Usage of Machine-based Translation Methods for Analyzing Open Data in Legal Cases

Nataliya Boyko[0000-1111-2222-3333], Lesia Mochurad[0000-0002-4957-1512],

Uliana Parpan[0000-0003-1424-050X], Oleh Basystiuk[0000-0003-0064-6584]

Lviv Polytechnic National University, Lviv, Ukraine
nataliya.i.boyko@lpnu.ua, lesia.i.mochurad@lpnu.ua,
uparpan35@gmail.com, obasystiuk@gmail.com

Abstract. Deep learning has completely changed approaches to machine translation. The initial ways of building machine translation software were based on rules, the next stage was based on statistics and probability theory. But nowadays, with new researches in the deep learning field has created simple solutions based on machine learning that outperform the best expert systems. This paper overviews the main features of machine translation for analyzing open data in legal cases based on recurrent neural networks. The advantages of systems based on RNN using the sequence-to-sequence model against statistical translation systems are also highlighted in the article. Two machine translation systems based on the sequence-to-sequence model were constructed using Keras and PyTorch machine learning libraries. Based on the obtained results, libraries' analysis was done, and their performance comparison.

Keywords: machine translation, deep learning, recurrent neural networks, performance, keras, pytorch, sequence-to-sequence.

1 Introduction

Systems of machine translation of unstructured data from one language to another are modeling work of a human translator. Their productivity depends on their ability to comprehend the language grammar rules. In the translation, the main units are not single words, but phrases or phraseological units expressing various concepts. Only by using them, more complex ideas can be expressed via the translated text [20].
The main feature of machine translation is the different length for input and output. To be able to work with different input and output length, you need to use a recurrent neural network [1-6].

Initially, the work of computer programs for translation is to replace words or phrases from one language with words or phrases from another. However, then there is a problem that such a replacement cannot provide a quality translation of the text because it requires the definition and recognition of words and whole phrases from the original language. Currently, multilingual ontological resources such as WordNet and UWN are used to handle collisions in translation.

Machine translation is one of the subgroups of computational linguistics that studies different languages text translation approaches based on software solutions.

 CybHyg-2019: International Workshop on Cyber Hygiene, Kyiv, Ukraine, November 30, 2019.

Machine translation basically performs the replacement of one language words to another language words, but usually, the translation made in this way is relevant, because in order to fully convey the meaning of the sentence and find the most suitable analog in the "target" language - it is often necessary to translate the whole phrase in general.

Solving this problem with statistical and neural translation systems is a rapidly growing field that leads to improved translation, upgrade differences in linguistic typology, better handling differences in linguistic typology, the translation of idioms, and the identification of anomalies [5,8].

Modern machine translation software has the function of changing the settings for the domain - industry or professional activity, for example, meteorological reports. By limiting the scope of permissible substitutions/substitutions, we are able to obtain a better translation result [10].

This method is especially effective in areas where the formal or template-style language is used. This means that machine translation is more efficient in government and legal documents, rather than translation any less standardized texts [7, 11].

Improving the quality of the final result can also be achieved through human intervention: for example, some systems will be able to provide a more accurate translation if the user will indicate in advance the correct translation of some words in the text.

There are two fundamentally different approaches to the construction of machine translation algorithms: rule-based and statistical-based. The first approach is traditional and is used by most machine translation system developers.

Rule-based MT (RBMT), “Classic Approach” (MT) is a machine translation system based on linguistic information from unilingual, bilingual, or multilingual dictionaries and grammar rules, source language and target language [13,15].

The system covers the basic semantic, morphological, and syntactic patterns of each language. Accordingly, in order to make a translation, the system must make a preliminary morphological, syntactic, and semantic analysis of the text, and only after that it generates a sentence. The biggest disadvantage of RB-translation is that in order for a program to perform a correct translation, its database must contain all spelling variations of word entry, and for all cases of ambiguity, lexical selection rules must be written. In itself, adaptation to new domains is not such a complicated process, because the basics of grammar for all domains are the same, and the settings of the areas of user activity are limited only by the correction of lexical selection. Thus, such a machine translation system is the classical method of its implementation, it allows to obtain a better result than the statistical method, but synthesizes translation more slowly [1,17].

Statistical machine translation is a type of text-based machine translation that is more effective in working with bigger volumes of language pairs. Language pairs - text data that contain sentences in one language and the corresponding sentences in another. Thus, statistical machine translation has a feature of self-learning. The more language pairs available to the program and the more accurately they correspond to each other, the better the result of statistical machine translation [2,19].

The term "statistical machine translation" refers to a general approach to solve the problem of translation, which is based on finding the most probable translation of a sentence using data obtained from a bilingual set of texts. An example of a bilingual set of texts is parliamentary reports, which are minutes of debates in parliament. Bilingual parliamentary reports are issued in Canada, Hong Kong, and other countries; official documents of the European Economic Community are issued in 11 languages, and the United Nations publishes documents in several languages. As a result, these materials are highly useful resources for statistical machine translation.

This system is based on the statistical calculation of the probability of coincidences. To translate, the program must have access to hundreds of millions of documents that have been translated by humans in advance. Such documents serve as templates for the system, on the basis of which it translates. The more documents, the higher the probability of better translation[18, 20] .

At the beginning of its existence, in 2006, Google Translate was based on the statistical method of machine translation, and its translation was of very low quality and was considered one of the worst translation options that can be done by an online translator. Today, Google uses the "neural" method of machine translation (MT) and is in serious competition with commercial enterprises, whose products are not free.

Neural networkapproach is based on the method of deep learning.Deep learning (also known as deep structured learning or hierarchical learning) is part of a broader group of machine learning methods based on the interpretation of learning outcomes, as opposed to algorithms for specific tasks. Training can be supervised or unsupervised.In recent years, Hybrid machine translation (HMT) has become increasingly popular, and the main technology of implementing HMT become RNN.Recurrent neural network (RNN) - is a class of artificial neural network, which has connections between nodes. In this case, the connection refers to the connection from the more distant node to the less distant node. The presence of connections allows RNN to memorize and reproduce the entire sequence of reactions to one stimulus. From the programming point of view in such networks there is an analog of the cyclic execution, and from the systems point of view - such networks are equivalent to a finite-state machine. RNNs, are generally used to handle the sequence of words in the processing of natural language [14-17]. Usually for word sequence processing using the Hidden Markov Model (HMM) and the N-program language model.

Hidden Markov Model (HMM) - the statistical model that simulates the work of a process similar to a Markov process with unknown parameters and the task is to guess unknown parameters on the basis of the observed ones. The obtained parameters can be used in further analysis. n a normal Markov model, the state is known to the observer, so the probability of transitions is one parameter. In NMM it is possible to observe only variables that are affected by this state. Each state has a probabilistic distribution among all possible output values. Therefore, the sequence of words generated by NMM gives information about the sequence of states. The NMM can be considered as the easiest Bayesian network.

Bayesian network - the graphical model in the form of a directed acyclic graph, each vertex of which corresponds to a random variable, and the arcs of the graph

encode the relations of conditional independence between these variables. The vertices can represent variables of any type, be weighted parameters, hidden variables, or hypotheses. There are effective methods that are used to calculate and study Bayesian networks. For conducting a probabilistic output in Bayesian networks, both precise and approximate algorithms are used [18-20].

2 Materials and Methods

At a high-level representation of a recurrent neural network (RNN), shown on figure 1, it's processes data sequences, such as sentences,one element at a time while retaining a memory (called a state) of what has come previously in the sequence.

Recurrent means the output at the current time step becomes the input to the next time step. At each element of the sequence, the model considers not only the current input, but what it remembers about the preceding elements. The most popular cell approach nowadays is the LSTM (Long Short-Term Memory) which maintains a cell state as well as a carry for ensuring that the signal (information in the form of a gradient) is not lost as the sequence is processed. At each time step the LSTM considers the current word, the carry, and the cell state.

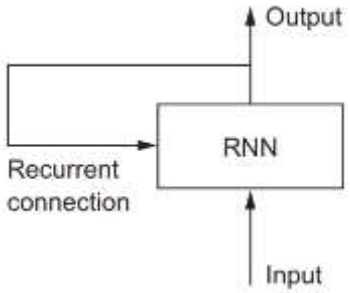

Fig. 1. Recurrent network loop.

The basic idea of an RNN is to use recursion to form the fixed dimension vector from the input sequence of symbols. Assume that in step t vector is h_{t-1} which is the history of all previous words. RNN will calculate new vector h_t (its internal state), which combines all previous words$(\boldsymbol{x_1}, \boldsymbol{x_2}, \dots, \boldsymbol{x_{t-1}})$and new character $x_t$using:

$$h_t = \varphi_\theta(x_t, h_{t-1}) \quad (1)$$

In this equation, the following parameters are present: $\boldsymbol{\varphi_\theta}$- function, parameterized with θ, which receive a new word input $\boldsymbol{x_t}$ and words history $\boldsymbol{h_{t-1}}$till (t - 1) - N word. First, we can assume that $\boldsymbol{h_0}$- zero vector. The recurrent activation function φ is usually implemented as an affine transformation, followed by non-linear function:

$$h_t = tanh(Wx_t + Uh_{t-1} + b) \quad (2)$$

In this equation, the following parameters are present: input weight matrix W, recurrence weight matrix U and bias vector b. Note, that this is not the only one variant. There is wide scope for developing new recurring activation functions [19].
More detailed about the work of the method for text translation based on neural networks. The idea of this algorithm is, in fact, simple and consists of the following steps:

1. Encoding the input data of language A into the data set;
2. Decoding the data set in language B.

Let's look at an algorithm for encoding unstructured data on an example text sentence: “Example of neural network”:

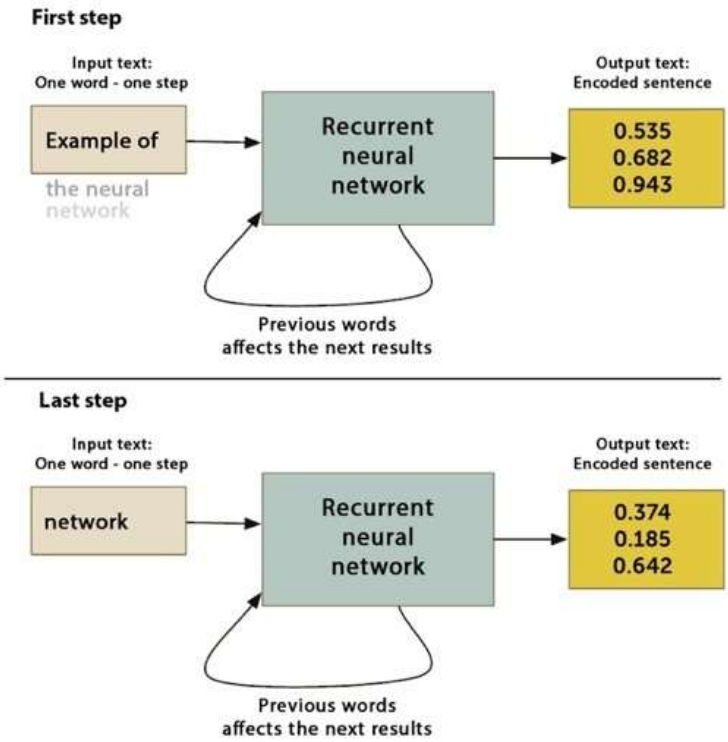

Fig. 2. Visualization of input unstructured data encoding

After performing such a simple operation, we obtain the encoded unstructured data, for example text, that looks like a numerical data set. At the initial stage of training, these numbers are random and generated by the algorithm also accidentally. Next passing of the text that has already encoded, RNN will be evaluated to the same numerical data set. The algorithm of decoding of the unstructured data works like encoding, only in the reverse - the input receives a numerical data set and outputs the probable text that corresponds to this data [7-12].

Once we understand the essence of encoding and decoding of the unstructured data, let's move to the very essence of our task - machine translation and its general algorithm. To do this, we just have to combine these two RNNs - for encoding and decoding - and get the following result:
Thus, we obtain the general way of transforming the sequence of Ukrainian words into an equivalent sequence of English words, this is the so-called, sequential method of language translation Sequence-to-Sequence. The main pros of the method are [13-16]:

- this approach is limited on the training data set amount and the computing power that you can allocate to the translation. Researchers of machine learning have invented this method only a few years ago, but such systems are already working better than the machine translation statistical systems, which was developing through last 20 years;
- the system does not depend on knowledge of any rules of the language. The algorithm itself defines these rules and constantly adapted. Lower level headings remain unnumbered; they are formatted as run-in headings.

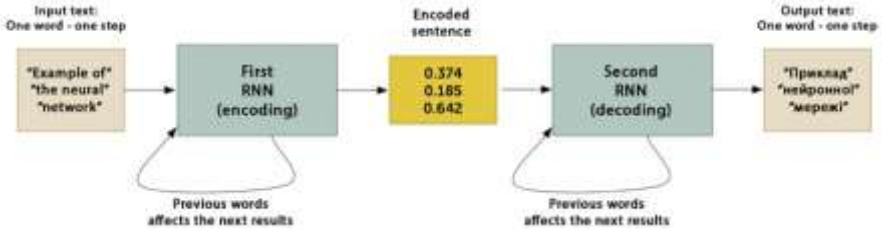

Fig. 3. Sequence-to-Sequence model.

3 Solution Analyze

Let’s conduct more information about our dataset and how we will collect that data. First and the most obvious way to collect data is to use open-source datasets, but this way of mining data is not so suitable, in case data will be noisy and will require a lot of economic resources to get from this data high accuracy results in any unique case.Another case is to create own dataset, this is a better way to create personalized solutions for any type of data. The main way to evaluate how noisy is current dataset is to calculate entropy [17].

$$H(x) = E[\log \frac{1}{p(X)}] \leq \log E[\frac{1}{p(X)}] = \log N \quad (3)$$

As you can see, the training data set consists of 10 phrases, that are widely used in open data sources related to legal cases, we will use that data to train and test our models, based on RNN approach, build on different ML libraries. After that will evaluate the speed and accuracy of the models.

Table 1. Dataset for training the RNN

English	Ukrainian
An example of a neural network .	Приклад нейронної мережі .
Statement by the Chairman of the UN Security Council.	Заява голови Ради Безпеки ООН.
Accepted the application.	Прийняв заяву.
The court made an order.	Суд ухвалив рішення.
Veto the law.	Ветувати закон.
Support the resolution.	Підтримати резолюцію.
Appeal the decision.	Оскаржити рішення.
Call for the fulfillment of obligations.	Закликати до виконання зобов'язань.
Emphasizes the need for strict compliance with regulations.	Підкреслює необхідність суворого дотримання постанов.
Support all initiatives.	Підтримувати всі ініціативи.

Let’s conduct experiments based on two machine learning libraries written in Python - PyTorch and Keras. The basis of the algorithm is the method of sequential learning.

Table 2. Comparison of Keras and PyTorch libraries results

Library title	Learning time	Training loops	Loss coefficient	Translation accuracy
Keras	4150 millis	400	0.0027	100%
PyTorch	5800 millis	650	0.0021	100%

Let's look at these data in more detail:

- Learning time. The value that shows the model's training time. Mainly depend on the environment where the script was run. Environment mean the current PC specifications; processor computing power and it upload by other processes.
- Training loops. The value that shows training cycles of the model. We give it ourselves.
- Loss coefficient. The value that shows the accuracy of the trained model. It is a measure of how good your model is.
- Translation accuracy. The value that shows in percentage term value of correct translation sentences.

So, the model build on the Keras library was more effective than the PyTorch model, the comparison based on the training time, training loops and error rate. Because of the small training data set, both algorithms show the maximum translation accuracy. In the case of increasing of training data set amount, models will provide completely

different loss coefficient and accuracy of the translation, training time and loss coefficient will increase and accuracy will decrease.

4 Results

The article explains the main stages of the development of machine translation technologies, describes the main architectural solutions used in machine translation nowadays. The advantages and disadvantages of several approaches, such as rule-based, statistical, and neural network-based are described. Considering all the factors, the most relevant way of organization and software approach for creating methods for analyzing open data in legal cases. Moreover, overviewed design and software approach of the two systems for numbering unstructured data based on different ML frameworks was chosen. For example, this solution will be suitable for translating sentences from one language to another. In the case of an RNN-based language translation approach, the most popular ML libraries are Keras and PyTorch.

In order to perform the English-Ukrainian language case study, we have used English- Ukrainian and Ukrainian-English as base language pairs, as it was shown in table 3. Based on that, we have the final result, of three different approaches comparison to the current set.

Table 3. Used English- Ukrainian and Ukrainian-English as base language pairs

English-Ukraine MT systems	Adequacy	Fluency
Rules-based	55.6%	47%
Statistical	77.2%	87%
RNN	98%	96%

5 Conclusion

RNN, like other classes of neural networks, are developing so fast that it's increasingly difficult to track new, more interesting, and more sophisticated models for solving more complex and complicated tasks.

These sequential methods of teaching neural networks can be used in other areas, not only in machine translation. Simple examples are models that could make verbal descriptions of the image, recognize the voice and maintain the conversation. In our opinion, the development of RNN will lead to the emergence of smart assistants that can recognize the owner's voice and correctly perceive the task. At the moment RNNs are the most frequently used in machine translation and we think this field will be also upgraded in the nearest future.

According to the results of the experiment, the model based on Keras library is more efficient for the current training data set. Note, that the research results may be considered relevant only for small data sets and there will be changes in translation

quality and training time after increasing the training data set amount. Next phase of this research may consist of model training in large data volumes with analyzing and comparing the quality and speed of its work.

References

1. Gahegan, M.: On the application of inductive machine learning tools to geographical analysis, Geographical Analysis, vol. 32, pp. 113–139 (2000)
2. Zhang, C., Murayama, Y.: Testing local spatial autocorrelation using, Intern. J. of Geogr. Inform. Science, vol. 14, pp. 681–692 (2000)
3. Estivill-Castro, V., Lee, I.: Amoeba: Hierarchical clustering based on spatial proximity using Delaunay diagram, 9th Intern. Symp. on spatial data handling, pp. 26–41, Beijing, China (2000)
4. Kryvenchuk Y., Boyko N., Helzynskyy I., Helzhynska T., Danel R.: Synthesis control system physiological state of a soldier on the battlefield. CEUR. Vol. 2488. Lviv, Ukraine, p. 297–306. (2019)
5. Kang, H.-Y., Lim, B.-J., Li, K.-J.: P2P Spatial query processing by Delaunay triangulation, Lecture notes in computer science, vol. 3428, pp. 136–150, Springer/Heidelberg (2005)
6. Boehm, C., Kailing, K., Kriegel, H., Kroeger, P.: Density connected clustering with local subspace preferences, IEEE Computer Society, Proc. of the 4th IEEE Intern. conf. on data mining, pp. 27–34, Los Alamitos (2004)
7. Wang, Y., Wu, X.: Heterogeneous spatial data mining based on grid, Lecture notes in computer science, vol. 4683, pp. 503–510, B.: Springer/Heidelberg (2007)
8. Harel, D., Koren, Y.: Clustering spatial data using random walks, Proc. of the 7th ACM SIGKDD Intern. conf. on knowledge discovery and data mining, pp. 281–286, San Francisco, California (2000)
9. Turton, I., Openshaw, S., Brunsdon, C.: Testing spacetime and more complex hyperspace geographical analysis tools, Innovations in GIS 7, pp. 87–100, London: Taylor & Francis (2000)
10. Boyko N., Pylypiv O., Peleshchak Y., Kryvenchuk Y., Campos J.: Automated document analysis for quick personal health record creation. 2nd International Workshop on Informatics and Data-Driven Medicine. IDDM 2019. Lviv. p. 208-221. (2019)
11. Kryvenchuk Y., Mykalov P., Novytskyi Y., Zakharchuk M., Malynovskyy Y., Řepka M.: Analysis of the architecture of distributed systems for the reduction of loading high-load networks. Advances in Intelligent Systems and Computing. Vol.1080. p.759-550. (2020)
12. Tung, A.K, Hou, J., Han, J.: Spatial clustering in the presence of obstacles, The 17th Intern. conf. on data engineering (ICDE'01), pp. 359–367, Heidelberg (2001)
13. Veres, O., Shakhovska N.: Elements of the formal model big date, The 11th Intern. conf. Perspective Technologies and Methods in MEMS Design, pp. 81-83, Polyana (2015)
14. Agrawal, R., Gehrke, J., Gunopulos ,D., Raghavan, P.: Automatic sub-space clustering of high dimensional data, Data mining knowledge discovery, vol. 11(1), pp. 5–33 (2005)
15. Guimei, L., Jinyan, L., Sim, K., Limsoon, W.: Distance based subspace clustering with flexible dimension partitioning, Proc. of the IEEE 23rd Intern. conf. on digital object identifier, vol. 15. Iss. 20, pp. 1250–1254 (2007)
16. Aggarwal, C., Yu, P.: Finding generalized projected clusters in high dimensional spaces, ACM SIGMOD Intern. conf. on management of data, pp. 70–81 (2000)

17. Procopiuc, C.M., Jones, M., Agarwal, P.K., Murali, T.M.: A Monte Carlo algorithm for fast projective clustering, ACM SIGMOD Intern. conf. on management of data, pp. 418–427, Madison, Wisconsin, USA (2002)
18. Kryvenchuk Y., Vovk O., Chushak-Holoborodko A., Khavalko V., Danel R.: Research of servers and protocols as means of accumulation, processing and operational transmission of measured information. Advances in Intelligent Systems and Computing. Vol.1080. p.920-934. (2020)
19. Ankerst, M., Ester, M., Kriegel, H.-P.: Towards an effective cooperation of the user and the computer for classification, Proc. of the 6th ACM SIGKDD Intern. conf. on knowledge discovery and data mining, pp. 179–188, Boston, Massachusetts, USA (2000)
20. Peuquet, D.J.: Representations of space and time, N. Y.: Guilford Press (2002)
21. Guo, D., Peuquet, D.J., Gahegan, M.: ICEAGE: Interactive clustering and exploration of large and high-dimensional geodata, Geoinfor-matica, vol. 3, N. 7, pp. 229–253 (2003)
22. Boyko, N., Shakhovska, Kh., Mochurad, L., Campos, J.: Information System of Catering Selection by Using Clustering Analysis, Proceedings of the 1st International Workshop on Digital Content & Smart Multimedia (DCSMart 2019), pp. 94-106, Lviv, Ukraine (2019)
23. Boyko, N., Komarnytska, H.,Kryvenchuk ,Yu., Malynovskyy, Yu.: Clustering Algorithms for Economic and Psychological Analysis of Human Behavior, Proceedings of the International Workshop on Conflict Management in Global Information Networks (CMiGIN 2019), pp. 614-626, Lviv, Ukraine (2019)
24. Fedushko S., Syerov Yu., Tesak O., Onyshchuk O., Melnykova N. (2020) Advisory and Accounting Tool for Safe and Economically Optimal Choice of Online Self-Education ServicesProceedings of the International Workshop on Conflict Management in Global Information Networks (CMiGIN 2019), Lviv, Ukraine, November 29, 2019. CEUR-WS.org, Vol-2588. pp. 290-300. http://ceur-ws.org/Vol-2588/paper24.pdf
25. Yavorska T., Prihunov O., Syerov Yu. Efficiency of Using Social Networks in the Period of Library Activity in Remote Mode. CEUR Workshop Proceedings. Vol 2616: Proceedings of the 2nd International Workshop on Control, Optimisation and Analytical Processing of Social Networks (COAPSN-2020), Lviv, Ukraine, May 21, 2020. p. 214-226. http://ceur-ws.org/Vol-2616/paper18.pdf
26. Boyko, N., Basystiuk, O.: Comparison Of Machine Learning Libraries Performance Used For Machine Translation Based On Recurrent Neural Networks, 2018 IEEE Ukraine Student, Young Professional and Women in Engineering Congress (UKRSYW), pp.78-82, Kyiv, Ukraine (2018).