International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-9 Issue-1, May 2020

Realistic Handwriting Generation using Generative Adversarial Networks (Rnn)

Rajakumar K, Spreeha Dutta, Bipsa Nayak, Bindhiya N Koliwad

***Abstract:** Generating handwritings of different kinds is quite a challenging task, an area in which not much work has been done yet. Though there has been substantial research done in the area of text recognition, the opposite of handwriting generation. Handwriting generation can prove to be extremely useful for children from blind schools where their speech can get converted into text and be used to generate handwritings of different kinds for them. Handwriting generation also has an important role in field of captcha generation. Our study exhibits in what way recurrent neural networks (RNN) of the type Long Short Term Memory (LSTM) could be used in order to create a composite sequence with structure covering a long range. We propose to use that the Generative Adversarial Network algorithm can be used to generate more realistic handwriting styles with better accuracy than other algorithms. Here, we will be trying to predict one point of data at a time. Our approach is shownfor text, where the type of data is discrete. It can also be used for online handwriting, that is real-valued data. It will then be further drawn out to handwriting generation. The created network will be conditioning its predictions based on a sequence of text. We will be using the resulting system to generate highly realistic cursive handwriting in a wide variety of styles. Experiments that have been carried out on online handwriting databases that are public predict that the method that has been proposed can be used to achieve satisfactory performance, the resultant writing samples achieved a high level of similarity with original samples of handwriting.*

***Keywords :** Recurrent Neural Networks, Long Short Term Memory, Generative Adversarial Networks, handwriting generating, online handwriting.*

I. INTRODUCTION

The blind often face problems when they go to schools, colleges and they need special schools for them where teachers understand Braille. We feel that the method of generating different styles of realistic handwriting can be used to solve this very problem, so that teachers can correct the papers of the blind children just like others and they don't have to learn Braille separately. And hence uniformity in teaching and corrections can be maintained this way.

We have used recurrent neural networks (RNNs) for this purpose, which are a rich class of dynamic models that have been known to generate sequences in various disciplines as diverse as music and motion capture data. So we thought of using this very model for generating handwriting too. We trained RNNs for generating sequences by filtering real valued sequences of data one at a time and then anticipating what would come next. We assume that all the predictions that we have made are probabilistic in nature and new sequences which can be generated from a trained network by sampling through multiple iterations from the output given by the network. Then for the input of the next step, we just entered the sample obtained from the previous step. To put it a different way, the network will be treating it's inventions like real ones. Although the nature of the network deterministic, stochasticity has been entered in them by picking samples which helps it distribute over the generated sequences. This distribution is due to the internal representation of the network and is conditional and therefore its distribution that we predict is hugely dependent on the inputs we supply to it from the earlier iterations.

RNNs are fuzzy, this implies that they don't utilize what has been obtained from the training data to match the same templates and make further anticipations. Similar to other neural networks, the utilize it's internal representation and an interpolation is performed between the available training samples. That is what sets our approach apart from compression models or n-gram models, for example Prediction by Partial Matching which have distributions that have been obtained by counting total number of similarities between previous tested examples along with the training set. What we get as the conclusion is apparent from the examples that we have used in our study that shows that RNNs aren't like other algorithms dependent on templates, they synthesize. Next the training data is reconstituted the training in a complicated way, and that ensures that there is no repetition in what is being generated. This provides diversity in the handwriting styles and ensures it can be used by many, this diversity is not supported by most other sequence generation algorithms. Furthermore, fuzzy predictions are not affected by dimensionality and hence they are much better at modeling real-valued or multivariate data than exact matches which works as an advantage.

Manuscript received on April 02, 2020.
Revised Manuscript received on April 15, 2020.
Manuscript published on May 30, 2020.
* Correspondence Author
Rajakumar k*, Computer Science, Vellore Institue of Technology, Vellore, India. Email:rajakumar.krishnan@vit.ac.in
Spreeha Dutta*, Computer Science, Vellore Institue of Technology, Vellore, India. Email:spreehadutta@gmail.com
Bipsa Nayak*, Computer Science with specialization in Bio informatics,Vellore Institue of Technology , Vellore, India. Email:bipsa1998@gmail.com
Bindhiya N Koliwad*, Computer Science with specialization in Bio informatics,Vellore Institue of Technology , Vellore, India. Email:bindhiyan.koliwad2016@vitstudent.ac.in

II. SCOPE ANALYSIS

In order to test whether the said prediction network can also be used to generate convincing realistic sequences, we applied the same to online handwriting data.

Online handwriting means that the writing is recorded as a sequence of pen-tip locations whereas in oine handwriting, only the images of pages are available. Online handwriting was taken by us sequence generation because of its low dimensionality, it has for every point of data, two real valued numbers and also because it is easy to visualize. The data that we have used for this paper has been collected from an online handwriting database called IAM-OnDB . It consists of many lines of handwriting that have been taken from 221 writings. They were recorded using a smartboard. Position of the pens of the writers have been traced using an infra-red device that had been put in the corner of the board. The input would comprise of coordinates (x and y) according to points where the pin tips were placed in the sequence and all the times when the pen would be removed from its previous position. The errors encountered in recording of the data coordinates (x, y) were made right by interpolation to account for the readings that were missing. All the iterations whose length would exceed a certain limit were not considered. Apart from that, no other data preprocessing was used and the both the X and Y co-ordinates as well as the points to mark the end of each stroke can be predicted by the network. That is different from other approaches in character recognition and handwriting generation that hugely depend on feature extraction techniques that can prove to be costly. We abjured such techniques since they minimise the difference in the data by normalisation the size and slant of every character.

Anticipating where the pen traces would be at a point in time gives the network the maximal capacity to create a new kind of handwriting everytime, but a downside of it is that it a lot of memory is used up. A letter on an average occupies more than twenty five timestamps and an a line on an average occupies around seven hundred. A challenge is to predict the delayed strokes like for i's and t's the dots and dashes are added after the remaining word has been written down. Every line in our experiment has been taken as a different sequence and alpossible dependences between consecutive lines were not considered. So that we can increase the quantity of training data, we used a set for training, testing set and bigger of the authorization sets to train it and the smaller authorization set was used in the process of premature stopping if required. The main challenge in the application of the prediction network to the online handwriting data was to establish a predictive distribution suitable for real valued inputs.

III. PREDICTION NETWORK

The prediction network shows the basic structure of the RNN which has been used for our paper. We have considered an input vector sequence that is x = x1 ….. xT which is sent through through mean connections. It is transferred onto a stack which has N connected but hidden layers recurrently. This is used to find (r)st from the unseen vector sequences hn = hn1 …. hnT. The output sequence of vector y = y1….yT . Every output vector yt is in turn used as parameters for the predicted distributions Pr(xt+1jyt). It is taken over the inputs which have been entered one after the other(xt+1). Rst element x1 from every sequence of input. It is always a null vector whose value is zero. The network provides a and anticipates x2. So the network is deep so that piece of information can pass in both directions that is either vertically or horizontally. It is done with the help of computation graphs. These are then computed on by many weight matrices and nonlinearities.

IV. TEXT PREDICTION

Text data maybe discontinuous in nature. It is classically represented to RNN with input vectors of the type onehot. Suppose we have K text classes and the class k is entered in at a time t. Xt is taken a length K vector where entries are zero. Only the kth is one. Pr(xt+1jyt) represents a multinomial distribution. We parameterize it by a softmax function at the time of output.

V. HANDWRITING GENERATION

We have used recurrent neural networks (RNNs) for this purpose, which are a rich class of dynamic models that have been known to generate sequences in various disciplines as diverse as music and motion capture data. So we thought of using this very model for generating handwriting too. We can process the RNNs for creating series of handwritten text by processing data sequences in real time a step at a time. It is then used for anticipating what might come next. We have made an assumption that the predictions are new sequences and are probabilistic. This can be used to create a distribution of output from a network that we train through sampling. Then we enter in the sample as input at the next step. To put it in a different way, we make the network treat it’s innovations as if it was real. Although this nature of the network is passive, we enter stochasticity into the samples that helps introduce a sequence distributions. That generated distribution is again depencdent because of the internal framework of the network. Hence its distribution that is predicted is dependent on what is input from the previous steps.

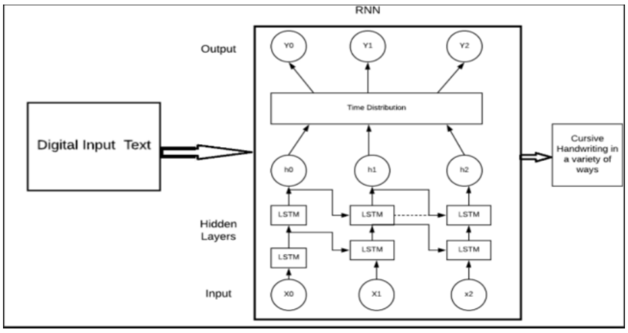
VI. HANDWRITING SYNTHESIS

Handwriting synthesis is the creation of handwriting. The algorithm of Generative Adversarial Networks (GAN) can be used to describe an accession which allows a network that predicts and generates sequences of data that are dependent on sequences that are annotated. The series of results are thoroughly convincing so much so much so that they cannot be differntiated from realistic handwriting. What works as a bonus point is that, this realistic result is obtained while getting different kinds of handwriting.

VII. FRAMEWORK

- The Generator will take a stochastic input. This will be further used to generate a sample of data. G(z) or the generator network takes a input z from the probability distribution that is p(z). It is then used to create a set of data that is then entered into D(x). D(x) is the discriminator network..
- The input i.e., x is taken from pdata(x). pdata(x) is the

distribution for the real valued data. The binary classification problem is solved using D(x). The function used is sigmoid function that gives an output in the range from 0 to 1.

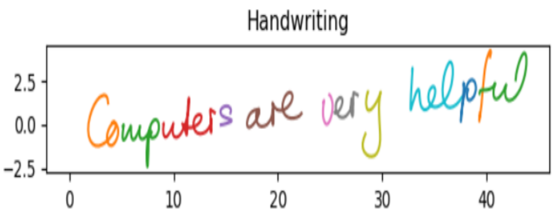


Architecture Diagram of the text generation system

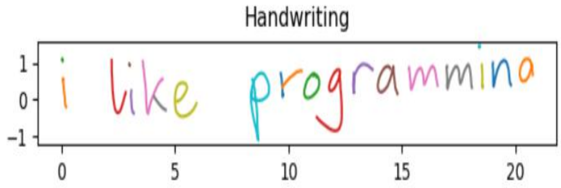
Text is input which is then fed into a recurrent neural network of the type LSTM. Partial matching is performed between the training set and outputs of the previous layers to generate the next layers. We are using this method since they don't rely on a an exact template which helps in generating a varied range of outputs. Interpolation is performed to construct a high dimensional internal structure. The pen positions of the originally recorded handwriting are used to generate an offset of 0.75 from it to generate the next layers of output. GAN is used where data is fed into a discriminator network. Another input is taken from a real valued data. The inputs are then fed into a sigmoid function to give outputs in the variation from 0 to 1. As a final result different styles of highly realistic handwriting are generated after successive iterations.

A. Figures and Tables

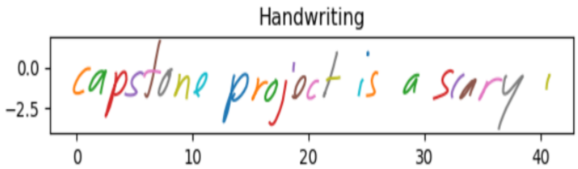
1. Handwriting generation where the offset has been taken from 0 to +2.5



2. Handwriting generation where the offset has been taken from -2.5 to +2.5



3. Handwriting generation where the offset has been taken from -1 to +1



4. Handwriting generation where the offset has been taken from -2.5 to 0

VIII. RESULT AND DISCUSSION

Different varieties of highly realistic handwriting are generated with legibility and accuracy very near to original handwriting. This paper demonstrates the capacity of recurrent neural networks of the type LSTM can be used to create both kinds - distinct and realistic data sequences that have complicated and structures having a long-range and is used to predict the next iteration. It then introduces a new convolutional mechanism which enables an RNN to condition and anticipate its predictions on a supplementary annotation sequence. This method using Generative Adversarial Networks can be used to synthesize diverse and realistic samples of online handwriting. This paper demonstrates how certain examples could be biased towards more legibility and how well the style of a writer can be modelled.

The methodology we have proposed have produced accurate results for images with handwritten text of different slant and style, various sizes along with alignment by varying the background. It can be developed using MATLAB. It can be tested for many sets of sample images. They comprise handwritten text from Intel dual core computer. This methodology is beneficial because it uses various features for training. Training is done using character geometry and gradient technique is used.

IX. CONCLUSION

The blind often face problems when they go to schools, colleges and they need special schools for them where teachers understand Braille. We feel that the method of generating different styles of realistic handwriting can be used to solve this very problem, so that teachers can correct the papers of the blind children just like others and they don't have to learn Braille separately. And hence uniformity in teaching and corrections can be maintained this way.

Several directions for future work can be explored from this. One is the application of the network to speech synthesis, which is likely to be more difficult in processing than handwriting synthesis because of the greater dimensionality of the data points. Another is to get a better insight into the internal representation of the data and to utilize this to change and see its effects on the sample distribution. Another area to explore can be to develop a mechanism to extract extract high-level annotations from long sequences of data automatically.

REFERENCES

1. Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is dicult. IEEE Transactions on Neural Networks, 5(2):157{166, March 1994.
2. C. Bishop. Mixture density networks. Technical report, 1994.
3. C. Bishop. Neural Networks for Pattern Recognition. Oxford University Press, Inc., 1995.
4. N. Boulanger-Lewandowski, Y. Bengio, and P. Vincent. Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. In Proceedings of the Twenty-nine International Conference on Machine Learning (ICML'12), 2012.
5. J. G. Cleary, Ian, and I. H. Witten. Data compression using adaptive coding and partial string matching. IEEE Transactions on Communications, 32:396{402, 1984.
6. D. Eck and J. Schmidhuber. A rst look at music composition using lstm recurrent neural networks. Technical report, IDSIA USI-SUPSI Instituto Dalle Molle.
7. F. Gers, N. Schraudolph, and J. Schmidhuber. Learning precise timing with LSTM recurrent networks. Journal of Machine Learning Research, 3:115{143, 2002.
8. A. Graves. Practical variational inference for neural networks. In Advances in Neural Information Processing Systems, volume 24, pages 2348{2356. 2011.
9. A. Graves. Sequence transduction with recurrent neural networks. In ICML Representation Learning Worksop, 2012.
10. A. Graves, A. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In Proc. ICASSP, 2013.

AUTHORS PROFILE

Dr. K. Rajkumar, is an Associate Professor in the School of Computer Science and Engineering at VIT University Vellore with over 17 years of experience. He is a Ph.D in image processing. He has worked on a project from ISRO. He has published more than 50 research papers in Thomson Reuters, IEEE, ScienceDirect, many conferences and several other National as well as International Scopus indexed journals.

Bipsa Nayak, is an undergraduate student pursuing Computer Science with specialization in Bio Informatics at Vellore Institute of Technology, Vellore,
Tamil nadu, India. She has done several projects like Automation of User-Interface, Restaurant Management System, Phone-mode secured conversion.Her interests include DBMS, SQL, Web Development, Internet of Things and Automation.

Spreeha Dutta, is an undergraduate student studying Computer Science and Engineering from VIT University, Vellore. She has done several projects like detection of malicious web pages, fake review generation and detection using RNN based text generators to name a few. Her interests include machine learning, natural language processing and web development.

Bindhiya N Koliwad, is an undergraduate student studying Computer Science with specialization in bioinformatics from VIT University, Vellore. She has done several projects like attendance aided by face recognition, restaurant management system, fake review filter for online shopping , forest fire prediction and detection using IoT .Her interests include artificial intelligence , image processing and human computer interaction.