

MINISTRY OF FINANCE OF UKRAINE
STATE TAX UNIVERSITY

FACULTY OF FINANCE AND DIGITAL TECHNOLOGIES
DEPARTMENT OF COMPUTER AND INFORMATION
TECHNOLOGIES AND SYSTEMS

QUALIFICATION WORK

на тему: ПЕРСПЕКТИВА РОЗВИТКУ МАШИННОГО ПЕРЕКЛАДУ
ТЕХНІЧНИХ ТЕРМІНІВ ІТ ГАЛУЗІ ІЗ ЗАСТОСУВАННЯМ
ШТУЧНОГО ІНТЕЛЕКТУ

on the topic: PROSPECTS FOR THE DEVELOPMENT OF MACHINE
TRANSLATION OF TECHNICAL TERMS IN THE IT INDUSTRY WITH
THE USE OF ARTIFICIAL INTELLIGENCE

Student of higher education of the second (Master's) level,
second year of studies, group KMIz-22-1, specialty 122 "Computer science",
educational programme "Information Management Systems and Technologies in
the Economics"

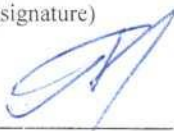
Higher education
student



(signature)

Kostina Olesia Volodymyrivna
(full name)

Scientific adviser



(signature)

Horbovyi Artur Yulianovych,
Doctor of Technical Science,
Professor
(full name, academic degree, academic title)

Head of educational and
professional programme



(signature)

Horbovyi Artur Yulianovych,
Doctor of Technical Science,
Professor
(full name, academic degree, academic title)

Admitted to defense by the decision of the Department
protocol dated December 28, 2023 No. 8

Head of the department



(signature)

Filonenko Mykhailo
Mykolayovych, Ph.D., Associate
Professor
(full name, academic degree, academic title)

Registration No. 1830.01.2024

(date)



(signature of the responsible person)

Check results

for defence

(for defence, for revision)

30.01.2024

(date)



(signature)

Defense result

5 (Excellent)

(assessment)

09.02.2024

(date)



(signature)

Chairman of commission:



(signature)

V.M. Franchuk

Members of the commission:



(signature)

A. Yu. Horbovyi

(signature)

O.V. Redych

(signature)

V.O. Nizhehorodtsev

QUALIFICATION WORK ASSESSMENT FORM

Objects of control	Assessment range in points	Actual number of points	Notes	Date, full name: Head, chairman of the commission, members of the commission, secretary, (signatures)
ASSESSMENT OF THE QUALIFICATION WORK BY THE DEFENSE COMMISSION				
1. Contents of the graduation work (quality of the theoretical material, its connection with the practical material, ability to present the material in a logical sequence, presence of conclusions and proposals)	from 0 to 30	30		09.02.2024 Franchuk V.M. <i>[Signature]</i> Horbovyi A.Yu. <i>[Signature]</i> Redych O.V. <i>[Signature]</i> Nizhehorodtsev V.O. <i>[Signature]</i> Sukhovetska L.P. <i>[Signature]</i>
2. Compliance with registration requirements	from 0 to 10	8		
3. Timely completion of work on at all stages within the deadlines set by the department	from 0 to 5	4		
4. Reviewer	from 0 to 5	5		
TOTAL POINTS	from 0 to 50	47		

Assessment of the qualification work by the Scientific adviser - 48 points.

CONVERTING THE ASSESSMENT ON A 100-POINT SCALE TO THE NATIONAL SCALE AND ECTS SCALE

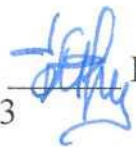
100-point scale assessment	ECTS assessment	National scale assessment	Date	Full name: chairman, members of the commission, secretary, (signature)
95	A	5 (excellent)	09.02.2024	Franchuk V.M. <i>[Signature]</i> Horbovyi A.Yu. <i>[Signature]</i> Redych O.V. <i>[Signature]</i> Nizhehorodtsev V.O. <i>[Signature]</i> Sukhovetska L.P. <i>[Signature]</i>

RULES FOR CONVERTING THE ASSESSMENT INTO THE NATIONAL AND ECTS SCALES

100-point scale assessment	ECTS assessment	National scale assessment
90-100	A	5 (Excellent)
80-89	B	4 (Good)
70-79	C	
60-69	D	
50-59	E	3 (Satisfactory)
35-49	FX	2 (Unsatisfactory)

Faculty of Finance and Digital Technologies
 Department of Computer and Information Technologies and Systems
 Field of knowledge: 12 "Information Technologies"
 Specialty: 122 "Computer Science"
 Educational programme "Information Management Systems and Technologies in Economics"

I hereby approve

Head of the Department  Filonenko M.M.
 "04" 04 2023

Assignment for qualification work to a student of higher education

Kostina Olesia Volodymyrivna

(full name)

1. The topic of the qualification work

Перспектива розвитку машинного перекладу технічних термінів it галузі із застосуванням штучного інтелекту.

Prospects for the development of machine translation of technical terms in the IT industry with the use of artificial intelligence.

Approved by the Rector's order dated April 4, 2023 No. 363.

2. The deadline for submitting the completed work is December 20, 2023.

3. Purpose of the work is the research of machine translation technologies for technical terms in the IT industry using artificial intelligence.

The object of the work are the most popular neural machine translation systems.

The subject of the work is the use of machine translation to improve the quality, speed and cost of translation of technical terms in the IT industry.

The estimated research base is the innovative new IOTOR product for the Internet of Things (IoT) developed by OKIN Facility's software engineers.

4. Background to the work are the methodology and evaluation of machine translation systems.

5. List of graphic material: text of the main page of the IOTOR product for the Internet of Things (IoT), code for BLEU metric evaluation, graphs of the BLEU score and precision results.

6. Contents of the qualification work (list of questions for each section)

CONTENTS

INTRODUCTION

SECTION 1. MACHINE TRANSLATION OF TECHNICAL TERMS IN THE IT INDUSTRY

1.1. Theoretical Analysis of the Study. Related Works

1.2. Machine Translation Models and Evaluation Metric

SECTION 2. MATERIALS AND METHODS

2.1. BLEU Metric, NLTK (Natural Language Toolkit) library

2.2. Practical Outcomes

SECTION 3. RESEARCH RESULTS

3.1. Capabilities of Machine Translation

3.2. Prospects for the development of MT for IT industry

SUMMARY
LIST OF REFERENCES
APPENDIXES

7. Work schedule

No.	Name of the stages of qualification work	Deadlines for completion of work stages	
		According to the plan	In fact
1	2	3	4
1	Choosing a topic, selecting list of references, forming the structure of a qualification work	Until 31.03.2023	31.03.2023
2	Section 1 (to reveal the essence and state of research of the scientific problem of the work, and then give its general characteristics; review of the list of references on the topic and the choice of specific areas of research)	28.06.2023	28.06.2023
3	Section 2 (analysis of the experimental part on research methods)	25.10.2023	25.10.2023
4	Section 3 (theoretical and/or experimental studies and project solutions)	01.12.2023	01.12.2023
5	Conclusions	10.12.2023	10.12.2023
6	Finalization of the work taking into account the comments of the Scientific adviser	20.12.2023	20.12.2023
7	Getting feedback from your adviser	31.01.2024	31.01.2024
8	Getting an external review	02.02.2024	02.02.2024

8. The issue date of the assignment is on April 04, 2023

9. Head of the graduation qualification work


 (signature)

Horbovyi A. Yu.
 (surname, initials)

10. Head of educational and professional programme


 (signature)

Horbovyi A. Yu.
 (surname, initials)

11. The assignment was given to the higher education student


 (signature)

Kostina O. V.
 (surname, initials)

АНОТАЦІЯ

Перспективи розвитку машинного перекладу технічних термінів в ІТ-галузі з використанням штучного інтелекту

Мета роботи полягає в дослідженні технологій машинного перекладу технічних термінів в ІТ-галузі з використанням штучного інтелекту.

Завданнями роботи є дослідження найбільш ефективних моделей машинного перекладу з англійської на українську мову, проведення аналізу найпоширеніших моделей машинного перекладу з використанням метрики BLEU для оцінки якості перекладу, визначення найкращої моделі машинного перекладу для технічних термінів ІТ-галузі для нового продукту OKIN Facility IOTOR у сфері Інтернету речей (IoT), щоб оптимізувати якість перекладу.

Предметом дослідження є використання машинного перекладу для оптимізації якості, швидкості та вартості перекладу технічних термінів в ІТ-галузі.

Об'єктом дослідження є найбільш поширені нейронні машинні перекладачі.

Базою дослідження виступає новітній продукт IOTOR для Інтернету речей (IoT), розроблений інженерами-програмістами компанії OKIN Facility.

За результатами дослідження сформульовані методика та оцінка машинних перекладачів.

Одержані результати можуть бути використані для покращення якості машинного перекладу технічної документації в ІТ-галузі та для веб-сайтів, а також для оптимізації часу і витрат на переклад.

Кваліфікаційна робота містить 60 сторінок, 1 таблиць, 10 рисунків, список використаних джерел 47 найменувань, 3 додатка (вкл. 11 рис.).

Ключові слова: Машинний переклад, нейронний машинний переклад, штучний інтелект, BLEU метрика, машинне навчання.

Рік виконання кваліфікаційної роботи: 2023.

Рік захисту роботи: 2024.

ABSTRACT

Prospects for the development of machine translation of technical terms in the IT industry with the use of artificial intelligence

The purpose of this work is the research of machine translation technologies for technical terms in the IT industry using artificial intelligence.

The tasks of the work are to research the most effective models of machine translation from English to Ukrainian, analyze the most common machine translation models using the BLEU metric to assess translation quality, opt the best machine translation model for technical IT terms for OKIN Facility IOTOR's new Internet of Things (IoT) product to improve the translation quality.

The subject of the work is the use of machine translation to improve the quality, speed and cost of translation of technical terms in the IT industry.

The object of the work are the most popular neural machine translation systems.

The work is based on the innovative new IOTOR product for the Internet of Things (IoT) developed by OKIN Facility's software engineers.

The work provides a methodology and evaluation of machine translation systems.

The scientific and practical significance of the results obtained is to improve the quality of machine translation of technical documentation in the IT industry and for websites, as well as reduce translation time and costs.

The research consists of 60 pages, 1 Table, 10 Figures, a list of 47 References, and 3 Appendixes (including 11 Figures).

Keywords: Machine Translation, Neural Machine Translation, Artificial Intelligence, BLEU Metric, Machine Learning

Year of qualification work: 2023.

Year of work defense: 2024.

CONTENTS

INTRODUCTION.....	9
SECTION 1. MACHINE TRANSLATION OF TECHNICAL TERMS IN THE IT INDUSTRY	12
1.1. Theoretical Analysis of the Study. Related Works	12
1.2. Machine Translation Models and Evaluation Metric.....	15
SECTION 2. MATERIALS AND METHODS	24
2.1. BLEU Metric, NLTK (Natural Language Toolkit) library	24
2.2. Practical Outcomes	32
SECTION 3. RESEARCH RESULTS	42
3.1. Capabilities of Machine Translation	42
3.2. Prospects for the development of MT for IT industry	45
SUMMARY	47
LIST OF REFERENCES	49
APPENDIXES.....	54

INTRODUCTION

The relevance of the topic is determined by the dynamics of the IT industry, which generates large volumes of documentation, code, technical specifications, and other materials that need to be translated for global deployment and collaboration. Machine translation is becoming a key tool for solving this problem, but the accuracy of technical terms remains an important issue.

The use of artificial intelligence in machine translation opens up great opportunities for improving translation efficiency and quality. The use of deep learning, neural networks, and other advanced AI technologies can help automate the translation process and adapt to the rapidly changing requirements of the IT industry.

The prospect of developing machine translation of technical terms in the IT industry using artificial intelligence is quite promising [40]. Due to the constant growth of information and the globalization of the IT industry, the need for fast and accurate translation of technical documentation is becoming increasingly important [7].

Artificial intelligence and machine learning have already significantly improved the efficiency of machine translation in general [16]. However, technical terms in the IT industry have their own peculiarities, such as complexity, specificity, and rapid changes. Therefore, to achieve high translation quality in this industry, some challenges need to be taken into account:

- 1) vocabulary: IT industry is developing rapidly, and new technical terms are constantly emerging. Machine translation needs to keep its vocabulary up to date, including new terms and their definitions;
- 2) achieving accuracy: technical terms have precise definitions, and incorrect translation can lead to misunderstanding or even errors in task performance. Achieving high accuracy in machine translation requires continuous training of models on a large volume of high-quality technical data;
- 3) context: some terms in the IT industry can have different meanings depending on the context in which they are used. Considering the context and using context-sensitive models can improve translation quality.

In recent years, significant progress has been made in machine translation of technical terms in the IT industry. The use of deep neural networks and the increasing amount of available data has helped to improve translation accuracy and speed [12].

Future development prospects include continuously training models on larger and more diverse datasets, expanding the vocabulary and improving contextual understanding. It is also important to develop interactive tools that allow users to make corrections and additions to the translation so that the models can learn from this data and continuously improve.

All in all, machine translation of technical terms in the IT industry using artificial intelligence has the potential for further development, and it is expected that even greater results in terms of relevance, accuracy, and translation speed will be achieved in the future.

The purpose of the work is to highlight current technological challenges in the field of machine translation, analyze existing approaches, and develop new strategies to improve the translation of technical terms in the IT industry.

In order to achieve this goal, the following tasks are being solved:

- 1) use the most effective models of machine translation from English to Ukrainian;
- 2) analyze different machine translation models using artificial intelligence by evaluating the quality of machine translation using the BLEU metric;
- 3) select the best machine translation model for technical IT terms for OKIN Facility IOTOR's new Internet of Things (IoT) product to improve the high quality translation.

The object of the work is the most common machine translators widely used by experienced human translators.

The subject of the work is the use of machine translation to improve the quality, speed and cost of translation of technical terms in the IT industry.

The following research methods were used in the work:

- 1) BLEU (Bilingual Evaluation Understudy): measures the accuracy of a translation compared to manual translations. It is used to evaluate the quality of translation according to the similarity with reference texts;
- 2) tokenization and quantitative calculation of the coincidence of the units under study; the accuracy of the coincidence compared to the reference (precision).

The scientific and practical significance of the results obtained is that the use of machine translation can reduce the financial costs and time spent on translation and localisation of technical documentation in the IT industry and information posted on a website.

Practical significance of the results. The results of the qualification project on machine translation evaluation are of great practical importance for business. High-quality translation and information that is localised for the Ukrainian user helps to attract new customers, increase sales and enhance the company's brand.

The work was approbated in publications: "State-of-the-art machine translation systems using artificial neural network" and "Assessing AI-driven machine translation performance from English into Ukrainian using BLEU metrics for IoT product localization" [6, 7] at the Ukrainian and international conferences.

SECTION 1. MACHINE TRANSLATION OF TECHNICAL TERMS IN THE IT INDUSTRY

1.1. Theoretical Analysis of the Study. Related Works

During the initial stages of contemporary machine translation systems, observers highlighted their tendency to be trained for producing a "standard" target language. However, it's crucial to recognize that various languages encompass diverse forms, including regional variants, dialects, social languages, and non-native forms, which deviate from the standard language. Sachin Kumar stated that such varieties are often low-resource, and hence do not benefit from contemporary NLP solutions, MT included [11]. Recent research papers of Myroslav Konyk and Victoria Vysotska provide an effective solution for a typical architecture of an intelligent Ukrainian-English machine translation system based on recurrent neural networks [12]. With the development of machine learning technologies, it is becoming possible to improve the results in an outstanding way. Neural Machine Translation (NMT) is a new approach to machine translation that differs from traditional statistical machine translation in that it emphasises the creation of a single artificial neural network. This network can be collectively tuned to optimise translation performance. The use of automated metrics plays a key role in both the development and evaluation of machine translation systems. However, assessing the extent to which these automated metrics meet the gold standard of human evaluation is a challenging problem [5, 32].

The majority of the systems in the last three years have been neural models, for which most metrics have a high correlation with human judgements. A BLEU Method for Automatic Evaluation of Machine Translation by Kishore Papineni has been surpassed by various other metrics at every iteration of the MT metrics shared task. Despite this, and extensive analytical evidence of the limitations of BLEU in particular and automatic metrics in general [14, 32], the metric remains the de facto standard of evaluating research hypotheses.

In recent years, significant progress has been made in machine translation of technical terms in the IT industry. The word "term" comes from the Latin "terminus"

and means a boundary or border; it outlines a concept that can be used in a particular field with the most precise semantic definition [1]. There are many classifications of terms. Bilozerska L. suggested to divide them: by the industry in which they are used, by structural features, by the way they are built, etc. According to the classification, terms can be divided into word terms, phrase terms, and multi-component terms. The term should be emotionally neutral and provide accurate definitions of concepts or phenomena. Terminology, like its development, is inherent in the progress of civilization. All social, political, and economic changes, as well as scientific and technological discoveries, give rise to new concepts, processes, devices, and functions that need to be named in some way to enable the exchange of information and specialized knowledge. Technical inventions such as video, DVDs or flash drives are not new now, but when they entered the market, the terms describing them were neologisms in English. A terminology should not be just a "list of words" but an organized conceptual system with a systematic set of rules that facilitate the process of understanding, creating, and translating terms [1, 3].

The development of technology and the absence of tangible territorial boundaries in scientific research make it necessary to rapidly develop terminology, organizing new concepts, and find equivalents of the resulting terms in other, terminologically equipped languages of the world.

English-language terms are widely used in many industries and sciences. English is the language of international communication, which means that terms originally appear in English and only then are spread to other languages. The development of society requires deepening the basic scientific concepts and creating terms to denote scientific and technical neologisms. The English terminology system is one of the most developed, constantly changing and filled with new terms. English is a global language code of the twentieth century that has acquired the status of an international language. English is the main language on the Internet. Terminological vocabulary is grammatically adapted to the language of the recipient.

Translation of technical terms in the information technology (IT) industry has its own peculiarities due to the specificity of terminology and the rapidly changing

nature of this industry. Some of the main features include new terms and technologies. IT is constantly evolving, and the emergence of new technologies is accompanied by the emergence of new terms. A translator must be aware of the latest trends and technologies to use newly created terms correctly. International standards also determine the specifics of translation. Many technical terms in the IT industry are defined by international standards, such as ISO, IEEE. A translator should take these standards into account and follow them when translating. Specific terminology in IT can be difficult to understand for those who are not experts in the field. A translator must have in-depth knowledge of information technology to correctly convey the meaning of the terms. Many technical terms have multiple meanings or can be used in different contexts. It is important to take this polysemy into account and choose the appropriate translation in a particular context. When translating technical terms, it is also important to consider the audience for which the translation is being done. If it is documentation for specialists, you can use more technical terms, but if it is instructions for the end user, you should avoid too technical language. Correct translation of technical terms in the IT industry is important to ensure clarity and correct use of terminology in an international context.

Machine translation (MT) has become a popular tool for translating texts, including technical terms in the information technology (IT) industry.

However, it is important to understand that MT can have limitations, especially when it comes to specific terminology [1, 2]. When using machine translation for technical terms in the IT industry, context should be taken into account. Machine translators may have difficulty recognizing context, especially when translating words with multiple meanings or abstract terminology. New translation technologies may not always be able to take into account the latest technologies and terms, as IT is rapidly evolving. Consequently, some terms may not have an exact equivalent in another language. After automatic translation, it is always recommended to correct and edit the text, especially if the accuracy and specificity of the terms are important. Some terms may have different translations in different language communities. However, MT may

try to use a generally accepted translation, which may not take into account the specifics of a particular language community.

The application of deep neural networks and the utilisation of the growing amount of available data is improving the accuracy and speed of translation.

Prospects for further development include continuous training of models on large and diverse datasets, expanding vocabulary and improving contextual understanding. In addition, it is essential to create interactive tools that allow users to make corrections and additions to translations. This feedback allows models to learn from user data, contributing to continuous improvement in translation quality [2].

All in all, the application of artificial intelligence for machine translation of technical terms in the IT industry holds significant potential for future advancements. Anticipated improvements include enhanced relevance, accuracy, and translation speed, paving the way for even more impressive outcomes.

1.2. Machine Translation Models and Evaluation Metric

As machine translation has been developing, different types of translation systems have emerged, each with its own unique advantages and challenges. The three most common categories are rule-based machine translation, statistical translation, and neural translation.

Rule-based machine translation (RBMT) is an early stage of machine translation that relies on an extensive set of predefined linguistic rules to transfer textual meaning between languages. [22]. Regardless of its historical importance, RBMT exhibited poor translation quality, requiring manual addition of languages and considerable human effort for subsequent editing. Rule-based MTs are currently being used less and less frequently.

Statistical Machine Translation (SMT) builds a statistical model that identifies associations between words, phrases and sentences in a given text. This model serves to translate these elements into another language, representing an advance over rule-based machine translation, although it faces a number of parallel challenges. Employing statistical models trained to translate text from a source language to a target

language, they utilize a substantial corpus of examples. SMT has held sway as the predominant translation paradigm for many years. Common practical implementations of SMT often involve phrase-based machine translation (PBMT) systems, which translate sequences of words or phrases of variable length [Ошибка! Источник ссылки не найден.].

Regardless of their efficiency, statistical machine translation methods face limitations due to their narrow focus on translated phrases and ignoring the overall structure of the target text. The emphasis on data-driven approaches also led to the potential neglect of crucial syntactic differences recognised by linguists. In addition, statistical approaches required careful customisation of each module in the translation pipeline.

In modern applications, neural machine translation (NMT) has been largely substituted for statistical machine translation (SMT) and is sometimes incorporated into regular machine translation systems. NMT uses artificial intelligence to "learn" languages and improve their comprehension iteratively, mirroring the learning process of neural networks in the human brain. Unlike traditional methods based on rigid rules, the neural network in a machine translation system encodes and decodes the source text.

Neural Machine Translation (NMT), characterised by high accuracy, support for additional languages and increased speed after training, is the main standard in the continuous evolution of machine translation technologies.

Prominent leaders in machine translation, including Google, Microsoft, and Amazon, prioritise NMT for its ability to provide fine-grained translations and easily integrate new language pairs. Such adaptability is attributed to the learning and improvement mechanisms built into machine translation systems, allowing them to improve performance with increasing utilisation.

Artificial Intelligence (AI) is playing a key role in shaping global content strategies, and the language services industry recognises its potential to revolutionise the processing and distribution of multilingual content. Accordingly, the sector is at the forefront of the development of neural machine translation technologies.

In today's interconnected world, AI and machine translation (MT) are becoming increasingly indispensable tools for the management and distribution of multilingual content. Due to rapid technological advancements, AI and MT enable near-human-quality translations at a fraction of the time and cost associated with traditional human translation. This has driven significant growth and investment in the language services industry as companies look to utilise these tools to enhance their global content strategies.

While AI and MT have enormous potential, their widespread adoption also comes with challenges and limitations that deserve attention. This paper reviews the current state of AI and MT, examines their applications (such as Google Translate and DeepL), discusses their benefits and limitations, and assesses their impact on the language services industry and other sectors.

Neural Machine Translation (NMT) involves the use of neural network models to train a statistical machine translation model. The main advantage of this methodology is the ability to train a single system directly on source and target texts, which eliminates the need to create a whole pipeline of specialised systems, as is the case in traditional statistical machine learning. Unlike conventional phrase translation systems, which consist of many independently customisable subcomponents, neural machine translation seeks to build and train one large neural network capable of understanding a sentence and producing an accurate translation.

Thus, neural machine translation systems are considered end-to-end systems because only one model is required for translation. The strength of NMT lies in its ability to learn directly, in an end-to-end manner, the mapping of input text to the corresponding output text **[Ошибка! Источник ссылки не найден.]**.

Over the last decade there appeared some significant machine translation systems: Google Translate, Bing Microsoft Translator, Transformer Models such models as BERT and GPT, and DeepL.

Comparing different machine translation systems (Google Translate, Microsoft Translator and ChatGPT), the scientists came to the following results [35, 12]: Google Translate and Microsoft Translator both outperformed ChatGPT in all the evaluation

metrics. These systems demonstrated higher BLEU and other scores than ChatGPT, indicating better translation qualities. Google Translate demonstrated slightly better performance than Microsoft Translator, but the difference was marginal. Although the performance of ChatGPT was lower than the other two systems, it is important to note that ChatGPT was primarily designed as a conversational AI model rather than a translation system. The results show that ChatGPT's performance is not on par with specialised translation systems such as Google Translate and Microsoft Translator. These results provide a valuable benchmark for users who decide to use translation systems for specific purposes, e.g. those who prioritise translation quality may choose Google Translate or Microsoft Translator. Based on the above research we have chosen the Google Translate for our qualification work. The other machine translation system for our qualification work is DeepL. It was chosen because the competition between Google Translate and DeepL for the number one position as the best machine translation system has been running for the past decade [Ошибка! Источник ссылки не найден., 35], and moreover DeepL is still number one among the human translators [340].

Launched in April 2006 as a statistical machine translation service, Google Translate switched to neural machine translation — Google Neural Machine Translation (GNMT) in 2016 [40]. GNMT improves the quality of translation by applying an example-based machine translation (EBMT) method in which the system "learns from millions of examples." GNMT's system learning architecture was first tested in more than a hundred languages supported by Google Translate. Thanks to its powerful internal structure, the system learns to produce better and more natural translations over time. GNMT is able to translate sentences as a whole, not in parts. GNMT is able to perform interlingual machine translation by encoding the semantics of a sentence instead of memorizing translations of individual phrases [18].

Google Translate is a multilingual neural machine translation service developed by Google to translate texts, documents, and websites from one language to another. It offers a web interface, a mobile app for Android and iOS, and an API that helps developers create browser extensions and software applications. As of April 2023, Google Translate supports 133 languages at various levels [Ошибка! Источник ссылки не найден.].

Google Neural Machine Translation (GNMT), a sequence-to-sequence (seq2seq) model that is currently used in Google Translate production systems [21, **Ошибка! Источник ссылки не найден.**]. Seq2seq (Sequence-to-Sequence) is a type of neural network architecture used in machine translation tasks such as Google Translate. It was introduced in 2014 by Sutskever et al. and has since become a popular approach for neural machine translation [24, 44].

Google Translate uses seq2seq models to translate text from one language to another. The basic idea is to use a network of encoders to encode the source language text into a fixed-length vector, and then use a network of decoders to generate the corresponding translation from this vector. The encoder network takes a sentence of the source language as input and transforms it into a sequence of vectors. Each vector represents a word in the sentence and is computed by applying a learned function to the word insertion and a hidden state vector that summarizes the previous words in the sentence. The final vector produced by the encoder is the summation of the entire sentence of the source language. The decoder network then takes this summary vector as input and generates the translation sentence word by word. At each time step, the decoder generates a probability distribution of words in the target language dictionary and selects the most probable word as the output. The hidden state of the decoder is updated based on the output word and the previous hidden state, which allows it to track the context and generate a free translation [21].

Encoder-decoder with attention is a neural network architecture that has become popular due to its ability to improve the quality of machine translation. The encoder-decoder model consists of two components: an encoder and a decoder. The encoder takes the source text as input and generates a fixed-length representation of it. The decoder then generates the output text based on the encoded representation. Attention is a mechanism added to an encoder-decoder architecture to help the decoder focus on relevant parts of the input text during the decoding process [21]. The attention mechanism allows the decoder to selectively pay attention to different parts of the input text at each step of the decoding process. This allows the model to better match the input and output text and improves the overall translation quality.

During training, the attention engine learns to weight the relevance of different parts of the input text based on the current state of the decoder. At each decoding step, the decoder uses this attention weighting to determine which parts of the encoded input text to focus on, improving the quality of the output translation. An attention-based encoder-decoder is a neural network model that improves machine translation by allowing the decoder to selectively focus on relevant parts of the input text during the decoding process. The attention mechanism allows the model to align the input and output text more efficiently, resulting in a higher quality translation [13, 24].

Google Translate's seq2seq model is trained on large parallel corpora of source and target language sentences. During training, the model is optimized to minimize the difference between predicted translations and true translations in the training data. Overall, seq2seq models have shown promising results in machine translation tasks and have become a key component of Google Translate's success. These models have great potential for the development of the natural language processing industry and are likely to be increasingly used in many applications in the future [24].

DeepL is a neural machine translation service launched in August 2017 and owned by DeepL SE, a Cologne-based company [27]. It initially offered translations between seven European languages and was gradually expanded to support 31 languages. Its algorithm uses convolutional neural networks (CNNs) [27], offers a paid subscription for additional features and access to an application programming interface for translation.

According to the developers, the service uses a newer, more advanced neural network architecture that provides more natural-sounding translations than competing services **[Ошибка! Источник ссылки не найден.]**.

The translation is generated by a supercomputer. In general, ANNs are slightly better suited for long word sequences, but are not yet used by competitors due to their weaknesses compared to recurrent neural networks. Weaknesses in DeepL are compensated for by additional methods **[Ошибка! Источник ссылки не найден., 26]**.

Convolutional neural networks (CNNs) are analogous to traditional artificial neural networks (ANNs) in that they consist of neurons that self-optimize through training. Each neuron still receives an input signal and performs an operation (e.g., a scalar product

followed by a nonlinear function) - this is the basis of countless ANNs. From the input vectors of raw images to the final result of class evaluation, the entire network will still express a single evaluation function (weight). The last layer will contain the loss functions associated with the classes, and all the usual tips and tricks developed for traditional ANNs still apply. The only noticeable difference between ANNs and traditional ANNs is that ANNs are mainly used in the field of pattern recognition in images. This allows us to encode image-specific features in the architecture, making the network Introduction to Convolutional Neural Networks more suitable for image-oriented tasks - while further reducing the parameters required to tune the model [8].

Convolutional neural networks differ from other forms of artificial neural networks in that instead of focusing on the entire problem domain, knowledge about a specific type of input data is used. This, in turn, allows for a much simpler network architecture.

To evaluate the quality of the translation, we chose the BLEU (Bilingual Evaluation Understudy) machine translation quality metric. The BLEU score measures the similarity between a machine translation and a reference translation performed by a human translator [5]. Although the BLEU score is a valuable tool for assessing the overall quality of a translation, it is important to remember that it is not a perfect measure, as it does not take into account factors such as idiomatic expressions, cultural nuances, and context. Since technical terms in the IT industry do not contain idioms and phrases, we can consider this metric suitable and valid for evaluation [31, 31].

BLEU can take values from 0 to 1 (Fig. 1.1.), where 0 means poor translation quality, i.e. that the resulting translation of the input sequence does not match the expected one. A value of 1 means that the machine translation result is exactly the same as the expected sentence. The mathematical model of the BLEU metric consists of two parts, namely, the brevity penalty and the n-gram overlap. BLEU scores correlate well with human judgment of translation quality. We should note that even human translators do not always achieve the perfect score of 1.

The BLEU metric counts the number of word combinations of different lengths, the so-called n-grams, that are present in the expected sentences and in the machine translation result. Longer n-grams are used to determine the correct sequence of words in the translation, or to check the meaning of the resulting sentence.

BLEU Score	Interpretation
< 10	Almost useless
10 - 19	Hard to get the gist
20 - 29	The gist is clear, but has significant grammatical errors
30 - 40	Understandable to good translations
40 - 50	High quality translations
50 - 60	Very high quality, adequate, and fluent translations
> 60	Quality often better than human

Fig. 1.1. Interpretation of BLEU score values

This figure displays the description of Blue score values, where 0 means poor translation and 1 means that the machine translation result is exactly the same as the expected sentence.

However, this metric has several drawbacks that should be kept in mind when comparing models:

- 1) does not work well when evaluating sentences that reflect the meaning of the sentence but are made up of other words;
- 2) does not evaluate the grammatical meaning of the sentence, i.e. if the translation does not contain a negation or other part of speech that greatly changes the meaning of the sentence, the metric will only reduce the number of points for the fact that such n-grams do not exist in the translation;
- 3) the absence of service words, such as articles in English, has the same negative impact on the grade as the absence or misuse of a word that affects the meaning;

4) sentence tokenization has a dramatic effect on the score, i.e. if the expected sentence has a different tokenization from the translation result, the score will be low, even though the textual content of the sentences may be the same.

The proposed study aims to evaluate the possibility of using the existing approaches presented in Section 1 to solve the text sentence translation task and determine which neural network model produces a sentence translation that is most accurate to the expected one. It was decided to conduct a study for two neural networks: Google Neural Machine Translation (GNMT) and Convolutional Neural Network (CNN) DeepL [6]. Both systems use deep neural networks. Google's architecture uses different models for translation, and GNMT uses deep recurrent networks (RNNs) and LSTM (Long Short-Term Memory) architecture for translation. DeepL uses convolutional layers in its models.

For this work, we used a set of natural language tools, or more commonly called NLTK, which is a set of libraries and programs for symbolic and statistical natural language processing (NLP) for English, written in the Python programming language. We were granted access to the information provided on the website in English, namely the innovative product IOTOR [46] using IoT (Internet of Things) technology, for the purpose of further translation and localization of this information into Ukrainian. The problems of translation and localization of technical terms in the IT industry using neural machine translators are the basis of my diploma project. Useful sources that came in handy were Kishore Papineni's research paper "BLEU: A method for automatic evaluation of machine translation" [32] and Google's notebook resources in Colab and GitHub, which we used for the research [47].

SECTION 2. MATERIALS AND METHODS

2.1. BLEU Metric, NLTK (Natural Language Toolkit) library

Software engineers at OKIN Facility have developed a new product for the Internet of Things (IoT) market - IOTOR. In order to optimize the costs of localizing the product in the Ukrainian market, it was decided to use a machine translator. The task is to choose the best model of machine translation of technical IT terms for the company's new product. The data used is taken from the public website (IOTOR) [46].

Google Translator and DeepL are chosen for the work as the most common machine translators widely used by experienced human translators. The task is to use different machine translation models from English into Ukrainian, analyze different machine translation models using artificial intelligence by evaluating the quality of machine translation using the BLEU metric, and select the best machine translation model for technical terms in the IT industry for OKIN Facility’s new product IOTOR, the Internet of Things (IoT). The text to be translated describes the environment of the new product (IOTOR).

Table 2.1.

Source, reference translation, Google Translate, DeepL

Source	Reference translation	Translation 1 (Google)	Translation 2 (DeepL)
1	2	3	4
SARAhub (Smart Assets Rapid Action) is easy-to-use environment for highly responsive IoT. SARAhub acts as a central nervous system for your assets – receiving information, processing it and triggering action when necessary in response to important events. It enables you to displays sensors and devices in hierarchical dashboards,	SARAhub (Smart Assets Rapid Action) — це просте у використанні середовище для інтернету речей (IoT) з високим рівнем реагування. SARAhub діє як центральна нервова система для ваших ресурсів: отримує інформацію, обробляє її та при необхідності запускає дії у відповідь на важливі події.	SARAhub (Smart Assets Rapid Action) — це просте у використанні середовище для швидкого реагування на IoT. SARAhub діє як центральна нервова система для ваших активів – отримує інформацію, обробляє її та запускає дії, коли це необхідно, у відповідь на важливі події.	SARAhub (Smart Assets Rapid Action) - це просте у використанні середовище для високочутливого Інтернету речей. SARAhub діє як центральна нервова система для ваших активів - отримує інформацію, обробляє її та запускає дії, коли це необхідно, у відповідь на важливі події. Він дозволяє відображати

Continued Table 2.1.

1	2	3	4
creating digital twin of your environment. It allows you to set up automated responses to specific situations, data consolidation, and many other features. Integration and responsiveness are key advantages of SARAHub. Integration of both different systems on input and output and different agendas into one environment. There is no need to deploy various systems for environmental monitoring, equipment maintenance or readings and energy control. All is in one environment, in which it is then possible to define reactions to different situations.	Таке середовище дозволяє відображати датчики та пристрої на ієрархічних інформаційних панелях, створюючи цифрового двійника вашого середовища. Це дозволяє налаштувати автоматичне реагування на конкретні ситуації, консолідацію даних і багато інших функцій. Інтеграція та оперативність є ключовими перевагами SARAHub. Інтеграція різних систем на вході та виході, а також різних програм в одне середовище. Немає необхідності розгортати різні системи для моніторингу навколишнього середовища, обслуговування обладнання або зняття показань і контролю енергоспоживання. Все знаходиться в одному середовищі, в якому потім можна визначити реакцію на різні ситуації.	Це дозволяє відображати датчики та пристрої на ієрархічних інформаційних панелях, створюючи цифрового двійника вашого середовища. Це дозволяє налаштувати автоматичні реакції на конкретні ситуації, консолідацію даних і багато інших функцій. Інтеграція та швидкість реагування є ключовими перевагами SARAHub. Інтеграція як різних систем на вході, так і на виході та різних планів в одне середовище. Немає необхідності розгортати різноманітні системи моніторингу навколишнього середовища, обслуговування обладнання чи зчитування показань та контролю енергії. Все в одному середовищі, в якому потім можна визначити реакції на різні ситуації.	датчики та пристрої на ієрархічних дашбордах, створюючи цифрових двійників вашого середовища. Це дозволяє налаштувати автоматичне реагування на конкретні ситуації, консолідацію даних та багато інших функцій. Інтеграція та оперативність - ключові переваги SARAHub. Інтеграція як різних систем на вході та виході, так і різних програм в одне середовище. Немає необхідності розгортати різні системи для моніторингу навколишнього середовища, обслуговування обладнання або зняття показань і контролю енергоспоживання. Все знаходиться в одному середовищі, в якому потім можна визначити реакцію на різні ситуації.

The above table displays the source translation (text taken from the website), reference translation (performed by the human) and two machine translations (Google Translate and DeepL). The above text is the description of the product IOTOR.

To assess the quality of machine translation, we use the BLEU (Bilingual Evaluation Understudy) metric (Formula 2.1). This metric was developed by a group of IBM researchers in 2002, and since then it has been the most widely used metric for evaluating machine translation quality [32]. The BLEU metric uses statistical tools without taking into account linguistic knowledge [31]. It is not the only and complete indicator of translation quality, but it still allows you to get a general idea of the result.

The BLEU metric formula:

$$BLEU = BP \times \exp \left(\sum_{n=1}^N W_n \cdot \log(P_n) \right), \quad (2.1)$$

where,

BP – the Brevity Penalty, which penalizes short translations. It is defined as $\min \left(1, \frac{\text{output length}}{\text{reference length}} \right)$;

N – the maximum n-gram order considered in the evaluation;

W_n – the weight assigned to the precision of n-grams set to $\frac{1}{N1}$ for uniform weighting;

P_n – the precision of n-grams, calculated as the ratio of the number of n-grams in the generated translation that match a reference translation to the total number of n-grams in the generated translation.

We use this Formula 2.1 to assess the quality of a given translation, it is needed to collect the following data:

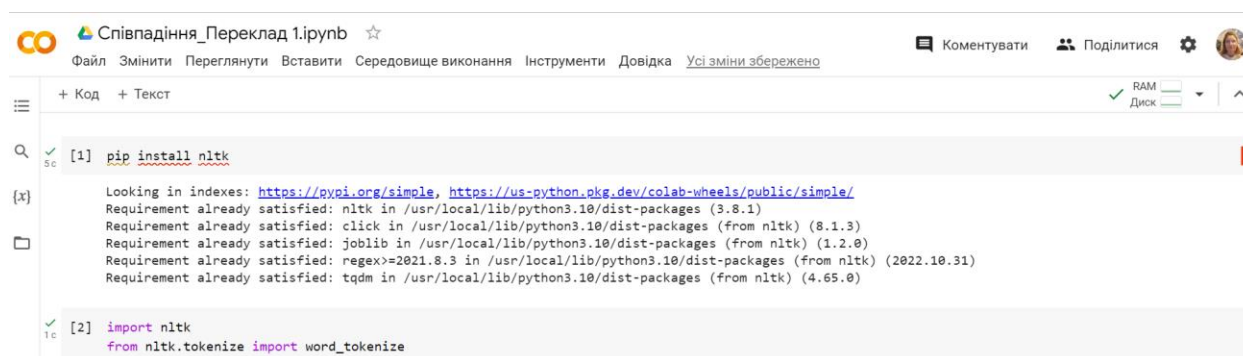
- 1) collect two sets of texts: a reference translation (the text that is considered correct) and a machine translation (we have 2 translations);
- 2) break each text into separate segments, for example, sentences or phrases (we broke it into token words);
- 3) calculate weights for BLEU metrics. Usually, the same weight is used for all segments;

- 4) using segments from each set of texts, count the number of word matches (n-grams) between the machine translation and the reference translation;
- 5) calculate the precision score for each n-gram, which reflects the correspondence between the machine translation and the reference translation;
- 6) calculate the geometric mean of the precision for all n-grams;
- 7) calculate the BLEU metric using the geometric mean of the precision and weighting factors.

Machine translation quality scores based on BLEU values are usually expressed as a percentage or on a scale from 0 to 1, where a higher value indicates better translation quality.

To evaluate the quality of machine translation using the BLEU metric of the sentences provided, you need to break them down into separate segments that will be used for comparison. In my case, these are token words. All the calculations are carried on the Colaboratory platform (Colab) [39].

The first step is to install the NLTK (Natural Language Toolkit) library (Fig. 2.1.), which can be installed using a Python package management tool such as pip (NLTK). Once installed, we can use NLTK to calculate the BLEU metric and other natural language tasks, such as tokenization. NLTK provides functions for breaking text into words, sentences, or other meaningful units. The NLTK library also provides datasets and resources for language processing, making it a valuable tool for researchers, developers, and data scientists working on projects involving textual data [45].



```
[1] pip install nltk

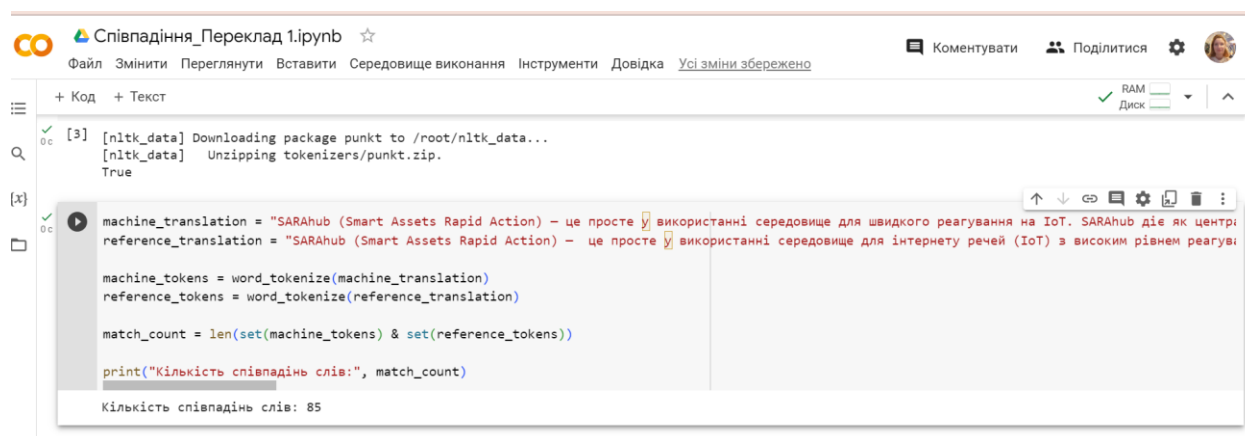
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: nltk in /usr/local/lib/python3.10/dist-packages (3.8.1)
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from nltk) (8.1.3)
Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages (from nltk) (1.2.0)
Requirement already satisfied: regex<=2021.8.3 in /usr/local/lib/python3.10/dist-packages (from nltk) (2022.10.31)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from nltk) (4.65.0)

[2] import nltk
    from nltk.tokenize import word_tokenize
```

Fig. 2.1. NLTK library installation

This figure displays installation of a tool `pip install nltk`. The `pip install nltk` command is used to install the Natural Language Toolkit (NLTK) library for Python. You can import and use NLTK in your Python scripts or interactive sessions to perform tasks such as tokenization, part-of-speech tagging, and more.

Next, download the necessary NLTK resources for word tokenization (Fig. 2.2.). Now you can calculate the number of word matches. Add the machine translation in the `machine_translation` variable and the reference translation in the `reference_translation` variable.



The screenshot shows a Jupyter Notebook titled "Співпадіння_Переклад 1.ipynb". The interface includes a top bar with navigation icons and a status bar showing "RAM" and "Диск". The notebook contains the following code cells:

```
[3] [nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data] Unzipping tokenizers/punkt.zip.
True
```

```
machine_translation = "SARAhub (Smart Assets Rapid Action) – це просте використання середовища для швидкого реагування на IoT. SARAhub діє як центри:
reference_translation = "SARAhub (Smart Assets Rapid Action) – це просте використання середовища для інтернету речей (IoT) з високим рівнем реагув:

machine_tokens = word_tokenize(machine_translation)
reference_tokens = word_tokenize(reference_translation)

match_count = len(set(machine_tokens) & set(reference_tokens))

print("Кількість співпадіннь слів:", match_count)
```

The output of the code is displayed below the code cell:

```
Кількість співпадіннь слів: 85
```

Fig. 2.2. Tokenization

This figure displays tokenization of reference translation, Translation 1 (Google Translate), and Translation 2 (DeepL). Tokenization is used to divide the text into tokens and count the number of word matches.

We saved the word tokens of all translations, as they will be needed for further processing. Examples of tokenization are shown in Appendix A.2 (there are sets of tokens for each translation: reference translation, Translation 1, and Translation 2).

Calculation of the number of word tokens are shown in Appendix B.2 (function `count_tokens` allows to count tokens in each translation from Appendix A.2). The results are as follows:

- 1) number of tokens in the referen. translation (length of the reference): 143 (Appendix B.2; Fig. B.2.1);
- 2) number of tokens in the Translation 1 (length of translation 1): 141 (Appendix B.2; Fig. B.2.2);

3) number of words in Translation 2 (length of translation 2): 142

(Appendix B.2; Fig. B.2.3).

Next, to calculate the BLEU metric, you need to count the number of n-gram matches between the machine translation 1 and 2, and the reference translation. To do this, we need to take the unigram (1-gram), bigram (2-gram), trigram (3-gram), and quadgram (4-gram). Let's calculate the number of matches for unigram ($n=1$), bigram ($n=2$), trigram ($n=3$), and quadgram ($n=4$).

To calculate the number of matches when using the BLEU metric with the NLTK library, we took the function `matching_count=count_matching_n-grams(reference_tokens, translation_tokens)` for each n-gram (Appendix C.2).

The weights parameter is set to (1, 0, 0, 0) to use only unigram (Appendix C.2; Fig. C.2.1, Fig. C.2.2):

- 1) translation 1 (Google Translate): 24 matches,
- 2) translation 2 (DeepL): 15 matches.

The weights parameter is set to (0, 1, 0, 0) to use only bigram (Appendix C.2; Fig. C.2.3, Fig. C.2.4):

- 1) translation 1 (Google Translate): 13 matches;
- 2) translation 2 (DeepL): 6 matches.

The weights parameter is set to (0, 0, 1, 0) to use only the trigram (Appendix C.2; Fig. C.2.5, Fig. C.2.6):

- 1) translation 1 (Google Translate): 12 matches;
- 2) translation 2 (DeepL): 5 matches.

The weights parameter is set to (0, 0, 0, 1) to use only quadgram (Appendix C.2; Fig. C.2.7, Fig. C.2.8):

- 1) translation 1 (Google Translate) 1: 11 matches;
- 2) translation 2 (DeepL): 4 matches.

The number of matches in the BLEU metric reflects the number of words that are matched between the translation and the reference translation (the benchmark). Precision or accuracy in the BLEU metric is defined as the number of matches compared to the reference.

Thus, precision reflects how accurately machine translation reproduces the grams compared to the reference translation. Next, the precision was calculated for each n-gram for Translation 1 (Fig. 2.3.):

- 1) unigram: 0.8936;
- 2) bigram: 0.6857;
- 3) trigram: 0.5755;
- 4) quadgram: 0.4855.



```
def calculate_precision(reference_tokens, translation_tokens, n):
    reference_ngrams = [tuple(reference_tokens[i:i+n]) for i in range(len(reference_tokens)-(n-1))]
    translation_ngrams = [tuple(translation_tokens[i:i+n]) for i in range(len(translation_tokens)-(n-1))]

    matching_count = sum(1 for ngram in translation_ngrams if ngram in reference_ngrams)

    precision = matching_count / len(translation_ngrams)

    return precision

reference_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "викори"]
translation_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "вико"]

precision_unigram = calculate_precision(reference_tokens, translation_tokens, 1)
print("Прецизія unigram:", precision_unigram)

precision_bigram = calculate_precision(reference_tokens, translation_tokens, 2)
print("Прецизія bigram:", precision_bigram)

precision_trigram = calculate_precision(reference_tokens, translation_tokens, 3)
print("Прецизія trigram:", precision_trigram)

precision_quadgram = calculate_precision(reference_tokens, translation_tokens, 4)
print("Прецизія quadgram:", precision_quadgram)
```

Прецизія unigram: 0.8936170212765957
 Прецизія bigram: 0.6857142857142857
 Прецизія trigram: 0.5755395683453237
 Прецизія quadgram: 0.4855072463768116

Fig. 2.3. Precision for each n-gram for Translation 1 (Google Translate)

This figure 2.3. reflects how accurately Translation 1 reproduces the n-grams compared to the reference translation.

Precision was calculated for each n-gram for Translation 2 (Fig. 2.4.):

- 1) unigram: 0.8921;
- 2) bigram: 0.7536;
- 3) trigram: 0.6642;
- 4) quadgram: 0.5882.

```

def calculate_precision(reference_tokens, translation_tokens, n):
    reference_ngrams = [tuple(reference_tokens[i:i+n]) for i in range(len(reference_tokens)-(n-1))]
    translation_ngrams = [tuple(translation_tokens[i:i+n]) for i in range(len(translation_tokens)-(n-1))]

    matching_count = sum(1 for ngram in translation_ngrams if ngram in reference_ngrams)

    precision = matching_count / len(translation_ngrams)

    return precision

reference_tokens = ["SARAhub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просто", "у", "використанні", "
translation_tokens = ["SARAhub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просто", "у", "використанні",

precision_unigram = calculate_precision(reference_tokens, translation_tokens, 1)
print("Прецизія unigram:", precision_unigram)

precision_bigram = calculate_precision(reference_tokens, translation_tokens, 2)
print("Прецизія bigram:", precision_bigram)

precision_trigram = calculate_precision(reference_tokens, translation_tokens, 3)
print("Прецизія trigram:", precision_trigram)

precision_quadgram = calculate_precision(reference_tokens, translation_tokens, 4)
print("Прецизія quadgram:", precision_quadgram)

```

Прецизія unigram: 0.8920863309352518
 Прецизія bigram: 0.7536231884057971
 Прецизія trigram: 0.6642335766423357
 Прецизія quadgram: 0.5882352941176471

Fig. 2.4. Precision for each n-gram for Translation 2 (DeepL)

This figure 2.4. shows how accurately Translation 2 reproduces the n-grams compared to the reference translation.

Next, let's calculate the geometric mean (Fig. 2.5.). Firstly, we need to take the product of the precisions:

- 1) product of precisions for Translation 1 (Google Translate):

$$0.8936 * 0.6857 * 0.5755 * 0.4855 = 0.1713;$$

- 2) product of precisions for Translation 2 (DeepL):

$$0.8921 * 0.7536 * 0.6642 * 0.5882 = 0.2627.$$

The calculation of the geometric mean of the precision for both machine translations is shown on the figure below (Fig. 2.5.):

- 1) translation 1 (Google Translate) is about 0.6432;
- 2) translation 2 (DeepL) is about 0.7159.

```
[ ] import math

translation1_precisions = [0.8936, 0.6857, 0.5755, 0.4855]
translation2_precisions = [0.8921, 0.7536, 0.6642, 0.5882]

def geometric_mean(precisions):
    product = 1.0
    num_precisions = len(precisions)
    for precision in precisions:
        product *= precision
    geometric_mean = product ** (1.0 / num_precisions)
    return geometric_mean

translation1_geometric_mean = geometric_mean(translation1_precisions)
print("Геометричне середнє прецизій для Перекладу 1:", translation1_geometric_mean)
translation2_geometric_mean = geometric_mean(translation2_precisions)
print("Геометричне середнє прецизій для Перекладу 2:", translation2_geometric_mean)
```

```
Геометричне середнє прецизій для Перекладу 1: 0.6432472940605846
Геометричне середнє прецизій для Перекладу 2: 0.7158871716586938
```

Fig. 2.5. Geometric mean of the precision for Translation 1 and Translation 2

This figure shows the code which calculates the geometric mean of precision scores for two different translations. It defines a function `geometric_mean` that takes a list of precision values, multiplies them together, and then calculates the geometric mean. The code then uses this function to find and print the geometric mean precision for Translation 1 and Translation 2, based on the given precision values for each translation.

2.2. Practical Outcomes

Finally, we can calculate the BLEU metric using geometric averages of precisions, products of precisions, and weighting factors.

In this case, since unigram, bigram, trigram, and quadgram are used, the weights will be:

- 1) unigram (1-gram): 1/4;
- 2) bigram (2-gram): 1/4;
- 3) trigram (3-gram): 1/4;
- 4) quadgram (4-gram): 1/4.

Calculating the BLEU metric using the NLTK library (Fig. 2.6). The `calculate_bleu()` function accepts the precision values for the reference translation, translation, and weight, as well as the lengths of the translation and reference translation. It calculates the BLEU metric, including the geometric mean of the precisions, the penalty for translation and weight shortness.

The results of the BLEU score and weighted BLEU score are displayed for each Translation (Fig. 2.6.). The parameters of machine translation according to the BLEU metric are as follows:

- 1) translation 1 (Google Translate) received a parameter of 0.6432;
- 2) translation 2 (DeepL) received a parameter of 0.7159.

The code in Fig. 2.6. calculates BLEU (Bilingual Evaluation Understudy) scores for two translations (Translation 1 and Translation 2) using a reference translation. It first computes precision scores by dividing each translation's precision by the reference precision. Then, it calculates the geometric mean of these precision scores and combines it with a brevity penalty to get the overall BLEU score. Additionally, it calculates a weighted version of the BLEU score using specified weights for individual precisions. Finally, it prints the BLEU and weighted BLEU scores for both translations.

The BLEU metric parameters for different translations, such as Translation 1 (Google Translate) and Translation 2 (DeepL) indicate the quality of these translations in comparison to the reference translation.

The BLEU metric ranges from 0 to 1, where a higher value indicates better translation quality. In our case, Translation 1 (Google Translate) has a value of 0.6432, and Translation 2 (DeepL) has a value of 0.7159. These values show that Translation 2 (DeepL) has a higher translation quality compared to Translation 1 (Google Translate). The value of 0.7159 is closer to 1, which is interpreted as a better translation using DeepL (Fig. 2.7.).

```

# Значення прецизій для Перекладу 1
translation1_precisions = [0.8936, 0.6857, 0.5755, 0.4855]

# Значення прецизій для Перекладу 2
translation2_precisions = [0.8921, 0.7536, 0.6642, 0.5882]

# Значення прецизій для референсного перекладу
reference_precisions = [1.0, 1.0, 1.0, 1.0]

# Довжина Перекладу 1
translation1_length = 100

# Довжина Перекладу 2
translation2_length = 105

# Довжина референсного перекладу
reference_length = 95

# Ваги
weights = [0.25, 0.25, 0.25, 0.25]

# Обчислення метрики BLUE для Перекладу 1
translation1_blue_score, translation1_weighted_blue_score = calculate_blue(reference_precisions, translation1_precisions, translation1_length,
print("BLUE-скор для Перекладу 1:", translation1_blue_score)
print("Зважений BLUE-скор для Перекладу 1:", translation1_weighted_blue_score)

# Обчислення метрики BLUE для Перекладу 2
translation2_blue_score, translation2_weighted_blue_score = calculate_blue(reference_precisions, translation2_precisions, translation2_length,
print("BLUE-скор для Перекладу 2:", translation2_blue_score)
print("Зважений BLUE-скор для Перекладу 2:", translation2_weighted_blue_score)

```

```

BLUE-скор для Перекладу 1: 0.6432472940605846
Зважений BLUE-скор для Перекладу 1: 0.6432472940605846
BLUE-скор для Перекладу 2: 0.7158871716586936
Зважений BLUE-скор для Перекладу 2: 0.7158871716586936

```

Fig. 2.6. Calculating the BLEU metric using the NLTK library

This figure displays the function `calculate_blue` for computing BLEU scores and weighted BLEU scores for two different translations (Translation 1 (Google Translate) and Translation 2 (DeepL)) based on precision values, lengths, and weights. The weighted BLEU score is influenced by user-defined weights on different n-gram precisions.

However, they are both still far from a perfect translation, which would have a value of 1. These parameters of the BLEU metric help to quantify translation quality and compare different machine translation systems or models. You can check the full code here: https://colab.research.google.com_BLEU_metric.

The BLEU scores obtained from the evaluation of translations performed by Google Translate and DeepL were compared to a reference translation performed by a human translator. The results showed that both machine translation models achieved fairly high BLEU scores, which indicates a significant level of relevance with the reference translation at the n-gram level (Fig. 2.8.).

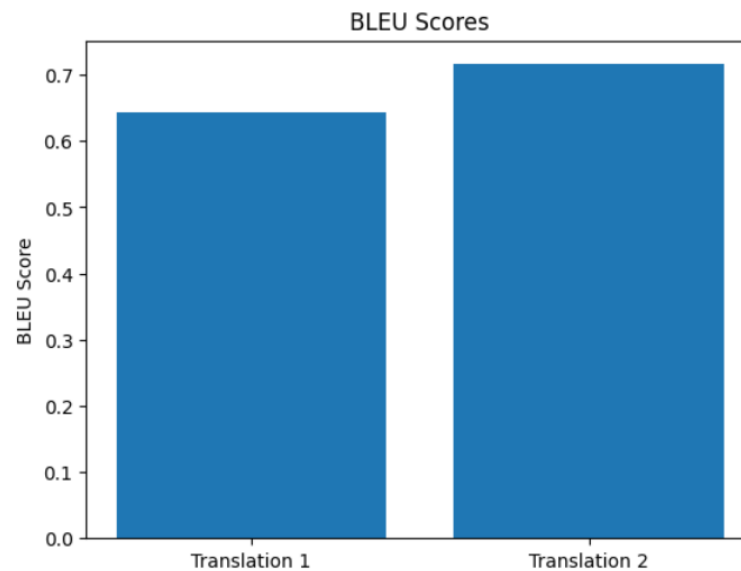


Fig. 2.7. Blue Scores

This figure is a bar chart. It illustrates the BLEU scores for Translation 1 (Google Translate) and Translation 2 (DeepL), with the x-axis denoting the translation types, and the y-axis representing the BLEU scores, providing a comparative view of the overall translation quality between the two.

The higher BLEU score (0.7159) for Translation 2 (DeepL) indicates that it has better translation quality than Translation 1 (Google Translate). The score of 0.7159 is closer to 1, which means that Translation 2 (DeepL) is closer to a good translation, but both translations still fall short of perfection, which is expressed in a BLEU score of 1.

It is important to understand that machine translation results are not perfect and may contain inaccuracies or linguistic inconsistencies. To eliminate these shortcomings and improve the quality of the translation, the finalization stage is very important. Translation editors, such as professional translators or native speakers, can help to correct and improve machine translation. Their experience and linguistic knowledge will help correct mistakes and ensure a more accurate translation.

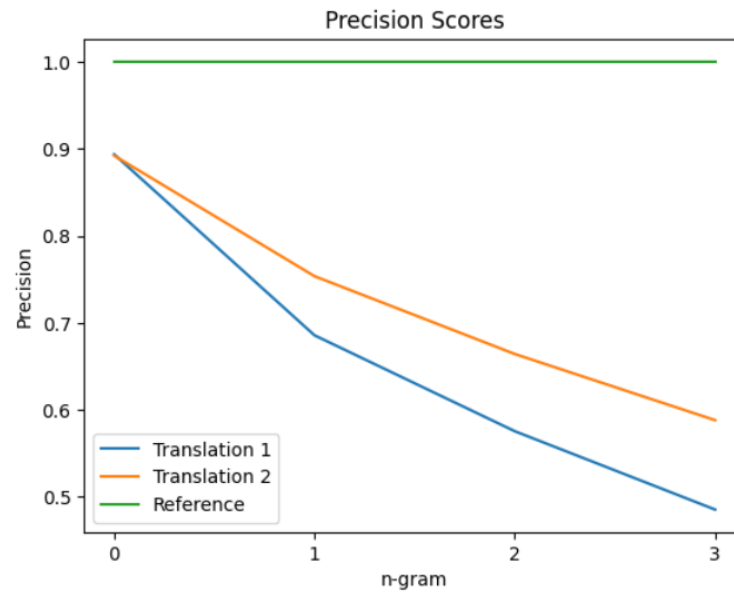


Fig. 2.8. Precision Scores

This figure shows precision scores for n-grams in Translation 1 (Google Translate), Translation 2 (DeepL), and the reference translation. The x-axis represents different n-grams, while the y-axis represents the precision scores ranging from 0 to 1. Each line on the graph corresponds to a different translation, showcasing how well each n-gram in the translations aligns with the reference.

In addition, machine translation post-editing (MTPE) is an effective approach to improving translation results. Manual editing and corrections can lead to a significant improvement in the overall quality and readability of the translation.

So, although machine translation models show promising results with relatively high BLEU scores, they still cannot match the quality of human translation. The BLEU metric is a valuable tool for quantifying and comparing different machine translation systems or models. By involving translators or editors and using post-editing techniques, we can achieve more accurate and contextually relevant translations, moving closer to the goal of creating a perfect translation with a BLEU score of 1.

However, while the BLEU metric provides a useful quantitative assessment, it is important to note that it has certain limitations. BLEU does not take into account such aspects as fluency, coherence, and semantic accuracy, which are crucial for creating a high-quality translation. Therefore, a high BLEU score does not guarantee that a translation is fully contextually relevant or linguistically natural.

To complement the BLEU score and get a more complete picture of the translation quality, it is recommended to use human evaluation. Experts can provide information on factors such as cultural relevance and overall comprehensibility that cannot be fully captured by automated metrics.

Human analysis of Translation 1 (Google Translate) and Translation 2 (DeepL) according to the reference translation showed, that both translations are generally correct and convey the main meaning of the reference sentence. The choice between them may depend on stylistic preferences or terminological nuances. Taking into account accuracy, structure and terminology, Translation 2 showed the best results, which also coincide with the BLEU metric. Here below we can consider and compare the translations.

Translations of sentence 1:

Reference translation: SARAhub (Smart Assets Rapid Action) — це просте у використанні середовище для інтернету речей (IoT) з високим рівнем реагування.

Translation 1: SARAhub (Smart Assets Rapid Action) — це просте у використанні середовище для швидкого реагування на IoT.

Translation 2: SARAhub (Smart Assets Rapid Action) – це просте у використанні середовище для високочутливого Інтернету речей.

Comparing both translation options with the reference translation, we can note the following:

1) translation 1 retains the main meaning and information by using the term "швидкого реагування", which reflects a high level of response. However, it is less precise and specific than the original;

2) translation 2 also conveys the general meaning, but uses the term "високочутливого Інтернету речей". This may be correct, but it is important to keep in mind that the term "highly sensitive" may have some ambiguity, as it can mean different aspects of sensitivity. There is also a lack of clarification about the level of response. Instead of a short dash, there should be a long dash, in accordance with Ukrainian punctuation rules [43].

Both translations reproduce the general meaning, but there are nuances that may be important depending on the context and specific translation requirements.

Translations of sentence 2:

Reference translation: "SARAhub діє як центральна нервова система для ваших ресурсів: отримує інформацію, обробляє її та при необхідності запускає дії у відповідь на важливі події."

Translation 1: "SARAhub діє як центральна нервова система для ваших активів — отримує інформацію, обробляє її та запускає дії, коли це необхідно, у відповідь на важливі події."

Translation 2: "SARAhub діє як центральна нервова система для ваших активів — отримує інформацію, обробляє її та запускає дії, коли це необхідно, у відповідь на важливі події."

Comparing both translations with the reference version, we can note the following: both translations include the main idea of the original, that SARAhub acts as a central nervous system for resources or assets, receives and processes information, and triggers actions in response to important events. Both translations contain corrections that make the phrase more natural. For example, "запускає дії, коли це необхідно" instead of "запускає дії при необхідності."

Translation 1 and Translation 2 use the term "активи" instead of "ресурси", but both terms may be acceptable depending on the context and specifics of the text.

In general, both translations faithfully convey the meaning of the original sentence, with some corrections to improve the clarity and naturalness of the expression.

Translations of sentence 3:

Reference translation: "Таке середовище дозволяє відображати датчики та пристрої на ієрархічних інформаційних панелях, створюючи цифрового двійника вашого середовища."

Translation 1: "Це дозволяє відображати датчики та пристрої на ієрархічних інформаційних панелях, створюючи цифрового двійника вашого середовища."

Translation 2: "Він дозволяє відображати датчики та пристрої на ієрархічних дашбордах, створюючи цифрових двійників вашого середовища."

Analyzing these translations:

Both translations convey the main idea of the ability to display sensors and devices on hierarchical dashboards to create a digital twin of the environment.

Translation 1 retains the structure of the original, but some words and phrases are reworded.

Translation 2 uses the term "дашборди" instead of "інформаційні панелі". This may reflect the more modern and technical terminology, but it is important to take into account the specifics and context of the text.

Both translations are logical and convey the main idea, although they use slightly different vocabulary.

Translations of sentences 4-5:

Reference translation: "Це дозволяє налаштувати автоматичне реагування на конкретні ситуації, консолідацію даних і багато інших функцій. Інтеграція та оперативність є ключовими перевагами SARAhub."

Translation 1: "Це дозволяє налаштувати автоматичні реакції на конкретні ситуації, консолідацію даних і багато інших функцій. Інтеграція та швидкість реагування є ключовими перевагами SARAhub."

Translation 2: "Це дозволяє налаштувати автоматичне реагування на конкретні ситуації, консолідацію даних та багато інших функцій. Інтеграція та оперативність - ключові переваги SARAhub."

Analyzing these translations: both translations convey the main idea of the original about the ability to set up automatic reactions, data consolidation, and other functions in SARAhub.

Translation 1 uses the term "швидкість реагування" instead of "оперативність," which can be understandable and accurate, especially in the context of a technical environment.

Translation 2 retains the term "оперативність" but uses dashes to emphasize key benefits. This can add emphasis and clarity to a statement. There should be a long dash in this case.

Both translations generally convey the meaning of the reference sentence correctly, but the choice between them may depend on which term and expression is considered more appropriate in a particular context or which terminology you prefer.

Translations of sentences 6-7:

Reference translation: "Інтеграція різних систем на вході та виході, а також різних програм в одне середовище. Немає необхідності розгортати різні системи для моніторингу навколишнього середовища, обслуговування обладнання або зняття показань і контролю енергоспоживання."

Translation 1: "Інтеграція як різних систем на вході, так і на виході та різних планів в одне середовище. Немає необхідності розгортати різноманітні системи моніторингу навколишнього середовища, обслуговування обладнання чи зчитування показань та контролю енергії."

Translation 2: "Інтеграція як різних систем на вході та виході, так і різних програм в одне середовище. Немає необхідності розгортати різні системи для моніторингу навколишнього середовища, обслуговування обладнання або зняття показань і контролю енергоспоживання."

Analyzing these translations: both translations convey the main idea of the reference sentence about the integration of different systems and programs into one environment, as well as the absence of the need to deploy different systems to perform different tasks.

In translation 1, instead of "різних систем на вході та виході" we used "різних систем на вході" which may slightly change the meaning. Also, "різноманітні" is used instead of "різні".

Translation 2 retains the structure of the reference sentence and the terminology is more similar to the original.

Translations of sentence 8:

Reference translation: "Все знаходиться в одному середовищі, в якому потім можна визначити реакцію на різні ситуації."

Translation 1: "Все в одному середовищі, в якому потім можна визначити реакції на різні ситуації."

Translation 2: "Все знаходиться в одному середовищі, в якому потім можна визначити реакцію на різні ситуації."

Both translations convey the basic idea of the reference sentence that all elements are in the same environment, in which reactions to different situations can then be determined.

Translation 1 uses the form "реакції" instead of "реакцію". This may be due to the fact that reactions can be a multilayered phenomenon.

Translation 2 retains the form "реакцію" as in the reference translation.

Both translations (1 and 2) are generally correct and convey the main meaning of the reference sentence. The choice between them may depend on stylistic preferences or terminological nuances. Taking into account accuracy, structure and terminology, Translation 2 showed the best results, which also coincide with the BLEU metric.

Most issues can be effectively resolved by using human expertise, such as machine translation post-editing (MTPE), and increasingly by customising in-house machine translation tool.

SECTION 3. RESEARCH RESULTS

3.1. Capabilities of Machine Translation

During the study we acquired skills of independent work and a bunch of knowledge such as the basics of working with Google Colab, skills of executing Python codes, using interactive notebooks Colab and GitHub, and calculating the BLEU metric.

Google Colab is a powerful and easy-to-use tool for developing and executing machine learning and natural language processing projects. One of the important advantages of Google Colab is its accessibility to many powerful libraries, including NLTK (Natural Language Toolkit). NLTK is one of the most popular libraries for working with natural language processing, and it is included in the standard installation of Google Colab. This means that you can easily install and use NLTK right in the Colab environment, performing all the necessary tasks related to text processing [19, **Ошибка! Источник ссылки не найден.**].

One of the interesting features is that Google Colab provides is the ability to evaluate the quality of machine translation using the Bilingual Evaluation Understudy (BLEU) metric. The BLEU metric is one of the most commonly used to evaluate the quality of machine translation, and it is often used to compare the results of different translation systems [4]4.

Thanks to the ability to run Python code directly in Google Colab, you can easily evaluate the quality of machine translation using the BLEU metric. Upload your translated texts and source data, perform the necessary calculations using NLTK functions, and get the BLEU metric value to evaluate the quality of your translation.

Overall, Google Colab is an extremely useful tool for developers, researchers, and students working with natural language processing and machine learning. In addition, the ability to evaluate the quality of machine translation using the BLEU metric makes Google Colab an indispensable tool for developing and improving text translation systems.

Google or DeepL. Both Google Translate and DeepL are powerful machine

translation systems with their own unique advantages and disadvantages. The best choice between the two will depend on your specific needs and use case.

Google Translate is a widely used machine translation system that supports a large number of languages and offers a range of features, including voice input, handwriting recognition, and image translation. It uses a combination of a rule-based machine translation approach and neural machine translation, and has been trained on a huge amount of data. Google Translate is a good choice for simple, everyday translations, and it's free to use [35, 17].

On the other hand, DeepL Translator is a newer machine translation system that uses convolutional neural networks, which tends to produce more accurate and natural translations. DeepL also offers support for fewer languages than Google Translate, but it does a better job with certain language pairs, such as English-German, English-French, and English-Spanish [35]. Additionally, DeepL offers a number of additional features, such as document translation and integration with computer-aided translation (CAT) tools. However, DeepL's premium features come at a price, and a subscription is required for professional or intensive use.

In general, if we need a free and widely available machine translation system for everyday use, Google Translate is a good choice. However, if we need a more accurate and natural translation or translation of professional documents, DeepL may be a better option, although a paid one.

Both versions of machine translation require revision by an editor, translator/linguist, i.e. machine translation post-editor (MTPE), who can make corrections and improve the machine translation. Their knowledge and expertise can help correct errors and ensure a more accurate translation. The Ukrainian IT company Ajax Systems held its annual presentation of its developments. The Special Event was localised in 24 languages. Previously, Ajax Systems used to hire specialists from translation agencies or foreign volunteers to translate the presentation and speakers' speeches. AI helped to reduce costs by 30-40% in 2023 compared to 2022. Over the past year, they saved \$100,000 on translations of various materials: presentations, documents, articles, etc. The key challenge in translation is understanding the context.

Ajax Systems has eliminated about 22 translators and editors from the agency and 10 dubbing actors. They have partially replaced translation agencies with artificial intelligence in their work on Special Event, but still have a human inside the company to check the AI. Ajax Systems uses DeepL to translate texts and PlayHT to dub speakers' speeches 51.

The development of neural translators in the field of translation of information technology technical terms is ongoing and has certain prospects for future development. The neural machine translators described in this work use a combination of different neural networks to produce more efficient and accurate results. Open sources indicate that convolutional neural networks (CNNs) are used in DeepL. They are traditionally used for image processing, but they are also used to process sequences, including text data. The development of models that effectively utilize CNNs for text processing can improve the translation of IT terms. Combining convolutional and recurrent neural networks in hybrid architectures can allow models to effectively account for both local and global dependencies in texts, which can lead to more accurate translations. It can also allow for both sequence and local dependencies in texts. For example, using RNNs to work with word sequences and CNNs to detect local features can improve translation quality [21, 15].

It is known from open sources that Google Translate uses a combination of Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), which are improved versions of RNNs that help avoid problems with gradient loss. The use of these architectures can improve the ability of models to understand and reproduce context in IT term translations [8].

Speaking of separate specialized models for specific tasks, it is advisable to develop specialized models for specific tasks in the field of IT term translation. For example, you can create separate models for translating code, technical documentation, and general texts. Increasing the amount of training data can improve the overall translation quality. In particular, expanding the corpus of texts with technical IT terminology can help to train the models better.

3.2. Prospects for the development of MT for IT industry

Future development prospects include continuous training of models on larger and more diverse data sets, expanding the vocabulary, and improving contextual understanding. To train a model, you can use available implementations of popular models or create your own based on machine learning libraries such as TensorFlow. In order to build your own translation model based on the TensorFlow machine learning library, you need to include several stages: data preparation, model building, training, and evaluation [Ошибка! Источник ссылки не найден.].

At the data preparation stage, it is necessary to collect and prepare parallel data for training, where each example should include text in the source language and its translation into the target language. Divide the data into training, validation, and test sets. Tokenization will help transform text data into a sequence. The model may include an encoder and a decoder, for example, as in the Transformer model (a deep neural network architecture introduced in 2017 by Google Brain developers) [Ошибка! Источник ссылки не найден.]. Since its inception, Transformers have increasingly become the model of choice for Natural Language Processing, NLP tasks, replacing RNN models such as LSTM. The additional parallelization of training makes it possible to train on larger datasets than was once possible. This has led to the development of pre-trained systems such as BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer) [23]. For training, it is necessary to define layers for input, output, and hidden layers of the model. Including an attention mechanism in the model can improve translation.

To train the model, it is appropriate to use the convenient API in TensorFlow. For example, `tf.keras.Model` and `tf.GradientTape`. Training is performed on a training set, a human translator performs a reference translation, and the translation quality is analyzed [37]. The quality of the model is assessed using the BLEU metric, which is the task of this paper.

For further use and/or integration of the model, you need to save the trained model to a file of another format (e.g., `SavedModel`) that can be used for translation.

To integrate the model into your program or web application, you can use TensorFlow Serving, TensorFlow Lite, or other tools [Ошибка! Источник ссылки не найден., 11].

It is also important to develop interactive tools that allow users to make corrections and additions to the translation so that models can learn from this data and constantly improve. For example, built-in error marking. The user can mark errors in the translation by selecting phrases or words and pointing to the correct version. This data can be collected and used to train the model. A tool such as the Selected Correction System can also be useful. When a user can modify a suggested translation by adding or replacing a suggested translation, the newly created translation is automatically included in the training set for the model. Using interactive tools, it is necessary to continuously analyze and visualize changes in the translation to identify patterns and understand which parts of the text were more often corrected. When implementing such tools, it is important to ensure the usability and reliability of the data entered by the user. It is also important to consider data privacy and security, as users may enter personal information into the model training process.

All in all, machine translation of technical terms in the IT industry using artificial intelligence has the potential for further development, and it is expected that even greater results in terms of relevance, accuracy, and translation speed will be achieved in the future.

SUMMARY

In the qualification work "Prospects for the development of machine translation of technical terms in the IT industry with the use of artificial intelligence" the functionality of the most common machine translators was successfully analysed and made proposals for the improvement and development of such translators, which significantly improves the quality of translation, speed up the process of translation which saves time and money for the developer, and, most importantly, provides the user with simple and clear information of the use of IT products.

The first chapter explored the subject area. Analysing the history and current state of machine translation, it becomes obvious that the development of machine translation, where machines learn on their own, is very relevant nowadays. The competition among developers of neural machine translators using artificial intelligence is growing, and the demand for such translations is also increasing.

The analysis of the selected machine translators was carried out using modern technologies and platforms, such as Google Colaboratory interactive notebooks, NLTK (Natural Language Toolkit) - a Python library for working with human language data, BLEU (Bilingual Evaluation Understudy) is a metric used to evaluate the quality of machine-generated translations, especially in the context of machine translation (MT). The use of these tools in combination helps to provide an objective and comprehensive analysis of the effectiveness of various machine translation systems.

The second section analyses the stages of evaluation using the BLEU metric and analyses the quality of human translation. We have also analysed the most vulnerable segment of translation that is not taken into account by machine translators - context. The results of the human translator analysis coincide with the results of the BLEU metric (although different indicators were taken into account for the analysis).

The last section describes the capabilities of each of the analysed machine translators (Google Translate and DeepL) currently available, as neural machine translators are developing and changing very rapidly. Using a combination of different

architectures and interactive built-in tools, machine translators learn and improve on their own. The BLEU translation quality metric was used to evaluate the translation quality. Although this is not a new metric, it is easy to use and still widely demanded for translation quality assessment.

As a result of the qualification work, we acquired key skills for assessing the quality of translation and analysed the work of various machine translators for translating technical terms of the IT industry, namely, the innovative product of OKIN Facility - IOTOR, IoT (Internet of Things). Based on the results of the work, it was proposed to use the machine translator DeepL as the best tool that meets the needs of the company and is suitable for translating technical terminology in the IT industry.

The use of the DeepL neural machine translation system in combination with the human post-editor improves the quality of the translation. The speed of translation using machine translation system can be increased up to 60% (the approximate speed of a human translator is 250 words per hour. Using machine translation, the speed can be increased to 400 words per hour, depending on the context and consistency of the source text), and the cost of translation can be reduced accordingly if productivity is calculated in words per hour. Recent survey of the Ukrainian company Ajax Systems showed, that they managed to save up to 40% on translators and dubbing actors for the annual Special Event. This is the case of using AI for translation. Though they did not refuse to use human editors and there was a person inside the company who checked AI performance.

The prospects for machine translation in the IT industry are promising, and tools such as DeepL are becoming key contributors to this purpose. The choice of this translator as the best machine translator underlines its effectiveness in meeting the needs of companies and translating technical IT terminology. Using such advanced machine translation system not only increases efficiency, but also provides significant time and cost savings in the translation process.

LIST OF REFERENCES

1. Білозерська Л.П. Термінологія та переклад. Навчальний посібник. Міністерство освіти і науки України. Вінниця: НОВА КНИГА, 2010, 232 с.
2. Ваховська О.В. Прикладні аспекти комп'ютерної лінгвістики. Навчально-методичний посібник. К.: Видавничий центр КНЛУ, 2023, 136 с.
3. Завадська В. Коли "вікно" не є вікном, або ще раз про сучасну українську ІТ-термінологію. Українське мовознавство, 43, 2013, стор. 20-27.
4. Aaron Smith, Christian Hardmeier et. al. Climbing mont BLEU: The strange world of reachable high-BLEU translations. *In Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, pages 269–281. URL: <https://aclanthology.org/W16-3414.pdf> (retrieved 13.01.2024).
5. Amanda Stent, Matthew Marge et al. Evaluating evaluation methods for generation in the presence of variation. *In Computational Linguistics and Intelligent Text Processing*, pages 341–351, Berlin, Heidelberg. URL: <https://link.springer.com/chapter/10.1007> (retrieved 13.01.2024).
6. Olesia Kostina, Artur Horbovyi. State-of-the-art machine translation systems using artificial neural network. *III International Scientific and Practical Internet-Conference*. Dubliany-Irpin, 2023, pages. 178-183. URL: <https://lnup.edu.ua/> (retrieved 13.01.2024).
7. Olesia Kostina, Artur Horbovyi. Assessing AI-driven Machine Translation Performance from English into Ukrainian using BLEU metrics for IoT product localization. *V International Scientific and Practical Conference «Scientific practice: modern and classical research methods»*. Collection of scientific papers, 2023, Boston, USA), pages 205-213. URL: <https://archive.logos-science.com/index.php/conference-proceedings/article/view/1434> (retrieved 13.01.2024).
8. Ahmed, S.F., Alam Sakib Bin et al. Deep learning modelling techniques: current progress, applications, advantages, and challenges. *Artif Intell Rev* 56, 2023, 13521–13617. URL: <https://link.springer.com/article/10.1007/s10462-023-10466-8> (retrieved 13.01.2024).

9. Bo Pang, Erik Nijkamp et al. Deep Learning With TensorFlow: A Review. *Journal of Educational and Behavioral Statistics*, 45(2), 2020, 227-248. URL: <https://doi.org/10.3102/1076998619872761> (retrieved 13.01.2024).
10. Carlos Hidalgo-Ternero. Google Translate vs. DeepL: analysing neural machine translation performance under the challenge of phraseological variation. *MonTI. Monografías de Traducción e Interpretación*, 154-177, 2020. URL: [DOI:10.6035/MonTI.2020.ne6.5](https://doi.org/10.6035/MonTI.2020.ne6.5) (retrieved 13.01.2024).
11. Christian Boitet, Hervé Blanchon et al. MT on and for the Web (PDF), 2010. URL: <https://web.archive.org/web/20170329125916> (retrieved 13.01.2024).
12. Cioffi Raffaele, Marta Travaglioni et al. Artificial Intelligence and Machine Learning Applications in Smart Production: Progress, Trends, and Directions. *Sustainability*, 2020. URL: <https://doi.org/10.3390/su12020492> (retrieved 13.01.2024)
13. Chris Olah, Shan Carter. Attention and Augmented Recurrent Neural Networks, p. 462. *Distill*, 2016. URL: <https://distill.pub/2016/augmented-rnns/> (retrieved 13.01.2024).
14. Chris Callison-Burch, Miles Osborne et al. Re-evaluating the role of Bleu in machine translation research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, 2006, Trento, Italy. URL: <https://aclanthology.org/E06-1032> (retrieved 13.01.2024).
15. Denny Britz, Anna Goldie et al. (2017). Massive Exploration of Neural Machine Translation Architectures, 2017. URL: <https://arxiv.org/pdf/1703.03906.pdf> (retrieved 13.01.2024).
16. Devin Coldewey, Frederic Lardinois et al. DeepL schools other online translators with clever machine learning. *TechCrunch*, 2017. URL: <https://techcrunch.com/2017/08/29/> (retrieved 13.01.2024).
17. DeepL For Academic Translations. *Textworks Translations*. URL: <https://www.textworks.eu/eng/deepl-for-academic-translations/>
18. Dzmitry Bahdanau, Kyunghyun Cho et al. Neural Machine Translation by Jointly Learning to Align and Translate, 2016. URL: <https://arxiv.org/abs/1409.0473> (retrieved 13.01.2024).

19. Edward Loper and Steven Bird. NLTK: The Natural Language Toolkit. *Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*. URL: <https://arxiv.org/pdf/cs/0205028.pdf> (retrieved 13.01.2024).
20. 51Forbes.ua, December, 2023. URL: <https://forbes.ua/innovations/> (retrieved 13.01.2024).
21. Graham Neubig. Neural Machine Translation and Sequence-to-sequence Models: A Tutorial, 2017, p. 209. URL: <https://arxiv.org/abs/1703.01619> (retrieved 13.01.2024).
22. Hutchins J. Machine Translation. *History. Encyclopedia of Language and Linguistics*, Oxford: Elsevie. P. 375–383, 2006. URL: https://www.infoamerica.org/documentos_pdf/bar05.pdf (retrieved 13.01.2024).
23. Jacob Devlin, Ming-Wei Chang et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2018. URL: <https://arxiv.org/pdf/1810.04805.pdf> (retrieved 13.01.2024).
24. Ilya Sutskever, Oriol Vinyals et al. Sequence to Sequence Learning with Neural Networks, 2014. URL: https://proceedings.neurips.cc/paper_files/paper/2014 (retrieved 13.01.2024).
25. Introducing tf-seq2seq: An Open Source Sequence-to-Sequence Framework in TensorFlow, 2017. URL: <https://blog.research.google/2017> (retrieved 07.01.2024).
26. Keiron O'Shea, Ryan Nash. An Introduction to Convolutional Neural Networks, 2015. URL: <https://arxiv.org/abs/1511.08458> (retrieved 13.01.2024).
27. Languages included in DeepL Pro. *DeepL.com*. URL: <https://web.archive.org/web> (retrieved 13.01.2024).
28. Luisa Bentivogli, Arianna Bisazza et al. Neural versus Phrase-Based Machine Translation Quality: a Case Study. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016. URL: <https://aclanthology.org/D16-1025/> (retrieved 13.01.2024)
29. Myroslav Konyk, Victoria Vysotska et al. Technology of Ukrainian-English Machine Translation Based on Recursive Neural Network as LSTM. *7th International*

Conference on Computational Linguistics and Intelligent Systems, 2023. URL: <https://ceur-ws.org/Vol-3387/paper27.pdf> (retrieved 13.01.2024).

30. M. Post. A call for clarity in reporting bleu scores. In *Proceedings of the Third Conference on Machine Translation*. Research Papers, Brussels, Belgium, October 2018. URL: <https://aclanthology.org/W18-6319.pdf> (retrieved 13.01.2024).

31. Nitika Mathur, Timothy Baldwin et al. Tangled up in BLEU: Reevaluating the Evaluation of Automatic Machine Translation Evaluation Metrics. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4984–4997, 2020. URL: <https://arxiv.org/pdf/2006.06264.pdf> (retrieved 13.01.2024).

32. Papineni K., Roukos S. et al. BLEU: a method for automatic evaluation of machine translation (PDF). *40th Annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002. URL: <https://aclanthology.org/P02-1040.pdf> (retrieved 13.01.2024).

33. Philipp Koehn. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, 2005. URL: <https://aclanthology.org/2005.mtsummit-papers.11/> (retrieved 13.01.2024).

34. Plenter JI. Advantages and pitfalls of machine translation for party research: the translation of party manifestos of European parties using DeepL. *Front. Polit. Sci.* Vol. 5, 2023. URL: <https://doi.org/10.3389/fpos.2023.1268320> (retrieved 13.01.2024).

35. Ruth Burbat, María-José Varela. Google Translate and DeepL: Breaking Taboos in Translator Training. *First International Conference on Research in Multilingualism: Innovation and New Challenges*, 2018. URL: <https://www.academia.edu/37078537/> (retrieved 13.01.2024).

36. Sachin Kumar, Antonios Anastasopoulos et al. Machine Translation into Low-resource Language Varieties. *The Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 2021. URL: <https://arxiv.org/pdf/2106.06797v2.pdf> (retrieved 13.01.2024).

37. 37Son J, Kim B. Translation Performance from the User's Perspective of Large Language Models and Neural Machine Translation Systems. *Information*, 14(10):574, 2023. URL: <https://doi.org/10.3390/info14100574> (retrieved 13.01.2024).
38. Thomas Wolf, Lysandre Debut et al. Transformers: State-of-the-Art Natural Language Processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. p. 38–45. URL: <https://aclanthology.org/2020.emnlp-demos.6/> (retrieved 13.01.2024).
39. Tiago Carneiro, Raul V. Medeiros Da Nóbrega et al. Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications, *IEEE Access*, 2018. URL: [DOI:10.1109/ACCESS.2018.2874767](https://doi.org/10.1109/ACCESS.2018.2874767) (retrieved 13.01.2024).
40. Yonghui Wu, Mike Schuster. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation, 2016. URL: <https://arxiv.org/abs/1609.08144> (retrieved 13.01.2024).
41. Vaswani A., Shazeer N. et al. Attention is All you Need. *Advances in Neural Information Processing Systems*, 2017. URL: https://proceedings.neurips.cc/paper_files/ (retrieved 13.01.2024).
42. Yulianto, A., & Supriatnaningsih, R. Google Translate vs. DeepL: A quantitative evaluation of close-language pair translation (French to English). *AJELP: Asian Journal of English Language and Pedagogy*, 9(2), pages 109-127, 2021. URL: <https://ojs.upsi.edu.my/index.php/AJELP/article/view/6087/3303> (retrieved 13.01.2024).
43. Чинний (новий) правопис української мови. URL: <https://slovnyk.ua/pravopys.php> (retrieved 13.01.2024).
44. Google Translate, Encyclopedia, Science News & Research Reviews URL: <https://academic-accelerator.com/encyclopedia/google-translate> (retrieved 13.01.2024).
45. NLTK. URL: <http://www.nltk.org/> (retrieved 13.01.2024).
46. IOTOR. URL: <https://www.iotor.io/> (retrieved 13.01.2024).
47. Notebook in Colaboratory URL: <https://colab.research.google.com> (retrieved 13.01.2024).

APPENDIXES

Appendix A.2

Reference translation:

["SARAhub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "it", "simple", "in", "use", "environment", "for", "Internet", "things", "(", "IoT", ")"), "with", "high", "level", "response", ".", "SARAhub", "acts", "as", "central", "nervous", "system", "for", "your", "resources", ":", "receives", "information", ",", "processes", "it", "and", "when", "necessary", "triggers", "actions", "in", "response", "to", "important", "events", ".", "Such", "environment", "allows", "display", "sensors", "and", "devices", "on", "hierarchical", "information", "panels", ",", "creating", "digital", "twin", "your", "environment", ".", "It", "allows", "customize", "automatic", "response", "to", "specific", "situations", ",", "consolidation", "data", "and", "many", "other", "functions", ".", "Integration", "and", "efficiency", "are", "key", "advantages", "SARAhub", ".", "Integration", "different", "systems", "on", "input", "and", "output", ",", "and", "also", "different", "programs", "in", "one", "environment", ".", "No", "need", "deploy", "different", "systems", "for", "monitoring", "environment", ",", "maintenance", "equipment", "or", "taking", "readings", "and", "control", "energy consumption", ".", "All", "is", "in", "the", "same", "environment", ",", "in", "which", "then", "can", "determine", "reaction", "to", "different", "situations", "."]

Translation 1 (Google Translate):

["SARAhub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "—", "це", "просто", "у", "використанні", "середовище", "для", "швидкого", "реагування", "на", "IoT", ".", "SARAhub", "діє", "як", "центральна", "нервова", "система", "для", "ваших", "активів", "—", "отримує", "інформацію", ",", "обробляє", "її", "та", "запускає", "дії", ",", "коли", "це", "необхідно", ",", "у", "відповідь", "на", "важливі", "події", ".", "Це", "дозволяє", "відображати", "датчики", "та", "пристрої", "на", "ієрархічних", "інформаційних", "панелях", ",", "створюючи", "цифрового", "двійника", "вашого", "середовища", ".", "Це", "дозволяє", "налаштувати", "автоматичні", "реакції", "на", "конкретні", "ситуації", ",", "

Continued Appendix A.2

"консолідацію", "даних", "і", "багато", "інших", "функцій", ".", "Інтеграція", "та", "швидкість", "реагування", "є", "ключовими", "перевагами", "SARAhub", ".", "Інтеграція", "як", "різних", "систем", "на", "вході", ",", "так", "і", "на", "виході", "та", "різних", "планів", "в", "одне", "середовище", ".", "Немає", "необхідності", "розгортати", "різноманітні", "системи", "моніторингу", "навколишнього", "середовища", ",", "обслуговування", "обладнання", "чи", "зчитування", "показань", "та", "контролю", "енергії", ".", "Все", "в", "одному", "середовищі", ",", "в", "якому", "потім", "можна", "визначити", "реакції", "на", "різні", "ситуації", "."]

Translation 2 (DeepL):

["SARAhub", "(", "Smart", "Assets", "Rapid", "Action", ")"], "-", "це", "просто", "у", "використанні", "середовище", "для", "високочутливого", "Інтернету", "речей", ".", "SARAhub", "діє", "як", "центральна", "нервова", "система", "для", "ваших", "активів", "-", "отримує", "інформацію", ",", "обробляє", "її", "та", "запускає", "дії", ",", "коли", "це", "необхідно", ",", "у", "відповідь", "на", "важливі", "події", ".", "Він", "дозволяє", "відображати", "датчики", "та", "пристрої", "на", "ієрархічних", "дашбордах", ",", "створюючи", "цифрових", "двійників", "вашого", "середовища", ".", "Це", "дозволяє", "налаштувати", "автоматичне", "реагування", "на", "конкретні", "ситуації", ",", "консолідацію", "даних", "та", "багато", "інших", "функцій", ".", "Інтеграція", "та", "оперативність", "-", "ключові", "переваги", "SARAhub", ".", "Інтеграція", "як", "різних", "систем", "на", "вході", "та", "виході", ",", "так", "і", "різних", "програм", "в", "одне", "середовище", ".", "Немає", "необхідності", "розгортати", "різні", "системи", "для", "моніторингу", "навколишнього", "середовища", ",", "обслуговування", "обладнання", "або", "зняття", "показань", "і", "контролю", "енергоспоживання", ".", "Все", "знаходиться", "в", "одному", "середовищі", ",", "в", "якому", "потім", "можна", "визначити", "реакцію", "на", "різні", "ситуації", "."]

Appendix B.2



```
def count_tokens(tokens):
    num_tokens = len(tokens)
    return num_tokens

translation_tokens = ["SARAhub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для"]

num_tokens = count_tokens(translation_tokens)
print("Кількість tokenів у перекладі:", num_tokens)
```

Кількість tokenів у перекладі: 143

Fig. B.2.1 Number tokens in reference translation



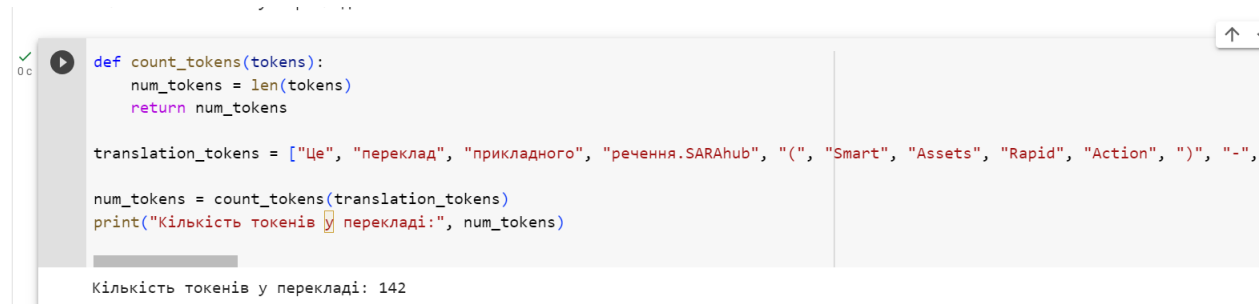
```
def count_tokens(tokens):
    num_tokens = len(tokens)
    return num_tokens

translation_tokens = ["SARAhub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для"]

num_tokens = count_tokens(translation_tokens)
print("Кількість tokenів у перекладі:", num_tokens)
```

Кількість tokenів у перекладі: 141

Fig. B.2.2 Number of tokens in Translation 1 (Google Translate)



```
def count_tokens(tokens):
    num_tokens = len(tokens)
    return num_tokens

translation_tokens = ["це", "переклад", "прикладного", "речення.SARAhub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для"]

num_tokens = count_tokens(translation_tokens)
print("Кількість tokenів у перекладі:", num_tokens)
```

Кількість tokenів у перекладі: 142

Fig. B.2.3 Number of tokens in Translation 2 (DeepL)

Appendix C.2

```
def count_matching_tokens(reference_tokens, translation_tokens):
    matching_count = 0
    for ref_token, trans_token in zip(reference_tokens, translation_tokens):
        if ref_token == trans_token:
            matching_count += 1
    return matching_count

reference_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для", "інтерн"]
translation_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для", "швид"]

matching_count = count_matching_tokens(reference_tokens, translation_tokens)
print("Кількість співпадінь:", matching_count)
```

Кількість співпадінь: 24

Fig. C.2.1 Unigram: Translation 1 (Google Translate)

```
def count_matching_tokens(reference_tokens, translation_tokens):
    matching_count = 0
    for ref_token, trans_token in zip(reference_tokens, translation_tokens):
        if ref_token == trans_token:
            matching_count += 1
    return matching_count

reference_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для", "інтерн"]
translation_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для", "висо"]

matching_count = count_matching_tokens(reference_tokens, translation_tokens)
print("Кількість співпадінь:", matching_count)
```

Кількість співпадінь: 15

Fig. C.2.2 Unigram: Translation 2 (DeepL)

```
def count_matching_bigrams(reference_tokens, translation_tokens):
    reference_bigrams = zip(reference_tokens, reference_tokens[1:])
    translation_bigrams = zip(translation_tokens, translation_tokens[1:])
    matching_count = sum(1 for bigram in translation_bigrams if bigram in reference_bigrams)
    return matching_count

reference_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для", "інтерн"]
translation_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для", "швид"]

matching_count = count_matching_bigrams(reference_tokens, translation_tokens)
print("Кількість співпадінь bigram:", matching_count)
```

Кількість співпадінь bigram: 13

Fig. C.2.3 Bigram: Translation 1 (Google Translate)

Continued Appendix C.2



```
def count_matching_bigrams(reference_tokens, translation_tokens):
    reference_bigrams = zip(reference_tokens, reference_tokens[1:])
    translation_bigrams = zip(translation_tokens, translation_tokens[1:])
    matching_count = sum(1 for bigram in translation_bigrams if bigram in reference_bigrams)
    return matching_count

reference_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просто", "у", "використанні", "середовище", "для", "інтерн"]
translation_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просто", "у", "використанні", "середовище", "для", "висо"]

matching_count = count_matching_bigrams(reference_tokens, translation_tokens)
print("Кількість співпадінь bigram:", matching_count)
```

Кількість співпадінь bigram: 6

Fig. C.2.4 Bigram: Translation 2 (DeepL)



```
def count_matching_trigrams(reference_tokens, translation_tokens):
    reference_trigrams = zip(reference_tokens, reference_tokens[1:], reference_tokens[2:])
    translation_trigrams = zip(translation_tokens, translation_tokens[1:], translation_tokens[2:])
    matching_count = sum(1 for trigram in translation_trigrams if trigram in reference_trigrams)
    return matching_count

reference_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просто", "у", "використанні", "середовище", "для", "інтерн"]
translation_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просто", "у", "використанні", "середовище", "для", "швид"]

matching_count = count_matching_trigrams(reference_tokens, translation_tokens)
print("Кількість співпадінь trigram:", matching_count)
```

Кількість співпадінь trigram: 12

Fig. C.2.5 Trigram: Translation 1 (Google Translate)



```
def count_matching_trigrams(reference_tokens, translation_tokens):
    reference_trigrams = zip(reference_tokens, reference_tokens[1:], reference_tokens[2:])
    translation_trigrams = zip(translation_tokens, translation_tokens[1:], translation_tokens[2:])
    matching_count = sum(1 for trigram in translation_trigrams if trigram in reference_trigrams)
    return matching_count

reference_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просто", "у", "використанні", "середовище", "для", "інтерн"]
translation_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просто", "у", "використанні", "середовище", "для", "висо"]

matching_count = count_matching_trigrams(reference_tokens, translation_tokens)
print("Кількість співпадінь trigram:", matching_count)
```

Кількість співпадінь trigram: 5

Fig. C.2.6 Trigram: Translation 2 (DeepL)



```
def count_matching_quadgrams(reference_tokens, translation_tokens):
    reference_quadgrams = zip(reference_tokens, reference_tokens[1:], reference_tokens[2:], reference_tokens[3:])
    translation_quadgrams = zip(translation_tokens, translation_tokens[1:], translation_tokens[2:], translation_tokens[3:])
    matching_count = sum(1 for quadgram in translation_quadgrams if quadgram in reference_quadgrams)
    return matching_count

reference_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просто", "у", "використанні", "середовище", "для", "інтерн"]
translation_tokens = ["SARAHub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просто", "у", "використанні", "середовище", "для", "швид"]

matching_count = count_matching_quadgrams(reference_tokens, translation_tokens)
print("Кількість співпадінь quadgram:", matching_count)
```

Кількість співпадінь quadgram: 11

Fig. C.2.7 Quadgram: Translation 1 (Google Translate)

Continued Appendix C.2

```

def count_matching_quadgrams(reference_tokens, translation_tokens):
    reference_quadgrams = zip(reference_tokens, reference_tokens[1:], reference_tokens[2:], reference_tokens[3:])
    translation_quadgrams = zip(translation_tokens, translation_tokens[1:], translation_tokens[2:], translation_tokens[3:])
    matching_count = sum(1 for quadgram in translation_quadgrams if quadgram in reference_quadgrams)
    return matching_count

reference_tokens = ["SARAhub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для", "інтерн
translation_tokens = ["SARAhub", "(", "Smart", "Assets", "Rapid", "Action", ")"), "-", "це", "просте", "у", "використанні", "середовище", "для", "висо
matching_count = count_matching_quadgrams(reference_tokens, translation_tokens)
print("Кількість співпадінь quadgram:", matching_count)

```

Кількість співпадінь quadgram: 4

Fig. C.2.8 Quadgram: Translation 2 (DeepL)

REVIEW OF THE SCIENTIFIC ADVISER

for the graduation qualification work

Student of higher education of the second (Master's) level of the Faculty of
Finance and Digital Technologies of the State Tax University

Kostina Olesia Volodymyrivna

на тему:

ПЕРСПЕКТИВА РОЗВИТКУ МАШИННОГО ПЕРЕКЛАДУ

ТЕХНІЧНИХ ТЕРМІНІВ ІТ ГАЛУЗІ ІЗ

ЗАСТОСУВАННЯМ ШТУЧНОГО ІНТЕЛЕКТУ

on the topic:

PROSPECTS FOR THE DEVELOPMENT OF MACHINE TRANSLATION OF TECHNICAL TERMS IN THE IT INDUSTRY WITH THE USE OF ARTIFICIAL INTELLIGENCE

The significance of the qualification work lies in the fact that it focuses on a relevant and contemporary topic in the IT industry - improving machine translation of technical terms using artificial intelligence. By looking at the work of machine translators, the study addresses an important part of translation technology that fits into the broader goal of increasing the competitiveness of the IT sector. The growing need for accurate and fast translations in the ever-changing IT industry shows why this research is relevant and important.

The aim of this project was to study machine translation technologies for technical terms in the IT industry using artificial intelligence, as well as to further evaluate the performance of the most common machine translation systems. The integration of technologies affects machine translation systems and business processes, showing how the development of translation IT technologies is linked to strategic goals.

The qualification work analysed the functionality of common machine translators in accordance with the tasks set. The research is intended to help improve the efficiency of translation processes, which will ultimately save developers time and resources. In addition, the work aims to provide users with clear and understandable information about the use of IT products, which is in line with the broader goal of improving users' understanding of the IT sector. Further research is needed on the possible issues and limitations that may arise when implementing machine translation systems, especially in diverse and complex technical contexts.

The graduation qualification work of the student of higher education of the second (Master's) level Kostina O. V. deserves the highest grade and Kostina O.V. deserves to be awarded the educational qualification level of Master in speciality 122 Computer Science.

Head of the graduation qualification work:

Horbovyy A. Yu., Doctor of Technical Science, Professor

"31" 07 2024 p.



**Conclusion of the Department
on the final qualification work**

The final qualification work of the student of higher education Kostina O.V. after the preliminary defence was reviewed at the meeting of the Department (protocol № 8 dd. "28" 12 2023 p.) and admitted to defence before the State Examination Commission.

Head of the department



(signature)

Filonenko M.M.

(surname, initials)

Head of educational and
professional programme



(signature)

Horbovyi A. Yu.

(surname, initials)

REVIEW

for the graduation qualification work

Student of higher education of the second (Master's) level of the Faculty of
Finance and Digital Technologies of the State Tax University

Kostina Olesia Volodymyrivna

on the topic:

PROSPECTS FOR THE DEVELOPMENT OF MACHINE TRANSLATION OF TECHNICAL TERMS IN THE IT INDUSTRY WITH THE USE OF ARTIFICIAL INTELLIGENCE

The relevance of the topic. The development of machine translation is relevant in today's context. The use of artificial intelligence in machine translation opens up significant prospects for improving translation efficiency and quality. The use of advanced technologies allows us to automate the translation process and adapt to the rapidly changing requirements of the IT industry.

Scientific and practical novelty. The use of tools such as Google Collaboratory, NLTK and the BLEU metric for comprehensive machine translation analysis contributes to the advancement of scientific knowledge. Choosing a machine translator DeepL underscores its effectiveness in meeting the needs of companies, increasing efficiency and significantly saving time and money on translation.

Quality of the problem analysis, text and graphic material. The analysis of machine translation systems allows us to identify the advantages and disadvantages of the most common machine translators. The use of modern tools demonstrates a comprehensive and systematic approach to evaluating machine translation systems. The inclusion of text and graphics effectively supports the analysis, making its content relevant.

The validity of proposals and conclusions and their practical value in the company's work. The results were published in the proceedings of Ukrainian conference Digital transformation of the financial system of Ukraine and V-4 countries in the context of European integration (April 26, 2023, Irpin) and International collection of scientific papers V International Scientific and Practical Conference Scientific practice: modern and classical research methods (December 22, 2023, Boston, USA). The results are proposed for implementation on the OKIN Facility company's website.

The availability of shortcomings. The potential challenges or limitations in the implementation of machine translation systems, especially in diverse and complex technical contexts, could be further explored, and there are some stylistic issues in the text that do not affect the informative quality of the work.

Overall conclusion and evaluation of the work. The graduation qualification work of the student of higher education of the second (Master's) level Kostina Olesia Volodymyrivna proves the high qualification of the author in solving the task, has scientific and practical value and deserves the highest grade.

Reviewer:



Prof. Yaroslav V. Koryukhin



CERTIFICATE OF IMPLEMENTATION

Issued to a student of higher education of the second (Master's) level Kostina Olesia Volodymyrivna on the use of the results of her master's qualification work on the topic: Prospects for the development of machine translation of technical terms in the IT industry with the use of artificial intelligence

The proposals made by the author in her work on the use of machine translation, namely the machine translator DeepL, have been put into practice on the basis of the research. The use of a machine translator will help to reduce translation time and costs by 40%. We propose to use the materials of the qualification work for the preparation of bachelor's and master's qualification works, as well as postgraduate students, in particular for the translation of technical terms in the IT industry.

Dean of the department



Korniciev V.V.
(surname, initials)

Head of the department


(signature)

Filonenko M.M.
(surname, initials)



Making facilities work better.



Вих. № Б/Н
Від 18.12.2023

CERTIFICATE OF IMPLEMENTATION

Issued to a student of higher education of the second (Master's) level Kostina Olesia Volodymyrivna of the Faculty of Finance and Digital Technologies of the State Tax University on the use of the results of her master's qualification work on the topic: Prospects for the development of machine translation of technical terms in the IT industry with the use of artificial intelligence.

The proposals made by the author in the qualification work on the use of the DeepL machine translator are currently relevant and appropriate. We started to use this machine translator for our work to translate information posted on the website. We also use and plan to continue applying the recommendations made by the author on the use of the machine translator in combination with a human editor.

This option improves our translation work by 40% compared to the previous period. We plan to apply this option not only for web-site translation but also for further work in the use of technical IT terminology, written communication, and writing texts on information technology projects.

Director



Olha Maletska

(first name, last name)

ТОВ ОКІН ФЕСІЛІТИ УА, 04119, Україна, м. Київ, вул. Юрія Іллєнка, 83-Д, офіс 702
+38 675 714 425 info.ua@okinfacility.com www.okinfacility.com
ІПН :352551926592 Код ЄДРПОУ: 35255193
Р/р №UA703005280000026005001367459 в АТ «ОТП Банк», МФО 300528