International Journal of Speech Technology (2020) 23:499–504
https://doi.org/10.1007/s10772-020-09724-9

RNN based machine translation and transliteration for Twitter data

M. K. Vathsala[1] · Holi Ganga[2]

Received: 13 February 2020 / Accepted: 2 June 2020 / Published online: 12 June 2020

Abstract
The present work aims at analyzing the social media data for code-switching and transliterated to English language using the special kind of recurrent neural network (RNN) called Long Short-Term Memory (LSTM) Network. During the course of work, TensorFlow is used to express LSTM suitably. Twitter data is stored in MongoDB to enable easy handling and processing of data. The data is parsed through different fields with the aid of Python script and cleaned using regular expressions. The LSTM model is trained for 1 M data which is further used for transliteration and translation of the Twitter data. Translation and transliteration of social media data enables publicizing the content in the language understood by majority of the population. With this, any content which is anti-social or threat to law and order can be easily verified and blocked at the source.

Keywords Long short-term memory (LSTM) · Recurrent neural network (RNN) · Sequence-to-sequence · Python · Translation · Transliteration · Twitter · Machine translation (MT) · BLEU · Tensorflow

1 Introduction

Machine Translation (MT) has evolved over five decades, which the developers have religiously followed since then. The most primitive approach in MT is the Statistical Machine Translation (SMT), which uses algorithms that are predictive in nature while teaching a system to translate the text. The existing translated text is used for translating the input text to the required language. The major drawback of this approach is that it requires a bi-lingual material for the model to predict the input text. This also hampers its ability to predict obscure languages.

Whereas, the evolution of Neural Machine Translation (NMT) approach, has addressed the major drawbacks associated with SMT with its more accurate translation. Though this approach is also based on Deep learning techniques, with the aid of existing statistical models, the input data is distributed among the layers enabling a faster response.

So based on the definitions thus stated related to two different approaches, it is quite clear that NMT can handle intricate computations when compared to the conventional statistical model. Despite having sufficient information about NMT, the extension of this towards handling the social media content has been looked at with meagre intent. The data posted on social networking sites have been the source for major cyber-attacks. Twitter statistics indicate a whopping 313 million tweets posted monthly. The data is very huge, which makes to difficult to analyse the information posted in different regional languages. A tweet posted in a regional language can neither be understood nor appreciated by the account holders of the same social media belonging to other regions. This make the users of other region deprived of the information posted. If the tweet is anti-national or anything that can threaten the social security can be averted by blocking the tweet at the source.

The present study aims at looking at the Twitter data more precisely with an intention of transliteration and translation of tweets, so that the social media users can be made aware of the content. This enables the end users to appreciate or criticize the information based on their understanding. Transliteration and translation also addresses the issue of social security and the information, paving way for any issue related to social security, can be screened-off at the nascent stage.

✉ M. K. Vathsala
vbsvathsala@gmail.com

[1] Dept of ISE, MSRIT, Bengaluru, VTU, Belagavi, Karnataka 560054, India

[2] Dept of ISE, Global Academy of Technology, Bengaluru, VTU, Belagavi 560098, India

2 Literature survey

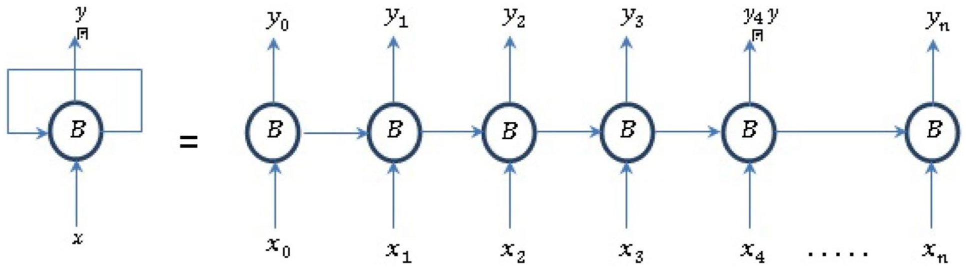
Enormous research has been carried out in the area of translation and transliteration since half-a decade. Some of the efforts include Statistical Machine Translation (SMT) methodology for translation via transliteration from Hindi to Urdu (Durrani et al. 2010), where the Bi-Lingual Evaluation Understudy (BLEU) scores of two different probability models viz. conditional probability model and joint probability model were found to be 19.35 and 19.00 respectively. The scores were compared with BLEU score obtained for DNN methodology to sequence-to-sequence problems using multi-layered Long Short-Term Memory (LSTM) approach (Sutskever et al. 2014), wherein it is proved that LSTM methodology not only outperforms SMT-based system but also standard Recurrent Neural Network (RNN) can be easily trained with a greater accuracy when the source sentences are reversed. In sequence-to-sequence LSTM framework, the text is read one byte at a time, producing span annotation over inputs (Gillick et al. 2016; Beck and Sales 2001). Also, some merits of production of span annotations are identified, which includes easy training of multi-lingual models without additional parameters and smaller output vocabulary. Most importantly it is also observed that the models turn out to be compact than conventional word-based systems, which discards the usage of tokenizers for text segmentation. Usage of RNN-LSTM for language modeling and extending it for identifying the image and providing suitable caption has been carried out (Al-muzaini et al. 2018). The results have been compared with CNN model, which indicated that RNN model gave more promising results. Also, Neural Machine Translation (NMT) for Vietnamese to English using sequence to sequence RNN, sequence to sequence CNN (ConvS2S) has been performed during which the BLEU scores were observed to be fairly good enough for low-resource data or language pairs (Phan-Vu et al. 2019). Though there are commendable efforts related to Text Summarization (TS), the latest work on Abstractive Text Summarization (ATS) using LSTM methodology based on Convolution Neural Network (CNN) (Song et al. 2019) is worth noting, where LSTM-CNN based ATSDL framework has been demonstrated. It is also proved to be state-of-art model for semantics and syntactic data structures. Finally, the most pertinent work for the present study includes the bangla sentence generation using LSTM-RNN with sequence-to-sequence model (Islam et al. 2019) which demonstrates the usage of LSTM for predicting the next word in a sentence. This uses.

On the other hand, the quality of translation is measured in terms of Bi-Lingual Evaluation Understudy (BLEU) factor which indicates the closeness and accuracy of translation. The aspects of machine translations viz. fluency, adequacy and fidelity are evaluated by humans (Hovy 1999; White and O'Connell 1994). For a bi-lingual human evaluation, two experts are required, in a way that they understand the other language but expert in one of them. This renders human evaluation costlier than MT, which is therefore facilitated by the corpus available for evaluation. Hence, machine translation comprises of two ingredients viz. numerical translation closeness matrix and a superior quality corpus of human reference translations (Papineni et al. 2002).

3 Methodology

Majority of the present study is aimed at utilizing the strengths of RNN models for an obvious reason that it can retain long term dependencies. RNN is complemented by LSTM model by enacting as a mechanism to ensure propagation of information by multiple time steps properly. Though umpteen efforts have been put-forth to understand the usage of RNN-based Language Models (Hochreiter and Schmidhuber 1997; Gers et al. 2000; Mikolov et al. 2010; Mikolov and Zweig 2012; Chelba et al. 2013; Zaremba et al. 2014; Williams et al. 2015; Ji et al. 2015a, b; Wang and Cho 2015), the present study aims at exploring RNN model for transliteration of Twitter data. Figure 1 depicts the architecture of RNN model considered for current study, where $x_0, x_1, x_2 \ldots x_n$ indicate the input to the neural network chunk, the output of which is indicated by $y_0, y_1, y_2 \ldots .y_n$ obtained at times $t_0, t_1, t_2 \ldots .t_n$ respectively. The input data is compared with the previous data which makes long term dependencies quite visible and allows the current methodology to stand out when compared to other conventional methodologies.

Fig. 1 Architecture of RNN model



RNN Methodology computes the outputs indicated in the architecture using the following equations iteratively:

$$h_n = \Sigma(W^{hx}x_n + W^{hh}h_{n-1})$$

$$y_n = W^{yh}h_n$$

where x_n and y_n are the lengths of input and output sequence respectively and h_N is the sequence used for mapping between x_N and y_N

The major issue with conventional RNN methodology is that, though it can handle long-term dependencies, fails to learn in the expected manner a human wishes to. Therefore, a special kind of RNN methodology called Long-Short Term Memory (LSTM) is used for the present study, which is explicitly designed for remembering and learning the information for longer durations. Though there are umpteen number of LSTM architectures available, the present study uses forget gate type of architecture, which forms an integral part of LSTM unit. An LSTM therefore comprises of a cell, an input gate, an output gate and a forget gate (Fig. 2). The forget gate layer is solely responsible for considering the previous cell state $\left(C_{n-1}\right)$ in the current cell calculation by assigning a value between 0and1. The value is assigned based on the comparison between the input vector $\left(x_n\right)$, output vector of the previous cell $\left(y_{n-1}\right)$ and the previous cell state $\left(C_{n-1}\right)$, which is indicative that the entire value of C_{n-1} has to be allowed to pass through to the Input Gate layer or not. If a value 0 is generated upon comparison, then C_{n-1} is discarded whereas a value 1 is to consider C_{n-1} for the current cell calculation. The generation of values is facilitated by a sigmoid neural layer and a pointwise multiplication operator in the forget gate layer (Fig. 2). Input gate layer performs two activities, first it updates the incoming vector data with the help of a sigmoid activation function and second, the output of sigmoid activation function is compared with the hyperbolic activation function, which generates new set of values, which are concatenated with the previous cell state (C_{n-1}). Finally, in the output gate layer, the input vector $\left(x_n\right)$ is passed through sigmoid activation function (σ), which is compared with the incoming updated cell state $\left(C_n\right)$ through hyperbolic activation function (*tanh*). The sigmoid function decides the portion that has to be put out of the cell (y_n).

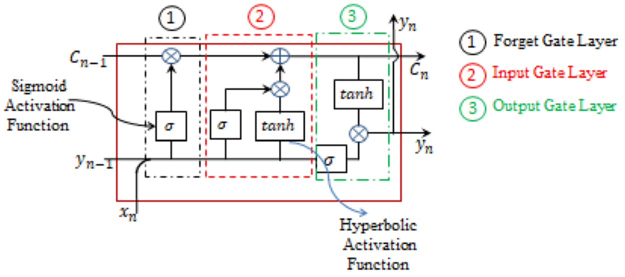


Fig. 2 Architecture of LSTM cell

LSTM mainly aims at estimating the conditional probability $p\left(x_0, x_1, x_2, \ldots\ldots x_N \middle| y_0, y_1, y_2, \ldots ..y_N\right)$ between the input sequence $x_0, x_1, x_2 \ldots\ldots x_N$ with the output sequence $y_0, y_1, y_2, \ldots\ldots y_N$, which is given as:

$$p\left(y_0, y_1, y_2, \ldots\ldots y_{N'} | x_0, x_1, x_2 \ldots\ldots x_N\right) = \prod_{n=1}^{n'} p(y_n | v, y_0, y_1, \ldots y_{n-1}) \tag{1}$$

This conditional probability is achieved by obtaining the processed data, the procedure of which may be elaborated. Firstly the data is collected from a reliable and authentic source. The data is collected from Twitter developer account, which is dumped into a database. In the present study, MongoDB is used for this purpose, which is a cross-platform or multi-platform document-oriented and NoSQL based database program. The data is stored in JSON format with integrity constraint imposed by database schema. Since the data is bulky and handling the same poses a challenge. It is easy to parse the entire dataover a field by writing simple queries in MongoDB. Secondly, the data is processed by cleaning, tokenizing and saving only the necessary fields of the raw-data into a new collection in the database, without overwriting the raw-Twitter data collection. The data is segregated based on the language and stored into their respective collections.

On the other hand, vocabularies for the participating languages are developed (say Hindi and English) (Table 1) by defining a set of all possible characters, which are encoded as vectors. The vectors are sequentially input to the LSTM model, where sequence length forms a critical parameter. Since it works on character level data, the access to dictionaries is eliminated. LSTM is bi-directional, where one layer reads the sequence from left to right and the hidden layer reads from right to left, the outputs of which are concatenated. Concatenation of input vector to output vector has a similar performance as that of residual connections introduced in deep residual networks. The batch size was initially set to 30 which indicated a slower performance, due to which the batch size was later reduced to 10. This was found to be optimum demonstrating a faster performance. Learning rate was kept constant at 0.001.

The result thus obtained using the above model is assessed using BLEU approach, where the greater the n-grams the researcher uses for the reference, the better is the accuracy. To obtain the BLEU score, the sourcelanguage is tested with a test-suite, suitably designed in the source language. The sentences are subjected to reference translations. This gives the deviation of the source language from the reference language.

Table 1 Sample English–Hindi pair for translation and transliteration

a	e	i	n	r	s	t	o	l
ए	इ	ऍ	न	र	स	ट	े	ल

The measure of BLEU score is performed by a modified precision score which is mathematically defined as (Papineni et al. 2002):

$$p_n = \frac{\sum_{C\in\{\text{Candidates}\}} \sum_{\text{n}-\text{gram}\in C} \text{Count}_{\text{clip}}(\text{n} - \text{gram})}{\sum_{C'\in\{\text{Candidates}\}} \sum'_{\text{n - gram}'\in C} \text{Count}(\text{n - gram}')} \tag{2}$$

where C is considered for all the sentences being translated, whereas $\text{Count}_{\text{clip}}$ indicates all the matching translations. Hence the ratio indicates how accurately the machine is translating when compared to human translations. Smaller the value of p_n, closer is the translation to the human translations, rendering a better accuracy. In order to attain the accuracy for shorter translations, *brevity penalty* is introduced which is mathematically given as:

$$B_p = \begin{cases} 1 & \text{if } c_l > r_l \\ e^{1-\frac{r_l}{c_l}} & \text{if } c_l \le r_l \end{cases}$$

where, c_l total or cumulative length of the translation; r_l is the length of the reference translation.

The BLEU is therefore calculated by the expression:

$$\log BLEU = \min\left(1 - \frac{r_l}{c_l}, 0\right) + \sum_{n=1}^{N} w_n \log(p_n)$$

where p_n is the geometric mean of modified precision score (Eq. 2) and w_n is the associated weight for the score.

Whereas, on the other hand, the BLEU score is very poor when considered for transliterated sentences due to the fact that some of the terms which seize to exist in the reference is transliterated and hence the BLEU score will be equal to the score of human translation. Due to this reason, a transliterated data has a lower BLEU score with an average value of 0.315 (Fig. 4, Table 2).

Table 2 Details of the model used for present study

Sl. no.	Description	Data
1	Model	LSTM
2	Vocabulary size	40,000
3	No. of layers	2
4	Size of layers	256
5	Batch size	10
6	No. of buckets	5
7	Learning rate	0.001
8	Learning decay factor	0.99

4 Results

During this study, one million Twitter data is stored in MongoDB. Storage in JSON format facilitates easy and quick writing and retrieval of data into and from the database respectively. The required fields of sample raw-Twitter data in JSON format is demonstrated below:

"text": "RT @varusarath: Happy birthday to the man of steel. Our #ThalaAjith Sir…May u haveee a lonnnnggggggggg lonnnggggggggg life filled with loa…",
"lang": "en".
"text":"सामना के मुताबकि अगर कट्टरवादी मुसलमान इस्लामि आतंकवाद कहलाते है|तो कट्टरवादी हनि्दूभी, हनि्दू आतंकवाद होने चाहिए|\n@NewsSamna",
"lang": "hi".
The cleaned data is obtained as follows:
"text": " Happy birthday to the man of steel. Our ThalaAjith Sir May u have a long long life filled with loa".
"lang": "en".
"text":"सामना के मुताबकि अगर कट्टरवादी मुसलमान इस्लामि आतंकवाद कहलाते है|तो कट्टरवादी हनि्दूभी, हनि्दू आतंकवाद होने चाहिए ",
"lang": "hi".

Table 2 indicates details of specifications considered for present study, where stacked LSTM is used. Based on the execution, perplexity curve is plotted with respect to the time step considered. It can be observed that the curve behaves exponentially and becomes an asymptote as it moves close to unity (Fig. 3). This indicates that the model is trained sufficiently and therefore can be used for translation and transliteration. The trained model demonstrates sequence to sequence learning model. Since the average training sentences length was around 18 to 19 and the average test set sentences length was around 22 to 23, the BLEU scores for both translation and transliteration were found to be in good

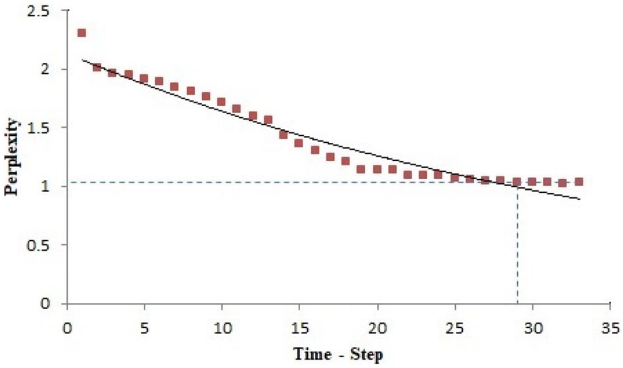


Fig. 3 Variation of perplexity with time-step

Table 3 Comparison of BLEU scores between SMT and LSTM Models

Sl. no.	Model description	Test BLEU scores
1	Baseline model (official)	10.778
2	LSTM En–Hi translation (reversed)	13.365
3	Hierarchical phrase based SMT En–Hi (Sen et al. 2016)	13.56
4	LSTM En–Hi transliteration	0.315
5	En–Hi transliteration (Ananthakrishnan et al. 2006)	0.27

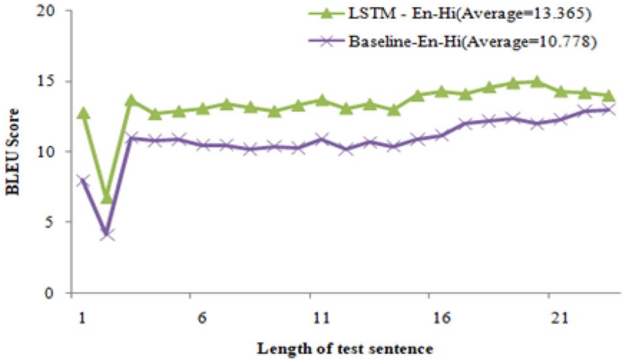


Fig. 4 Comparison for BLEU scores between LSTM translation and baseline model

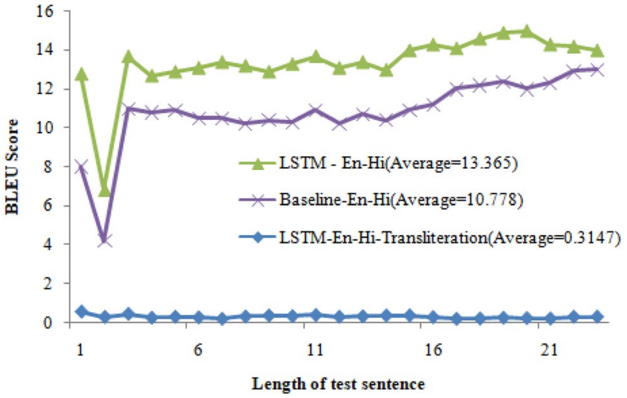


Fig. 5 BLEU comparison between translated and transliterated data

agreement with the available literature (Ananthakrishnan et al. 2006; Sen et al. 2016) (Table 3). For translation using SMT based translation with hierarchial phrase approach, the BLEU score was found to be 13.56, whereas using LSTM approach the BLEU score is found to be 13.365 which is around 1.4% deviation from the 13.56 (Fig. 4). On the other hand, the difference in transliteration precision score compared between the value in literature (Ananthakrishnan et al. 2006) and value obtained in present study was found to be 0.045 which is quite close to the values reported in the literature (Table 3, Fig. 5).

Sample translated and transliterated data is given as follows:

4.1 Input data

"text":" मुझे भूख लग रहा है”.
"lang": "hi".

4.2 Translated data

"text": "to me hunger feeling is",
"lang": "hi".

4.3 Transliterated data

"text": "Mujhe Bhook lag raha hi",
"lang": "hi".

5 Conclusion and future work

The present work demonstrates an idea of how to utilize RNN-LSTM model to address social security by identifying improper content being posted on to social media. It is observed that RNN-LSTM is more accurate than the conventional statistical machine translation (SMT) models (Table 2). Also, BLEU score is one of the reliable parameters that identify the quality of the machine translation. But, this parameter fails to determine the accuracy of transliterated data, due to the fact that some of the terms to be translated may not exist in the reference data.

The Twitter data stored in the database is parsed and cleaned for subjecting them to the process of translation and transliteration. During the course, it was observed that processing of data plays a very important role as the Twitter data comprises of slang words for which equivalent word is not available in the dictionary. Neglecting those words compromises the accuracy of transliteration. The present work can be extended to other social and professional media sites such as Facebook, Instagram, LinkedIn etc.

On the other hand, it can be extended to perform content search associated with improper video, audio and image content posted on social media. The video and image data can be obtained from social media developer accounts which can be used to train the LSTM model to analyse the content.

References

Al-muzaini, H. A., Al-yahya, T. N., & Benhidour, H. (2018). Automatic Arabic image captioning using RNN-LSTM-based language model and CNN. *International Journal of Advanced Computer Science and Applications, 9*, 6.

Ananthakrishnan, R., Bhattacharyya, P., Sasikumar, M., & Shah, R. M. (2006). *Some issues in automatic evaluation of English-Hindi MT: more blues for BLEU*. Proceedings of COLING/ACL 2006.

Beck, C. A. J., & Sales, B. D. (2001). Family mediation: Facts, myths, and future prospects (pp. 100–102). Washington, DC: American Psychological Association. https://doi.org/10.1037/10401-000.

Chelba, C., Mikolov, T., Schuster, M., Ge, Q., Brants, T., Koehn, P., & Robinson, T. (2013). *One billion word benchmark for measuring progress in statistical language modeling*. Retrieved from https://arxiv.org/abs/1312.3005.

Durrani, N., Sajjad, H., Fraser, A., & Schmid, H. (2010). *Hindi-to-Urdu machine translation through transliteration*. Stuttgart: Institute for Natural Language Processing, University of Stuttgart.

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual predictionwith LSTM. *Neural Computation, 12*(10), 2451–2471.

Gillick, D., Brunk, C., Vinyals, O., & Subramanya, A. (2016). *Multilingual language processing from bytes.* Google Research. Retrieved from https://arxiv.org/abs/1512.00103.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

Hovy, E. H. (1999). *Toward finely differentiated evaluation metrics for machine translation*. In Proceedings of the Eagles Workshop on Standards and Evaluation, Pisa, Italy.

Islam, M. S., Mousumi, S. S. S., Abujar, S., & Hossain, S. A. (2019). Sequence-to-sequence Bangla sentence generation with LSTM recurrent neural networks. *International Conference on Pervasive Computing Advances and Applications-PerCAA, 152*, 51–58. https://doi.org/10.1016/j.procs.2019.05.026.

Ji, S., Vishwanathan, S. V. N., Satish, N., Anderson, M. J., & Dubey, P. (2015a) *Blackout: Speeding up recurrent neural network language models Exploring the Limits of Language Modeling with very large vocabularies.* CoRR, abs/1511.06909. Retrieved from https://arxiv.org/abs/1511.06909.

Ji, Y., Cohn, T., Kong, L., Dyer, C., & Eisenstein, J. (2015b). *Document context language models.* Retrieved from https://arxiv.org/abs/1511.03962.

Mikolov, T., Karafiat, L., Burget, L., Cer-nocky, J., & Khudanpur, S. (2010). *Recurrent neural network based language model*. In INTERSPEECH, vol. 2, p. 3.

Mikolov, T. & Zweig, G. (2012). *Context dependent recurrent neural network language model*. In SLT, pp. 234–239.

Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). *BLEU: a method for automatic evaluation of machine translation*. In ACL.

Phan-Vu, H.-H., Tran, V.-T., Nguyen, V.-N., Dang, H.-V., & Do, P.-T. (2019) Neural machine translation between Vietnamese and English: An empirical study. *Journal of Computer Science and Cybernetics*. Retrieved from https://arxiv.org/pdf/1810.12557.pdf.

Sen, S., Banik, D., Ekbal, A., & Bhattacharyya, P. (2016). *IITP English-Hindi machine translation system at WAT 2016*. Proceedings of the 3rd Workshop on Asian Translation (pp. 216–222), Osaka, Japan, December 11–17 2016.

Song, S., Huang, H., & Ruan, T. (2019). Abstractive text summarization using LSTM-CNN based deep learning. *Multimedia Tools Application, 78*, 857–875. https://doi.org/10.1007/s11042-018-5749-3.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). *Sequence to sequence learning with neural networks*. Retrieved from https://arxiv.org/abs/1409.3215.

Wang, T. & Cho, K. (2015). *Larger-context language modeling*. Retrieved from https://arxiv.org/abs/1511.03729.

White, J. S. & O'Connell, T. (1994). *The ARPA MT evaluation methodologies: Evolution, lessons, and future approaches*. In Proceedings of the First Conference of the Association for Machine Translation in the Americas (pp. 193–205), Columbia, Maryland.

Williams, W., Prasad, N., Mrva, D., Ash, T., & Robinson, T. (2015). *Scaling recurrent neural network language models*. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5391–5395. IEEE.

Zaremba, W., Sutskever, I., & Vinyals, O. (2014). *Recurrent neural network regularization.* Retrieved from https://arxiv.org/abs/1409.2329.