USING ARTIFICIAL INTELLIGENCE STRANGENESS IN MACHINE TRANSLATION

[1]Dr.M.Helda Mercy, [2]P.Prakash, [3]Ms. N. Malathy
[1]*Professor and Head, M.C.A, Panimalar Engineering College, Chennai*
[2]*PG Scholar, M.C.A, Panimalar Engineering College, Chennai*
[1]*Faculty, M.C.A, Panimalar Engineering College, Chenna*

Abstract:

The project entitled as" using artificial intelligence strangeness in machine translation. The main objective is to recognize all type of characters, which may be of various font styles even of hand-written type by implementing the artificial intelligence (ai) providing various texts editable formats.

The process involves two modules template matching and character matching. Template matching is used for the input character representation images. Such systems are font dependent and suffer in accuracy when given documents printed in novel font styles. Then prefer character matching for recognition.

Character matching is to group together similar characters in the document and solves a cryptogram to assign labels to clusters of characters. This method does not require any character model and it is able to handle arbitrary font styles.

Template matching involves three processes. They are straightforward, multi threading and property grid. Character matching also involves three steps. They are advanced processing, set region and display character.

The project is able to display recognized characters, numbers and the reduced characters. The project can recognize single character, selected characters and entire characters of the file.

The entire process is developed using the software tool C#.Net. Then two dynamic link libraries called fuzzy optical character recognition (focr1 and focr2) are used.

1. Introduction:

The central objective of this project is demonstrating the capabilities of Artificial Neural Network implementations in recognizing extended sets of optical language symbols. The applications of this technique range from document digitizing and preservation to handwritten text recognition in handheld devices. The classic difficulty of being able to correctly recognize even typed optical language symbols is the complex irregularity among pictorial representations of the same character due to variations in fonts, styles and size. This irregularity undoubtedly widens when one deals with handwritten characters.

Neural networks are very sophisticated modeling techniques capable of modeling extremely complex functions. In particular, neural networks are nonlinear. For many years linear modeling has been the commonly used technique in most modeling domains since linear models have well-known optimization strategies. Where the linear approximation was not valid (which was frequently the case) the models suffered accordingly. Neural networks also keep in check the curse of dimensionality problem that bedevils attempts to model nonlinear functions with large numbers of variables.

Neural networks learn by example. The neural network user gathers representative data, and then invokes training algorithms to automatically learn the structure of the data. Although the user does need to have some heuristic knowledge of how to select and prepare data, how to select an appropriate neural network, and how to interpret the results, the level of user knowledge needed to successfully apply neural networks is much lower than would be the case using (for example) some more traditional nonlinear statistical methods.

2. A. Existing System:

In Early Days a data entry system generates an electronically stored coded representation of a character sequence from one or more electronically stored document images. The system comprising logic for generating, from the document image or images, character data specifying one of a plurality of possible character values for corresponding segments of the document images. The system also has an interactive display means for generating and sequentially displaying, one or more types of composite image, each composite image comprising segments of the document image or images arranged according to the character data, and a correction mechanism responsive to a user input operation to enable the operator to correct the character data associated with displayed segments.

LIMITATIONS

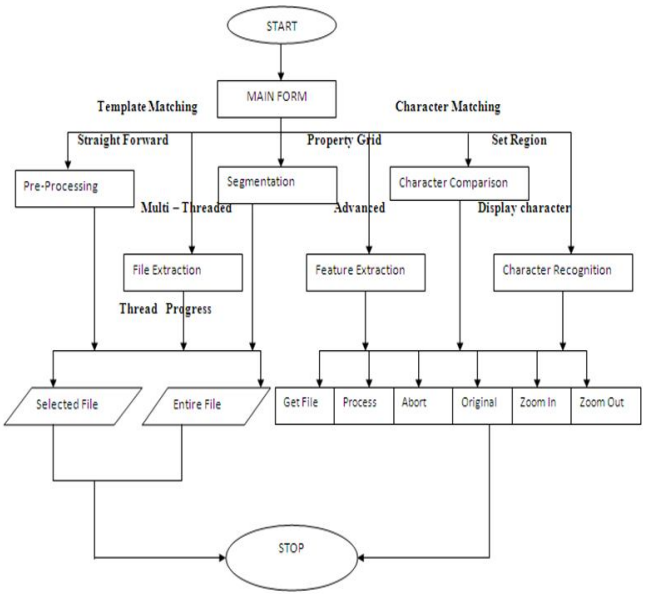
- Early systems required training (the provision of known samples of each character) to read a specific font.
- Systems are capable of reproducing formatted output that closely approximates the original scanned page including images, columns and other non-textual components.
- The problems in existing systems are that, they only recognize the machine print character in the image and not the hand-written character or text on the image, which is of various font styles- resulting in non-editable text format types.

2. B. Proposed System:

Optical Character Recognition (OCR) system produces a text output based on the recognition of their specific inputs. The text output contains errors due to wrong recognition. The project uses natural language processing technology to automatically correct 60% to 90% of these errors and is attached to any commercial OCR. The OCR system will work on the English language only. The OCR system works in the following way. The OCR system output is redirected on the input of the Recognition. It automatically uses dictionaries, language syntax models and advanced representation of text meaning to detect errors and make corrections. The OCR system already includes some language technology to improve output. It intends to use more advanced model, which enables to reduce errors much more efficiently.

BENEFITS

- The main improvement over existing system is the incorporation of an advanced representation of text meaning.
- It can recognize all types of character, which may be of various font styles even of hand-written type.



3. Back Propagation Algorithm:

1)Let A be the number of units in the input layer, as determined by the length of the training input vectors. Let C be the number of units in the output layer. Now choose B, the number of units in the hidden layer. The input and hidden layers each have an extra unit used for threshold; therefore, the units in these layers will sometimes be indexed by the ranges (0,…..,A) and (0,…..,B). We denote the activation levels of the units in the input layer by xj , in the hidden layer by hj , and in the output layer by oj . Weights connecting the input layer to the hidden layer are denoted by w1ij, where the subscript I indexes the input units and j indexes the hidden units. Likewise, weights connecting the hidden layer to the output layer are denoted by w2ij, with I indexing to hidden units and j indexing output units.

2)Initialize the weights in the network. Each should be set randomly to a number between -0.1 and 0.1.Wij = random(-0.1,0.1) for all i =0,…..A, j = 1,……B Wij = random(-0.1,0.1) for all i =0,…..B, j= 1,……C

3)Initialize the activations of the network. The values of these thresholding units should never change. a)X0= 1.0 b)h0 = 1.0

4)Choose an input-output pair. Suppose the input vector is Xi and the target output vector is Yi Assign activation levels to the input units.

5)Propagate the activations from the units in the input layer to the units in the hidden layer using the activation functions

$$\Delta h_j = \frac{1}{1+e^{-\sum_{i=0}^{B} W1_{ij} h_i}} \quad for\ j = 1,\ldots,C$$

Note that i ranges from 0 to A. W10j is the threshold weight for hidden unit j (its propensity to fire irrespective of it inputs).X0 is always 1.0.

6)Propagates the activations from the units in the hidden layer to the units in the output layer

$$h_j = \frac{1}{1+e^{-\sum_{i=0}^{B} W2_{ij} h_i}} \quad for\ j = 1,\ldots,C$$

Again, the thresholding weight w20j for output unit j plays a role in the weighted summation. h0 is always 1.0.

7)Compute the errors of the units in the output layer denoted 2 $j\ \delta$. Error is based on the network's actual output (Oj) and the target output (Yi)

$$\delta 2_j = o_j(1-o_j)(y_j - o_j) \quad for\ all\ j = 1,...B$$

8)Compute the errors in the units in the hidden layer, denoted 1*j* δ .

$$\Delta 1_j = h_j(1-h_j)\sum_{i}^{c} \delta 2_i \times w2_{ji} \quad for\ all\ j = 1,...B$$

$$\Delta w1_{ij} = \eta \bullet \delta 1_j \bullet h_i \quad for\ all\ i = 0,...,A,\ j = 1,...,B$$

9)Adjust the weights between the hidden layer and the output layer. The learning rate denoted is denoted byη η ; its functions is in the same as in perception learning. A reasonable value of η is 0.35.

$$\Delta w2_{ij} = \eta \bullet \delta 2_j \bullet h_i \quad for\ all\ i = 0,...,B,\ j = 1,...C$$

10)Adjust the weights between the input layer and the hidden layer.

$$\Delta w1_{ij} = \eta \bullet \delta 1_j \bullet h_i \quad for\ all\ i = 0,...,A,\ j = 1,...,B$$

11)Go to step 4 and repeat. When all the inputs-output pairs have been presented to the network, one epoch has been completed.

Repeat steps 4 to 10 for as many epochs as Desired

4. Module Description

This project is designed using various forms for

- **Pre-Processing,**
- **File Extraction,**
- **Segmentation,**
- **Feature Extraction,**
- **Character Comparison,**

The project can segregated into various modules such as Tray to Import the Processed file, Character Extraction for the processed file.

4A. Pre-Processing

OCR pre-process technology enhances class 1 semantic content immediately before images are sent to an OCR engine. Semantic content in a class 1 sense refers to the correct classification of a blob of pixels as a specific character (or part of such), a line, a part of an image, or noise. By implementing several powerful and proprietary noise removal and character enhancement algorithms

4B. File Extraction

File by OCR automatically names files and places them in a file folder structure based on the document's OCR text contents. It can extract text from a searchable image and name and file it, or it can extract the OCR text and build a file.

4C. Segmentation

Characters are arranged in document lines following some typesetting conventions which we can use to locate characters and find their style. Typesetting rules can help in distinguishing such characters as s from 5, h from n, and g from 9, which can be often confused in multi font context. They can also limit the search area according to characters' relative positions and heights with respect to the baseline

4D. Feature Extraction

Feature extraction in bilingual OCR is handicapped by the increase in the number of classes or characters to be handled. This is evident in the case of Indian languages whose alphabet set is large. It is expected that the complexity of the feature extraction process increases with the number of classes.

4E. Character Comparison

Complete recognition for English 99% Plus Accuracy -- Achieves the highest character accuracy in the industry. With the addition of over 300 new international fonts, recognition is improved over 60% from previous international version of Type Reader. OCR Engine switches to achieve highest accuracy on degraded quality documents. Number Bias achieving highest accuracy on numbers.

5 Future Enhancement

The Automatic Reader features categorized as General Features, Image Processing & Recognition Features, and Technical & Integration Specification. Supports both automatic and manual framing modes, and the ability to save frames it files and apply the saved frames as templates. Supports non rectangular frames. Recognizes broken and connected characters, and colored documents.Recognizes underlined words, and automatically detects style for fonts (Regular or Bold).Works as standalone application with interactive interface or through a wizard from which user can perform all program functions, or as a menu item from other applications.

- RFID

- 3D Barcode
- Recognition of other languages
- Digital Signatures are also recognized

6 Conclusion

Further, the delta lognormal model provides a realistic and meaningful way to analyze and describe handwriting generation and provides information that can be used, in a perceptivomotor context to tackle recognition problems. Its first practical application has been the development of a model-based segmentation framework for the partitioning of handwriting and its use in the development of an automatic signature verification system. Based on this model, a multilevel signature verification system was developed, which uses three types of representations based on global parameters and two other based on functions. The overall verification is performed using a step wise process at three distinct levels, using personalized decision thresholds.

7 References

1. Simon Robinson, Christian Nagel, Karli Watson, **"PROFESSIONAL C#"**, Wiley Dreamtech India Pvt ltd., third edition.
2. Andy Harris, "**MICROSOFT C# PROGRAMMING**", Prentice hall of India Pvt ltd.,
3. Roger S.Pressman, **"SOFTWARE ENGINEERING"**, TataMcGraw Hill Publications, fifth edition.
4. Elias M.Award's, "**SYSTEM ANALYSIS AND DESIGN",** Galgotia Publications Private Limited Companies, 1997 Edition.
5. Herbert Schildt, **"THE COMPLETE REFERNCE C# 2.0"**, TataMcGraw Hill Publications, second edition.
6. V.K.Jain, **"THE COMPLETE GUIDE TO C# PROGRAMMING",** Dreamtech press.
7. E.Balagurusamy, **"PROGRAMMING IN C#",** TataMcGraw Hill Publications.