Translation from Russian to Everyday English using a Model of A recurrent Neural Network (TREEMANN 🌲👨‍💼)

EN 601.475/675 Machine Learning

Zoe Kim, Kyllie Furukawa, Stella Li, Brandon Wong

Introduction

- Machine translation is an important application of machine learning
- Neural machine translation (NMT) solves issues regarding speed and accuracy
- Data:
    - Bilingual English-Russian dataset from Kaggle
    - Translate Russian (source) to English (target)

Importance of Machine Translation

- Automated translation of large datasets
- Cross-language information retrieval
- Instant text, audio, image, and video translations

Aim

- To develop a Russian-English sequence to sequence (seq2seq) neural machine translation model using Recurrent Neural Networks
- To perform hyperparameter tuning in this particular problem space

Prior Work

- Neural Machine Translation by Jointly Learning to Align and Translate (Bahdanau et al. 2015)
  - Encoder/Decoder architecture
  - Attention mechanism
- Sequence to Sequence Learning with Neural Networks (Sutskever et al. 2014)
  - Reversing the input for optimization
- Effective Approaches to Attention-Based Neural Machine Translation (Luong et al. 2015)
  - Global and local attention mechanisms

Deliverables

1.1. Must Accomplish

✅1. Data pre-processing

✅2. Basic framework for the model that can take input and generate output

✅3. Utilize at least 1 ML technique from class (e.g. batching)

Deliverables

1.2 Expect to Accomplish

✅1. Batch size hyperparameter searching

✅2. Bidirectional LSTM

✅3. One more optimization (e.g. Teacher Forcing)

Deliverables

1.3 Would Like to Accomplish

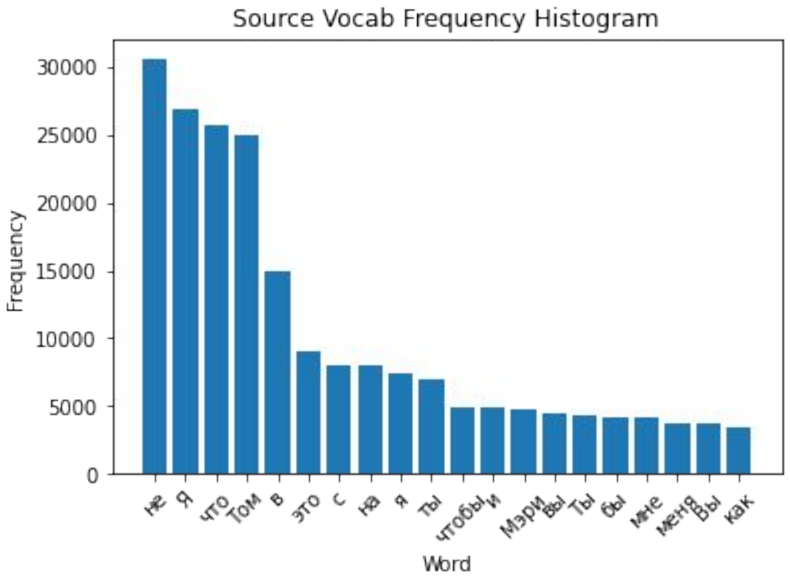
✅1. Implement and experiment with attention types
2. Perform systematic hyperparameter tuning
3. Generalize model and hyperparameters to another Slavic language

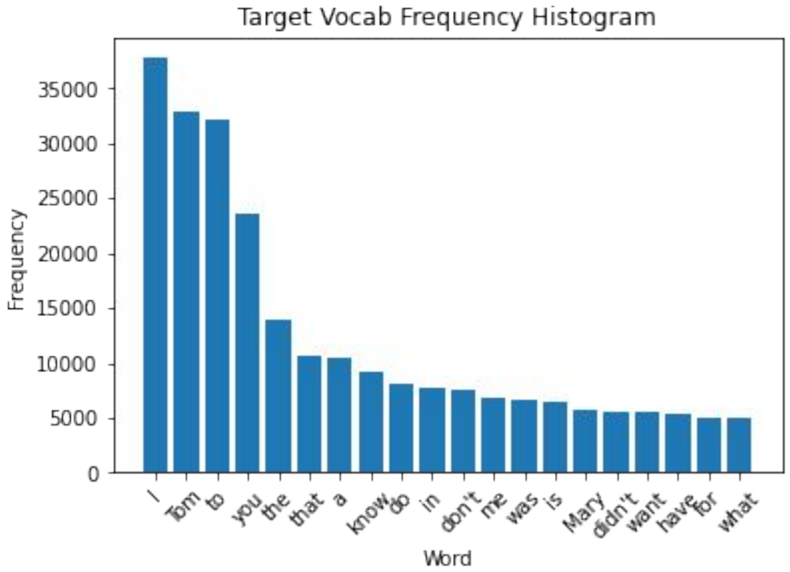
Methods - Data Pre-Processing

- MAX_LENGTH = 10 (96.6% of the corpus)
- Vocab class
  - **n_words**: number of unique words
  - **word2index**: uses a word as an index for the current n_words
  - **word2count**: keeps track of the frequency of each word
  - **index2word**: uses the current n_words as an index for the word (<SOS> and <EOS>)

Answer me.	Отвечай мне.
Birds fly.	Птицы летают.
Bless you.	Будь здоров!
Call home!	Позвони домой.
Call home!	Позвоните домой.
Calm down!	Успокойся!
Calm down!	Успокойтесь!
Calm down.	Успокойтесь.
Can I eat?	Я могу есть?
Can I eat?	Я могу поесть?
Can we go?	Мы можем идти?
Catch Tom.	Поймай Тома.

Methods - Exploratory Data Analysis





Methods - Encoder Decoder Model

- Recurrent Neural Network : nn.LSTM (Pytorch)
- Bidirectional LSTM (Encoder, EncoderBack)
- EncoderRNN : Reads in the input sequence and outputting the context vector
- AttnDecoderRNN: Stack of recurring units that predicts outcome at each time step
    - Global and local attention

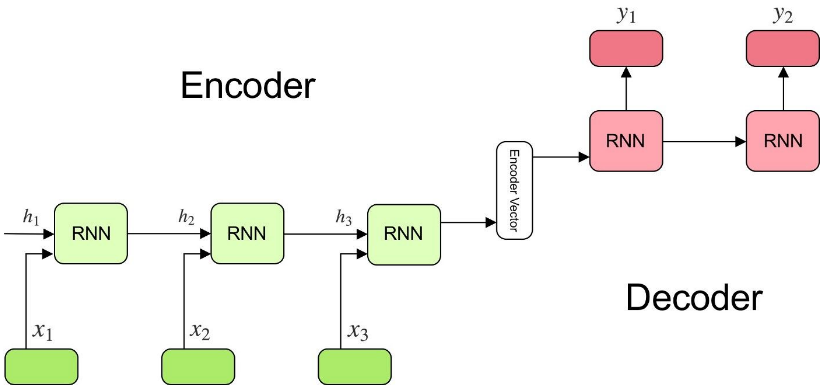


Image obtained from
“https://towardsdatascience.com/understanding-encoder-decoder-sequence-to-sequence-model-679e04af4346”

Methods - Training

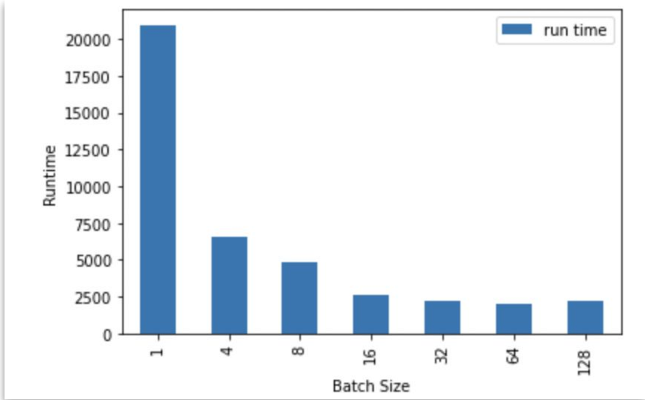
- Teacher Forcing
- Reversing the source
- Translate function = "evaluation" process
    - Predicted output from the decoder (an index) $\longrightarrow$ a word
- BLEU score calculated from the validation set using NLTK SacreBLEU

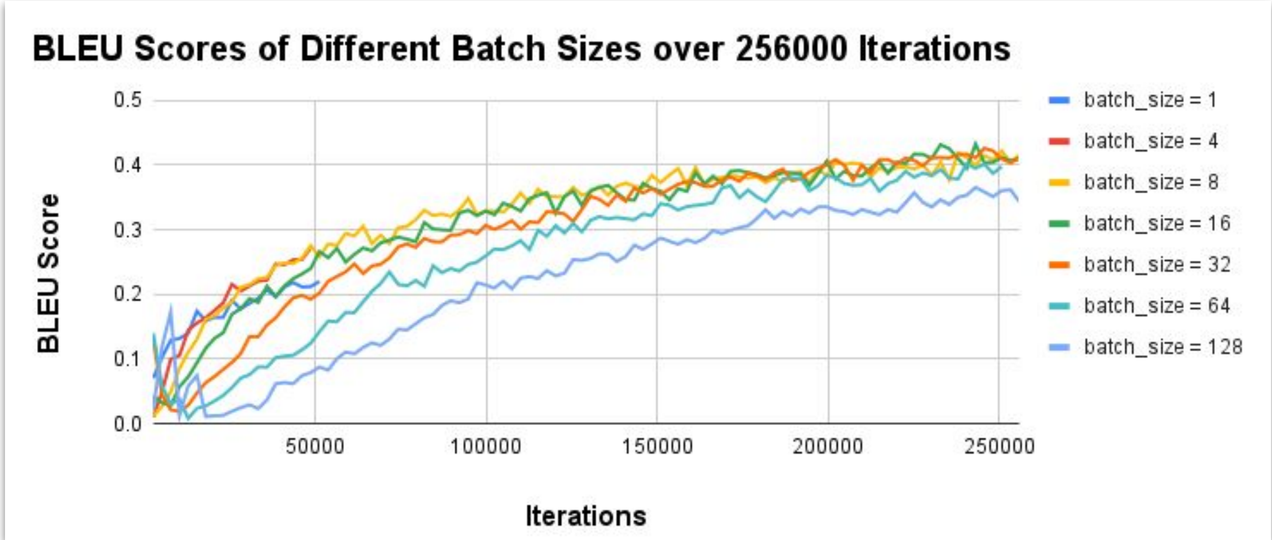
Obstacles

1. Understanding LSTM, Encoder-Decoder model
2. Integrating attention, had to modify several functions and classes
3. Long runtime for training (GPU), especially with 256000 iterations
    a. Encountered a crash, had to work with partial data (no time to re-run)

Results - Batch Size

- Full experimental results are here
- Optimal batch size of 16
  - BLEU score of 0.410
  - Training time of 2572 seconds (23040 iterations)

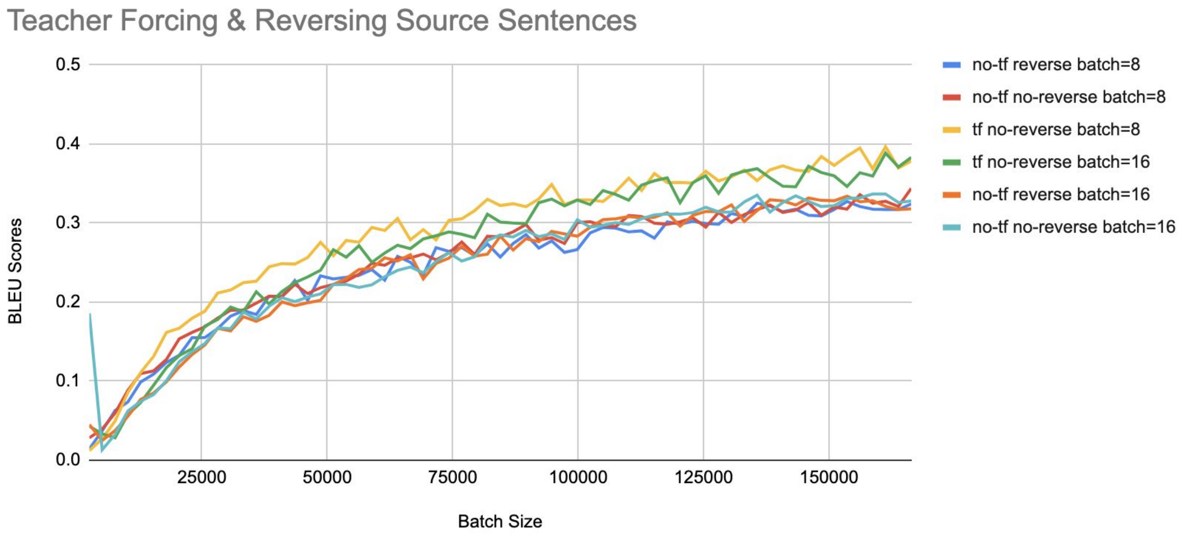




Results

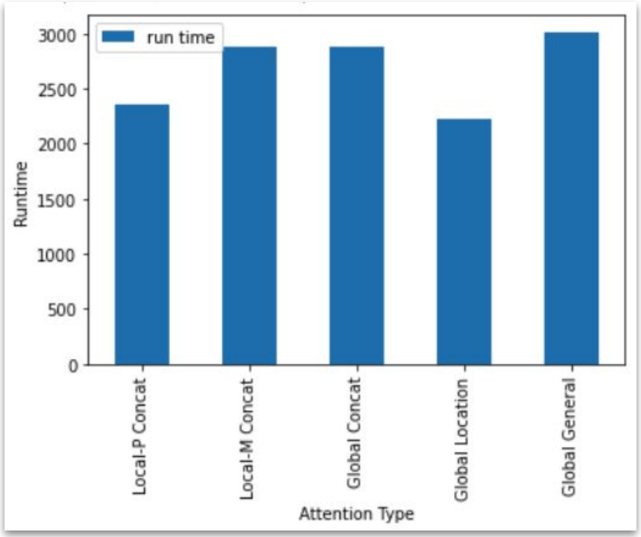
Reverse Input & Teacher Forcing

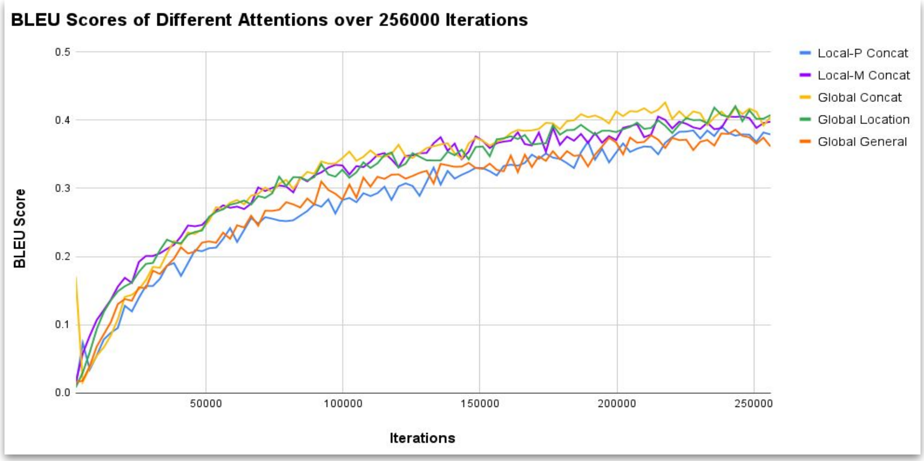
1. Teacher forcing helps w/ both runtime and BLEU
2. Reverse input has no significant effect



Results - Attention Model

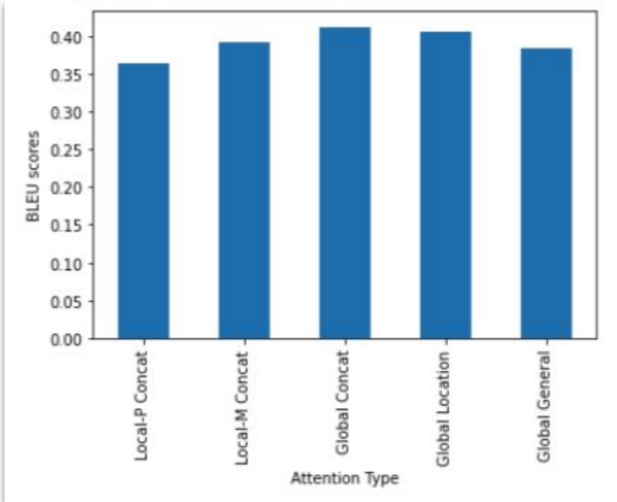
- Full experimental results are here
- Optimal attention model is global location
  - BLEU score of 0.405
  - Training time of 2223 seconds (23040 iterations)





Results - Attention Model

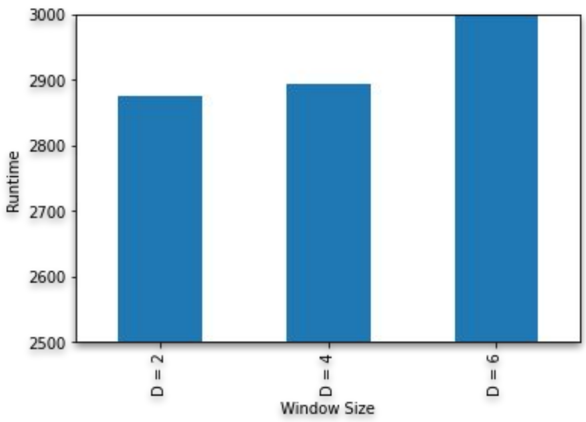
- Optimal attention model is global location
  - BLEU score of 0.405
  - Training time of 2223 seconds (23040 iterations)

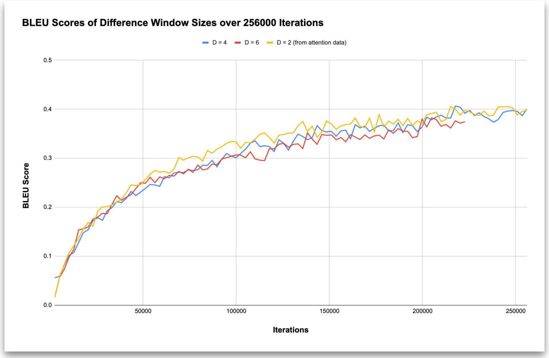



```
I know that you'll have a fun.
How did you get to to school?
You know we didn't do that.
Who did you break on this book?
Boston is where I want to go.
You promised you wouldn't be drink today.
Tom looks like he needs to be
Tell Tom that I don't have money.
I swear I'll never do such a thing.
How do you know Tom is sick?
Tom said he's glad glad you're back.
I'm am afraid she washing
What do you think about those about that?
We all want prices to kill
I played tennis with with
Do you have with with you?" car?
Tom's face is out of the
I bought a new car last week.
If you want me to go, I'll go.
I suspect Tom is allergic to peanuts.
I read this this book for three days.
Could you like an electric own own
Tom didn't need to go home.
What you saying saying is
I managed to be there by 2:30.
I need a place to change
We wanted us to you a song.
Children comes to swim in river. river.
Can you tell me where I is?
I doubt that Tom will be at time. time.
Tom doesn't talk to me about about that.
All our memory of of
Tom made the joke, but wasn't
My office is on at the the
I don't think any of talking except you.
Tom has a after school.
We expect us to be rain today.
I have a little about Tom.
Do you remember what you last night?
When was was surprised Tom dead.
Here's a teacher of an accident.
He is a diary in English.
```

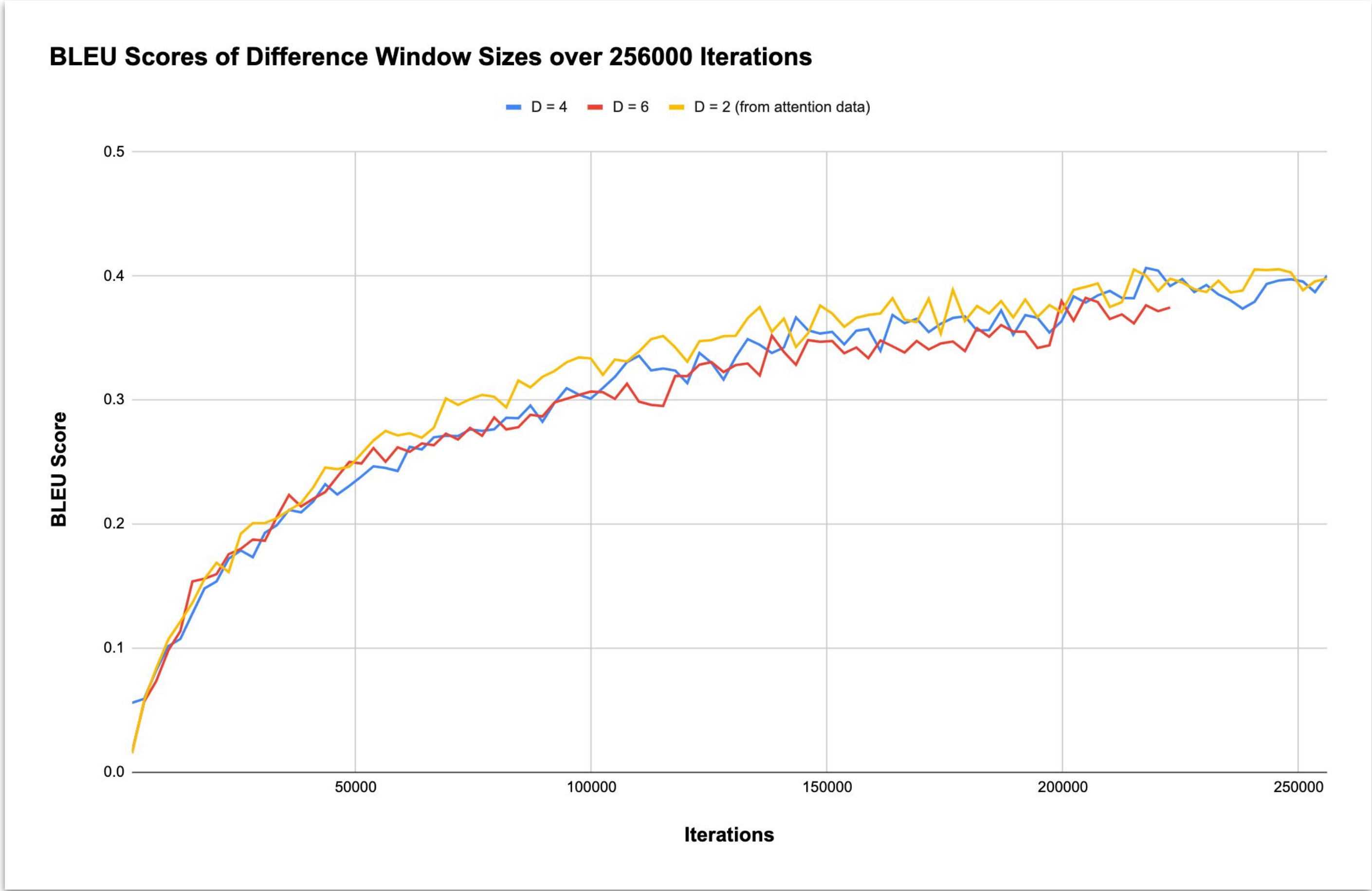
Results - Window Size

- Full experimental results are here
- Attention model: local-m concat
- Optimal window size is 2
  - BLEU score of 0.391
  - Training time of 2876 seconds (23040 iterations)





(enlarged in next slide)


BLEU Scores of Difference Window Sizes over 256000 Iterations
D = 4
D = 6
D = 2 (from attention data)
0.5
0.4
0.3
0.2
0.1
0.0
BLEU Score
50000
100000
150000
200000
250000
Iterations

Conclusions

- Batching speeds up training but sacrifices accuracy
- Context-aware global attention improves accuracy
- Corpus-specific optimal parameters:
    - Batch size = 16
    - With Teacher Forcing
    - Attention model = global location
    - Window size = 2

Future Directions

- Bias in the translation of proper nouns
- Plotting BLEU over the runtime of the program and not over iterations
- Pre-training for low-resource morphologically similar languages
    - e.g. Upper Sorbian, a Slavic minority language from Eastern Germany
- More rigorous experimentation, multiple trials per data point
- More extensive search in the entire hyperparameter space
- Experiment with optimizers and loss function

References

1. Bahdanau, D., Cho, K., \& Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
2. Jassem, Krzysztof, Dwojak, Tomasz. (2019). Statistical versus neural machine translation – a case study for a medium size domain-specific bilingual corpus. Poznan Studies in Contemporary Linguistics. 55. 491-515. 10.1515/psicl-2019-0018.
3. Kostadinov, S. Understanding encoder-decoder sequence to sequence model. Medium. 2019, February 4, https://towardsdatascience.com/understanding-encoder-decoder-sequence-to-sequence-model-679e04af4346.
4. Luong, M. T., Pham, H., \& Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025.
5. Sundermeyer, M., Alkhouli, T., Wuebker, J., \& Ney, H. (2014, October). Translation modeling with bidirectional recurrent neural networks. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 14-25).
6. Zosimov, Radmir. “English-Russian Dictionary for Machine Translate.” Kaggle, 17 June 2021, https://www.kaggle.com/hijest/englishrussian-dictionary-for-machine-translate.
7. “NLP from Scratch: Translation with a Sequence to Sequence Network and Attention.” *PyTorch*, https://pytorch.org/tutorials/intermediate/seq2seq_translation_tutorial.html.