The Scientific Temper (2023) Vol. 14 (3): 774-781
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2023.14.3.32
https://scientifictemper.com/

RESEARCH ARTICLE

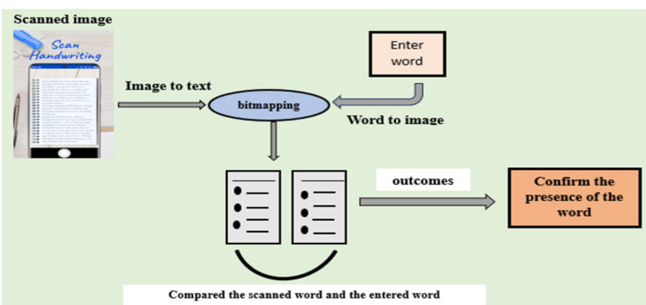
English language analysis using pattern recognition and machine learning

Jhankar Moolchandani, Kulvinder Singh*

Abstract

Pattern identification and classification in complicated systems are difficult. This study uses optical character recognition (OCR) to digitize handwritten data. OCR segments and categorizes characters using online and offline methods for different input sources. Hindi and Bangladeshi categorization results unite linguistic studies. Handwriting recognition systems create editable digital documents from touchscreens, electronic pens, scanners, and photographs. Statistical, structural, neural network and syntactic methods improve online and offline recognition. In "english language analysis using pattern recognition and machine learning," the accuracy of various approaches is examined, showing deep convolution neural networks (DCNN) 98% accuracy in recognizing subtle linguistic patterns. Nave Bayes, a trustworthy language analysis approach, has 96.2% accuracy. Table recognition (TR) algorithms retrieve structured information at 97%. This method outperforms others with 98.4% accuracy. This unique strategy could improve english language analysis using cutting-edge pattern recognition and machine learning techniques.

Keywords: Computer text, Handwriting data, OCR, Pattern recognition, Statistical structure.



Introduction

Pattern recognition and classification are the hardest system components. Categorizing recovered phrases' characters. Character names rebuilt the word. The new field optical character recognition (OCR) digitizes handwritten data. OCR segments handwriting. OCR categorizes most. Scanners and handwritten scripts use offline character recognition, while tablets use OCR. Their findings united investigators' Hindi and Bangladeshi language classification methods. Digitizing printed, scratched, or typed characters requires OCR. Pattern recognition classifies physiological data. It changes fastest. This technique tests speech recognition and understanding on "Notepad" computers. Examples include cloud patterns, facial recognition, handwritten lettering in printed systems, and satellite imagery. This paper proposed a model to check scanned text for English words. This employs a four-dimensional pixel-matching technique.

Touchscreens, pens, scanners, pictures, and paper feed handwriting recognition systems. Editable digital files (Zeng, 2021). The system struggles to classify handwritten words—tilted, cursive, or block. A working model digitizes handwritten formats. It improves human-machine communication (Entezami *et al.,* 2020). Pattern recognition, along with categorization, is the system's hardest. (Paolanti *et al.,* 2020). Digital handwriting recognition methods use a pad and a computerized pen. Keystrokes, velocity, and timing determine writing strokes (Jha, 2019). Digitizing texts permits handwriting editing. OCR reads various handwriting. Online/offline OCR (Rajalakshmi *et al.,* 2019). Tablets use online character recognition, while scanned papers and handwritten scripts are offline (Bartocci *et al.,* 2016). Machines help humans think and function. (Chen *et*

Department of Computer Science, Tantia University, Sri Ganganagar, Rajasthan, India.

***Corresponding Author:** Kulvinder Singh, Department of Computer Science, Tantia University, Sri Ganganagar, Rajasthan, India, E-Mail: Kulvinder.hundal@gmail.com

How to cite this article: Moolchandani, J., Singh, K. (2023). English language analysis using pattern recognition and machine learning. The Scientific Temper, **14**(3): 774-781.

Doi: 10.58414/SCIENTIFICTEMPER.2023.14.3.32

Source of support: Nil

Conflict of interest: None.

Received: 12/07/2023 **Accepted:** 15/08/2023 **Published** : 25/09/2023

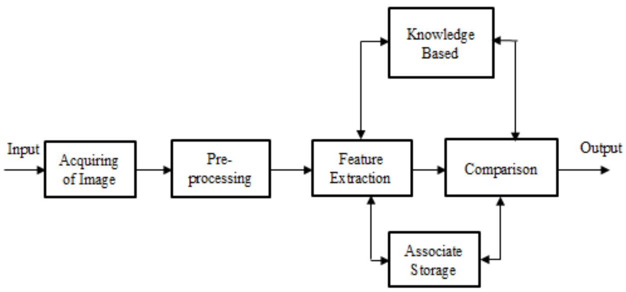


Figure 1: Image processing system (Sun *et al.,* 2019).

al., 2016). People give devices tasks. This must be opened to pattern recognition in optical devices, neural nets, deep learning, ML, and robots (Ande *et al.,* 2021).

Deep learning detects images. It names objects, people, and locations. Picture and facial recognition categorize friends automatically. Facebook auto-suggests friend labels (Milojkovic opsenica *et al.,* 2016) found Facebook friends in photos uploaded by the gadget. Facebook's latest deep face algorithm identifies faces (Bengfort *et al.,* 2018). Figure 1 shows image recognition.

Machines help humans work and boost cognitive abilities. Machines now type, study, recall, and substitute. People are giving devices tasks. Authors should teach computers human abilities to complete their tasks. Pattern recognition in optical devices, neural nets, deep learning, ML, and robotics all benefited from this requirement (Africa *et al.,* 2017). Handwriting can only be learned via experience and cannot be perfected by study or practice. Thus, any two people's handwriting is identical and cannot be copied. Variation is a writer's naturally occurring variance. Fingerprint verification by forensic document specialists is crucial (Rehman *et al.,* 2019). Digital handwriting recognition systems employ a pad and computerized pen to enter data. Keystrokes, velocity, and timing determine writing strokes. Thus, pen motions and writing speeds are displayed. Scanning offline data creates a 2-dimensional matrix (Agrawal *et al.,* 2022).

There is a well-defined structure to this paper. The first section introduces the topic, while the second examines related methods. The study techniques are discussed in section 3, while the tests and results are presented in section 4. Discussion is provided in section 5, and the paper concludes with suggestions for future study in section 6.

Literature Survey

The following study expands on a review of english language analysis from the scanned document using pattern recognition and ML. Several scientists explained their findings, as seen below.

Pre-trained deep convolution neural networks (DCNN) models for handwritten Devanagari alphabet recognition were tested (Aneja *et al.,* 2020). Exploration employs DenseNet, AlexNet, Inception ConvNet, and visual geometry group fixed feature extractors. Vgg 11, 19, 16, and Inception V3 have 15 epochs. Inception V3 averages 16.3 minutes per epoch with 99% accuracy. AlexNet has the quickest epoch (2.2 minutes) and best accuracy (98%). Medical breakthroughs are simply one-way technology influences every part of life (Khanday *et al.,* 2020). Data-driven healthcare AI decision-making is promising. Over 100 countries have contracted COVID-19 in days. Long-term impacts harm everyone. Create a coronavirus detector. AI may soothe the turmoil. Logistic regressions and multinomial Nave Bayes outperformed other ML methods with 96.2% efficiency. Yousef *et al.,* (2020) revealed that unconstrained computer vision text recognition contains several subtasks and challenges. DNN automate and converge, extracting characteristics from raw inputs for optimum performance without subject expertise. Performing cutting-edge OCR, Captcha recognition, and LPR on seven public benchmark datasets. The 2018 International Conference on Frontiers in Handwritten Recognition (ICFHR) recognized the READ dataset architecture for automated text recognition.

OCR is still difficult in uncontrolled natural environments with geometric distortions, complex backgrounds, and several fonts (Namysl *et al.,* 2020). Deep learning, synthetic training data, and data augmentation enable segmentation-free OCR. Synthetic training data employs over 2000 fonts in big text corpora. Tabular data organization was straightforward for Rashid *et al.,* (2018). Table recognition (TR) and document picture retrieval are crucial. Modern OCR algorithms don't recognize table structure for text extraction. It detected non-table and table components on a test dataset with over 97% accuracy after training on a subset of UNLV and UW3 dataset pictures. Avadesh *et al.,* (2018) state that ancient Sanskrit manuscripts include enormous mathematics, science, hindu mythology, and Indian history and peoples. These papers must be available to the public to exchange knowledge. Segmenting images determines pixel intensities. The OCR classifies common compound characters (half-letter combinations) to enhance segmentation. OCR is perfect for scanning unclean Sanskrit manuscripts since it resists image effectiveness, contrast, font style, and font size. Scanned document OCR has been widely explored (Zhao *et al.,* 2017). These photographs feature lighting issues, cluttered backgrounds, and landscape geometry distortions. This research discusses recognizing letters and words in scene photos using convolutional neural networks.

Handwritten character recognition (HCR) is a growing area with applications in various sectors (Ashiquzzaman *et al.,* 2017). Multilingualism has been studied before and after. The author invented Arabic handwriting digit recognition. This model has the best accuracy, 97.4%. Handwritten text recognition (HTR) is a popular computer vision and pattern recognition research topic (Alani *et al.,* 2017). Digitizing handwritten digits is the trickiest part. RBM, CNN, and deep learning create a novel Arabic handwritten digit recognition system. The proposed method was taught and tested using Arabic handwritten digits. Md Fazlul Kader *et al.,* (2012) created a basic, color- and size-invariant english alphabet character recognition system utilizing an artificial neural network. The recommended method works well with numbers and letters when taught and tested alone but badly when processed jointly. Inter-class similarity yields 99.99% accuracy for numeric digits (0–9), 98% accuracy for letters (A–Z), and over 94% accuracy for alphanumeric characters.

Research Methodology

Image segmentation is a critical component in many visual comprehension systems. Image (or video frame) partitioning into several sections or objects is a process component. Image analysis, autonomous vehicle navigation, video monitoring, and augmented reality are just a few examples of applications where segmentation plays a key role, such as tumor border extraction and volume measurement of tissue.

Technique Used

In digital image processing and computer vision, image segmentation is a procedure that divides a digital into several image segments. It is also known as image regions or image objects, which describe the color of each pixel in a rectangular array of pixels (sets of pixels). A bitmap is a mapping from some domain (for example, a range of integers) to bits. It is also termed a bitmap index or a bit array. The bitmap is related to a specific bit mapping application such as the pix-map, which relates to a map of pixels where everyone could contain more than two colors, employing more than one bit per pixel. The domain in question is the array of pixels that constitute a digital graphic output device (a screen or monitor). The pictures have several bits per pixel, and the word pix-map is used instead of the more generic term bitmap.

Image dataset

Computer vision algorithms depend heavily on collections of pictures known as datasets to simulate the cognitive capacities of humans and other animals. In computer vision, a dataset is a carefully selected collection of digital images that programmers use to test, train, and assess the effectiveness of their algorithms. The field of computer vision gives computers the capacity to recognize, label, and understand graphical representations. This technology will serve as the foundation for many of the discoveries and advancements that will occur in the future, such as the array of pixels that constitute a digital graphic output device.

Proposed Methodology

The complete recognition procedure involves some steps. Figure 2 shows the proposed methodology.

Step 1.1 Scanned Images

In this step, all the document pages are scanned and stored in one location. The scanned document is used in the further processing of the model.

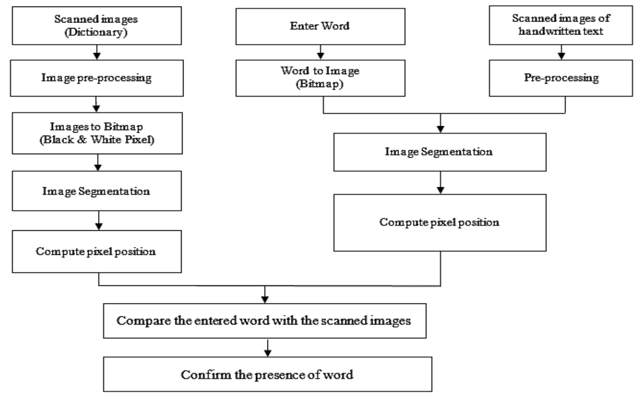


Figure 2: Flow chart of proposed methodology

Step 1.2 Image Pre-processing

In the second step, a modification, twist, or even an exaggeration that makes something look different from how it is an example of distortion. Actions might distort a picture, a notion, or even an idea. Distortion caused by image sensors generates various kinds of visual warping, including bending straight lines, chromatic aberration, and color shifts. Choose a lens that is not appropriate for the subject are photographing, particularly one that places less emphasis on the image's center will likely end up with distorted photographs.

Step 1.3 Images to Bitmap

The procedure encrypts a bitmap image from a normal image in the third step. A complete understanding of the bitmap file format is possible. It finds the beginning of a bitmap's pixels or array, and all the header file's components are identified (4.8 or 24-bit bitmap image). A single scan line in the image is represented by one byte in the bitmap bit array's bytes, which are kept in row order from left to right. It is possible to depict black-and-white pictures in a bitmap and vector form. A bitmap is a grid of black or white pixels that constitute an image. It uses an algebraic description of the picture's contour in the shape of Bezier curves, and a vector outline explains an image.

Step 1.4 Image Segmentation

In this step, an image segmentation method is employed to calculate pixel intensities to distinguish the letters in the images. Segmentation is breaking down a picture into smaller and more manageable chunks. Images are partitioned into many segments (curves, arcs, lines) to aid in locating objects and boundaries (curves, arcs, lines). Separating a picture into smaller, more manageable chunks aims to make it easier to understand. Each area's pixels should be characterized by color, intensity, or texture.

Step 1.5 Pixel Position

In this step, the location of a pixel, or array element, in the image is uniquely defined by its coordinates. There are two ways to calculate the pixel position:

- Calculate the pixel position on getting the black pixel.
- Calculate the minimum right, left, top, and bottom pixels found to be black.

Step 2.1 Word Input

Consider finding the handwritten text or manually entering a word in this step. The word is entered in the document. This word determines whether the document contains the English word being checked.

Turning handwriting into text requires using a tool to scan handwriting into text that can be edited.

Images can have text detected and extracted from them using the different annotation features that allow OCR. Text is identified and extracted from any photo using text detection.

Step 2.2 Word to Image (Bitmap)

The word bitmap image is again segmented to check the occurrence of the entered word within the scanned document.

Step 2.3 Word Segmentation

An individual morpheme could build up a single word in the English language. An algorithmic method for determining the borders between words is termed word segmentation.

Step 2.4 Pixel Position

In this step, how to calculate the pixel position-

- Calculate the pixel position on getting the black pixel.
- Calculate the min right, left, top, and bottom pixels found to be black.

Step 3 Compare the entered word with the scanned one

A comparison is made between the values of images in the dictionary and the word the user supplied. At this stage, a comparison process is carried out to identify similarities or matches between the input provided by the user and the pre-existing image data contained within the dictionary.

Step 4 Confirm the presence of the word

The program determines whether the word is listed in the dictionary. The system compares and matches the visual data to locate the user's words. At this point, the algorithm's output is checked, and a determination is made as to whether the term is present in the dictionary.

Results

Tools

The Python programming language is utilized in the implementation of the results. It is an object-oriented high-level, actively semantic, construed language. Dynamic linking and dynamic typing combine with their built-in high-level data structures to provide a perfect scripting language for quickly combining existing pieces during implementation. Readability is one of Python's primary focuses, which lowers the cost of maintaining programmers. Python supports using packages and modules to facilitate the reuse of code and the modularization of projects. Each general platform may download and distribute the Python interpreter and its standard library in source or binary form (Hu *et al.,* 2021).

- The scanned images are entered into the dataset. This indicates that the images from the dictionary are going to be entered. Figure 3 shows that the scanned image will be included in the database as a new row. Jhankar is the file name that will be uploaded as part of this procedure.
- At this moment, the scanned image will be converted into the Bitmap format. There is a file format known as bitmap. Microsoft Windows and operating system (OS) platforms employ the Bitmap file type to save digital images in the Bitmap file, irrespective of the display device. It has many different names, such as the bitmap image format, the bitmap file format, and bitmap. The transformation of the scanned image into a bitmap can be viewed in its entirety. Figure 4 shows the coding used to facilitate the completion of this conversion.
- Dividing a larger image into several smaller, more manageable parts is called segmentation. Images are broken up into different segments like lines, curves, and arcs, which help detect objects and boundaries within the image. The goal of breaking down a larger image into a series of more manageable and digestible parts is to make it easier to grasp. The pixels in each region must be categorized according to their color, intensity, or texture.
- The smallest addressable element in a raster image or the smallest addressable element in an all-points-addressable display system is a pixel, also known as a pixel or picture element. As a result, a pixel is the tiniest part of a picture that is adjusted on a computer screen. The shorthand for the pixel in digital photography is px.

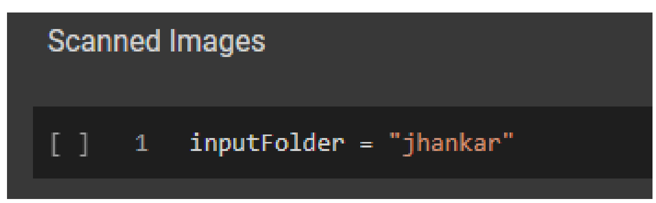


Figure 3: Scanned image

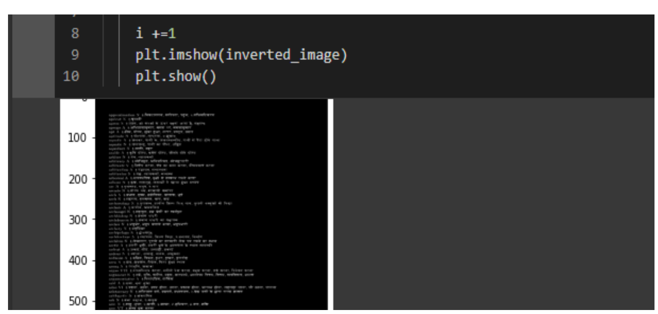


Figure 4: Image to Bitmap

The previous image, either converted or inverted, will now have its pixels changed to black and white in the third stage. Figure 5 shows the description of the pixel conversion into black and white.

- Image segmentation, a critical step in the processing of images, divides an image into relevant sections for examination. K-means clustering, a popular picture segmentation method, is used for this. The K-means algorithm groups n observations (image pixels) into clusters of k, where k is defined by the user's number reflecting the ideal number of segments. The centroid of each cluster is physically near each observation. K-means segmentation groups pixels by color or intensity, creating image segments. The segmented image shows related regions, helping with image analysis and processing. Figure 6 shows K-means picture segmentation and its segmented regions.
- In a computer, images are kept as a matrix of numbers. These numbers, referred to as pixel values, make up the matrix. These values for each pixel represent its respective level of intensity. White is represented by 255, whereas black is represented by 0. The scanned image will now go through the process of having its pixel values computed. Figure 7 presents an array representation of the scanned image's pixel values.
- Now, in this process stage, a word will be input so that it is looked up in the dictionary. The word is put here to check the presence of this word in the document or dictionary. The entered can be chosen randomly, and it doesn't matter whether it is present in the dictionary or not. Figure 8 demonstrates how this results in the word being typed being easily discernible, which is good quality.
- The K-Means clustering algorithm is an example of an unsupervised algorithm that is utilized to separate the area of interest from the background. It does so by grouping or dividing the data that has been provided into K clusters or sections depending on the K-centroids. The typed word will be transformed into an illustration at this point in the procedure. The segmented image of the word that was created using K-means and entered in the stage before this one can be seen in Figure 9.
- The two-dimensional grid of pixels displayed on a computer monitor or a digital image is generated to determine the shape and color of a digital image.

Figure 5: Black and white pixel

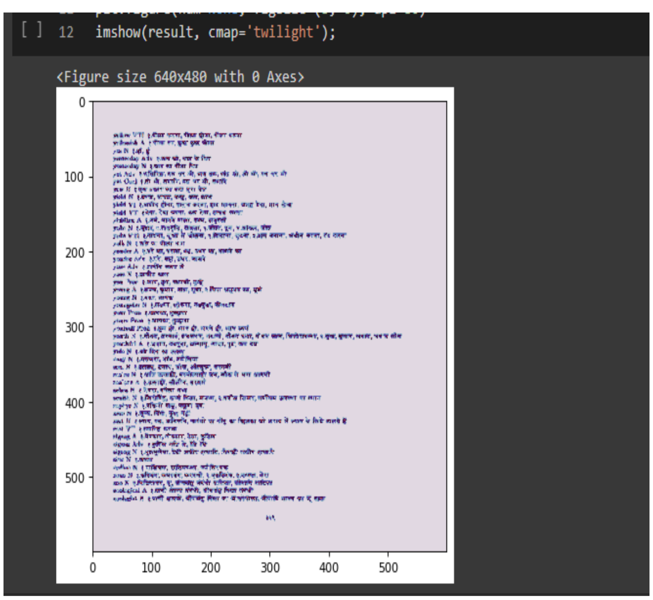


Figure 6: Image segmentation

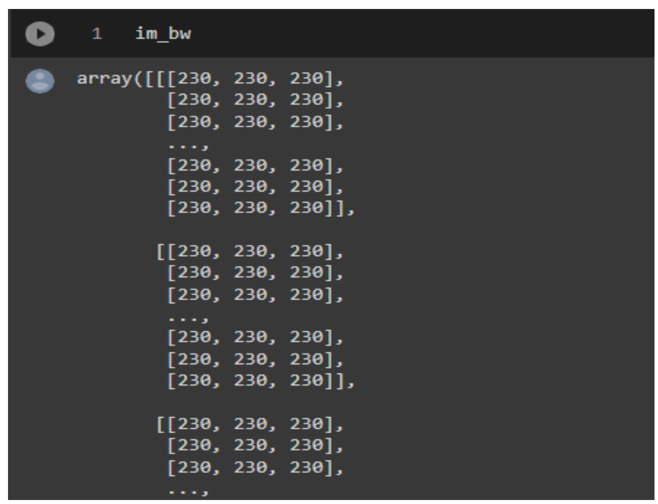


Figure 7: Pixel values

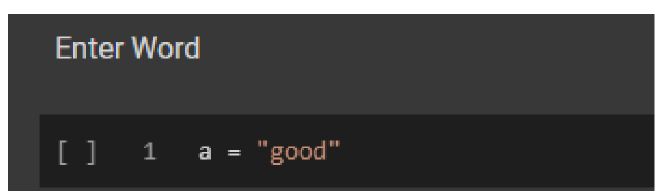


Figure 8: Enter the word

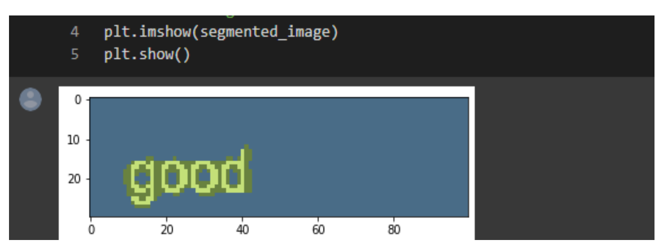


Figure 9: Segmented image

```
[ ] image

    array([[[ 73, 108, 136],
            [ 73, 108, 136],
            [ 73, 108, 136],
            ...,
            [ 73, 108, 136],
            [ 73, 108, 136],
            [ 73, 108, 136]],

           [[ 73, 108, 136],
            [ 73, 108, 136],
            [ 73, 108, 136],
            ...,
            [ 73, 108, 136],
            [ 73, 108, 136],
            [ 73, 108, 136]],

           [[ 73, 108, 136],
```

Figure 10: Pixels values

```
Confirm the Presence of word

[ ]  if a in nltk_tokens:
         print("Word found")
     else:
        print("Word Not found")

  Word found
```

Figure 11: Confirmation of word

After the image has been segmented, the pixel value calculation can now begin. These pixel coordinates can be found in Figure 10 under the heading array. The K-Means clustering method consists of an evolutionary manner used in isolating the objects of interest from the surroundings. This separation is accomplished to analyze the data more accurately. It achieves this goal by organizing the data supplied into K clusters or sections based on the K-centroids and grouping or dividing them. The typed word will be transformed into an illustration at this point in the procedure.

- After the picture has been segmented, calculating the values of the individual pixels begins. Users could locate these pixel coordinates in the heading array.
- The dictionary was then searched using the entered word, and the results were compared. In the programming language Python, this comparison will be carried out with the assistance of an if or else loop. If the word typed can be in the dictionary, the result would say word found; otherwise, it will say a word not found. The coding for this stage can be found in Figure 11, and the outcome of this stage is the word found.

Discussion

In the context of "English Language Analysis Using Pattern Recognition and Machine Learning," the comparison shown

Table 1: Comparative analysis based on accuracy

Techniques	*Accuracy (%)*
Deep Convolution Neural Networks (DCNN) [Aneja *et al.,* (2020)]	98
Nave Bayes [Khanday *et al.,* (2020)]	96.2
Table Recognition (TR) [Rashid *et al.,* (2018)]	97
Proposed algorithm	98.4

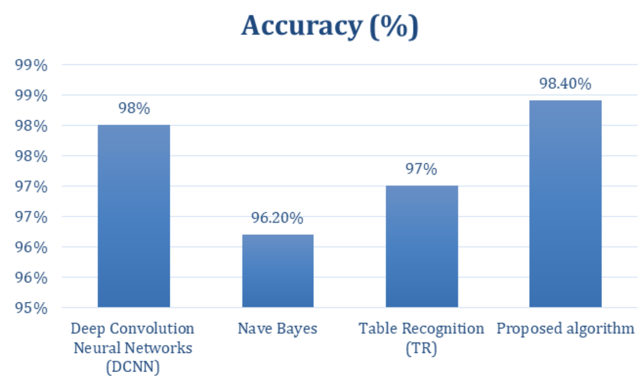


Figure 12: Graphical representation of the comparison between the proposed algorithm and several techniques

in Table 1 demonstrates the various degrees of accuracy that may be reached via the application of various approaches. DCNNs are particularly noteworthy due to their remarkable accuracy rate of 98%, which demonstrates their mastery in identifying complex linguistic patterns.

The Naive Bayes approach, famed for its accomplishments in the past, continues to perform brilliantly when used in language analysis tasks, with a rate of accuracy of 96.2%. In addition, algorithms centered on table recognition (TR) achieve a remarkable success rate of 97%, which illustrates their efficiency in extracting structural information from tabular data.

The inferences that can be made from Figure 12 provide more confirmation and bring to light the enhanced accuracy of the suggested strategy, which currently sits at 98.4%. This outperforms other state-of-the-art techniques, further proving its standing as a cutting-edge instrument in pattern recognition and machine learning. The fact that it has the potential to better analysis of the English language can be seen as both evident and encouraging.

Conclusion

Accuracy evaluation across many approaches in "English Language Analysis using Pattern Recognition and Machine Learning" yields important insights for enhancing linguistic comprehension. Amazingly, DCNN have a remarkable accuracy rate of 98%, demonstrating their prowess in decoding convoluted linguistic patterns. Regarding linguistic analysis, Naive Bayes is highly accurate, with a 96.2% success rate. The success rate of 97% for retrieving structured information is particularly impressive for TR techniques. The proposed method stands out as the top performer, with an accuracy of 98.4%, making it a potentially game-changing resource for improving analyses of the English language. In addition, this research is enriched using an innovative technology, OCR, which allows for digitizing hand-printed material by transforming handwritten scripts into editable electronic forms. This highlights OCR's potential for bridging the paper and digital media gap. The investigation into competing methods and developing better algorithms shows that the proposed strategy is superior.

Future difficulties may include spotting signatures on scanned documents and handwritten scripts and keeping data private when sharing is limited. Considering these factors could improve the system's performance and help guarantee that further work in pattern recognition and machine learning applied to the english language analysis is accurate and morally acceptable.

Acknowledgment

The author would like to express her sincere gratitude to the "Tantia University, Sri Ganganagar, Rajasthan" members for their invaluable contributions to this research work. Without their support, this study would not have been possible. The author would like to extend my special thanks to her co-author "Associated Prof. Dr. Kulvinder Singh" for their guidance, encouragement, and constructive criticism throughout the research process.

References

Africa, A., & Velasco, J. (2017). Development of a urine strip analyzer using artificial neural network using an android phone. *ARPN Journal of Engineering and Applied Sciences*, *12*(6), 1706-1712.

Agrawal, M., Chauhan, B., & Agrawal, T. (2022). Machine learning algorithms for handwritten Devanagari character recognition: a systematic review. *J Sci Technol*, *7*(01).

Alani, A. A. (2017). Arabic handwritten digit recognition based on restricted Boltzmann machine and convolutional neural networks. *Information*, *8*(4), 142.

Alrubayi, A. H., Ahmed, M. A., Zaidan, A. A., Albahri, A. S., Zaidan, B. B., Albahri, O. S., Alamoodi, A. H., & Alazab, M. (2021). A pattern recognition model for static gestures in Malaysian Sign language based on machine learning techniques. *Computers and Electrical Engineering*, *95*, 107383.

Aneja, N., & Aneja, S. (2019). Transfer learning using CNN for handwritten Devanagari character recognition. In *1st International Conference on Advances in Information Technology (ICAIT), 2019* (pp. 293–296). IEEE

Ashiquzzaman, A., & Tushar, A. K. (2017). Handwritten Arabic numeral recognition using deep learning neural networks. In *2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR)* (pp. 1-4). IEEE.

Avadesh, M., & Goyal, N. (2018). Optical character recognition for Sanskrit using convolution neural networks. In *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)* (pp. 447-452). IEEE.

Bengfort, B., Bilbro, R., & Ojeda, T. (2018). *Applied text analysis with Python: Enabling language-aware data products with machine learning*. " O'Reilly Media, Inc.".

Chen, T. L., & Chen, F. Y. (2016). An intelligent pattern recognition model for supporting investment decisions in the stock market. *Information Sciences*, *346*, 261–274.

Cho, Y., & Kim, J. (2021). Production of mobile english language teaching application based on text interface using deep learning. *Electronics*, *10*(15), 1809.

Dutta, K., Krishnan, P., Mathew, M., & Jawahar, C. V. (2018). Improving CNN-RNN hybrid networks for handwriting recognition. In *2018 16th international conference on frontiers in handwriting recognition (ICFHR)* (pp. 80-85). IEEE.

Entezami, A., Sarmadi, H., Behkamal, B., & Mariani, S. (2020). Big data analytics and structural health monitoring: A statistical pattern recognition-based approach. *Sensors*, *20*(8), 2328.

Gol, E. A., Bartocci, E., & Belta, C. (2014, December). A formal methods approach to pattern synthesis in reaction diffusion systems. In *53rd IEEE Conference on Decision and Control* (pp. 108-113). IEEE.

Hu, Z., Peng, J., & Zhao, H. (2021). Dynamic neural orthogonal mapping for fault detection. *International Journal of Machine Learning and Cybernetics*, *12*, 1501-1516.

Jha, S. K., Yadava, R. D. S., Hayashi, K., & Patel, N. (2019). Recognition and sensing of organic compounds using analytical methods, chemical sensors, and pattern recognition approaches. *Chemometrics and Intelligent Laboratory Systems*, *185*, 18-31.

Kader, M. F., & Deb, K. (2012). Neural network-based English alphanumeric character recognition. *International Journal of Computer Science, Engineering and Applications*, *2*(4), 1.

Khanday, A. M. U. D., Rabani, S. T., Khan, Q. R., Rouf, N., & Mohi Ud Din, M. (2020). Machine learning based approaches for detecting COVID-19 using clinical text data. *International Journal of Information Technology*, *12*, 731-739.

Luo, X. (2021). Efficient English text classification using selected machine learning techniques. *Alexandria Engineering Journal*, *60*(3), 3401-3409.

Milojković Opsenica, D., Ristivojević, P., Trifković, J., Vovk, I., Lušić, D., & Tešić, Ž. (2016). TLC fingerprinting and pattern recognition methods in the assessment of authenticity of poplar-type propolis. *Journal of chromatographic science*, *54*(7), 1077-1083.

Namysl, M., & Konya, I. (2019, September). Efficient, lexicon-free OCR using deep learning. In *2019 international conference on document analysis and recognition (ICDAR)* (pp. 295-301). IEEE.

Pande, S. M., & Jha, B. K. (2021, April). Character recognition system for Devanagari script using machine learning approach. In *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)* (pp. 899-903). IEEE.

Paolanti, M., & Frontoni, E. (2020). Multidisciplinary pattern recognition applications: A review. *Computer Science Review*, *37*, 100276.

Rajalakshmi, M., Saranya, P., & Shanmugavadivu, P. (2019, April). Pattern recognition-recognition of handwritten document using convolutional neural networks. In *2019 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)* (pp. 1-7). IEEE.

Rashid, S. F., Akmal, A., Adnan, M., Aslam, A. A., & Dengel, A. (2017, November). Table recognition in heterogeneous documents using machine learning. In *2017 14th IAPR International conference on document analysis and recognition (ICDAR)* (Vol. 1, pp. 777-782). IEEE.

Rehman, A., Naz, S., & Razzak, M. I. (2019). Writer identification using machine learning approaches: a comprehensive review. *Multimedia Tools and Applications*, *78*, 10889-10931.

Sun, Y., Peng, M., Zhou, Y., Huang, Y., & Mao, S. (2019). Application of machine learning in wireless networks: Key techniques and open issues. *IEEE Communications Surveys & Tutorials*, *21*(4), 3072-3108.

Wang, D., Su, J., & Yu, H. (2020). Feature extraction and analysis of natural language processing for deep learning English language. *IEEE Access*, *8*, 46335-46345.

Yousef, M., Hussain, K. F., & Mohammed, U. S. (2020). Accurate, data-efficient, unconstrained text recognition with convolutional neural networks. *Pattern Recognition*, *108*, 107482.

Zeng, J., Guo, Y., Han, Y., Li, Z., Yang, Z., Chai, Q., ... & Fu, C. (2021). A review of the discriminant analysis methods for food quality based on near-infrared spectroscopy and pattern recognition. *Molecules*, *26*(3), 749.

Zhao, H., Hu, Y., & Zhang, J. (2017, November). Character recognition via a compact convolutional neural network. In *2017 International conference on digital image computing: techniques and applications (DICTA)* (pp. 1-6). IEEE.