https://doi.org/10.31449/inf.v45i2.3564 Informatica **45** (2021) 309–314 **309**

Research on the Efficiency of Intelligent Algorithm for English Speech Recognition and Sentence Translation

Gang Zhang
School of Foreign Languages and Literature, Nanjing Tech University, Nanjing 211816, Jiangsu, China
E-mail: zg@njtech.edu.cn

Student paper

Keywords: intelligent algorithm, speech recognition, machine translation, long short term memory

Received: May 25, 2021

Machine translation has been gradually widely used to improve the efficiency of English translation. This paper briefly introduced the English speech recognition algorithm based on the back-propagation (BP) neural network algorithm and the machine translation algorithm based on the long short-term memory-recurrent neural network (LSTM-RNN) algorithm. Then, the machine translation algorithm was simulated and compared with BP-RNN and RNN-RNN algorithms. The results showed that the BP neural network algorithm had a lower word error rate and shorter recognition time compared with the manual recognition approach; the LSTM-RNN-based machine translation algorithm had the lowest error rate for the translation of speech recognition results, and the translation gained the highest rating in the evaluation of ten professional translators.

Povzetek: Predstavljena je analiza več pristopov umetne inteligence za prepoznavanje govora in prevajanje.

1 Introduction

As internationalization progresses faster and faster, economic and cultural exchanges between different countries have become more and more frequent. However, different countries have different languages, and even within the same country, there are languages with different accents in different regions, making the variety of languages further increase [1].

The variety of languages, while guaranteeing cultural diversity, can cause significant communication barriers in practice, especially in today's environment of deepening internationalization [2]. Under normal circumstances, the cost of learning multiple languages is enormous, so to ensure smooth communication, a common language is often chosen. English is currently one of the most common languages. Although the variety of language learning has been reduced from multiple to one, the cost of learning is equally high, and it is difficult to reach the level of free communication [3]. In the wave of globalization, it would be sufficient for face-to-face daily communication, but in formal situations and when a large amount of information needs to be exchanged, it is difficult for a single human translator to meet the increasing demand for language translation. Simultaneous interpretation, for example, requires high on the attention of translators; thus, they often cannot work for long hours. Therefore, a translation tool is needed to replace manual translation. Machine translation uses computers and Chinese-English thesaurus to perform batch translation based on, but it is too rigid. The emergence of intelligent algorithms has effectively promoted the efficiency and quality of machine translation. Luong [4] proposed an effective technique to solve the problem that traditional neural network machine translation can not accurately translate rare words. The experimental results showed that this method had a better translation effect compared with neural network machine translation without applying this technique. Lee et al. [5] used a character-level convolution network for machine translation. The character-level convolution network encoder outperformed the subword-level encoder in all of the multilingual experiments. Choi et al. [6] proposed the use of nonlinear word-packet representation of source statements to contextualize embedding vectors in neural machine translation. The experiment found that the proposed approaches of contextualization and symbolization significantly improved the translation quality of neural machine translation systems.

2 English speech recognition

The advancement of computer technology and the emergence of intelligent algorithms have effectively improved the efficiency and quality of machine translation. However, computers have neither vision nor hearing; therefore, machine translation of English requires the input of the text to be translated, which is relatively cumbersome [7]. Inputting text to be translated by voice is easier than text input; moreover, the input English voice can be translated, which means that it can achieve long periods of non-manual simultaneous interpretation. Before translating the English input by voice, it is necessary to first recognize the voice and convert the audio to text characters, and then translate the text. The commonly used algorithms for recognition of speech are

dynamic time warping method [8], hidden Markov method, and neural network method.

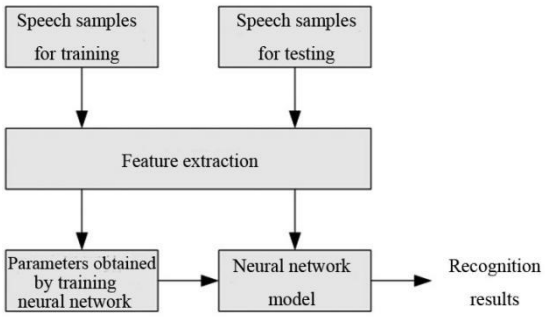

Figure 1: The principle of speech recognition based on neural network.

This study uses the neural network method to recognize English speeches. The neural network method is an imitation of the human brain and animal nervous system in reality; thus, it also has the ability of autonomous learning, i.e., it can effectively adapt to the randomness of pronunciation and excavate the corresponding hidden rules between pronunciation and text, as shown in Figure 1. Both training speech samples and testing speech samples require feature extraction first. This study extracts the features of speech samples using the Mel cepstrum coefficient method. After that, the selected neural network model is trained using the training speech samples to obtain the super-parameters in the neural network. The trained super-parameters are then fixed in the selected neural network, and the testing speech samples are input into the trained neural network model after feature extraction to obtain recognition results.

There are various types of neural network models that can be used in the speech recognition process in Figure 1, and this paper adopts the widely used BP neural network [9]. During training, the samples that have undergone feature extraction are input into the BP neural network, and the multi-layer forward calculation of the extracted features of the speech is performed in the hidden layer using the activation function. The results obtained after the layer-by-layer calculation are compared with the results corresponding to the training samples, and the super-parameters in the hidden layer are adjusted in the reverse direction according to the difference between the results. Then, the forward calculation is carried out layer by layer again, and the calculated results are compared with the actual results to reversely adjust the parameters. The steps are repeated until the difference between the calculated results and the actual results is reduced to the set threshold. The speech samples used in the test are input into the trained neural network model after feature extraction, and the recognition results are output after calculation.

3 Translation of English sentences

After speech recognition of the neural network algorithm, English speech is converted into English texts. The traditional machine translation translates English word by word with the Chinese English thesaurus. This translation method is simple in principle and has relatively high efficiency. It is more suitable for an essay composed of short sentences. However, this translation method is less effective for long texts composed of long sentences. On the one hand, due to the differences in grammar between Chinese and English, word-by-word translation will lead to grammar chaos and even completely opposite semantics; on the other hand, some auxiliary words that have no specific meanings in English sentences will also be translated, and word-by-word translation will affect the coherence of the translation. In order to solve the above problems and further improve the efficiency and quality of translation, this study chooses neural network algorithms for machine translation. A neural network algorithm is an imitation of a biological neural system. With the help of the parallel corpus of big data and the computational performance of computers, we can get the corresponding hiding rules between Chinese and English. As the parallel corpus of big data is used for training, hiding rules obtained contain the grammatical rules to a certain extent.

3.1 Recurrent neural network-based machine translation

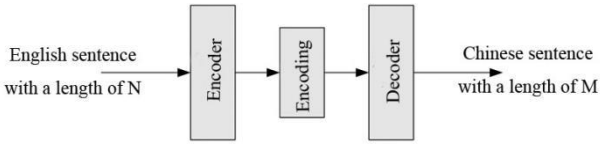

Figure 2: The basic structure of neural network-based machine translation.

The basic structure of conducting English machine translation with neural network algorithms is shown in Figure 2, including an encoder and a decoder [10]. The reason for using this structure is as follows. When translating English into Chinese, the length of the character sequence of English and Chinese cannot be the same; thus, it is necessary to transform an English sentence with a length of N to a code with a required length and decode the code with the decor to obtain a Chinese sentence with a length of M. In the above process, both the encoder and decoder realize the encoding and decoding of the sequence through neural network algorithms. As the length of sentences to be translated is different and the meaning of words will be affected by their sequence in the sentence, it is not suitable to use the traditional BP neural network. Recurrent neural network (RNN) [11] has memory function in training, and the result of the current moment will be affected by the result of the previous moment, which is very consistent with the characteristic that the word sequence will affect the meaning of words in languages; therefore, it is applied to the coding and decoding of machine translation. First, an English sentence is set as $\boldsymbol{\varepsilon} = \{\varepsilon_0, \varepsilon_1, \varepsilon_2, \cdots, \varepsilon_n\}$, where ε_n represents the $n+1$-th word in sentence $\boldsymbol{\varepsilon}$; then, RNN propagates the input data forward in the hidden layer according to the following formula:

$$\begin{cases} \boldsymbol{x}_t = e(\varepsilon_t) \\ \boldsymbol{a}_t = \boldsymbol{\omega} h_{t-1} + \boldsymbol{U}\boldsymbol{x}_t + \boldsymbol{b} \\ h_t = tanh(\boldsymbol{a}_t) \\ t = 0,1,2,3,4,\cdots,n \end{cases}, \quad (1)$$

where $\boldsymbol{x}_t$ stands for the word vector after inputting ε_t into the vector transformation function $e(\cdot)$, t stands for time step (every time step corresponds to the input moment of a word in the sequence), h_{t-1}, h_t are the hidden states of words vectors with a word order of $t-1$ and t, $\boldsymbol{a}_t$ is the output at time t, $\boldsymbol{\omega}$ and $\boldsymbol{U}$ are the hidden state at time $t-1$ and the weight matrix of the word vector of ε_t input at time t respectively, and $\boldsymbol{b}$ is a bias term.

After the forward operation of equation (1) in the encoder, every word (ε_t) in the English sentence obtains the corresponding coding vector $\boldsymbol{a}_t$, and the coding vector of every word is affected by the former word vector, ensuring the influence of the word order in the coding vector of the whole sentence on word meaning. Then, the coding vector is decoded to obtain the corresponding Chinese translation. The decoding is carried out in the decoder. In order to ensure the influence of word order on word meaning, the coding vector is decoded by RNN. The forward calculation formula in the hidden layer is:

$$\begin{cases} \mathbf{y}_{t-1} = e(Z_{t-1}) \\ z_t = \tanh(\boldsymbol{\omega} z_{t-1} + \mathbf{U}\mathbf{y}_{t-1} + \mathbf{b}) \\ \hat{\mathbf{y}}_t = \text{softmax}(\mathbf{d} + \mathbf{V}z_t) \\ Z_t = \arg\max \hat{\mathbf{y}}_t \end{cases}, \quad (2)$$

where Z_t is a word with an order of t in the target sentence, y_{t-1} is the vector of a word with an order of $t-1$ in the target sentence, $\boldsymbol{d}$ is a bias term, $\boldsymbol{V}$ is a weight matrix, $\hat{\boldsymbol{y}}_t\hat{\boldsymbol{y}}_t$ is the probability distribution of different characters in the translation [12], and z_t is the hidden state in the decoder.

3.2 Improving machine translation with long short term memory

The above is the introduction of the RNN-based machine translation algorithm. This machine translation algorithm adopted the basic structure of encoder-decoder. In the encoder, the English text is coded by RNN in the encoder to get the intermediate vector, and the intermediate vector is decoded by RNN in the decoder to get the Chinese translation. However, the encoder faces the problem of gradient explosion when using RNN to encode the English text, which makes the algorithm inefficient and less accurate in the training and use process. Therefore, this study used LSTM instead of RNN to encode the English text. LSTM [13] is also a kind of recurrent neural network algorithm. Compared with the traditional RNN, LSTM introduces the structural units of input gate, forgetting gate, and output gate to simulate the phenomena of deep impression and forgetting in the process of human brain memory, thus reducing the unimportant parts in the English text, highlighting the key points, reducing the amount of computation while enhancing the accuracy. The calculation formula of LSTM in the encoder is as follows:

$$\begin{cases} f_t = \sigma(b_f + U_f x_t + W_f h_{t-1}) \\ s_t = f_t s_{t-1} + g_t \sigma(b + U x_t + W h_{t-1}) \\ g_t = \sigma(b_g + U_g x_t + W_g h_{t-1}) \\ h_t = tanh(s_t) q_t \\ q_t = \sigma(b_q + U_q x_t + W_q h_{t-1}) \end{cases}, \quad (3)$$

where f_t is the output of the forgetting gate, b_f, U_f, and W_f are the bias term, input term weight, and forgetting gate weight in forgetting gate [14], s_t is the output of the cycle gate, b, U, and W are the bias term, input term weight, and cycle gate weight in the cycle gate weight, g_t is an external input gate unit, b_g, U_g, and W_g are the bias term, the weight of the input term, and the weight of the input gate in the input gate respectively, q_t is the output gate unit, b_q, U_q, W_q are the output gate bias term, input term weight, and output gate weight.

In the subsequent decoder, RNN is still used to decode the output vector of the encoder, but in actual use, the encoder of the machine translation algorithm compresses the information contained in the whole sentence to be translated in a vector when encoding English using LSTM, resulting in partial loss of information in the compressed vector. The larger the length of English text to be translated is, the more serious the loss is. Therefore, when decoding the LSTM-encoded vector using RNN, its effectiveness decreases with the increase of the length of the original text. In order to remedy the above defects, an attention mechanism [15] is added to the decoder to make the decoder interact with the original text in the decoding process to alleviate the information loss. The calculation formula of attention is as follows:

$$\begin{cases} \alpha_t = attention(z_t, h_t) \\ \sum_t \alpha_t = 1 \\ c = \sum_t \alpha_t h_t \end{cases}, \quad (4)$$

where α_t is the weight of the matching degree between the query vector and the key vector in the value vector (the query vector is the hidden state in the decoder, and the key vector is the hidden state in the encoder), and c is the attention weight. The attention calculation formula compares the hidden states in the decoder with the hidden states of the original text in the encoder to obtain the attention weights of the hidden states in the decoder, and the attention weights participate in the decoding process of the decoder afterwards to calculate the probability distribution of the translated text within equation (2). The probability distribution formula after transformation is:

$$\hat{\boldsymbol{y}}_t = \text{softmax}(\mathbf{c} + \mathbf{V}z_t + \boldsymbol{W}_c c). \quad (5)$$

4 Experimental analysis

4.1 Experimental environment

The experiment was carried out on a laboratory server, configured with Windows 7 system, I7 processor, and 16 G memory.

4.2 Experimental data

This study used the English speech data set from the UCI machine learning database. The speakers in the data set covered most of the age groups from 12 to 70. Ten thousand sentences with clean, clear, and standard pronunciation were selected, 9000 sentences were randomly selected as the training samples, and the remaining 1000 sentences were used as as the test samples. The experimenters read the sentences aloud. The speech characteristic parameters of the sentences were collected. The sampling rate was set as 16 kHz, and the 16-bit coding was used. Some of the sentences are as follows.

① It's a nice day today;
② How can I get to the airport, please;
③ What's the price of this product……

4.3 Experimental project

(1) Training and testing of the speech recognition algorithm: the BP algorithm-based speech recognition algorithm was trained by the training set. In the BP algorithm, there were five hidden layers containing the relu activation function and 1024 hidden layer nodes in each layer, and the maximum number of iterations during training was 500. The test set was used for testing.

(2) Training and testing of the machine translation algorithm: in the machine translation algorithm, the encoder used LSTM containing the attention mechanism. There were four hidden layers, and the number of nodes in each layer was 1024. The decoder used RNN with two hidden layers, and the size of the hidden layer was consistent with LSTM. Two machine translation algorithms are used for comparison, one used BP neural network as the encoder, and the other used a RNN as the encoder. The decoder of both algorithms used RNN. The test set was used for testing.

4.4 Evaluation criterion

In this paper, the machine translation algorithm was evaluated by the word error rate after recognition by the speech recognition algorithm. The calculation formula is:

$$WER = \frac{X+Y+Z}{P} * 100\%, \quad (6)$$

where X is the number of error words substituted, Y is the number of error words deleted, Z is the number of error words inserted, and P is the number of all words in the test set. However, the corresponding word order differs between the original text and the translated text because of the difference in grammar. Therefore, when evaluating the machine translation algorithm, not only the word error rate but also the overall translation level of the sentence should be considered. Therefore, ten professional translators were invited to evaluate the translation obtained after machine translation, and the score is based on the expression content and grammatical structure of the translated text. The total score was 100 points, and the average score of the ten translators was taken as the final result.

4.5 Experimental results

The accuracy of English speech recognition algorithms will greatly affect the quality of machine translation It was seen from Table 1 that the word error rate of speech recognition by manual recognition was 6.34%, and it took 35 minutes; the word error rate of speech recognition by BP neural network was 1.33%, and it takes one minute. The comparison of the results of two speech recognition methods showed that BP neural network had a lower word error rate and consumed less recognition time than manual recognition. The reason for the above result is as follows. Computers were more efficient in computational efficiency, and when faced with a large number of speech sounds in the test set, people cannot maintain their attention for a long time, resulting in a higher word error rate and longer recognition time; however, BP neural network used computers for speech recognition, which was not only computationally efficient but also did not have the disadvantage of inattention.

	Word error rate/%	Time consumed in recognition
Manual recognition	6.34	35 min
BP neural network recognition	1.33	1 min

Table 1: Word error rate and recognition time of artificial recognition and BP neural network recognition.

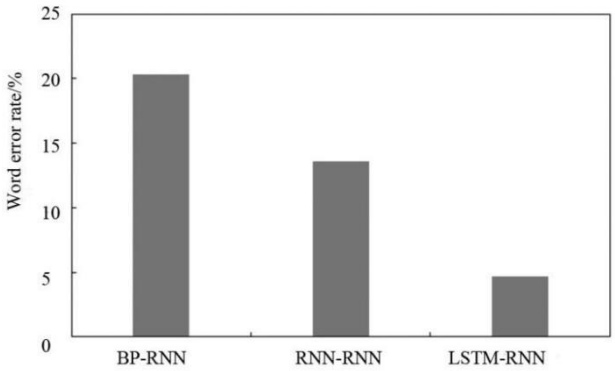

Figure 3: The word error rates of three machine translation algorithms in translating English speech recognition results.

The word error rates of the translations of English speech recognition results by the three machine translation algorithms are shown in Figure 3. The word error rate of the machine translation algorithm using BP neural network for encoder and RNN for decoder was 20.3%; the word error rate of of the machine translation algorithm with RNN for encoder and RNN for decoder was 13.6%; the word error rate of the machine translation algorithm with LSTM for encoder and RNN for decoder and attention mechanism was 4.7%. It was seen from Figure 3 that the LSTM-RNN-based machine translation had the lowest word error rate, while the BP-RNN-based machine translation had the highest word error rate. BP

neural networks encoded English directly word by word, and although it can also explore the hidden laws, it cannot accurately describe the influence of word order on word meaning, which leads to an increase in word error rate; RNN took into account the influence of the previous moment in the forward calculation, i.e., the influence of the previous word in this study; in the LSTM-RNN-based machine translation, LSTM, as a variant of RNN, can also summarize the influence of word order, while the forgetting gate, input gate, and output gate units introduced by it can filter the unimportant words among them and improve the accuracy, and the attention mechanism introduced by RNN in the decoder also made the decoding focus more on to the main information.

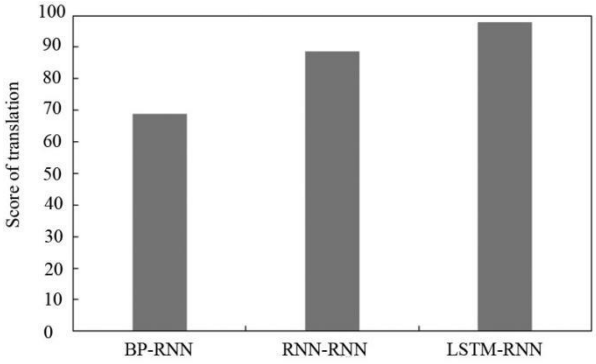

Figure 4: Translation scores of three machine translation algorithms for English speech recognition results.

For the machine translation algorithm, the criterion of whether the translation performance is excellent or not includes the grammatical goodness of the translation as a whole, in addition to the judgment of whether the words of the translation are accurate or not. Therefore, ten professional translators were invited to manually translate the English sentences in the test set and evaluated and scored the machine translation. The results are shown in Figure 4. The translation scores were 68.9 using the BP-RNN algorithm, 88.7 using the RNN-RNN algorithm, and 97.8 using the LSTM-RNN algorithm. It was seen from Figure 4 that the translation obtained by using the BP-RNN algorithm for machine translation had the lowest rating, and the translation obtained by using the LSTM-RNN algorithm for machine translation had the highest rating. The reason was similar to the above. The BP-RNN algorithm did not consider the influence of word order, resulting in a final translation that was similar to word-by-word translation, with no problem in overall comprehension but difficult to read smoothly; the RNN-RNN algorithm RNN-RNN considered the influence of word order when encoding English through RNN, so the translation was relatively more smooth and had a higher score; the LSTM-RNN algorithm also considered the influence of word order coding because LSTM is a kind of RNN, and the LSTM algorithm in the encoder and the attention mechanism in the decoder effectively highlighted the key points and improved the fluency of translation while reducing the translation computation.

5 Conclusion

In this study, LSTM network was used as the encoding algorithm of the encoder, RNN was used as the decoding algorithm of the decoder, and the attention mechanism was introduced into the decoder. The results are as follows: (1) BP neural network algorithm was used to recognize English speech, and the recognition results were used in machine translation; compared with human recognition, BP neural network had a lower word error rate and less recognition time; (2) the LSTM-RNN algorithm had the lowest word error rate for English speech recognition, the RNN-RNN-based machine translation algorithm had a higher word error rate, and the BP-RNN-based machine translation algorithm had the highest word error rate; (3) the LSTM-RNN-based machine translation algorithm had the lowest rating in the evaluation of professional translators, followed by the RNN-RNN machine translation algorithm and the BP-RNN machine translation algorithm.

References

[1] Graham Y, Baldwin T, Moffat A, Zobel J (2015). Can machine translation systems be evaluated by the crowd alone. *Natural Language Engineering*, 23, pp. 1-28. https://doi.org/10.1017/S1351324915000339.

[2] Wang Y, Li J, Gong Y (2015). Small-footprint high-performance deep neural network-based speech recognition using split-VQ. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4984-4988. https://doi.org/10.1109/ICASSP.2015.7178919.

[3] Wu C, Karanasou P, Gales M J F, et al (2016). Stimulated Deep Neural Network for Speech Recognition. *INTERSPEECH*, pp. 400-404. https://doi.org/10.21437/Interspeech.2016-580.

[4] Luong M, Sutskever I, Le Q, Vinyals O, Zaremba W (2015). Addressing the Rare Word Problem in Neural Machine Translation. *Bulletin of University of Agricultural Sciences and Veterinary Medicine Cluj-Napoca. Veterinary Medicine*, 27, pp. 82-86. https://doi.org/10.3115/v1/P15-1002.

[5] Lee J, Cho K, Hofmann T (2017). Fully Character-Level Neural Machine Translation without Explicit Segmentation. *Transactions of the Association for Computational Linguistics*, 5, pp. 365-378. https://doi.org/10.1162/tacl_a_00067.

[6] Choi H, Cho K, Bengio Y (2017). Context-Dependent Word Representation for Neural Machine Translation. *Computer Speech & Language*, 45, pp. 149-160. https://doi.org/10.1016/j.csl.2017.01.007.

[7] Chorowski J, Bahdanau D, Serdyuk D, Cho K, Bengio Y (2015). Attention-Based Models for Speech Recognition. *Computer Science*, 10, pp. 429-439.

[8] Miao YJ, Gowayyed M, Metze F (2015). EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding. *2015 IEEE Workshop on Automatic Speech Recognition and*

Understanding (ASRU), pp. 167-174. https://doi.org/10.1109/ASRU.2015.7404790.

[9] Schwarz A, Huemmer C, Maas R, Kellermann W (2015). Spatial diffuseness features for DNN-based speech recognition in noisy and reverberant environments. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4380-4384. https://doi.org/10.1109/ICASSP.2015.7178798.

[10] Kipyatkova I (2017). Experimenting with Hybrid TDNN/HMM Acoustic Models for Russian Speech Recognition. https://doi.org/10.1007/978-3-319-66429-3_35.

[11] Yoshioka T, Karita S, Nakatani T (2015). Far-field speech recognition using CNN-DNN-HMM with convolution in time. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4360-4364. https://doi.org/10.1109/ICASSP.2015.7178794.

[12] Wang Y, Bao F, Zhang H, Gao G (2017). Research on Mongolian Speech Recognition Based on FSMN. https://doi.org/10.1007/978-3-319-73618-1_21.

[13] Alam MJ, Gupta V, Kenny P, Dumouchel P (2015). Speech recognition in reverberant and noisy environments employing multiple feature extractors and i-vector speaker adaptation. *Eurasip Journal on Advances in Signal Processing*, 2015, pp. 50.

[14] Brayda L, Wellekens C, Omologo M (2015). N-best parallel maximum likelihood beamformers for robust speech recognition. *2006 14th European Signal Processing Conference*, pp. 1-4.

[15] Hammami N, Bedda M, Nadir F (2012). The second-order derivatives of MFCC for improving spoken Arabic digits recognition using Tree distributions approximation model and HMMs. *International Conference on Communications and Information Technology*, pp. 1-5. https://doi.org/10.1109/ICCITechnol.2012.6285769.