Optimizing CNN and RNN For Free-Hand Text Classification

Sonika Jha

Feroze Gandhi Institute Of Engineering And Technology

sonikajha2711@gmail.com

Abstract

People use sketches to express and record their ideas. Free-hand sketches are usually drawn by non-artists using touch sensitive devices rather than purpose-made equipments; thus, making them often highly abstract and exhibit large intraclass deformations. This makes automatic recognition of sketches more challenging than other areas of image classification because sketches of the same object can vary based on artistic style and drawing ability. Here We aim at using the both types of neural networks and optimizing them.The model takes the stroke data and 'preprocesses' it a bit using 1D convolutions and then uses two stacked LSTMs followed by two dense layers to make the classification. The model can be thought to 'read' the drawing stroke by stroke.

1. Introduction

Sketching is a natural way of expressing some types of ideas. It conveys information that can be really hard to explain using text, and at the same time it does not require a tremendous amount of effort. It is also a suitable communication tool for children or illiterate people. As human-computer interaction moves towards easier and more high level languages, sketching will certainly continue to have its place in all sorts of applications, including image [Eitz et al.2018] and shape [Eitz et al. 2012b] retrieval, shape modeling [Olsenet al. 2009] [Schmidt et al. 2006] and character animation [Daviset al. 2003].In a different perspective, sketching is possibly the most high-level and sparse type of visual media that can be understood by humans, which makes it an interesting object of study in computer vision.Why we can understand sketches so well and whether we can teach computers to do the same are research questions still in need of an answer.We present a technique for sketch classification that performs significantly better than the state-of-art.

Using the TU-Berlin sketch benchmark [Eitz et al. 2012a], we achieve a recognition rate of68.9%, which is an absolute improvement of13%over their results.Also, with these results, we come close to the accuracy achieved by humans, which is73%. Unfortunately, it might be too soon to say computers are performing comparably to humans in this task. Before looking into how humans understand sketches, we need to determine when it is possible for humans to understand sketches. More specifically, when does a sketch contain enough information to allow it to be unmistakably put into a specific category? As can be seen in Figure 1, this is not always the case. We discuss the specific reasons for the low performance achieved by humans in the TU-Berlin benchmark, and modify it to make it less sensitive to the types of problems we found.Finally, we perform a data-driven analysis of sketches based on the classification method. We get sound results when determining which sketches are good/poor representatives of a class, performing consistently better than [Eitz et al. 2012a]. Then, we analyze which parts of the sketch are most important for recognition.

These results usually describe the most basic features of a sketch, and provide intuition on how the computer understands the different categories. The two types of neural networks studied are convolutional neural networks (CNN) and recurrent neural networks (RNN). CNNs are typically used for image classification. 2D-convolutional layers, the type of layers used in the constructed CNN, interpret images as 2D-objects and preserves the grid-like structure of the pixels. RNNs are typically used for finding patterns in sequential data. There is a connection between the chain-like structure of the RNN and the sequential structure of the data it is suitable for.

2. Theory

2.1 Parametric Models

The aim of machine learning is to find a model that describes the relationship between the input and output data of a system by learning from the input data. The model f needs to be estimated in the following expression

$$Y=f(X) + \varepsilon \qquad [1]$$

Where Y is the output data, f is the model, X is the input data and ε is an error term. An example of a model could be Ohm's law which is a linear model that describes the relationship between the voltage and the current for a resistor. In this example the current could be considered as the input data and the voltage as the output data. Another example is the case of classifying sketches, as in this study, where the input data is the sketches and the output data is the class. Parametric models are a type of mathematical model which describes the relationship between the input and output data using finite-dimensional parameters. Linear regression, logistic regression and neural networks are some examples of parametric models.

3.2 Supervised Learning

Within machine learning, the data that is used for finding the model is often split into training and evaluation data. Labelled data contains both the input and the corresponding output and can be used for finding the model, also known as supervised learning. Based on the model predictions can be made for new, unlabelled input data .

3.3 Classification Results

We now discuss the results we obtained applying the technique described above to the TU-Berlin sketch benchmark.The benchmark consists of 250 object categories,with 80 sketches each. The category of each sketch was defined by the person drawing it. After the sketch creation, there was a phase where humans tried to recognize the sketches - achieving an accuracy of 73% .Results. The test settings were chosen to be consistent with thesetup from [Eitz et al. 2012a]. We test 3 different patch sizes, with and without using spatial pyramids, and 10 subset sizes (the subset is the part of the dataset that will be used in each test). We divide the subset in 3 parts: 2 to be used as the training set and 1 as the testing set. The results reported are the average accuracy of three runs, onewith each part being used as the testing set. Figure 2 shows theresults we obtained for patch sizes of 16×16 And 24×24. We omitted the inferior results (8×8) for better clarity.The results demonstrate that Fisher vectors significantly improve over the state-of-art. Also, with enough training samples, accuracy is now close to human performance. Our usage of bigger SIFT patches is also responsible for an important amount of the improvement (the best results for FV with 8×8 patches was 63.1%, on the subset with 80 images). The usage of spatial pyramids also gave us better accuracy - differently from what was reported by [Eitz et al.2012a]. We suppose their features are big enough to already encode most of the spatial information. Finally, note that they show in their paper the difference between soft and hard assignment. In our experiments, we only used soft assignment.

4. Conclusions

The RNN achieved a higher accuracy than the CNN. From this result it can be concluded that interpreting the sketches from the Quick, Draw! data set as sequences gives a higher accuracy than interpreting them as pixels when constructing fairly shallow neural networks. In order to achieve a higher accuracy the amount of training and evaluation data per category could be increased. Another improvement could be to pre-process the data set by removing sketches which can be considered as noise, see appendix for example. Removing unnecessary image processing in the implementation of the CNN might also be a way of improving the results. More categories could be added and the performance of the networks in each category could be evaluated. To achieve a more reliable result one could perform an iteration in each step of the optimisation process and average the result.

5. References

[1] M. Eitz, J. Hays, and M. Alexa. How do humans sketch objects?ACM Trans. Graph. (Proc. SIGGRAPH), 31(4):44:144:10, 2012.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition.IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[3]Goodfellow I, Bengio Y, Courville A.Deep Learning[Internet], MIT Press; 2016 [cited 2018 Mar 28] Available from: http://www.deeplearningbook.org

[4]Lindsten F, Svensson A, Wahlström N, Schön T.Lecture Notes[Internet]. Uppsala University Department of Information Technology [updated 2018 Feb 02; cited 2018 Mar18] Available from:http://www.it.uu.se/edu/course/homepage/sml/literature/lecture_notes.pdf

[5]Dahal P.Classification and Loss Evaluation - Softmax and Cross Entropy Loss[Internet].Deep Notes, 2017 Maj 28 [cited 2018 Apr 01] Available from:https://deepnotes.io/softmax-crossentropy

[6]P. Kingma D, Lei Ba J. Adam: A Method for Stochastic Optimization [Internet]. Proceedings of International Conference for Learning Representations, 2015, San Diego [updated 2017 Jan 30; cited 2018 Apr 17] Available from:https://arxiv.org/pdf/1412.6980.pdf

[7]Wahlström N. Statistical Machine Learning 2018, Lecture 8 - Deep learning and neural Networks ,[unpublishedlecturenotes].UppsalaUniversity;notesprovidedatlecturegiven 2018 Feb 22. [cited 2018 Apr 03] Available from :http://www.it.uu.se/edu/course/homepage/sml/lectures/lecture8_handout_updated.pdf

[8]Schön T.Statistical Machine Learning 2018, Lecture 5 - Bias-variance tradeoff, crossvalidation,[unpublishedlecturenotes].UppsalaUniversity;notesprovidedatlecturegiven 2018 Jan 31. [cited 2018 Maj 21] Available from :http://www.it.uu.se/edu/course/homepage/sml/lectures/lecture5_handout.pdf