier that outputs the final target labels. It overpowers the average ensemble because the weighted and maximum contribution of each pipeline is taken in this approach. The stacked ensemble meta-learning classifier proves to be efficient because the base pipelines, which are already acquainted with output desirable results, are concatenated, instead of averaging, to achieve maximum efficiency. Performance evaluation and analysis have been done on Marathi handwritten numeral dataset, and the experiment results are better than the existing proposed systems.

Keywords Marathi numeral · Pattern recognition · Deep learning · Convolutional neural networks · Stacked ensemble

✉ Deepak T. Mane
dtmane@gmail.com

Rushikesh Tapdiya
rushikesh.tapdiya47@gmail.com

Swati V. Shinde
swaatii.shinde@gmail.com

[1] JSPM's Rajarshi Shahu College of Engineering, Pune, Mararashtra 411033, India

[2] Pune Institute of Computer Technology, Pune, Maharashtra 411043, India

[3] Pimpri Chinchwad College of Engineering, Pune, Maharashtra 411044, India

1 Introduction

Pattern recognition is one of the rudimentary requirement of many applications based on deep learning. Pattern recognition is an important aspect of many application domains like optical character recognition, video surveillance, face recognition, medical diagnosis, human computer recognition and access control systems. Pattern recognition requires highly adroit pipelines that can tackle not only real time data but also it can recognize it accurately.

Handwritten Recognition is one of the basic entity in Pattern Recognition. Handwritten recognition is a fundamental requirement for OCR and Document Analysis and Extraction. Marathi language is one of the most spoken regional language in India and is the mother tongue of many Maharashtraians. Many researchers [1–3] already studied on the recognition of isolated handwritten characters. Marathi numerals classification is not a frequently touched topic. The accuracy obtained for our pipeline is about 97.91% which is much higher than the other existing pipelines.

The data is collected by scanning more than 80,000 handwritten numerals. The Marathi numerals follow non-linear nature as compared to the linear nature of English numerals. Various nomenclatures for writing the same numerals make digit recognition task as a challenging one. We have tried to cover all styles of writing in our data-set. The Marathi numerals data-set is not easily available on the Internet like the English numerals data-set (MNIST). The key highlights of the paper are

- Stacked ensemble for convolutional neural network is proposed for numeral recognition.

- The meta learning classifier has better performance than the existing systems.

In this paper, Sect. 2 throws insight into the work already done in this field, Sect. 3 describes about the dataset gathering and pre-processing techniques. Furthermore, Sect. 4 talks in detail about the proposed pipeline and implementation. Section 5 deals with Performance evaluation and analysis and finally, we conclude with Sect. 6 that talks about the future scope and final conclusions drawn.

2 Related work

The handwriting styles, the uniqueness in this jargon has uplifted the uniqueness and essence of Marathi language in Indian sub-continent. The development of handwritten numerals identification system is not an easy task, given the diverse writing styles and the sophisticated nature of curves in writing. Despite this, Dongre and Mankar proposed a solution using statistical discriminate functions and geometric attributes like line,line directions, perimeter, solidity, image area, eccentricity of the numerals for their identification in 2013 [4]. Kim et al. [5] proposed a system that used hybrid features for representation and combined classifier for classification. Acharya et al. [6] devised a handwritten recognition system that made the use of various features in multilevel classifiers. Vasantha et al. [7] implemented pre-processing and post-processing in order to augment accuracy above 99%. Kumar et al. [8] used the morphological features for identifying blobs in numerals with blobs and stem. Singh et al. [9] designed a artificial neural network pipeline that identifies five different types of fonts of the Devanagari script. Rajput and Mali [10] used Fourier Descriptors to describe the shape of quarantined Marathi handwritten numerals and the system was tested using various algorithms. Bhattacharya et al. [11] achieved the accuracy of 92.83% using artificial neural network (ANN) and hidden Markov model (HMM). Srivastava and Gharde [12] used support vector machines (SVM) on the dataset constructed by then automated numeral extraction and segmentation program (ANESP). Moment invariant techniques and affine moment invariant techniques were used extract 18 features that were passes to the SVM, which achieved accuracy about 99.48%. Mane and Kulkarni [13] proposed a customized convolutional neural network (CNN), which achieved accuracy of 94.93%. Mane and Kulkarni mentioned importance of CNN in pattern recognition [14]. Vaidya and Joshi [15] proposed a handwritten numeral identification system using the statistical distributions of the image feature vectors. Hanmandlu et al. [16] proposed a fuzzy model for Hindi characters identification, which achieved an accuracy of 90.65%. Khanale and Chitnis [17] used ANN for the Marathi characters identification,that achieved an accuracy of about 96%. Patil and Sinha [18] proposed basic ANN approach for classification, but the magnitude of their dataset i. e. 150 images is very small as compare to proposed dataset i.e 82,000 images. Duddela et. al. [19] proposed ANN classifier for Devnagri digits using PRTool which achieved 95% accuracy. Many researchers [20–23] used the support vector machine and its extended variations like weighted SVM, SVM-KNN etc for recognition of handwritten characters. Patil et al. [24] presented a recurrent neural network with an LSTM model for recognition of handwritten MARATHI digits which achieved 79% accuracy. Recently, Gupta et al. [25] used supervised deep learning techniques for handwritten digit recognition for eight different scripts which got a maximum 96% result. Additionally, our proposed stack assembler neural network is much more intricate and novel in comparison with ANN and other classifiers. The proposed architecture outperforms their ANN in terms of diversity and accuracy.

3 Dataset-collection and preprocessing

The Marathi handwritten numeral dataset is not available for deep learning applications. Hence, The dataset is manually generated by collecting handwritten samples of Marathi numerals from people of diverse age groups. The individuals were asked to write the numerals from 0 to 9 on a nine region A-4 size paper, which was partitioned into nine fixed-size areas. Representation of extracted and cropped images of collected samples is shown in Fig. 1. All the images are converted into grayscale images, and their dimensions are cropped to 28*28 i.e 784 features per

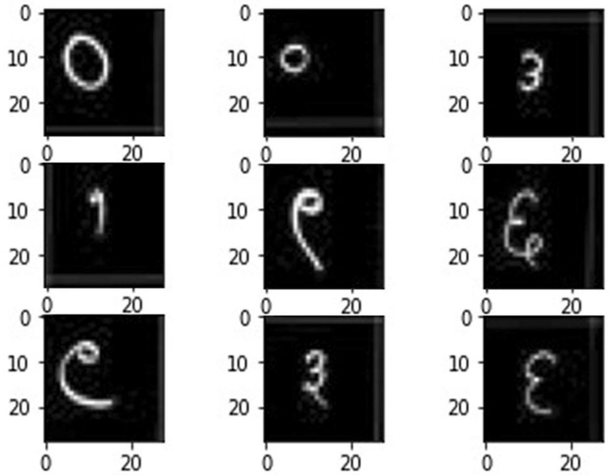

Fig. 1 Extracted and cropped images of collected samples

image. To avoid space congestion, all the features of the image are stored in a CSV file corresponding to their respective image ids. Now, the dataset is augmented using various image data set augmentation techniques like zooming (0.2), shear-range (0.04), Horizontal and Vertical Flips, rotation (8), width shift range (0.2), height shift range (0.2) etc. The images in the dataset after applying dataset augmentation techniques look as shown in Fig. 2.

4 Proposed pipeline and implementation

The proposed pipeline is the stacked ensemble model, whose base pipeline is customized CNN pipeline that is used to identify the handwritten Marathi numerals.

- The CNN pipeline does not make use of the pooling operation. Rather, stride convolutions with larger kernel sizes are applied so that the weights of the filter can be updated during backpropagation.
- The CNN pipeline used here is inspired by the VGG 16 CNN pipeline. The pipeline consists of two normal convolution blocks followed by a stridden convolution block to extract lower level features.

All the convolution layers and dense layers are using the Rectified Linear Unit (ReLu) activation function to introduce non-linearity in the pipeline designed.

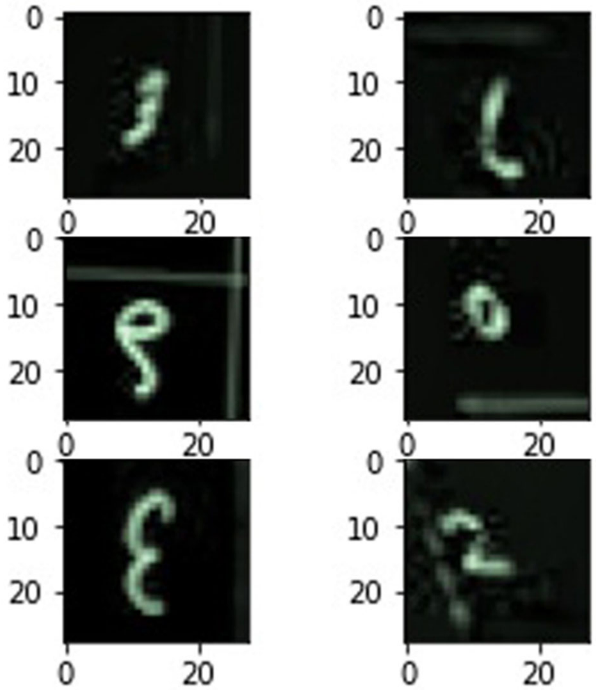

Fig. 2 Images after applying augmentation techniques

$$f(x) = \begin{cases} 0 & : x \leq 0 \\ x & : x \nleq 0 \end{cases} \tag{1}$$

ReLu is most widely used activation function in many CNN pipelines [26]. Softmax Classifier is used in this Pipeline. Softmax classifier is give as follows-

$$P(y = j) = e^{z}_{\mathrm{j}} / \sum_{\mathrm{k}} e^{z}_{\mathrm{k}} \tag{2}$$

where j = 1 to N(no of classes) The function normalizes the output values in the range from 0 to 1 so that the output can be decoded as categorical probability distribution over K classes.

The loss function used here is the categorical loss entropy function which is given by

$$H(p, q) = -\sum_{\mathrm{i}} p_{\mathrm{i}} \log q_{\mathrm{i}} - (1 - p_{\mathrm{i}}) \log (1 - q_{\mathrm{i}}) \tag{3}$$

Where p represents the actual output, and q shows predicted labels. We have proposed a stacking ensemble for customized CNN pipeline designed by us to augment the accuracy of the predictions. In stacking ensemble, all the pre-trained base pipelines (customized CNN pipelines) participating in the average ensemble are integrated or stacked into the multi-head deep learning neural network that makes a prediction based on the outputs given by the base pipeline. The base pipelines are stacked so to create a meta-learning pipeline classifier that combines the outputs of the base models and gives the final output result which is represented in Fig. 3.

Stacked ensemble for CNN makes sure that the all the best possible contributions are taken from the base CNN pipelines as compared to the same or average contribution made by the pipeline in the average ensemble. Moreover, the base pipelines can also be updated or trained again, depending upon the functional requirement and the computation capacity. The Fig. 4 illustrates the architecture diagram of the proposed pipeline. The Base classifier is the base customized CNN Pipeline. The keras API is used to implement the pipeline. The flowchart of the entire training

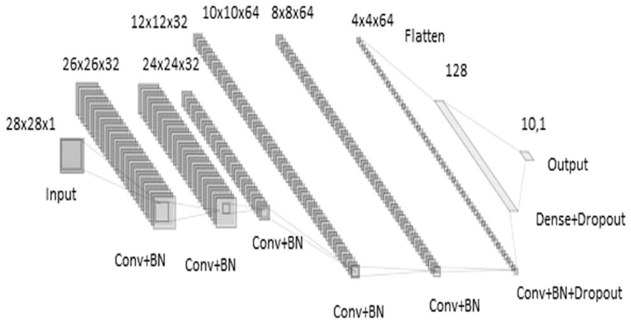

Fig. 3 Base pipeline

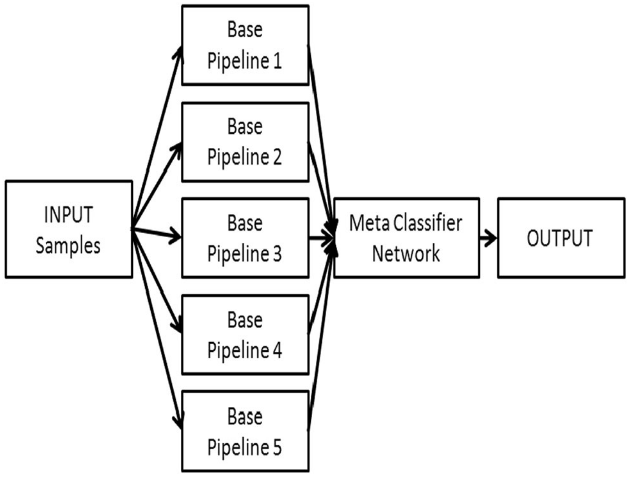

Fig. 4 Architecture diagram of proposed stacked ensemble model

process is represented in Fig. 5. The five base pipelines models are trained. The average ensemble accuracy of all the models is about 97.2%. All the base pipelines are merged to create a meta learning classifier that is contingent on stacked ensemble. Ensembling augments the performance metrics of the CNN Pipeline and is widely used in pattern classification. The batch size used in the implementation is 64. 63,000 samples are used for Training, 7000 samples are used for validation; 11,500 samples are used for testing. For each fold, the training and validation samples are randomly selected to make the model more exhaustive.

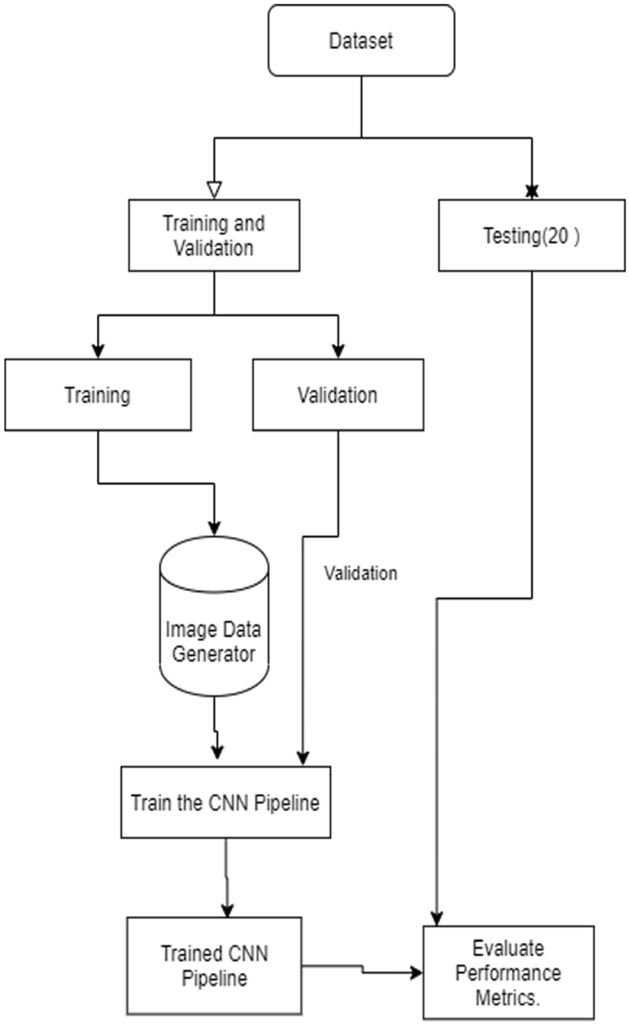

Fig. 5 Training flowchart for base pipeline/model

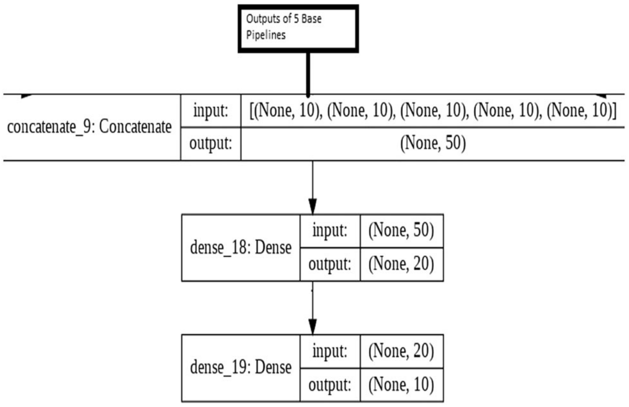

Fig. 6 Stacked ensemble meta learning classifier

The customized convolutional neural networks training algorithm is described using six steps.

- Step I: Initialize all filters and parameters with random values in the CNN pipeline.
- Step II: The pipeline takes an image batch as input and iterates through stages like convolution, flattening, and finally makes an output.
- Step III: Calculate the error using the categorical cross-entropy loss function mentioned above.

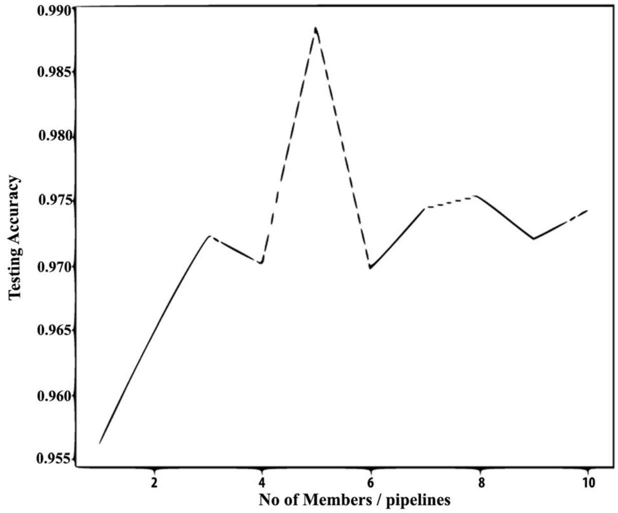

Fig. 7 Accuracy with no of member k-fold validation

Table 1 Comparison of proposed method with exiting systems

Feature extraction method	Technique	Dataset	Performance (%)
Strokes based on directional view [11]	Combination of HMM and MLP	Manually prepared containing 25,535 samples	92.83
Structural analysis of samples [28]	General fuzzy neural network	Manually prepared containing 2000 samples	99.5
Fourier descriptor [10]	Support vector machines	Self made containing 13,000 samples	97.85
Rotational mechanics concepts like density, moments [27]	K nearest neighbours, support vector machines	Self prepared with 3000 (KNN) and 2030 (SVM) samples	96.55 97.89
Invariant moments [8]	FFN, KNN	CPAR 2012 with 35,000 samples	83.4
Feature matching based on histogram [16]	Pixel distribution based on statistics	Self made 6000 samples	94.16
CNN based feature extraction [13]	Customly designed CNN Pipeline	Self created with 80,000 samples	94.93
Customized convolutional neural network	Proposed stack ensemble meta learning classifier	Self-created containing 81,500 samples	97.91

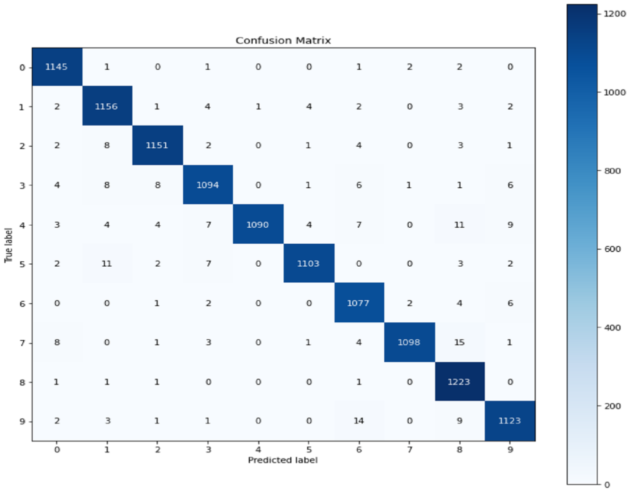

Fig. 8 Confusion matrix

- Step IV: Back propagate the error and update all parameters accordingly.
- Step V: Repeat the above steps for each base pipeline.
- Step VI: Test the base pipeline for the given Testing set.

We have trained ten base pipelines, but keeping in mind the computational constraints, we will use stack only five base pipelines. Now, the meta-learning classifier is created, which takes the stacked outputs of all the base models, and processes it and gives the final output. Learning algorithm for stacked ensemble meta learning classifier is described in Algorithm 1.

The base pipelines were not updated as a part of the training process of meta classifier. The meta-learning classifier was trained on the validation data set for 1 epoch. The no of samples used were 8000. The accuracy achieved was 97.91%. The Fig. 6 shows the pipeline of the meta-learning classifier.

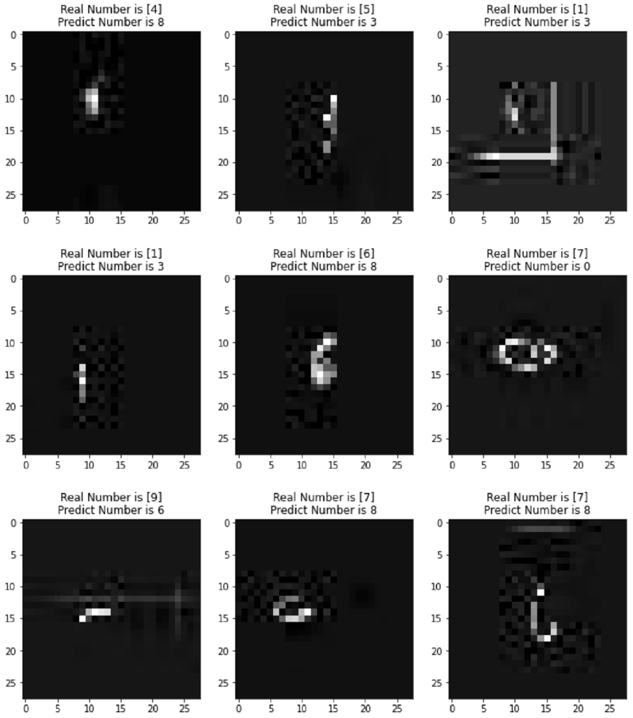

Fig. 9 Mis-classified samples

5 Results and analysis

The approach proposed by Patil in 2012 has the highest accuracy, but their sample size is tiny. Hence, it is tough to say whether their model will outnumber other models in the exhaustive testing strategy. The proposed approach is

Table 2 Classification report

Numeral	Precision	Recall	F1-score	Support
0	0.98	0.99	0.99	1152
1	0.97	0.98	0.98	1175
2	0.98	0.98	0.98	1172
3	0.98	0.97	0.97	1129
4	1.00	0.96	0.98	1139
5	0.99	0.98	0.98	1130
6	0.97	0.99	0.98	1092
7	1.00	0.97	0.98	1131
8	0.96	1.00	0.98	1227
9	0.98	0.97	0.98	1153
Accuracy			0.98	11500
Macro avg.	0.98	0.98	0.98	11,500
Weighted avg.	0.98	0.98	0.98	11,500

examined on 11,500 samples, which is much higher than the sample size used in other methods by the authors. Hence, in terms of both accuracy and Sample size, the proposed approach is optimum. The Fig. 7 shows the plot of accuracy with the count of members used for average ensemble. The comparison of the proposed method with existing systems is represented in Table 1. The average ensemble accuracy is about 97.21%. But, the stacked ensembled meta-learning classifier augmented the accuracy to 97.91%. The classification metrics used in the analysis of the proposed pipeline are described in Fig. 8.

- Confusion matrix: It is a square matrix of size equal to the number of the target classes. Here, the rows represent the actual labels, and columns represent the predicted labels. Each entity represents the number of samples predicted for their labels against their actual labels. For the classifier to be optimum, the diagonal of the matrix should contain maximum numbers, and rest all elements should be zero.
- Accuracy: It represents the percentage of samples whose predicted labels match their actual labels. It is one of the basic and important classification metric.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \tag{4}$$

- Precision: Precision tells us that, out of the population of a particular label, how many sample's predicted labels matched their actual labels.

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

- Recall: It gives the proportion of how many samples from a particular class label were identified correctly.

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

- F1-score: The harmonic mean of precision and recall gives the F1-score. The maximum value of F1-score is 1 and minimum is 0.

$$F1_score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{7}$$

Actual verses predicted testing samples of confusion matrix is shown Fig. 9. The Table 2 shows the classification report of the stack ensemble meta learning classifier and Figure 9 shows some misclassified samples. The samples that are not predicted correctly are either not clear or having vague writing nature as it is clear from the image above. The proposed pipeline fails to predict such handwritten samples accurately.

6 Conclusion

The paper focuses on the Marathi handwritten numeral recognition using stacked ensembles. The pipeline proposed achieved an average accuracy of 97.91%. The stacked ensemble learning approach augments the accuracy of the average ensemble model. However, in some cases, the pipeline does not work as desired. This is because of the complex writing curves, the lower resolution of the scanned images, and some unusual patterns involved in writing. However, pipelines works as expected in most of the different writing styles and patterns.

Optical character recognition with Devanagari scripts faces huge impediments because of the identification of letters and numbers. The proposed approach would contribute a lot for OCR with Devanagari scripts. Additionally, various applications like evaluation of Marathi answer sheets, Marathi sentiment analysis etc could benefit a lot from our approach.

In the upcoming future, the same pipeline could be extended to Marathi alphabets. Additionally, the accuracy could be augmented by increasing more of number of individual classifiers or epochs. Furthermore, dilated customized CNN could be used to decrease computational cost as well as to increase accuracy. The pipeline should be designed in a way to accommodate the curves that our pipeline failed to identify.

References

1. Sharma N, Pal U, Kimura F, Pal S (2009) Recognition of off-line handwritten Devanagari characters using quadratic classifier. In: Proceedings of ICVGIP, vol 31. Springer, 444–457

2. Bhattacharya U, Chaudhuri BB (2006) Handwritten numeral databases of Indian scripts and multistage recognition of mixed numerals. IEEE Trans Pattern Recognit Mach Intell 4338:805–816
3. Bhattacharya U, Shridhar M, Parui SK, Sen PK, Chaudhuri BB (2012) Offline recognition of handwritten Bangla characters: an efficient two-stage approach. Pattern Anal Appl 15:445–458
4. Dongre VJ, Mankar VH (2013) Devnagari handwritten numeral recognition using geometric features and statistical combination classifier. Int J Comput Sci Eng 2:856–863
5. Kim KM, Park JJ, Song YG, Kim IC, Suen CY (2004) Recognition of handwritten numerals using a combined classifier with hybrid features. In: SSPR and SPR. Springer, 992–1000
6. Acharya DU, Subba Reddy NV, Makkithaya K (2008) Multilevel classifiers in recognition of handwritten Kannada numerals. World Acad Sci Eng Technol 42:278–283
7. Vasantha C, Jain R, Patvardhan (2008) Fast and robust scheme for recognition of handwritten Devnagri Numerals. In: National System Conference, IIT Roorkee, pp 1–7
8. Kumar R, Vashishtha A, Agrawal I (2014) Devanagari handwritten numerals recognition based on invariant moments. Int J Comput Sci Manag Stud 14(6):8–11
9. Singh R, Yadav CS, Verma P, Yadav V (2010) Optical character recognition (OCR) for printed Devnagari script using artificial neural network. Int J Comput Sci Commun 1(1):91–95
10. Rajput GG, Mali SM (2010) Fourier descriptor based isolated Marathi handwritten numeral recognition. Int J Comput Appl 7:1–5
11. Bhattacharya U, Parui SK, Shaw B, Bhattacharya K (2006) Neural Combination of ANN and HMM for handwritten Devanagari numeral recognition. Tenth international workshop on frontiers in handwriting recognition. La Baule, France, pp 613–618
12. Srivastava SK, Gharde SS (2010) Support vector machine for handwritten Devanagri numeral recognition. Int J Comput Appl 7:9–14
13. Mane DT, Kulkarni UV (2018) Visualizing and understanding customized convolutional neural network for recognition of handwritten Marathi numerals. Procedia Comput Sci 132:1123–1137
14. Mane DT, Kulkarni UV (2017) A survey on supervised convolutional neural network and its major applications. Int J Rough Sets Data Anal 4(3):71–82
15. Vaidya MV, Joshi YV (2015) Marathi numeral recognition using statistical distribution features. In: International Conference on Information Processing, Pune, India, pp 586–591
16. Hanmandlu M, Murthy OV, Madasu V (2008) Fuzzy model based recognition of handwritten Hindi characters. J Pattern Recognit Res 2:454–461
17. Khanale PB, Chitnis SD (2011) Handwritten Devanagari character recognition using artificial neural network. J Artif Intell 4(1):55–62
18. Patil S, Sinha GR (2012) Real time handwritten Marathi numerals recognition using neural network. Int J Inf Technol Comput Sci 4(12):76–81
19. Prashanth DS, Mehta RVK, Sharma N (2020) Classification of Handwritten Devanagari Number-An analysis of Pattern Recognition Tool using Neural Network and CNN. Procedia Comput Sci 167:2445–2457
20. Kadam AA, Bhalerao MV, Tanurkar MN (2019) Handwritten Marathi compound character reconition. Int J Eng Res Technol 8
21. Chikmurge D, Shriram R (2019) Marathi handwritten character recognition using SVM and KNN classifier. Hybrid Intell Syst HIS 2019:319–327
22. Ramteke S, Gurjar A, Deshmukh DS (2019) A Novel Weighted SVM Classifier Based on SCA for Handwritten Marathi Character Recognition. IETE J Res 1–13. https://doi.org/10.1080/03772063.2019.1623093
23. Mahapatra D, Choudhury C, Karsh RK, (2020) Handwritten character recognition using KNN and SVM based classifier over feature vector from autoencoder. In: Machine learning, image processing, network security and data sciences, MIND, (2020) Communications in computer and information science, vol 1240. Springer, Singapore, pp 304–317
24. Patil Y, Bhilare A (2019) Digits recognition of marathi handwritten script using LSTM neural network. In: Proceedings of the 5th international conference on computing, communication, control and automation, Pune, India, pp 1–4
25. Gupta D, Bag S (2021) CNN-based multilingual handwritten numeral recognition: a fusion-free approach. Expert Syst Appl 165:
26. Yann L, Yoshua G, Hinton G (2008) Deep learning. Nature 521:436–444
27. Mali SM (2012) Moment and density based handwritten Marathi numeral recognition. Indian J Comput Sci Eng 3(5):707–712
28. Patil PM, Sontakke TR (2007) Rotation, scale and translation invariant handwritten Devanagari numeral character recognition using general fuzzy neural network. Pattern Recognit 40:2110–2117