A Framework for German-English Machine Translation with GRU RNN

Levi Corallo[1], Guanghui Li[2], Kenna Reagan[3], Abhishek Saxena[4], Aparna S. Varde[5] and Brandon Wilde[6]

[1]Computational Linguistics, Montclair State University, Montclair, NJ, United States
[2]Computational Linguistics, Montclair State University, Montclair, NJ, United States
[3]Computational Linguistics, Montclair State University, Montclair, NJ, United States
[4]Data Science, Montclair State University, Montclair, NJ, United States
[5]Computer Science, Montclair State University, Montclair, NJ, United States
[6]Computational Linguistics, Montclair State University, Montclair, NJ, United States

Abstract

Machine translation (MT) using Gated Recurrent Units (GRUs) is a popular model used in industry-level web translators because of the efficiency with which it handles sequential data compared to Long Short-Term Memory (LSTM) in language modeling with smaller datasets. Motivated by this, a deep learning GRU based Recurrent Neural Network (RNN) is modeled as a framework in this paper, utilizing WMT2021's English-German data-set that originally contains 400,000 strings from German news with parallel English translations. Our framework serves as a pilot approach in translating strings from German news media into English sentences, to build applications and pave the way for further work in the area. In real-life scenarios, this framework can be useful in developing mobile applications (apps) for quick translation where efficiency is crucial. Furthermore, our work makes broader impacts on a UN SDG (United Nations Sustainable Development Goal) of Quality Education, since offering education remotely by leveraging technology, as well as seeking equitable solutions and universal access are significant objectives there. Our framework for German-English translation in this paper can be adapted to other similar language translation tasks.

1. Introduction

[1] **Motivation and Goal:** The open task created by EMNLP provides datasets of sentences from news articles in multiple language pairs with parallel translated data [1]. The work generated by the task seeks to advance current machine translation (MT) research by using the latest performance scores as a comparison for future research, to investigate the applicability of current varying methods of MT, to examine challenges in word translation for specific language pairs, and to elicit more research on low-resource, morphologically rich languages. This provides the motivation for our research. Our goal is to investigate a specific machine translation problem in a morphologically rich language and model a framework to provide a feasible solution. In this context, we address the issue of German-English news translation. While there is much work on translation, there are gaps in existing tools, e.g. Google Translate has a limit on characters (see Fig. 1) with translation from a German News source [2]. In order to make news and other such text accessible globally, it is important to address large-scale translation, for which issues such efficiency are significant. We present the following.

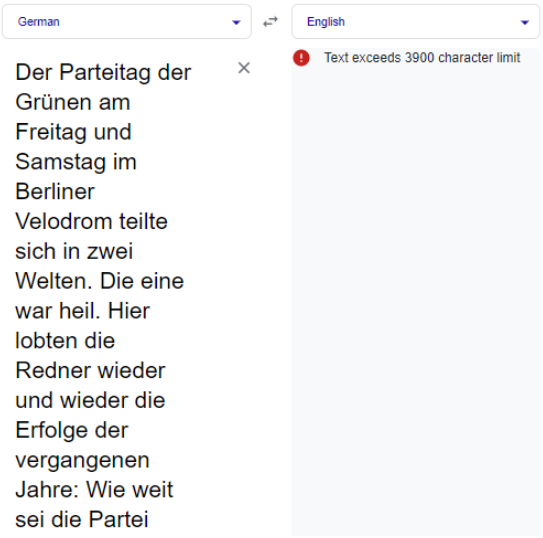

Figure 1: Google Translate attempt from Frankfurter Allgemeine Zeitung (German newspaper) with limitations [2]

Models and Methods: We address the issue of transla-

Published in the Workshop Proceedings of the EDBT/ICDT 2022 Joint Conference (March 29-April 1, 2022), Edinburgh, UK
corallo1@montclair.edu (L. Corallo); lig1@montclair.edu (G. Li); reagank1@montclair.edu (K. Reagan); saxenaa1@montclair.edu (A. Saxena); vardea@montclair.edu (A. S. Varde); wildeb11@montclair.edu (B. Wilde)

[1]Authors are in alphabetical order with equal contributions

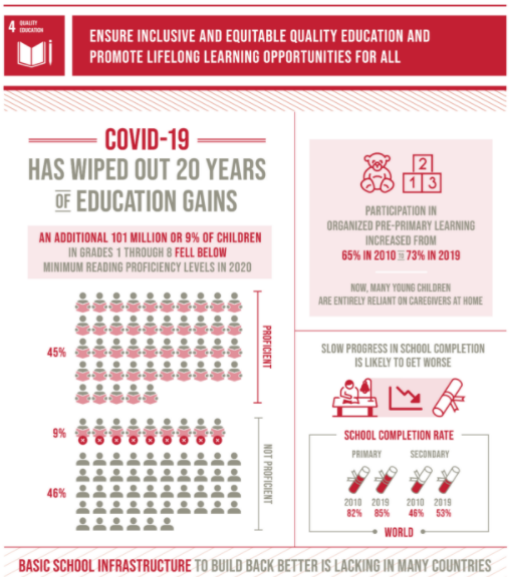

Figure 2: UN SDG on Education and its recent concerns

tion via a framework modeled by GRU RNN deep learning methods on a parallel German-English translated news corpus. The RNN (Recurrent Neural Network), originally conceptualized by Rumelhart et al. [3], with the concept of GRU (Gated Recurrent Unit) proposed by Cho et al. [4], is selected to model this framework based on its current performance in machine translation due to its efficiency as compared to the Long Short-Term Memory (LSTM) model. In order to create a reasonable training time, we experiment with our framework on batches of 64 sentence pairs at a time. We choose to work with Keras, an open-source Python library to implement this framework [5]. We obtain interesting results that set the stage for building applications and conducting further research for enhancement. Our framework in this paper for German-English translation is usable for translation in other morphologically rich languages.

Applications: From a real-life perspective, translation of news is important for ensuring accessibility of current events for readers across the world who read in different languages, and even fighting censorship of news-media by bridging information divides to countries without freedom of press. To that end, our paper broadly impacts UN SDG 4: Quality Education since its facets include the following. (1) "Help countries in mobilizing resources and implementing innovative and context-appropriate solutions to provide education remotely, leveraging hi-tech, low-tech and no-tech approaches"; (2) "Seek equitable solutions and universal access" [6]. In the aftermath of COVID, some of these goals have been negatively impacted (see Fig 2 from the United Nations source [6]), including language-related issues. This makes it even more important for us to address such concerns in order to enhance education. In addition, the framework in this paper has the real-life standpoint of being useful in mobile application (app) development due to its efficiency. Application of machine learning in mobile apps is broached in a variety of works, e.g. as summarized in a survey paper [7]. During online news translation in a mobile application, it is important to obtain fast results that capture the crux of the material presented in the news. Our framework is useful in such tasks.

2. Related Work

Avramis et al. presented work at WMT2020 utilizing the German-English news corpus provided by EMNLP for that year's open task [8]. The paper details the development of a test suite, containing multiple different linguistic phenomena relevant to the German to English translation process. The most difficult concepts highlighted in the test suite when using MT to translate German into English include ambiguous sentences, multi-word expressions, verb valency, and "false friends" which refers to words in two languages that appear similar in composition and are often mistaken as sharing the same meaning, but do not. The example their paper provides is the German word "Novella" commonly having its target translation mistaken for "novel," which it does not translate into or semantically represent, but instead "novella," or "short story." The paper points out that it is a surprising fact that MT models are prone to false friends when making mistakes in translating because this is an observed human error. This was insightful when analyzing the validity and accuracy of our translated sentences, where we were able to understand phenomena that could be influencing the margin of error.

There is research that points to Rumelhart et al. for the early conceptualization of Recurrent Nets that were able to evolve into modern RNN programming [3]. This early work is a predecessor introducing vital concepts in neural machine translation using an RNN such as the hidden layer between input and output units, sigma-pi units, and so on. More recent work by Chung et al. [9] is able to give empirical comparisons to LSTM in RNNs. The original concept for GRUs was introduced by Cho et al. [4] who proposed a novel neural network model called RNN Encoder-Decoder which uses two different neural networks as encoder and decoder respectively. The encoder is used to read the source sentence and map it into a vector of fixed length, while the decoder reads the vector and maps it back to a corresponding target sentence. Along with the new architecture, they also proposed an improved version of standard RNN called a

Gated Recurrent Unit (GRU) which uses a reset gate and an update gate to decide how much information should be passed to the output sequence. They can be trained to keep information from long ago if the information is critical to the prediction or forget information if it is irrelevant to the prediction. They experimented with this model on a task of translating English to French, found that the overall translation performance was improved in terms of BLEU (BiLingual Evaluation Understudy) scores [10] and linguistic regularities at both word level and phrase level were captured. After their work, this model has become a mainstream model framework.

Zhang et al. [11] proposed an alternative to the widely-used bidirectional encoder with the merits of incorporating future and history contexts into the source representation. This novel encoder is called a context-aware recurrent encoder (CAEncoder) which consists of two levels. The bottom level summarizes the history information and the upper level assembles this information together with future context into the source representation. Through their experiment on translation tasks with two different language pairs, they found this novel encoder to be as efficient as the bidirectional encoder and to demonstrate better performance.

Previous work has been done on multilingual neural machine translation (NMT) that demonstrates the difficulties in translating between languages of the same language family and languages in different language families. The study determined that it is difficult for one model to handle every language to be considered for translation. The reasoning for this, in part, is because the model could be negatively impacted during training when considering language pairs, such as Chinese to English and German to English. For this reason, the study explores language clustering, where languages that are closely related are clustered together, to boost the model during training. They determine that language embeddings, which considers genealogy and typology in clustering, outperforms random family, which only considers genealogy [12]. Handwritten Chinese character recognition by distance metric learning is approached in [13] that cites work pertinent to pictorial scripts, considering OCR and machine translation.

Efforts in improving machine translation quality between typologically similar languages have long been witnessed in the field. For those very close language pairs, a direct word-for-word translation method was tested and received promising results [14]. More advanced multilingual neural machine translation system has been created to address one to many or many to many translations within language groups which share inherent similar structures. Azpiazu and Pera [15] put forward a novel encoder-decoder machine translation framework called HNMT specifically exploited the hierarchical nature of a typological language family tree. The natural connection among languages enables effective knowledge transfer, while avoiding negative effects caused by incorporating very distant languages. Recent work done by Oncevay et al. [16] tried to embed typological features in language vector space for multilingual machine translation tasks and reported to achieve competitive translation accuracy.

Recent work by Popović [17] details and compares language-structure related issues that arise in NMT specifically between German and English. The author's work finds that key structural differences between German and English causing ambiguities and inconsistent target translations are the handling of prepositions, the translation of ambiguous English (source) words, and generation of English (target) continuous tenses. English and German both follow SVO (Subject-Verb-Object) sentence structure, so the obstacles found in Popović's work highlighting prepositional phrasing, ambiguity, and tense account for inaccuracies.

Other work in this general area entails addressing article errors and collocation errors in written text translation from a source language into English [18, 19, 20, 21], by addressing issues of ESL (English as a Second Language) learners. Preposition prediction and idiom detection are addressed in some works [22, 23, 24]. Problems on knowledge discovery from big data including those on machine translation are discussed in [25]. Deep learning techniques are used widely in machine translation via paradigms such as LSTM (Long Short-Term Memory) [26], BERT (Bidirectional Encoder Representations from Transformers) [27], GPT (Generative Pre-trained Transformer) [28] and T5 (Text-To-Text Transfer Transformer) [29]. Depending on the task, one of these paradigms would be selected and adapted within solution approaches. There are studies that emphasize commonsense knowledge in the realm of machine intelligence, addressing translation among several tasks [30, 31, 32]. Comparison is presented in [33] between symbolic knowledge graphs (KGs) and deep learning with neural models, explaining their pros versus cons, and how they can potentially complement each other. Our work in this paper fits in the broad spectrum of such exhaustive research. Its main contribution is the framework modeled to conduct German-English news translation with efficiency as needed in real-life applications.

3. Models and Methods

The deep learning paradigm is one of the most widely used facets for Machine Translation. We model a framework for morphologically rich language translation deploying a GRU based RNN, given its success with real-life scenarios such as industry level web-translators, and adapt it specifically to our problem of German-English news translation in this paper. Our framework is imple-

mented within the Python Keras platform [5] to perform translations from German to English. The methodology for the model discussed in this paper involves text preprocessing, model design and model training. This is discussed next with reference to our data in this work.

3.1. Dataset and Text Preprocessing

The data used to train our model is sourced from the News Commentary dataset, obtained on the EMNLP 2021 website for the machine translation conference WMT21 [1]. The data, provided specifically for the task of machine translation, is an aligned corpus of German and English news stories. The collection comprises approximately 400,000 German-English sentence pairs sourced from news articles.

The text preprocessing phase entails data cleaning, tokenization and sentence padding. First, the dataset is passed through data cleaning filters. Since all sentences would be padded to the same final length, extremely long sentences are removed. This includes sentence pairs for which either the German or English sentence is more than 50 words long. Errors in the creation of the dataset can also occasionally incorrectly map one German sentence to two English sentences or vice versa. This is partially corrected by passing the data through two filters. The first removes all sentence pairs in which one sentence has more than twice as many words as its counterpart and a minimum length of 25 words. The second filter removes all sentence pairs in which one sentence has more than four times as many words as its counterpart. The combined filters reduce the dataset to a size of approximately 378K German-English sentence pairs.

Tokenization is then performed with the Keras Tokenizer function, dividing sentences into their component words, and assigning each unique word an integer for further processing. Each sentence is thus converted into a list of integers. Dummy <PAD> tokens are then added at the end of each tokenized sentence, so that each sentence conforms to the same length and can be processed by the neural machine translation model.

3.2. Translation Model Design

We predetermined to approach this machine translation task with an RNN model as justified earlier. After reviewing the literature and assessing approaches by others, we resolved to build a GRU-based RNN. Our framework for translation is illustrated in Fig. 3.

The model is built within the Keras platform and is composed of two principal components: a GRU and a dense layer. Input data are entered into the GRU, and processed in matrices with a configurable dimensionality referred to here as GRU units (not to be confused with the number of GRUs, which was only one). The

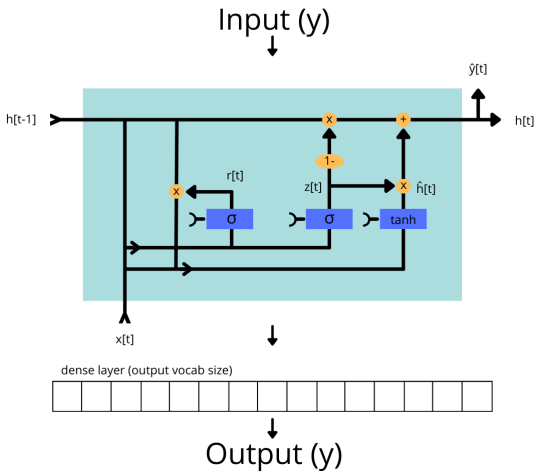

Figure 3: Framework for Machine Translation

GRU is designed for sequential processing and so maintains dependencies between different parts of a given entered sentence. The GRU output is then fed into a time-distributed dense neural layer, which produces a series of logit vectors for each sentence. Each logit effectively represents the probability of a given English word occurring in that position within the sentence, so the output is decoded by calling the English word which corresponds to the position of the largest logit in each vector.

Several model and training parameters are left as variables, to ensure the easy reconfiguration of components. The parameter list and our chosen parameter configurations are included in Tables 1 and 2. We choose to remain relatively constant with some of the configurable model functions that are well-established: Softmax is used as the activation function, sparse categorical cross-entropy is used as the loss function, and Adam was used as the optimization function [5].

3.3. Model Training

After preparing the dataset and RNN model, the data is divided into two parts. Using Python Sklearn's train-test-split method, the dataset is shuffled and split: 80% of the data for training and 20% for testing, to add robustness to the framework.

Model training then commences with a configuring of model parameters and subsequent passing of the training data into the Keras Model.fit() method. Accuracy and loss are used as standard metrics [5] to monitor model performance during training and provide a basis for modifications to the model's hyper-parameters.

A new validation set is created at the beginning of each batch, on which the training data of the batch is tested following the completion of batch training. This provides a way to obtain more reliable metrics than simple training statistics. The methodology in our work including the text preprocessing and actual machine translation is summarized in Algorithm 1.

Algorithm 1: Text Preprocessing and Translation

INPUT: English-German corpus
DEFINE: L(S) as Length of Sentence S

FOREACH sentence-pair (Sx, Sy) in corpus:
  IF L(Sx) > 50 OR L(Sy) > 50
    REMOVE (Sx, Sy)
  ELSEIF L(Sx)J/L(Sy) ≥ 4
  OR (L(Sx)J/L(Sy) ≥ 2 AND $L(Sx) \geq 25$)
    REMOVE (Sx, Sy)
  ELSEIF L(Sy)J/L(Sx) ≥ 4
  OR (L(Sy)J/L(Sx) ≥ 2 AND $L(Sy) \geq 25$)
    REMOVE (Sx, Sy)
ELSE TOKENIZE (Sx, Sy)

MAP each unique token to an integer (token ID)
PAD encoded token IDs to max length
DEFINE model hyper-parameters
DEFINE model architecture via GRU RNN
INSTANTIATE model with architecture and hyper-parameters

FOREACH epoch:
  FOREACH encoded (Sx, Sy) batch in training data:
    FIT model to encoded (Sx, Sy)
  EVALUATE model on validation data

FOREACH encoded (Sx, Sy):
  MODEL-PREDICT encoded output
  DECODE encoded output to text
  OUTPUT: Translated sentences

4. Experiments and Discussion

Initial experimentation is conducted with abbreviated datasets (5k - 50k sentence pairs) to reduce test time and allow for the testing of more hyper-parameter combinations. This provides a precursory glimpse of the fully trained model. Two parameter configurations are then selected for training on the full dataset, creating what would be named Simple RNN Model I (Table 1) and Simple RNN Model II (Table 2) in our overall framework.

The learning rate is a configurable hyper-parameter that controls how quickly the model is adapted to the problem, often in the range between 0.001 and 0.05. Our experiments are set up with learning rates of 0.01 and 0.05 correspondingly. The number of GRU units are set to 128 and 512 for Model I and Model II respectively. We save the history of the model throughout the training process and subsequently plot the changes in loss and accuracy (see Figs. 4 – 7). We conduct experiments with two setups for the running of the RNN model. Comparing these two setups, the principal differences lie in the learning rate and GRU parameters.

No.	Parameters	Value
1	Learning Rate	0.01
2	GRU Units	128
3	Activation Function	Softmax
4	Loss Function	Categorical Cross Entropy
5	Validation Percentage	0.2
6	Epochs	10
7	Batch Size	64

Table 1
Training Parameters for Simple RNN Model I

No.	Parameters	Value
1	Learning Rate	0.05
2	GRU Units	512
3	Activation Function	Softmax
4	Loss Function	Categorical Cross Entropy
5	Validation Percentage	0.2
6	Epochs	10
7	Batch Size	64

Table 2
Training Parameters for Simple RNN Model II

Model	Train	Test
Model I	5 hours	10 minutes
Model II	7 hours	12 minutes

Table 3
Total Training and Testing Times Combined (For All Experiments Conducted)

4.1. Experimental Results

The total training and testing times for all the executions combined in our experiments with Models I and II are synopsized in Table 3. In Fig. 4, we can observe that both the training and validation loss decreased overall for Model I. Despite the occasional spikes in loss, this is what we expect to see while training the model. Meanwhile,

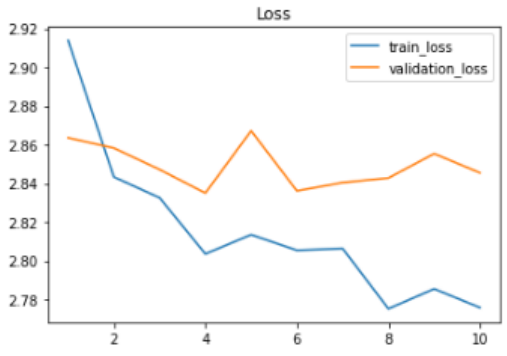

Figure 4: Model I Loss

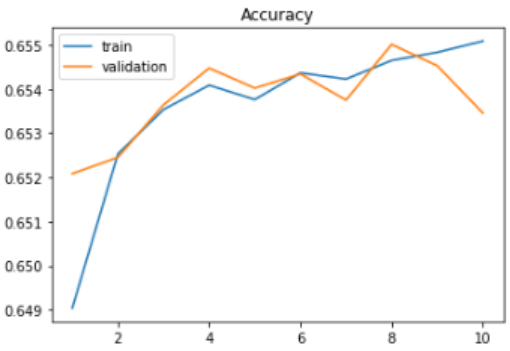

Figure 5: Model I Accuracy

Figure 6: Model II Loss

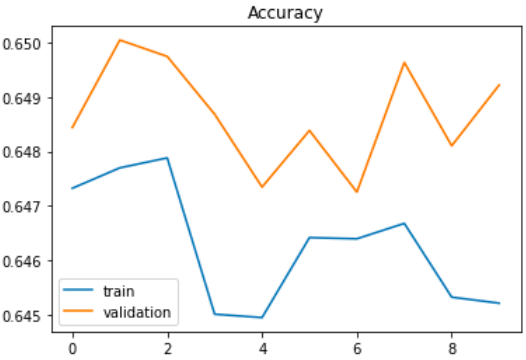

Figure 7: Model II Accuracy

both the training and validation accuracy increase for Model I, seen in Fig. 5.

However, we also notice that the validation accuracy drops at the end of running, demonstrating that the model weights and biases have not yet reached a stable optimum. Interestingly, the validation loss does not increase during the same period, as would be expected. Such occurrences might indicate when the model weights and biases are more precisely able to replicate several of the previously correct results, while losing ground on some of the less certain results. In other terms, the model is becoming more confident producing target sentence words easy to predict, while simultaneously losing confidence in words that are more difficult to predict.

For Model II, we change to a larger learning rate of 0.05 and a larger number of GRU units, 512. The results differ drastically from Model I. The loss for both the training and validation set have an overall increase across all 10 epochs, as can be seen in Fig. 6. In the second, third, fifth, eighth, and ninth epochs, the training loss decreases. In the second, fifth, ninth, and tenth epoch, the validation loss decreases. In all other epochs, the training and validation losses both increase. The accuracy for both training and validation sets shows a fluctuating change across all 10 epochs, as can be seen in Fig. 7, indicating robustness. The overall accuracy decreases as is expected with rising loss.

4.2. Discussion on Experiments

We observe in all our experimentation that Model I somewhat outperforms Model II in both loss and accuracy. Model I has a final training accuracy of 0.655 and final validation accuracy of 0.653. Model II had a final validation accuracy of 0.645 and a final validation accuracy of 0.649. Model I has a final training loss of 2.78 and the final validation loss was 2.85. Model II has a final validation loss of 4.66 and a final validation loss of 5.55.

Model I depicts a consistent decrease in loss and a consistent increase in validation. Model II portrays a consistent increase in loss while the accuracy increases and decreases throughout the training process without any consistency. Despite the markedly different behavior, the two models both finish with a difference in translation accuracy less than 1%. Overall, it seems as though the

lower learning rate of 0.01 in Model I produces better results than the learning rate of 0.05 in Model II. The accuracy of Model I is higher than in Model II and the loss in Model I is lower than in Model II. The learning rate is a significant factor in how well the models perform. In our previous attempts to find the best parameters to train on, we find that 512 GRU units provide the best preliminary results. However, despite the fact that Model II uses 512 GRU units, Model I still outperforms Model II on the whole. It is clear that the higher learning rate hinders Model II much more than the use of 512 GRU units is able to help it. It is likely that with the high learning rate, Model II over-corrects and is not able to narrow in on optimal results. This is reflected in Figs. 6 and 7 where we can see that the loss increases and the accuracy is inconsistent. The learning rate of both of our models hovers around the 65% range. Though Google Translate gives an accuracy in the 80% range, it faces the issue of a maximum character limit which is not feasible for translating news articles. Similar critiques can be applied to other tools and methods in the literature. Hence, our work, though at an early stage, can address such issues and pave the way for building efficient, larger scale, and easily accessible mobile apps in news translation for morphologically rich languages. This would complement other state-of-the-art apps.

One limitation on our model's performance may have been the technique used to transform our data into feature sets. A simple word-integer assignment method is used here that may have been a detriment due to its representation of words in an ordinal system as opposed to a categorical one. Alternate approaches could include one-hot encoding [34] or word vector generation word2vec [35]. We could also implement an alternative architecture such as a bidirectional RNN [36] into the framework to explore if it enhances model performance We chose to work with a simple approach first in line with the logic of preferring simpler theories over complex ones as per Occam's razor principle [37], and also given the fact that we need reduced complexity and high efficiency for translation tasks in this context. While our simple approach allows the model to observe general context patterns, it does not offer semantic representation of the words to be translated. Furthermore, for better understanding the performance of the translation model, we could consider adopting BLEU scores in the evaluation of our future model, since this is widely recognized as a reliable evaluation criteria in the machine translation field [10]. On the whole, our current framework creates a good baseline for translating German news to English, capturing reference to context.

5. Conclusions

In this paper, we model a framework using a GRU-based RNN to perform German-English news translation, depicting a method of efficient translation of text pieces in morphologically rich languages. We notice high efficiency in training and testing the model. While the accuracy levels obtained here seem good for a starting point, there is scope for further improvement.

In future work, apart from considering approaches such as word2vec and bidirectional RNN, as well as tuning some hyper-parameters, we could recommend using more training epochs. Selecting an appropriate learning rate and number of GRU units, as well as securing sufficient training time are challenges for training deep learning MT models. During our attempts to tune the model, we observe that smaller learning rates require more training epochs, given the smaller changes made to the weights each update, whereas larger learning rates result in rapid changes and require fewer training epochs. Later, this work might benefit from using a learning rate that decreases with each epoch, allowing the initial training to advance quickly while letting the fine-tuning take the time it needs. These are some recommendations based on our study in this paper. Furthermore, we could potentially incorporate commonsense knowledge into the learning process. As depicted in recent works, deep learning based models and commonsense based models can complement each other for enhanced performance.

Files for this project are available on GitHub and can be provided to interested users upon request. On the whole, this work provides the ground for developing mobile apps for news translation orthogonal to existing work in the area. It caters broadly to the United Nations Sustainable Development Goal of Quality Education.

6. Acknowledgments

Authors are in alphabetical order. A. Saxena has been funded by a GA from the Comp. Sc. dept. at MSU. A. Varde has NSF grants 2018575 and 2117308. She is a visiting researcher at the Max Planck Institute for Informatics, Germany.

References

[1] E. WMT, Translation Task— German-English corpus, 2021. URL: https://www.statmt.org/wmt21/translation-task.html.

[2] Zeitung, Frankfurter Allegemeine, 2021. URL: https://www.faz.net/aktuell.

[3] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning internal representations by error propagation, Technical Report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[4] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using rnn encoder-decoder for statistical machine translation, arXiv preprint arXiv:1406.1078 (2014).

[5] F. Chollet, Deep learning with Python, Simon and Schuster, 2021.

[6] UN, SDG website, 2021. URL: www.un.org/sustainabledevelopment/sustainabledevelopment-goals/.

[7] P. Basavaraju, A. S. Varde, Supervised learning techniques in mobile device apps for androids, ACM SIGKDD Explorations 18 (2017) 18–29.

[8] E. Avramidis, V. Macketanz, U. Strohriegel, A. Burchardt, S. Möller, Fine-grained linguistic evaluation for state-of-the-art machine translation, arXiv preprint arXiv:2010.06359 (2020).

[9] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, arXiv preprint arXiv:1412.3555 (2014).

[10] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, Bleu: a method for automatic evaluation of machine translation, in: Proceedings of the 40th annual meeting of the Association for Computational Linguistics, 2002, pp. 311–318.

[11] B. Zhang, D. Xiong, J. Su, H. Duan, A context-aware recurrent encoder for neural machine translation, IEEE/ACM Transactions on Audio, Speech, and Language Processing 25 (2017) 2424–2432.

[12] X. Tan, J. Chen, D. He, Y. Xia, T. Qin, T.-Y. Liu, Multilingual neural machine translation with language clustering, arXiv preprint arXiv:1908.09324 (2019).

[13] B. Dong, A. S. Varde, D. Stevanovic, J. Wang, L. Zhao, Interpretable distance metric learning for handwritten chinese character recognition, CoRR abs/2103.09714 (2021). arXiv:2103.09714.

[14] J. Hajic, Machine translation of very close languages, in: Sixth Applied Natural Language Processing Conference, 2000, pp. 7–12.

[15] I. M. Azpiazu, M. S. Pera, A framework for hierarchical multilingual machine translation, arXiv preprint arXiv:2005.05507 (2020).

[16] A. Oncevay, B. Haddow, A. Birch, Bridging linguistic typology and multilingual machine translation with multi-view language representations, arXiv preprint arXiv:2004.14923 (2020).

[17] M. Popović, Comparing language related issues for nmt and pbmt between german and english, The Prague Bulletin of Mathematical Linguistics 108 (2017) 209.

[18] D. Dahlmeier, H. T. Ng, Correcting semantic collocation errors with l1-induced paraphrases, in: Proceedings of the 2011 conference on empirical methods in natural language processing, 2011, pp. 107–117.

[19] N.-R. Han, M. Chodorow, C. Leacock, Detecting errors in english article usage by non-native speakers, Natural Language Engineering 12 (2006) 115–129.

[20] A. M. Pradhan, A. S. Varde, J. Peng, E. M. Fitzpatrick, Automatic classification of article errors in l2 written english, in: Twenty-Third International FLAIRS Conference, 2010.

[21] A. Varghese, A. S. Varde, J. Peng, E. Fitzpatrick, A framework for collocation error correction in web pages and text documents, ACM SIGKDD Explorations 17 (2015) 14–23.

[22] P. Bhagat, A. S. Varde, A. Feldman, Wordprep: Word-based preposition prediction tool, in: 2019 IEEE International Conference on Big Data (Big Data), IEEE, 2019, pp. 2169–2176.

[23] J. Briskilal, C. Subalalitha, An ensemble model for classifying idioms and literal texts using bert and roberta, Information Processing & Management 59 (2022) 102756.

[24] A. Elghafari, D. Meurers, H. Wunsch, Exploring the data-driven prediction of prepositions in english, in: Coling 2010: Posters, 2010, pp. 267–275.

[25] G. De Melo, A. S. Varde, Scalable learning technologies for big data mining, in: 20th International Conference on Database Systems for Advanced Applications, DASFAA 2015, Springer Verlag, 2015.

[26] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, J. Schmidhuber, Lstm: A search space odyssey, IEEE transactions on neural networks and learning systems 28 (2016) 2222–2232.

[27] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. arXiv:1810.04805.

[28] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, Improving language understanding by generative pre-training (2018).

[29] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, 2020. arXiv:1910.10683.

[30] N. Tandon, A. S. Varde, G. de Melo, Commonsense knowledge in machine intelligence, ACM SIGMOD Record 46 (2017) 49–52.

[31] C. Matuszek, M. Witbrock, R. C. Kahlert, J. Cabral, D. Schneider, P. Shah, D. Lenat, Searching for common sense: Populating cyc from the web, UMBC Computer Science and Electrical Engineering Department Collection (2005).

[32] E. Onyeka, A. S. Varde, V. Anu, N. Tandon, O. Daramola, Using commonsense knowledge and text mining for implicit requirements localization, in: 2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI), IEEE, 2020, pp. 935–940.

[33] S. Razniewski, N. Tandon, A. S. Varde, Information to wisdom: commonsense knowledge extraction and compilation, in: ACM International Conference on Web Search and Data Mining (WSDM), 2021, pp. 1143–1146.

[34] J. Liang, J. Chen, X. Zhang, Y. Zhou, J. Lin, One-hot encoding and convolutional neural network based anomaly detection, Journal of Tsinghua University (Science and Technology) 59 (2019) 523–529.

[35] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781 (2013).

[36] M. Schuster, K. K. Paliwal, Bidirectional recurrent neural networks, IEEE transactions on Signal Processing 45 (1997) 2673–2681.

[37] T. Mitchell, M. L. McGraw-Hill, Edition, 1997.