Authorship Identification using Recurrent Neural Networks

Shriya T.P [1] ,J.K. Sahoo [2] ,R.K.Roul [3]

[1] Student, Department of Computer Science, BITS Pilani Goa, email: shriyatp99@gmail.com
[2] Professor, Dept. of Mathematics, BITS Pilani Goa, email: jksahoo@goa.bits-pilani.ac.in
[3] Professor, Dept. of Computer Science, Thapar Institute of Technology, email: raj.roul@thapar.edu

Outline

- Problem Statement and Motivation
- Author identification: Sample Cases
- Datasets
- Implementation
- Experimentation and Results

Problem statement and motivation

- Authorship Identification using Recurrent Neural Networks like Gated Recurrent Unit (GRU) and Long Short Term Memory (LSTM).
- Comparison of word-index based embedding vs pre-trained embeddings
- Useful for tasks of :
  1. cybercrime investigation
  2. psycho-linguistics
  3. political socialization etc

Author Identification : Sample Case

Chapter No.	01	02	03	04	05	06	07	08	09	10	11	12
Lexical	WR	LC	LC	WR	LC	WR	LC	WR	WR	LC	WR	LC
Punctuation	LC	LC	LC	WR	WR	LC	LC	LC	WR	LC	WR	LC
Bag of words	WR	LC	WR	LC	WR	WR	LC	LC	WR	LC	WR	LC

Building Machine Learning Systems with Python. by Willi Richert (WR) and Luis Pedro Coelho (LC).

Source link:http://www.aicbt.com/authorship-attribution/

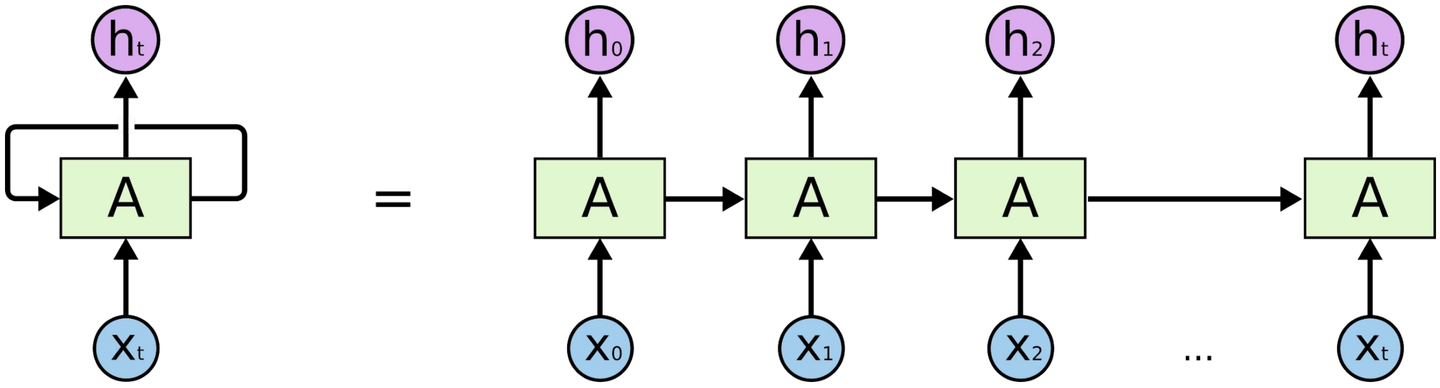
Background

Key characteristics of embeddings:

- Every word/sentence has a unique embedding.
- Embeddings are multidimensional
- For each word/sentence, the embedding captures the “meaning” of the word/sentence.
- Similar words/sentences end up with similar embedding values.

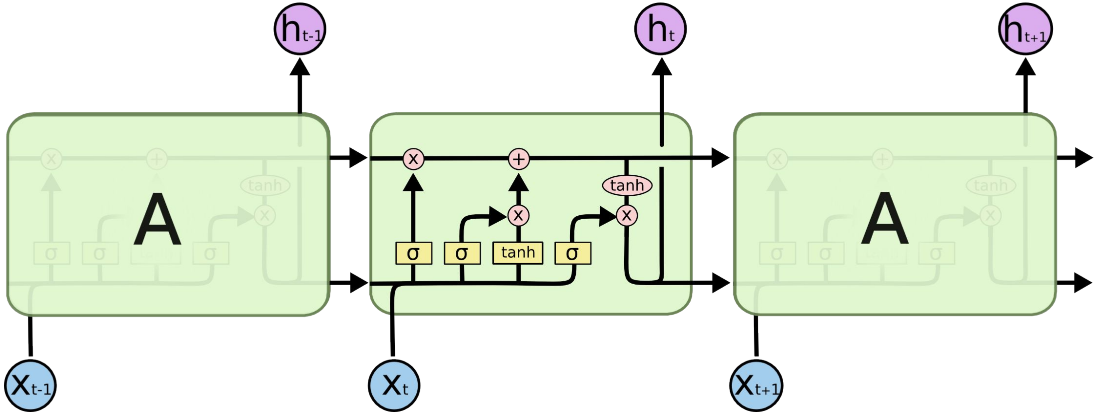
Recurrent Neural Network (RNN)

$$q_t = \sigma(b_{qt} + W_{vq} v_t + W_{qq} q_{t-1})$$



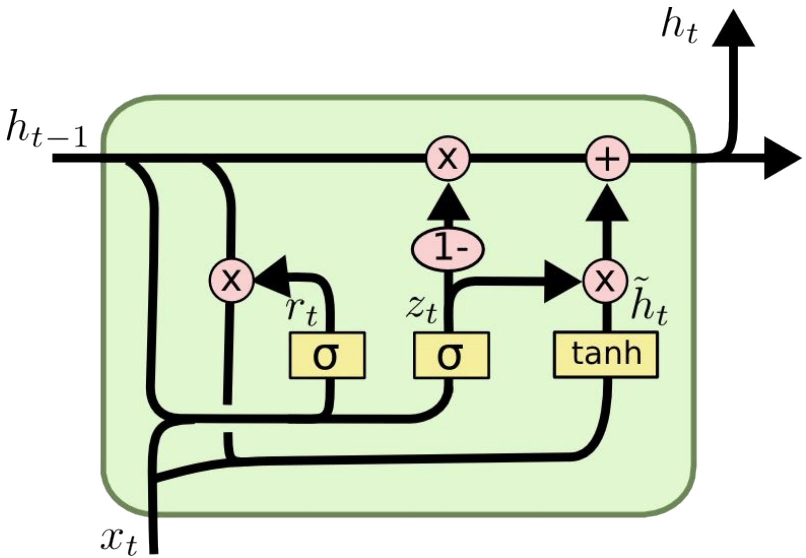
Long Short Term Memory Network (LSTM)

$$\begin{cases} i_t &= \sigma(W^{(i)}x_t + U^{(i)}h_{t-1} + b^{(i)}) \\ f_t &= \sigma(W^{(f)}x_t + U^{(f)}h_{t-1} + b^{(f)}) \\ o_t &= \sigma(W^{(o)}x_t + U^{(o)}h_{t-1} + b^{(o)}) \\ \tilde{c}_t &= \tanh(W^{(c)}x_t + U^{(c)}h_{t-1} + b^{(c)}) \\ c_t &= f_t \circ c_{t-1} + i_t \circ \tilde{c}_t \\ h_t &= o_t \circ \tanh(c_t) \end{cases}$$



Gated Recurrent Unit Network (GRU)

$$\begin{cases} z_t &= \sigma(W^{(z)} x_t + U^{(z)} h_{t-1} + b^{(z)}) \\ r_t &= \sigma(W^{(r)} x_t + U^{(r)} h_{t-1} + b^{(r)}) \\ \tilde{h_t} &= \tanh(r_t \circ U^{(h)} h_{t-1} + W^{(h)} x_t + b^{(h)}) \\ h_t &= (1 - z_t) \circ \tilde{h_t} + z_t \circ h_{t-1} \end{cases}$$



Dataset

The Reuters_50_50 (C50) dataset:

- Subset of the Reuters Corpus Volume I(RCV1) by Reuters, Ltd..
- archive of over 800,000 manually categorized newswire stories
- Corpus consists of 2,500 texts i.e. 50 per author .

The BBC dataset :

- News article dataset, originating with 2,225 documents of five topical
- Class Labels: business, entertainment, politics, sport, tech.

Related Work

Word embedding techniques:

1. **TF-IDF :** term frequency-inverse document frequency.
2. **Pre-trained embeddings**: dense, low-dimensional, and learned from data. Eg: GloVe

Deep learning models for classification:

1. Autoencoders for feature extraction + Support Vector Machine (SVM) classifier
2. CNNs for sentence classification and authorship attribution.

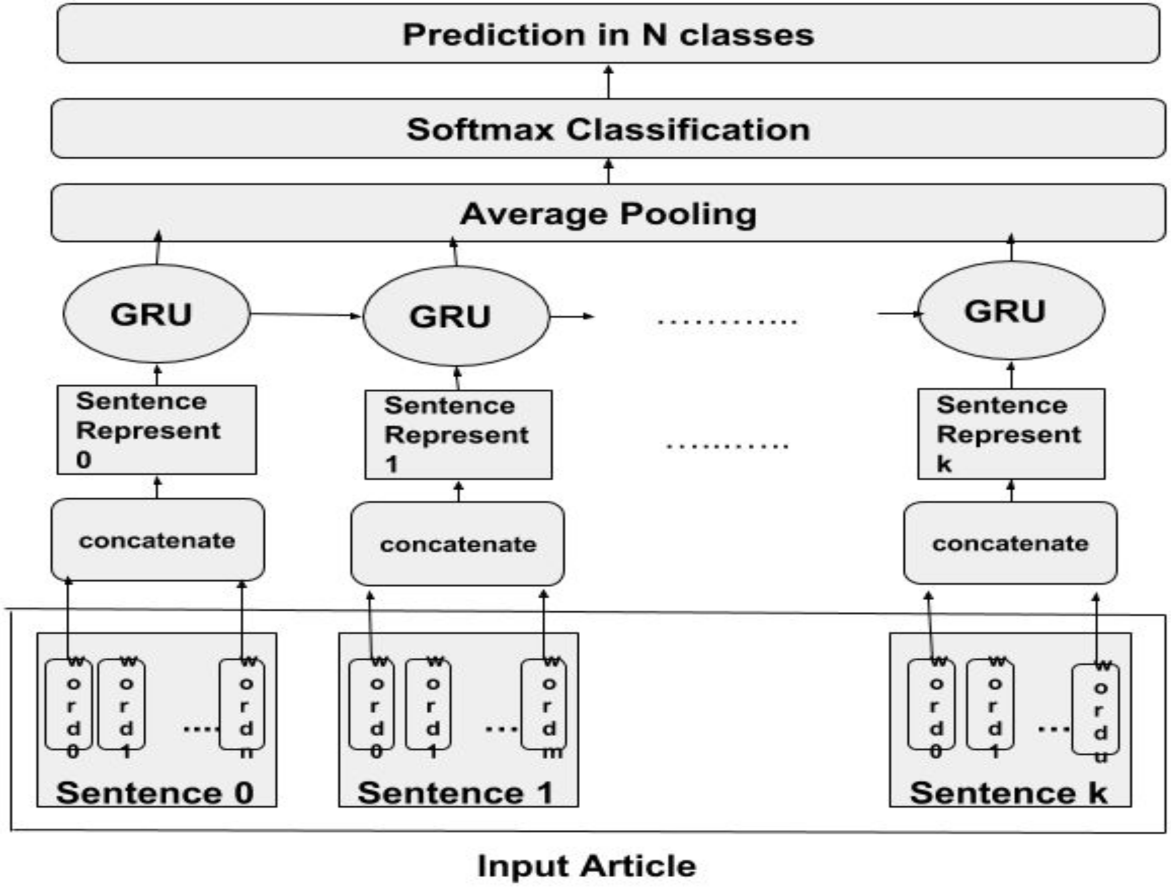
Implementation

Pre-processing for article level GRU network

- Tokens to integer format : word indices from the GloVe look-up table

$$v_k = \left(w_{1,k}, w_{2,k} \ldots w_{i,k} \ldots w_{l_k,k}\right)$$

- Batch input is adopted
- Input truncated if it exceeds specified length.
- Sentence input to subunits- concatenation of word vectors



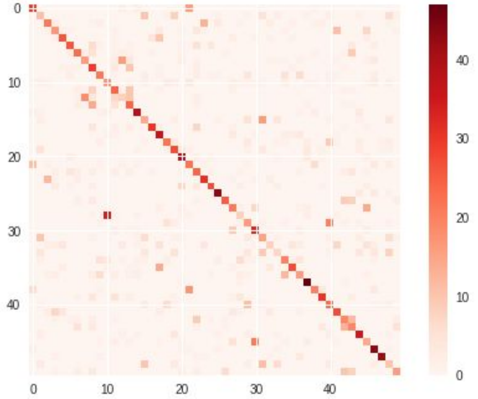
Main structure of the article level GRU model

Experiments and Results

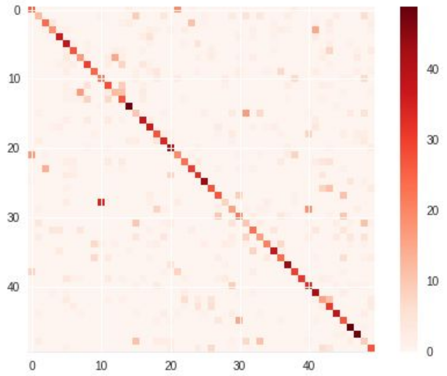
Scenario (With Word Index)	LSTM	GRU
Test Accuracy (C50)	66.67%	78.1%
Train Accuracy (C50)	98.2%	100%

Scenario (With GloVe)	LSTM	GRU
Test Accuracy (C50)	61.47%	69.2%
Train Accuracy (C50)	98.33%	100%

LSTM:



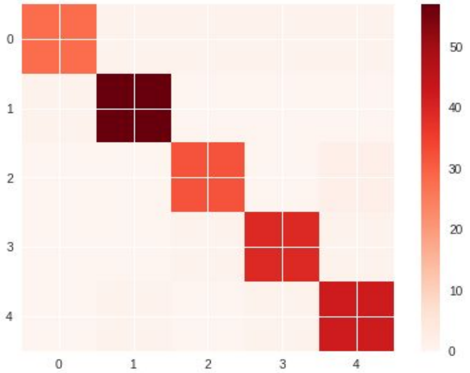
GRU:



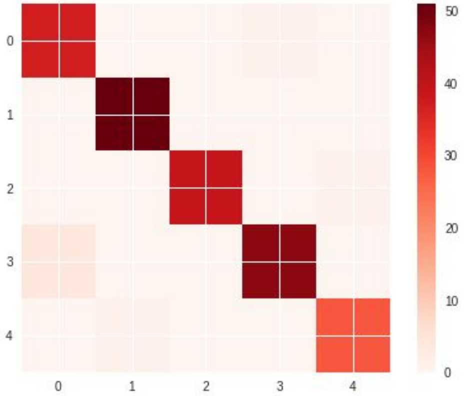
For BBC Dataset

Scenario (With Word Index)	LSTM	GRU
Test Accuracy	94.73%	96.65%
Train Accuracy	100%	100%

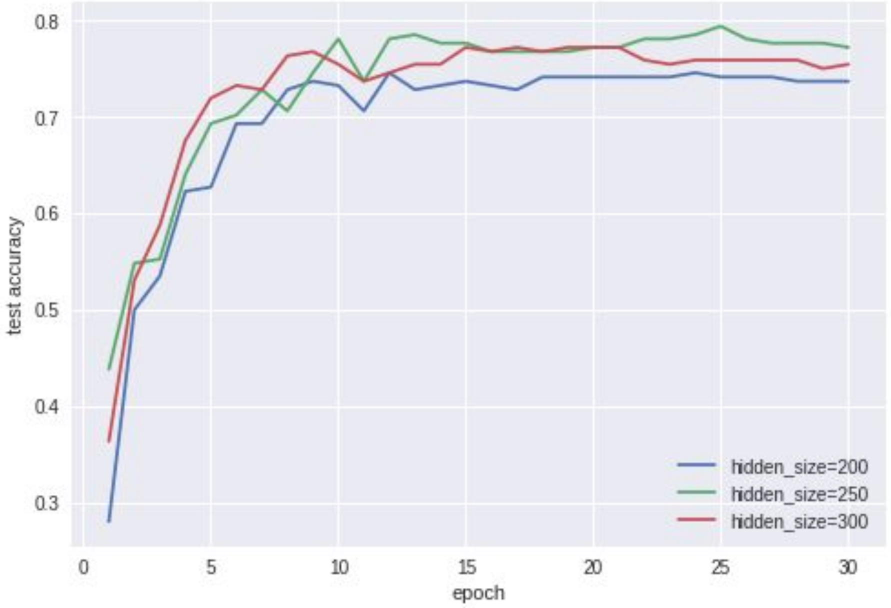
LSTM:

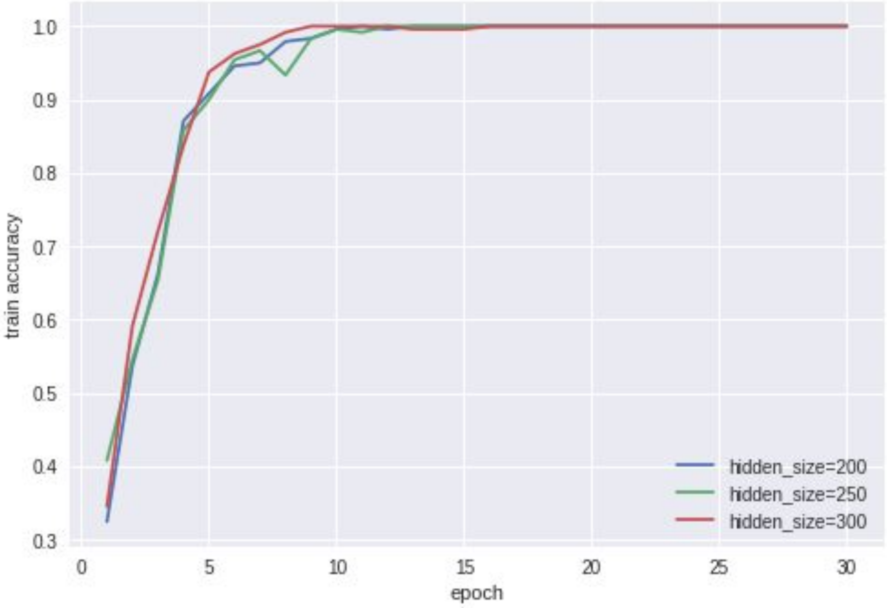


GRU:



Hyperparameter tuning





Conclusion and Future work

- Article level GRU model performs significantly better than the LSTM model.
- Index based embedding outperforms the pre-trained embeddings for these datasets.

- Future work include exploring variants of RNNs for the author identification
- Can be tried for larger datasets for better generalization of the network.

THANK YOU

References

[1] Young, T., Hazarika, D., Poria, S. and Cambria, E., 2018. Recent trends in deep learning based natural language processing. ieee Computational intelligenCe magazine, 13(3), pp.55-75.

[2] Mohsen, A.M., El-Makky, N.M. and Ghanem, N., 2016, December. Author identification using deep learning. In Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on (pp. 898-903). IEEE.

[3] Kim, Y., 2014. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882.

[4] Shrestha, P., Sierra, S., Gonzalez, F., Montes, M., Rosso, P. and Solorio, T., 2017. Convolutional neural networks for authorship attribution of short texts. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers (Vol. 2, pp. 669-674).

[5] Ding, S.H., Fung, B.C., Iqbal, F. and Cheung, W.K., 2017. Learning Stylometric Representations for Authorship Analysis. IEEE Transactions on Cybernetics.

[6] Qian, C., He, T., & Zhang, R. Deep Learning based Authorship Identification.

[7] Yao, L. and Liu, D., Wallace: Author Detection via Recurrent Neural Networks.

[8] https://archive.ics.uci.edu/ml/datasets/Reuter 50 50

[9] D. Greene and P. Cunningham. ”Practical Solutions to the Problem of Diagonal Dominance in Kernel Document Clustering”, Proc. ICML 2006.

[10] J. Pennington, R. Socher, C. Manning, GloVe: Global Vectors for Word Representation, EMNLP, 2014

[11] Zaremba, W., Sutskever, I. and Vinyals, O., 2014. Recurrent neural network regularization. arXiv preprint arXiv:1409.2329.