Urdu Handwriting Recognition Using Deep Learning

Shehryar Malik [1*], M. Naeem Maqsood[1], Abdur Rehman Ali[1]
Ubaid Ullah Fiaz[1], Qurat-ul-Ain Akram[2]

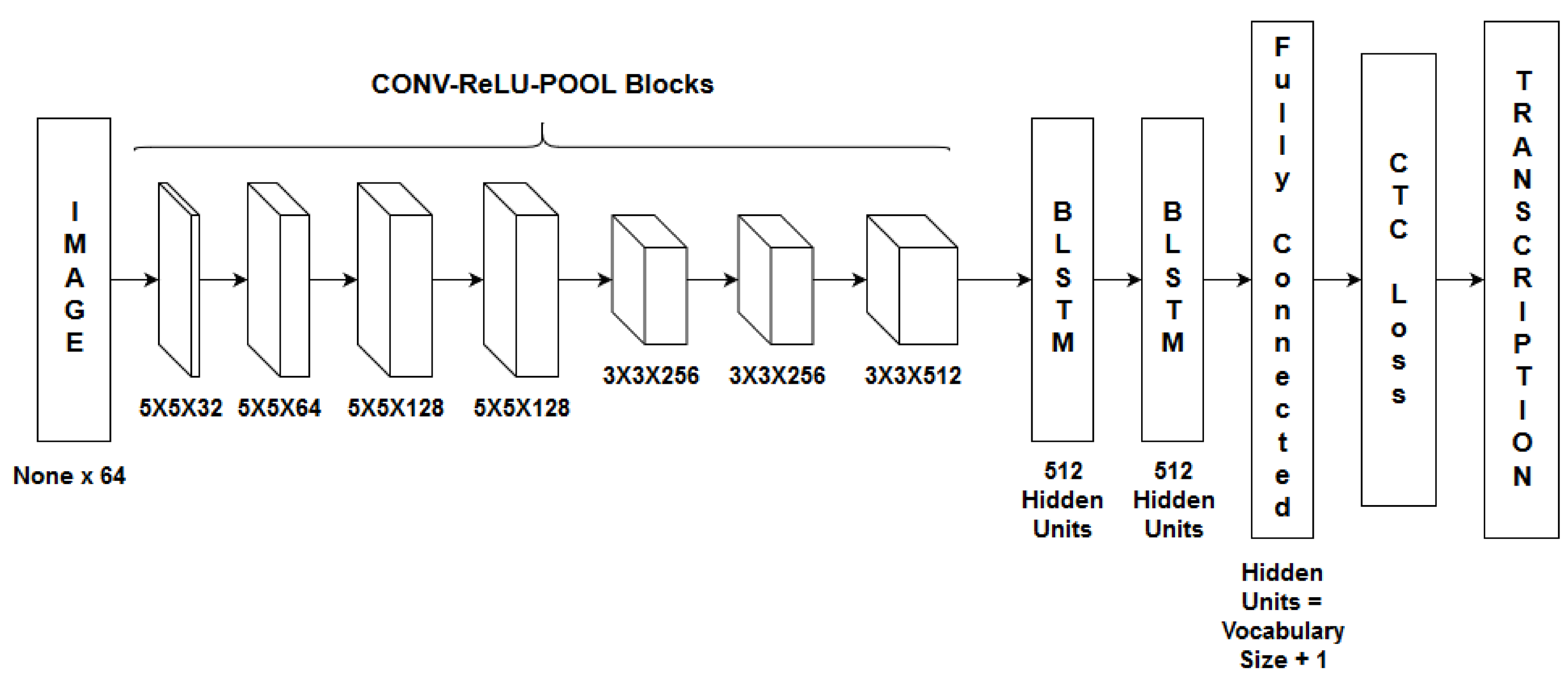
1 Department of Electrical Engineering, University of Engineering and Technology, Lahore, Pakistan
2 Centre for Language Engineering, Al-Khwarizmi Institute of Computer Science, Lahore, Pakistan
*shehryarmalik04@outlook.com

Introduction

Optical Character Recognition (OCR) aims to recognize text in images. Recent breakthroughs in Deep Learning have revolutionized OCR systems for languages such as English. However, their impact on Urdu has been minimal. This project aims to bridge this gap. We develop a new dataset comprising of 15,000 images of Urdu handwritten text lines and use it to train two existing Deep Learning architectures. The first is the standard CNN-RNN architecture with the Connectionist Temporal Classification (CTC) objective function. The second is an Attention-Based Encoder-Decoder architecture with a Cross-Entropy (CE) objective function. We also incorporate an n-grams based language model to further improve performance.

Methodology

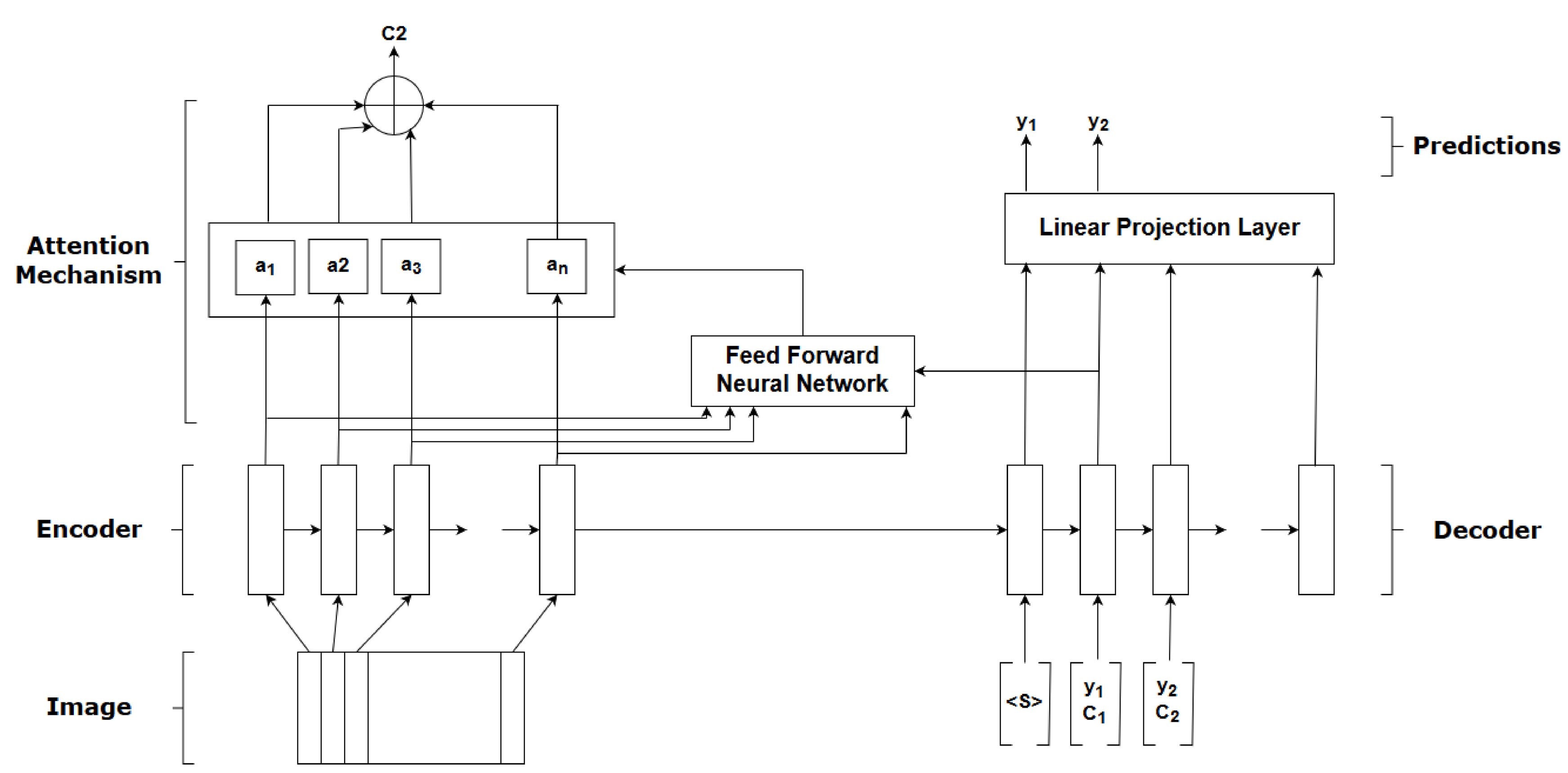
(1) CNN-RNN-CTC Architecture



We minimize the following objective function given a set S consisting of images x and corresponding labels z:

$$O = -\ln \prod_{(x,z)\in S} p(z|x)$$

(2) Attention-Based Encoder-Decoder Architecture



The encoder is a bidirectional two-layer stacked layer-norm LSTM and the decoder is a unidirectional two-layer stacked layer-norm LSTM.

We minimize the following objective function for each image:

$$O = -\ln \prod_i p(y_i|y_1, \dots, y_{i-1}, I)$$

(3) N-Grams Language Model

During decoding of the outputs of the models above, we can incorporate a language model by calculating appropriate n-gram probabilities. Consider a sequence $\{w_1, \dots, w_n\}$. Then:

$$p(w_i|w_{i-n+1}^{i-1}) = \frac{N\{w_{i-n+1}, \dots, w_{i-1}, w_i\}}{N\{w_{i-n+1}, \dots, w_{i-1}\}}$$

Where $N(°)$ is the number of times ° appears in corpus. We train a ligature-based N-Grams model on an Urdu corpus of about 10,000 lines. To account for new ligatures that may be encountered during testing and deployment, we use the Kneser-Ney Smoothing recursive equation [5]:

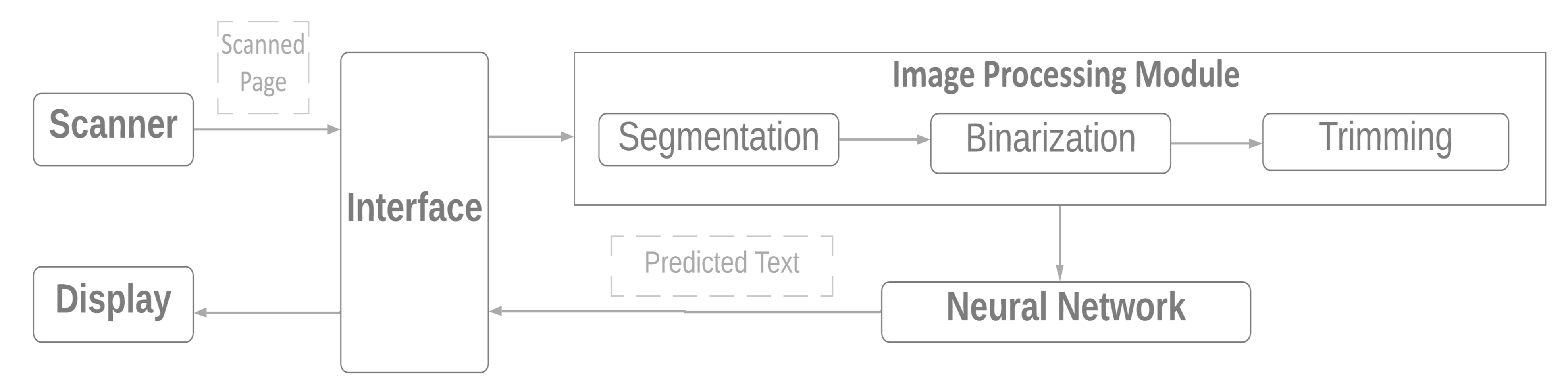
$$p(w_i|w_{i-n+1}^{i-1}) = \frac{\max(c_{KN}(w_{i-n+1}w_i) - d, 0)}{c_{KN}(w_{i-n+1}^{i-1})} + \lambda(w_{i-n+1}^{i-1})p(w_i|w_{i-n+2}^{i-n})$$

which bases lower-gram estimates on the number of different contexts a ligature appears in.

Data Collection

- Comprises of 15,164 images of distinct text lines scanned at 300 dots per square inch (dpi)
- Written by 490 different writers
- Selected 10,000 lines from Urdu literature that maximized unigrams, bigrams and trigrams coverage
- The resulting dataset includes 61, 1,674 and 13,497 unigrams, bigrams and trigrams respectively and covers 140 different Urdu letter shapes
- The dataset will be publicly available for further research

Block Diagram

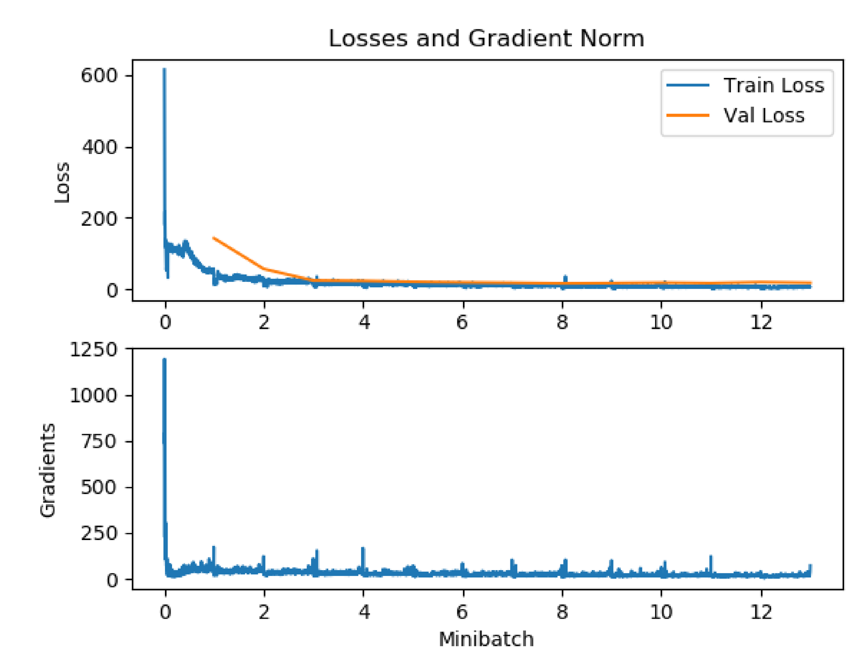


Results

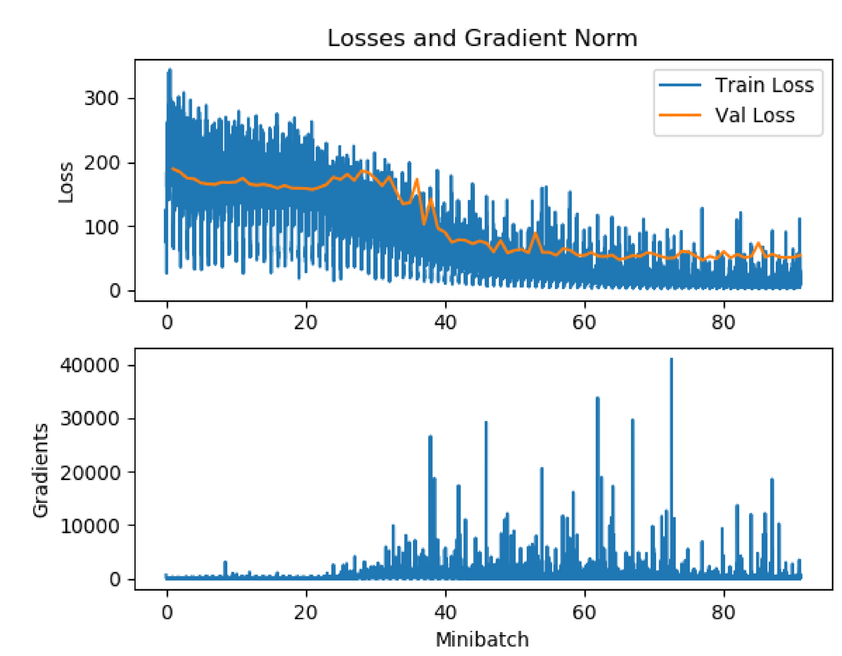
We calculate the Levenshtein (edit) distance to find the accuracy of the model:

$$\text{Accuracy} = \frac{\text{Number of Insertions, Deletion and Substitutions Required}}{\text{Length of Label}} \times 100\%$$

	CNN-RNN-CTC	Attention-Based Encoder-Decoder
Greedy Search	88.37	85.75
Beam Search	88.57	**87.06**
Beam Search with LM	**91.20**	-
On English (IAM dataset) [6]	93.8	91.9



CNN-RNN-CTC



Attention-Based Encoder-Decoder

For language modelling we use a trigram model and achieve a perplexity of 47.621 on the held-out set.

Sample Outputs

نے ان کی کسی شادع پہ کوئی گھونسلل بنایا۔

CNN-RNN-CTC

نے ان کی کسی شاخ پہ کوئی گھونسلہ بنایا۔

CNN-RNN-CTC with Language Modelling

اے ان کی کسی شادیہ کوئی گھونسلبنایا۔

Attention-Based Encoder-Decoder

Visualization of Attention Mechanism

Conclusions and Future Work

Experiments on the newly-created dataset have yielded results comparable to those on English datasets. Future work may include experimentation with newer architectures on this dataset. The dataset itself may also be extended to include images of multilingual handwritten texts, which is common to documents of languages such as Urdu.

Acknowledgments

This Centre for Language Engineering, Al-Khawarizmi Institute of Computer Science, University of Engineering and Technology, Lahore provided technical and financial support for this project.

References

1. Hochrieter, S. & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Comput. 9, 8, 1735-1780.
2. LeCun, Y. K., Haffner, P., Bottou, L., & Bengio, Y. (1999). Object Recognition with Gradient-Based Learning. In Shape, Contour and Grouping in Computer Vision. Springer-Verlag, London, UK, UK, 319-.
3. Graves, A., Fernández, S., Gomez, F., & Schmidhuber, J. (2006). Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks. In Proceedings of the 23rd ICML, 369-376.
4. Bahdanau, D., Cho, K., Bengio, Y. (2015) Neural Machine Translation by Jointly Learning to Align and Translate. arXiv:1409.0473. In International Conference on Learning Representations.
5. Kneser, R. and Ney, H. (1995). Improved backing-off for M-gram language modeling. In International Conference on Acoustics, Speech, and Signal Processing, Vol. 1, 181-184.
6. Chowdhury, A., Vig L. (2018). An Efficient End-to-End Neural Model for Handwritten Text Recognition. arXiv:1807.07965v2.