Features of Information System Development of Interactive Communication in Foreign Languages

Taras Basyuk[1], Andrii Vasyliuk[1], Olha Vlasenko[2]

[1]*Lviv Polytechnic National University, Bandera str.12, Lviv, 79013, Ukraine*
[2]*Osnabrück University, Institute of Education Science, Heger-Tor-Wall 9, Osnabrück, 49074, Germany*

Abstract

The article analyzes existing methods and known systems that provide means of communication in foreign languages. Technologies and software tools that implement interactive communication in foreign languages were analyzed, which made it possible to identify the main shortcomings of existing approaches and showed the relevance of the research. The use of a log-linear combination of functions with features and direct posterior probability and a translation model were described. The design of the software system was carried out using the object approach and displaying the created diagrams: options for use, activities, sequence. The synthesis of mathematical support was performed using the algebra of algorithms apparatus. The process of implementing the information system of interactive communication in foreign languages was described. A software tool has been created that works in prototype mode and implements the described functionality.

Keywords

Ukrainian language, digital education, translation, digital visualization, information system, interactive communication, data driven decision making

1. Introduction

The development of information technologies in society is mainly scientific and technical. An example of this is electronic communication [1, 2]. E-communication makes it possible to obtain the necessary information, helps with online communication, and it is also a powerful impetus in the transition to a new level of communication. Today, the global network is the most popular among society of various age categories and social groups, especially among young people, and when there is a need to communicate with foreign language users, many interlocutors face a language barrier. Learning a new language can take years, and the need to communicate information is immediate. Using the services of professional translators requires significant financial costs and very few translators are fluent in more than one or two languages, so there is a need to find a new translator which translates from and to different languages according to the needs [3].

The special influence of a lack of an international common language is felt in different organizations, where there is a need for fast daily translation of thousands of letters from different languages. Such organizations can cooperate with many foreign partners [4, 5]. Also, the problem arises acutely during personal communication on the network, if a regular user cannot use professional translation services. In such cases, it is advisable to use machine translation, which allows you to quickly translate between any two languages. Automatic translation can also be used to translate large amounts of information on the Internet for multilingual information. In general, modern online translators provide the possibility of translation in more than 100 languages (Google Translate).

Even an imperfect translation can be useful, for example, to translate Internet pages, news or make an appointment. It can also be used to obtain common information when there is no need for an exact translation. Also, such a translation can be useful for speeding up the text translation. You can translate the text and, in such cases, when the received information is not accurate, use machine translation.

In machine translation, three main approaches are used for text translation [6, 7]. Direct translation or word-for-word translation, where each word in one language is translated into another word in the target language using a translation dictionary between the two languages, followed by simple rules for permuting words in a sentence. A literal translation approach that takes the resulting information about the structure of the input sentence, the structure of the translation of the sentence, and then into the generation of the target language sentence. A third approach involves parsing sentence information to form an abstract representation, known as multilingual, before generating the original sentence.

The development of an interactive correspondence system in foreign languages is necessary to solve the problem of the language barrier. The system should provide high-quality and fast translation between many languages. Such a system helps users communicate on the Internet. This approach can be used both for official correspondence and for communication with friends. The algorithm of the system can also be applied in other fields, for example, for translating numerous documents and sending them between users, or for speech recognition and automatic translation. As the accuracy of machine translation improves, the possible field of application of the system increases.

1.1. Analysis of recent researches and publications

1.1.1. Analysis of known machine translation techniques

The conducted analysis showed that there are many methods and techniques of machine translation. In order to determine the optimal ones, we will conduct their research. In general, the following methods of machine translation can be distinguished [8-10]:

- translation based on a dictionary. This translation method takes into account sections of the language dictionary, where the translated text is formed using the corresponding word or equivalent words contained in the dictionary. This approach means that words will be explained as a dictionary does - literally, mostly without a specific relationship between them. Searching for a suitable word can be done by using analysis or lemmatization (related to the form and structure of words). Dictionary-based machine translation is effective for translating long lists of phrases or simple products and services. This method can also be used to speed up manual translation if the person doing it has knowledge of both dialects and can adjust the language structure and usage accordingly. This approach is still useful for translating phrases, albeit with some limitations at the sentence level. Most of the translation approaches developed later use bilingual lexicons with a syntactic approach;
- translation based on rules. This machine translation engine takes into account etymological data about dialects and sentence structure, covering the primary semantic, morphological and syntactic patterns of each dialect separately. During translation, the system, based on the rules, processes the sentence from the source (which may be in a certain dialect) in order to create a sentence in the objective dialect. This process involves taking into account morphological, syntactic and semantic aspects in both source and target dialects. The main methodology of such translation systems is to establish a connection between the structure of the information sentence and the structure of the requested sentence, which preserves their new meaning [11, 12];
- machine translation based on knowledge. This type of system is centered around the lexical concept, which represents a specific domain. For example, the KANT system serves as an example of a knowledge-based machine translation system designed for multilingual translation. It was established in 1989 with the aim of research and development of large and

practical technical documentation translation systems. To achieve high translation accuracy, KANT uses controlled vocabulary and grammar for each source language. It also has clear and targeted semantic models for each technical sector;

• corpus machine translation. Since 1989, corpus-based machine translation methodology has become one of the most actively studied areas in the field of machine translation and uses a variety of methods capable of providing high-accuracy in three-way translation. If only one address is required, all text will be centered around it. If there are two addresses, two centered tabs are used, and so on [13-15];

• statistical approach. A statistical approach to machine translation was first proposed by Warren Weaver in 1949. This method uses statistical techniques to generate a translated form using bilingual corpora. Statistical machine translation is based on actual translation models, the parameters of which are determined based on monolingual and bilingual corpora analysis. Building translation statistical models is a fast process, but the success of such an approach strongly depends on the availability of large multilingual corpora [16];

• a statistical approach to the word-based machine translation model. The basic unit of this methodology is the word. Algorithms aimed at the arrangement of words should achieve the most accurate translations of sentences. Compound words, expressions and homonyms add versatility to basic word-based translation. It is characterized by a statistical approach to phrase modeling. The crucial unit of this model is a phrase or grouping of words. In phrase-based approach, the main goal is to reduce the limitations of word-based translation by translating whole sequences of words where the length can vary. Such sequences of words are called phrases, however, they usually do not correspond to linguistic phrases, but are obtained using statistical methods from bilingual text corpora [17];

• machine translation based on an example. Example-based translation is built on the analysis or identification of situations that strongly resemble a dialectical pair. The idea of "translation by analogy" was first expressed by Makoto Nagao in 1981. Sentences are submitted in the source dialect from which the translation takes place, and translations of each sentence in the objective dialect are compared using a cartographic representation. These samples are used to decode the comparative sentence types of the source dialect with the target dialect. The basic idea is that if a previously translated sentence occurs again, the same translation can be recognized as correct again [18];

• neural machine translation. This is an innovative method of machine translation, where machines "learn" to translate through one large neural network consisting of many computing devices, studying the structure of the brain [19, 20];

• hybrid machine translation. It represents the integration of different approaches to machine translation with different options, resulting in high-quality translation at high speed. In general, the statistical approach becomes the most advanced, and the hybrid approach - the most qualitative among the available methods of machine translation. By combining the best aspects of different methodologies, a significant improvement in results can be achieved. The ideal is to get a grammatically correct translation, rule-based, lexical selection, a statistical approach to machine translation, and tolerance for unexpected structures. Such a hybrid approach becomes an interesting challenge and can help solve the problems encountered in existing machine translation methods. The main purpose of this hybrid approach is the advantage of using both linguistic rules and statistical methods. Hybrid Machine Translation reports on an effort to combine the best features of high-performance pure rule-based and corpus-based approaches [21].

1.1.2. Analysis of known systems of integrated correspondence

In today's world, there are many messengers, and each of them is distinguished by its unique characteristics and features. Well-known platforms such as WhatsApp, Telegram, Viber, Signal, and many others have become an integral part of our everyday life, providing us with a convenient means of communication and information exchange [22].

Each messenger offers its own unique set of features, which may include text messages, voice and video calls, sharing multimedia content, statuses, and other features. Some of them are distinguished by high confidentiality and encryption of messages, others by advanced possibilities for group or team communication.

The Viber system is widely distributed among users of mobile and web applications. It is a VoIP application for mobile phones on Android, iOS, Windows Phone, BlackBerry OS, Bada platforms and computers running such operating systems as Windows, OS X and Linux. The application automatically integrates with the address book and authorizes the user by phone number. Provides the ability to make free high-quality calls between smartphones using Viber, as well as send text messages, images, video and audio messages.

The next system is WhatsApp Messenger. It is a cross-platform text messaging app that provides messaging without SMS carrier charges. WhatsApp Messenger is available for mobile platforms such as iPhone, BlackBerry, Android, Windows Phone and Nokia. WhatsApp Messenger can be used under the same tariff plan that is charged for using the Internet, that is, additional fees for sending messages are not charged and users can always stay in touch. It is also possible to create group chats and send each other an unlimited number of images, videos and audio messages.

Facebook Messenger is a mobile application that allows you to communicate with your friends on Facebook. Notifications are sent to users' mobile devices. You can also send chat messages to users who are logged into their Facebook accounts. The application is available for users of Android phones, iPhones, iPads and BlackBerry devices and works on such platforms as IOS, Windows (Windows 10 and 11) and Android. Facebook Messenger is the official Facebook instant messaging product. Using Facebook Messenger, users can view their posts, comments and messages from their Facebook friends and will be notified of new messages. The advantages of this application are dynamic group session functions, integration with mobile devices and location mapping tools. Users also have the opportunity to exchange file content of various kinds. The function of transmitting video and voice messages in real time is available.

ROTR is the world's first full-fledged and independent VoIP service for simultaneous translation of voice, video calls and chats. Previously known as Droid Translator, the app has been enhanced to become a full VoIP service. Services, which are provided include video calls with translation, voice calls with translation, and communication with translation, where automatic translation is used. Currently, this service is available for Android and Apple devices. The name of the application has been changed, and it has acquired a shorter form, and now the application is called "DROTR". Using DROTR you can communicate in your native language with the world, using automatic translation of voice messages. For now Drotr is reborn under the new Vidby brand.

Google Hangouts is a text messaging and video chat system. This application was developed by Google and is popular among users of Android systems. The system is designed to replace Google products such as Google Talk, Google+ Messenger (formerly: Huddle), and Hangouts, the video chat system of today's Google+ that was used for messaging. Hangouts provides the ability to communicate between two or more users. All features are available online using the Gmail Google+ website or through mobile applications. The application is available for Android and IOS platforms. This application uses its own protocol, instead of the open standard XMPP protocol used in Google Talk. This has led to the fact that third-party applications that have access to and communicate with Google Talk do not have the same ability when using Google+ Hangouts.

Despite the wide selection and variety of functions offered by modern messengers, many of them still have their shortcomings, which emphasize the need to develop new and improve existing communication systems.

1.2. The main tasks of the research and their significance

The purpose of the research is to develop a prototype of a mobile system of interactive correspondence in foreign languages to simplify the communication process between foreigners. The main goal of this system is not to compete with professional translators, since machine

translation has not yet reached the appropriate level, but the system settings and selected means of implementation should provide the most accurate translation. A convenient and understandable user interface is also an important task that is solved. Such a system should require a minimum of effort from the user.

The results of the research solve the actual scientific and practical task of providing a tool that can be used both when communicating with a business partner and when group correspondence between friends, relatives or acquaintances. The result of the work is a system prototype that can replace already existing correspondence systems, especially taking into account its extended functionality. Many organizations already use static machine translation in their daily activities, for example the European Union (EU) translates all documents into 23 official languages. Organizations such as the EU have many connections, and therefore can use this system to communicate with representatives from different countries. The developed system will be relevant and can be used both for communication in a private circle and in the business sphere.

2. Major research results

The first step is to design the system [23]. It is the process of creating a detailed plan or model for the implementation of a system with certain functional requirements and characteristics. This phase of system development defines exactly how the system will be built, interact with users and other components, and how to provide the required functionality and performance.

While fulfilling the task, a number of diagrams were created, which are designed to facilitate the process of the created software product perception. The use case diagram is intended to represent the interaction of different actors with the proposed system, as well as how the actors interact with each other using this system [24, 25].
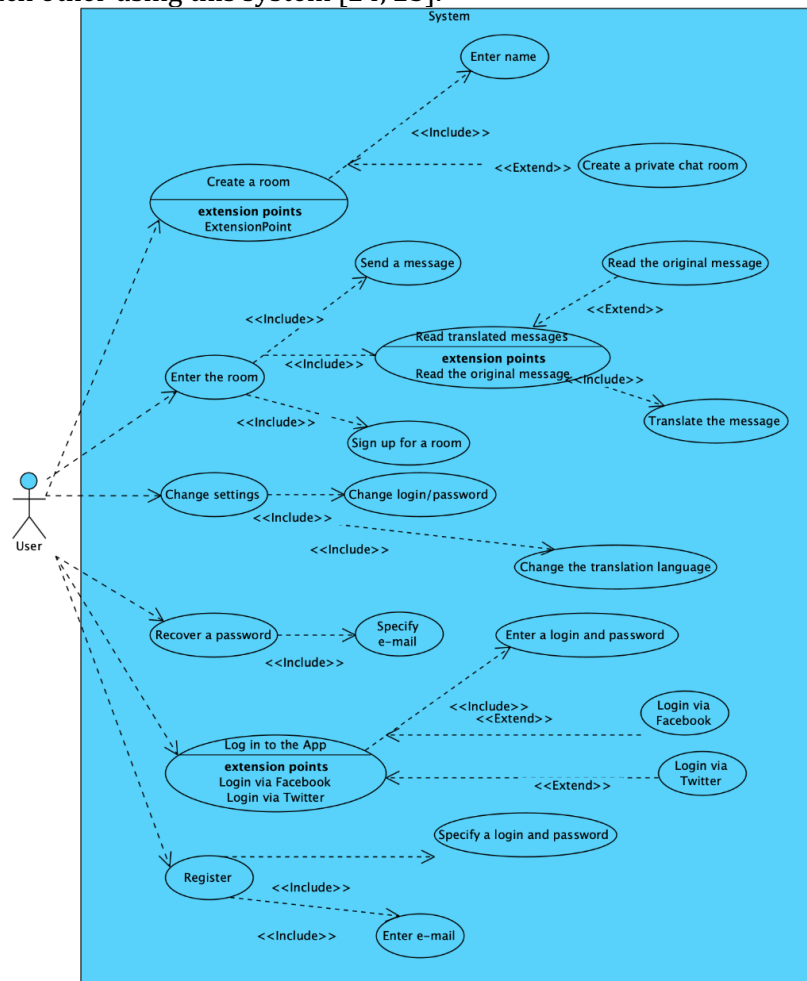

Figure 1: Use case diagram

Activity diagram is one of the types of diagrams in UML (Unified Modeling Language), which is used to model the sequence of actions, processes and behavior of a system or object [26]. Activity diagrams help visualize and analyze work processes, work flows, and interactions between different parts of a system.
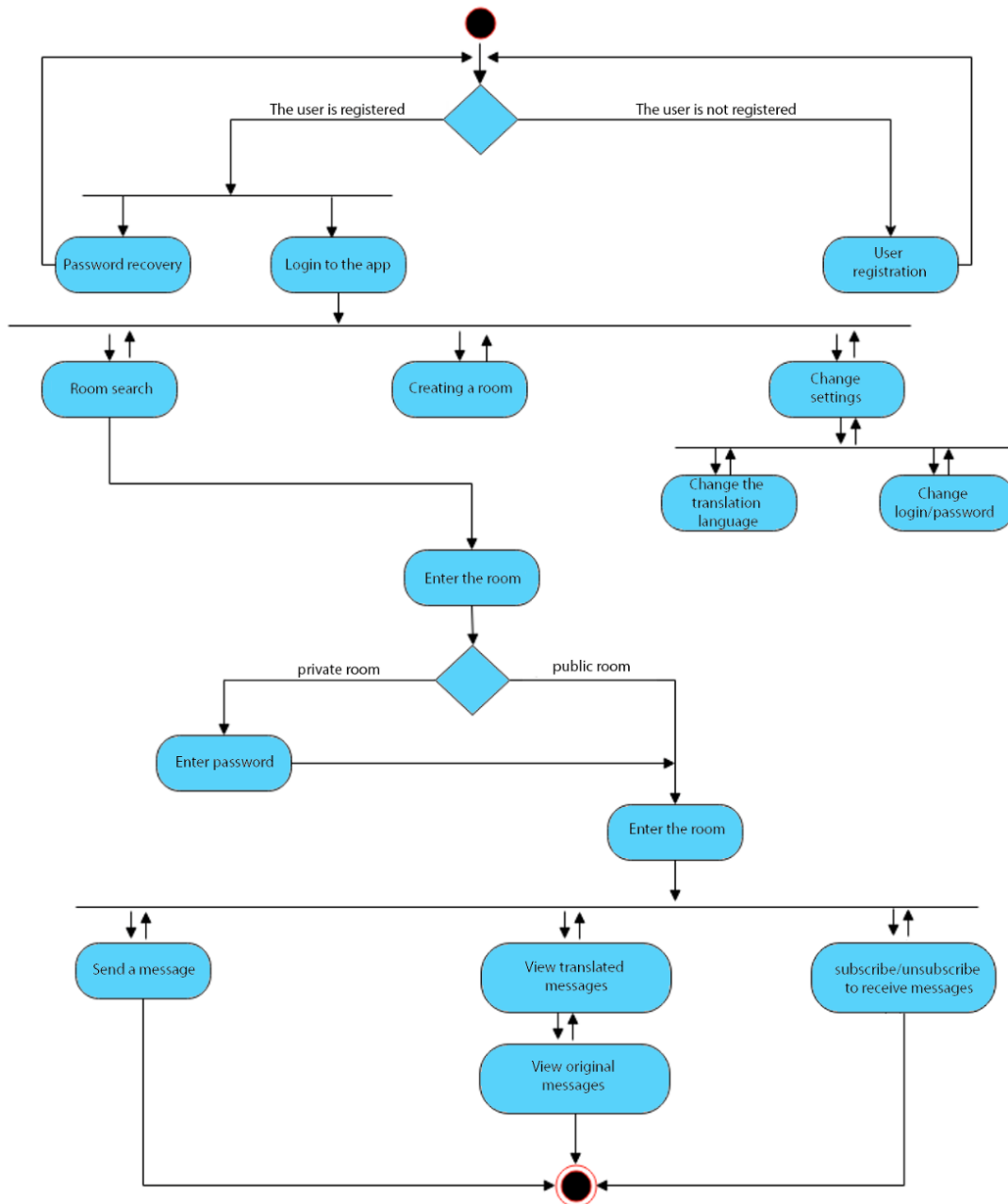

Figure 2: Activity chart

The developed diagram clearly shows how the system works and what its components are. In this diagram, step-by-step execution is performed with the possibility of branching and returning to previous stages.

A sequence diagram is one of the types of diagrams in the UML (Unified Modeling Language) modeling language [27]. This diagram is used to visualize the interaction of various objects or components in the system depending on the time order of events. It helps to determine the sequence of messages exchanged between objects during the execution of a specific scenario or functionality.

Sequence diagrams help clarify the order of operations, as well as identify parallel processes and interactions between objects in the system.
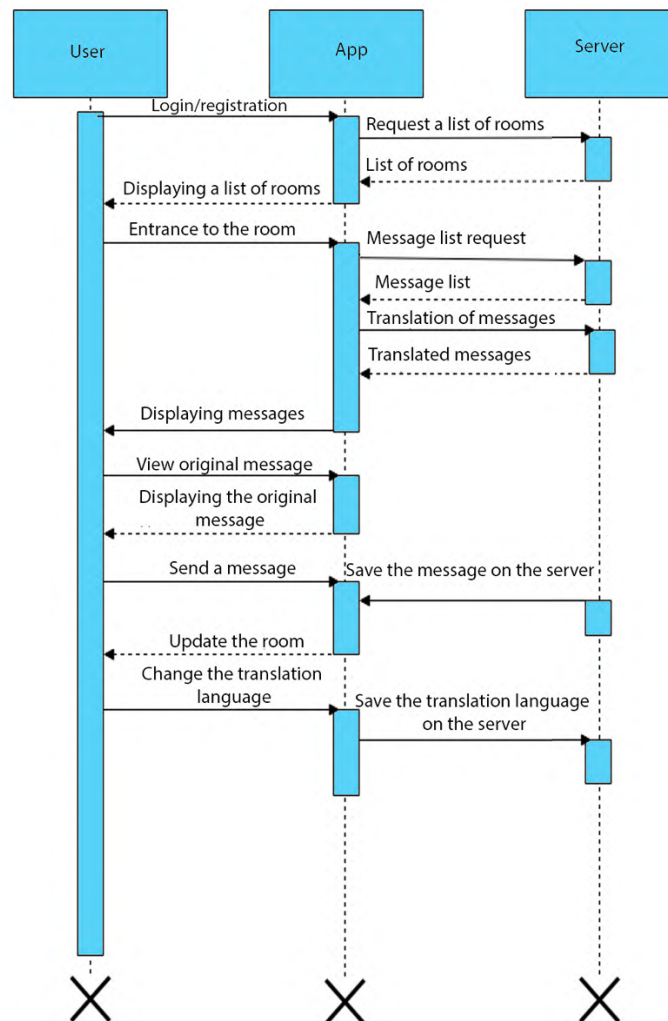

Figure 3: Sequence diagram

In this type of diagram, the interaction of the selected system objects at a certain point in time is demonstrated. Here the system individual objects order of actions is demonstrated. The objects in the sequence diagram form a specific order, which is determined by the degree of activity of these objects when interacting with each other.

The next stage was the selection of mathematical support for the given task implementation. In general, the task of machine translation is to translate an input sentence f in one language into an output sentence e in another language that must have the same meaning as f. This is done by building a static model that represents the translation process.

Brown P. presented a source-channel model where the language of the original sentence e is considered as generated by the source with a probability P(e) determined by the language model, and this is passed through the translation channel to generate sentences of the input language f, according to the translation probability P(f|e). The task of the translation system is to determine e from the received sentence f.

A large amount of data is required to accurately train these models, but they can be trained separately. Estimating the translation probability P(f|e) requires training using two languages in parallel. There should be a large volume of text in electronic parallel corpora that can be used for this purpose, and is growing rapidly, for example, Chinese news is available in many languages and all data is stored on electronic media. The language model P(e) is trained with a large monolingual corpus with the required language, as this allows the language to be represented as

accurately as possible. In general, the more data for training, the more accurate the speech representation will be.
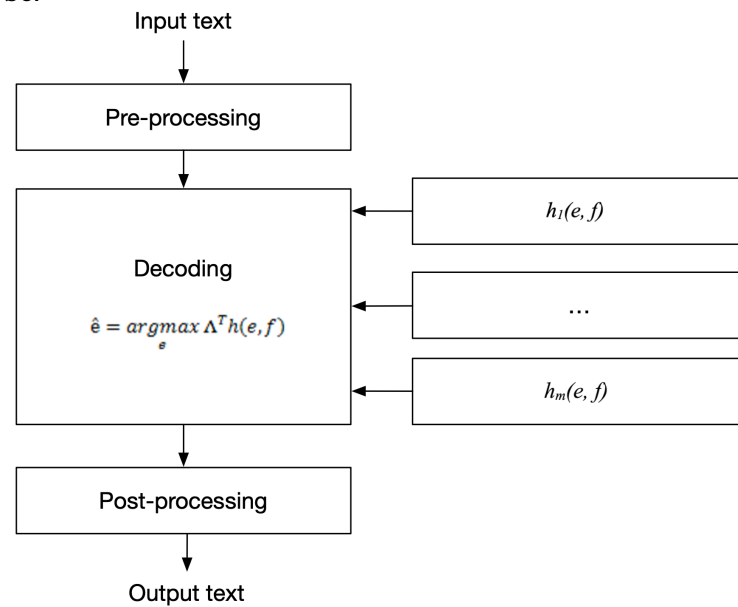
Input text

$$\downarrow$$

Pre-processing

$$\downarrow$$

Decoding $$\hat{e} = \underset{e}{argmax}\ \Lambda^T h(e,f)$$	←	$h_1(e,f)$
	←	...
	←	$h_m(e,f)$

$$\downarrow$$

Post-processing

$$\downarrow$$

Output text

Figure 4: Decoding process

Today, the most popular approach is to use a log-linear combination of functions with features and the immediate posterior probability *P(e|f)*. Let's define M functions with signs $h_m(e, f)$, *m = 1, ..., M*, then the probability of finding e from f can be calculated using the following formula:

$$P_\Lambda(e|f) = \frac{\exp(\sum_{m=1}^{M}\lambda_m h_m(e,f))}{\sum_{e`}\exp(\sum_{m=1}^{M}\lambda_m h_m(e`,f))} = \frac{\exp(\Lambda^T h(e,f))}{\sum_{e`}\exp(\Lambda^T h(e`,f))},$$

where each feature has a corresponding weight λ_m. $\Lambda = (\lambda_1 \ldots \lambda_M)$ *T* is used for the vector with feature weights and *h* for the feature vector. The process of finding the best translation (that is, the one that contains the highest probability) is known as decoding. Let's choose *a* *ê* that maximizes $P_\Lambda(e|f)$, for example:

$$A = \underset{e}{argmax}\ \Lambda^T h(e,f).$$

The decoding process is shown in Figure 4. And it is equivalent to the source-channel model if we set *M = 2, λ₁ = λ₂ = 1* and use the following functions:

$$h_1(e,f) = \log P(e),$$
$$h_2(e,f) = \log P(f|e).$$

The log-linear model was first used for machine translation by Dale Berger (1996), and then improved and popularized by Franz Josef Och and Hermann Ney (2002).

Assumption sentences are formed from the input sentence using the generative translation model, which depends on the specific input system used. Although we could theoretically make a guess from any sequence of words from the source language and let the model decide to find the best translation, we will only consider sequences that would be expected to translate based on what was given in the training data. Thus, we will learn the rules or get tables of phrases with the corresponding text. For a given input sentence of the assumption, we will consider only the output of the word that occurred in the parallel text as a translation of the input word.

A big problem in the decoding process is computational complexity. The search space in the translation process is a set of possible assumptions and can become very large for a number of reasons. It is possible that words in the source sentence do not have a direct translation with any words in the input sentence, so models must accommodate such situations. In the same way, the models must allow the extraction of the input word from the direct translation. Words and phrases in the input sentence can be translated differently; algorithms must be designed to find a better guess. Decoding usually uses dynamic programming algorithms and efficient search procedures, such as radial search or *A** search.

An important aspect is the development of the translation system functions. Functions that are complex enough to accurately model the translation process and provide useful information are required, but they also have to be efficiently computed and the algorithms used for decoding can be worked out. One feature common to all translation systems is the logging of the source language model $P(e)$. Often more than one feature language model is used, with a relatively simple language model used in the first pass of decoding and a more accurate, but computationally more expensive, language model used to re-evaluate the assumption to refine the result.

It is possible to define a feature that depends on any properties of the two sentences, and the features can be of different complexity. Signs from the generated models $P(e|f)$, $P(f|e)$ can be used in any translation direction. Other, simplified features used include sentence length distributions, distortion patterns, which are treated as permuted word and word order, features whose word pairs are found in common bilingual dictionaries, and lexical features whose word pairs are found in the training data. The word insertion penalty is a fee that is added for each word in a sentence and is effective in controlling output length and makes a big difference in translation quality. Once we have defined our model $P_\Lambda(e|f)$, we need to estimate the parameters from the training data. This is usually done in two steps. In the first step, parameters are estimated for the features that make up the model from a large set of training data. Most functions are evaluated on the basis of a corpus of parallel texts, although the model language and other functions for which $h(f, e) = h(e)$ are evaluated on the source language.

The second step is to estimate the weights for the features Λ of the log-linear model. Discriminant training is performed when designing a set that is less than parallel to the corpus, and typically translates the source text to minimize errors compared to the reference translation.

Models can be used to transform data of various types from news on the Internet to output with speech recognition systems running on television with each type having different properties required for its output. Recognition learning can be used to help with domain adaptation, such as giving more importance to the language pattern relevant to the desired domain and changing the penalty for insertions. The design of the kit is compared to the text of the model that will be used for translation, and the parameters are adjusted to optimize the system performance on this type of data.

In Automatic Speech Recognition (ASR), the transcription is evaluated according to the acoustic data O using the following formula (Dan Jurafsky and James H. Martin, 2000):

$$\hat{e} = \underset{e}{argmax} P(O|e)P(e).$$

This can be thought of as a source-channel model, with the sentence e as the source, modeled by the speech model $P(e)$, which passes through a noisy channel to produce O: the task is to determine e which provides only the information contained in O; we use sophisticated statistical models whose parameters are estimated from training data to ensure that they accurately reproduce the data.

The translation model $P(f|e)$ can be considered as an analogue of the acoustic model $P(O|e)$. In machine translation, however, there is an additional level of complexity that is not evident in ASR. While the words in the ASR transcription are found in the same order in which they were spoken and in the same order in which their representations appear in O. The word order in the input and output languages does not necessarily have to be the same. It is necessary to find an arrangement of words and as a result, to find their translation. Alignment is considered at three levels: the document (the translation of which input document corresponds to the output document); sentence (determines correspondence between sentences in the document); and word/phrase level. Dan (2005) addresses the problem of determining alignment at the sentence level with a related paper provided, but we will focus on determining alignment at the word and phrase level after the sentence pairs have been matched.

In the process of research, it will be used as the productive model of alignment and considered f as generated from e. For alignment, the source language is the language from which it is generated, and the target language is the language that is generated. Alignment models can be used in both directions of translation, that is, a sentence in the input language can be treated as having been created from a sentence in the source language and vice versa. Given a source

language sentence e and a target language sentence f, one can introduce a hidden alignment variable A that determines the correspondence between words and phrases in the two sentences, and calculates the translation probability as

$$P(f|e) = \sum_A P(f, A|e),$$

where the sum is determined from all possible alignment values of A.

The number of possible alignments between e and f is $2|e||f|$ and increases geometrically with increasing sentence length. This poses a problem for modeling due to the complexities of the models used. The field of alignment modeling is concerned with the development of techniques to overcome problems and produce translational correspondences between sentences.

An important part of the translation system is the modeling of the source language, as soon as it allows to use the directly translated assumption, which is a grammatical sentence. It is assumed that the sentence is created from left to right and that each word depends on the previous words, i.e. the probability of the sentence $e = e_1, e_2, ..., e_l = e_1^l$ is calculated by the formula:

$$P(e_1^l) = \prod_{i=1}^{l} P(e_i|e_1, e_2, ..., e_{i-1}).$$

For computational reasons, it is assumed that each word depends only on $n-1$ previous words; this is known as the N-gram language model. The probability of verdict is approximated by:

$$P(e_1^l) \approx \prod_{i=1}^{l} P(e_i|e_{i-n+1}, ..., e_{i-1}),$$

and N-gram probabilities $P(e_i|e_{i-n+1}^{i-1})$ are estimated from a large amount of monolingual data in the source language. The maximum likelihood estimate is:

$$P(e_i|e_{i-n+1}) = \frac{c(e_{i-n+1}^i)}{c(e_{i-n+1}^{i-1})},$$

where c is the number of sequences in the training data. These maximum likelihood estimators may suffer from data sparsity, i.e. they are inaccurate when there are few N-gram examples in the training data. They also assign zero probability to N-grams or words that do not occur in the training data, whereas we do not want to exclude a possibility from the data. Various smoothing methods are used to adjust the maximum likelihood to obtain a more accurate probability distribution. We can use probability interpolation where a lower order distribution is interpolated with a higher order distribution; delaying is the use of N-grams of lower-order probabilities when higher-order ones are available; discounting subtracts from non-zero counter values such that the probability mass can be distributed among N-grams in the order that the probability mass can be distributed among N-grams with zero counters, usually according to the lower partition order. General view of a large number of language models:

$$P_{smooth} = (e_i|e_{i-n+1}^{i-1}) = \begin{cases} \rho(e_i|e_{i-n+1}^{i-1}), & if\ c(e_{i-n+1}^i) > 0 \\ \gamma(e_i|e_{i-n+1}^{i-1})P_{smooth}(e_i|e_{i-n+2}^{i-1}), & otherwise \end{cases},$$

for some probability distributions ρ are defined by N-grams that occur in the training data, where $\gamma(e_i|e_{i-n+1}^{i-1})$ output weight and is normalized to ensure a valid probability distribution. All such models require significant computation to ensure that the probabilities are normalized.

The largest language models were built on 2 trillion (2×10^{12}) words of data and contain 200 billion different N-grams using «stupid backoff», a scheme that assigns points to each word but does not require normalization of probabilities. An assessment «stupid backoff» is:

$$S(e_i|e_{i-n+1}^{i-1}) = \begin{cases} \frac{c(e_{i-n+1}^i)}{c(e_{i-n+1}^{i-1})}, & if\ c(e_{i-n+1}^i) > 0 \\ a_n S(e_i|e_{i-n+2}^{i-1}), & otherwise \end{cases},$$

where a_n is the delay mass, which can depend on N-gram orders. The main advantage of this scheme is that the training is efficient according to the fact that the normalization of the probabilities is not performed. Despite its simplicity, it is competitive with more complex methods, especially when the amount of training data grows, and can be trained on large amounts of data where other models would be too complex.

Primary statistical translation models work only with correspondences between pairs of words in two languages, that is, translation units of individual words. However, some sequences of words are often translated as a whole, so it is preferable to consider sentences in terms of expressions, that is, sequences of words that follow one another. Phrase-based translation uses

just such an approach, allowing a group of adjacent words in one language to be translated as a continuous sequence of words in another language. It also allows the context of the word to exert its influence, and local variations in word order between languages can be studied. In addition, the phrases are extracted from the real text, so that the words can be grammatical inside the phrase and the system can provide a faster translation.

The next stage was the creation of a functioning model using the algebra of algorithms [28]. The first step is the synthesis of uniterms: $S(s)$ is the uniterm of loading the system and setting the initial settings; R – new user registration uniterm; L is the uniterm of entry into the system; C is a uniterm of a request to the server to receive a list of rooms; $L(c)$ is the uniterm of the entrance to the selected room; K is the uniterm for the start of the communication session; M is the uniterm for receiving a message in the original language of the interlocutor; $D(l)$ is a uniterm for determining the language and direction of automated translation; $I(m)$ is the uniterm of input of information about the direction of translation and data for translation to the input of the created model; $E(k)$ is the uniterm of the end of the communication session; u_1 – condition for checking the presence of a user account; u_2 – checking the availability of the received translation. As a result of the use of the apparatus of the algebra of algorithms, the following sequences and eliminations were synthesized:

S_1 – system operation sequence in the case of a user account and receiving a translation:

$$S_1 = \overparen{S(s), L}, C, L(c), S(k), M, E(k)$$

S_2 – sequence of system operation in the case of a user account and not receiving a translation:

$$S_2 = \overparen{S(s), L, C, L(c), S(k), M, D(l), I(m), E(k)}$$

S_3 – sequence of system operation in case of no user account and receiving a translation:

$$S_3 = \overparen{S(s), R, L, C, L(c), S(k), M, E(k)}$$

S_4 – sequence of system operation in case of no user account and no translation received:

$$S_4 = \overparen{S(s), R, L, C, L(c), S(k), M, D(l), I(m), E(k)}$$

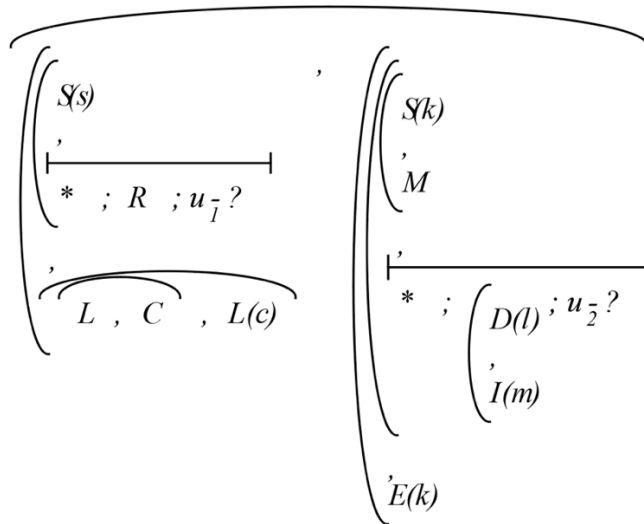The next stage is the synthesis of eliminations:

L_1 and L_2 – elimination of checking for the presence of a user account:

$$L_1 = \overline{S_1 \; ; \; S_3 \; ; \; u_1 - ?} \qquad L_2 = \overline{S_2 \; ; \; S_4 \; ; \; u_1 - ?}$$

L_3 and L_4 – elimination of the check for receiving a translation of a message:

$$L_3 = \overline{S_1 \; ; \; S_3 \; ; \; u_2 - ?} \qquad L_4 = \overline{S_2 \; ; \; S_4 \; ; \; u_2 - ?}$$

After substituting the corresponding sequences in the elimination operation and carrying out transformations, the following formula of the algebra of algorithms is obtained:

$$\left(\!\left(\begin{array}{c} S(s) \\ ,\\ \overline{*\quad;\ R\ ;u_{\bar{1}}?} \\ ,\\ L\ ,\ C\ ,\ L(c) \end{array}\right)\!\!,\ \left(\begin{array}{c} S(k) \\ ,\\ M \\ ,\\ \overline{*\ ;\left(\begin{array}{c} D(l) \\ ,\\ I(m) \end{array}\right);u_{\bar{2}}?} \\ ,\\ E(k) \end{array}\right)\!\right)$$

To develop an application for the Android mobile platform, use the Android Studio development environment, since Eclipse is no longer supported by developers, this platform is the only available option [29,30]. Android Studio is free software and is available for download at developer.android.com. This product includes a wide selection of high-end tools for code editing, debugging, rapid assembly and deployment of mobile applications. You can further expand the toolkit by additionally installing plugins.

As for the development of an application for the iOS mobile platform, it is advisable to use the XCode IDE, which is officially supplied by Apple. Built for software development for MacOS, iOS, WatchOS and tvOS. XCode supports programming languages such as C, C++, Objective-C, Objective-C++, Java, AppleScript, Python, Ruby, ResEdit (Rez), and Swift, including but not limited to Cocoa, Carbon. Thanks to the Mach-O format, you can compile universal binaries that can be executed on PowerPC and Intel-based platforms [31]. According to the selected methods and means of solving the problem, it is necessary to create a complete system of interactive correspondence in foreign languages.

The developed system is characterized by an intuitive interface and described functionality. When logging into the system from the mobile application, the user is prompted to select the functionality that needs to be obtained.
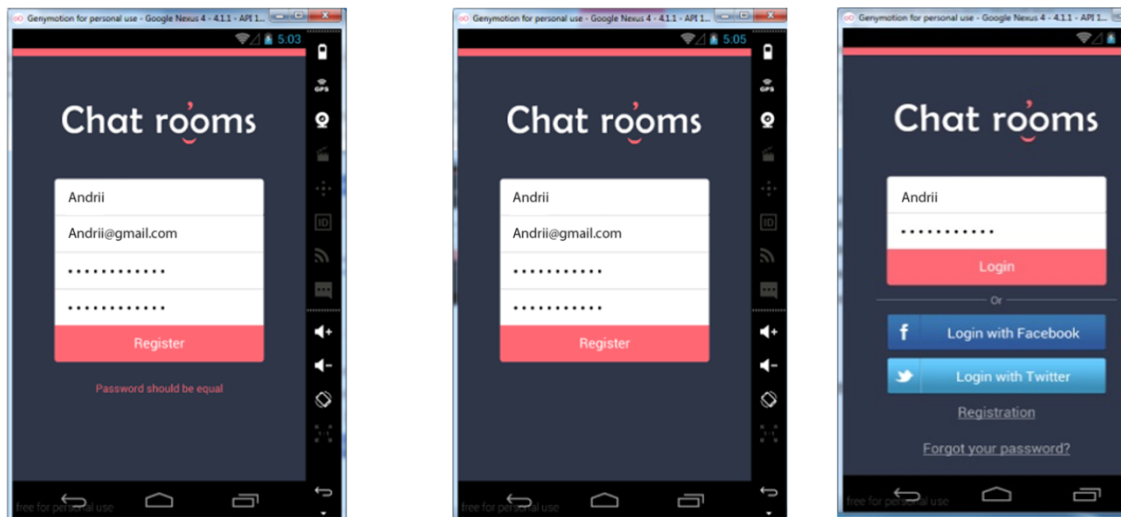

Figure 5: User registration and login

After logging in, a list of available rooms is displayed, allowing you to choose and start chatting. Also in Figure 6 you can see the use of such available functions as creating a new room or quick search.
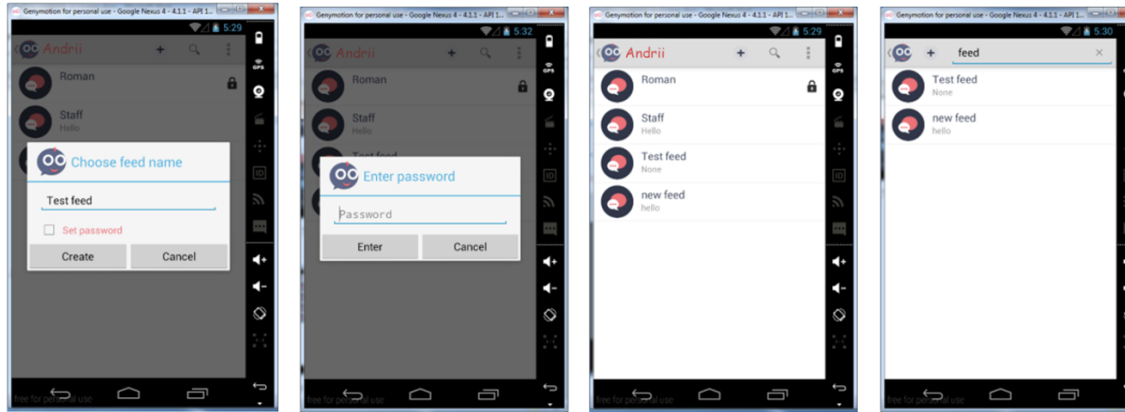
Figure 6: List of rooms

By opening the user profile, you can view or change the name, e-mail, password and translation language.

Before starting a conversation or viewing messages, you must select the desired room. After opening the room, all sent messages are displayed and you can start communicating in your native language, all messages will be automatically translated into the user's language. An example of communication between several users is shown in Figure 7.
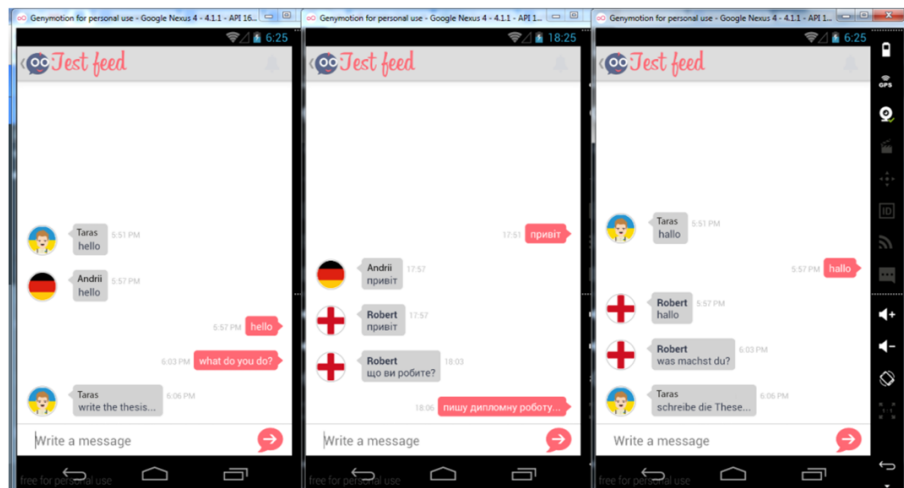

Figure 7: User communication in different languages

From the above example, you can see that the system is simple and intuitive to use, there is no need for additional skills in using the system, anyone can easily master it. The system interface is well-thought-out and easy to use, while using the system, the necessary hints for the user are displayed. Quick registration and login will not take much time, which allows you to start communicating with your foreign friends very quickly. And automatic language detection and translation of messages do not require any additional effort during communication.

Conclusion

As a result of the conducted research, the existing methods and known systems that provide means of correspondence in foreign languages were analyzed. Technologies and software tools for correspondence in foreign languages were analyzed as well, which made it possible to identify the peculiarities of existing approaches. As the analysis showed, today there are many software systems, but all of them are characterized by certain shortcomings, from the commercial prospect of the application to limited functionality, which makes the task of developing an information system for correspondence in foreign languages urgent. The next stage was the design of the

software system using the object approach and displaying created diagrams in accordance with the UML standard. The study presents diagrams of use cases, activities, sequences, and classes. An intelligent system of interactive communication in foreign languages has been developed in the form of a client-server application for mobile devices. During development, all system requirements were met. The application is a chat with extended functionality, a convenient user interface and the ability to automatically translate text messages into the user's chosen language in real time. Such a system makes it possible to communicate with foreigners without the language knowledge and without any additional effort.

Further research will be directed to testing and improving the system, eliminating conflicts and expanding functionality in accordance with the specified requirements.

References

[1] Serge Linckels, Yves Kreis, Robert A.P. Reuter, Carole Dording, Claude Weber, and Christoph Meinel. 2009. Teaching with information and communication technologies: preliminary results of a large scale survey. In Proceedings of ACM SIGUCCS fall conference: communication and collaboration (SIGUCCS '09). Association for Computing Machinery, New York, NY, USA, 157–162. https://doi.org/10.1145/1629501.1629530

[2] Yuling Huang, Chaoqun Leng, and Liping Zhan. 2022. The Application of Computer Electronic Information Technology in Engineering Management. In Proceedings of the 2021 5th International Conference on Electronic Information Technology and Computer Engineering (EITCE '21). Association for Computing Machinery, New York, NY, USA, 93–96. https://doi.org/10.1145/3501409.3501427

[3] Feng Han, Jianyu Cai, and Xiaoshuang Wang. 2022. The development of computer communication technology and its application in electronic information engineering. In 2021 10th International Conference on Internet Computing for Science and Engineering (ICICSE 2021). Association for Computing Machinery, New York, NY, USA, 56–59. https://doi.org/10.1145/3485314.3485333

[4] Yixuan Du. 2021. Interpersonal Meaning Analysis of Foreign Literature Communication and Modality Based on Language Processing Theory under Computer Network Technology Environment. In 2021 2nd International Conference on Computers, Information Processing and Advanced Education (CIPAE 2021). Association for Computing Machinery, New York, NY, USA, 1573–1576. https://doi.org/10.1145/3456887.3459723

[5] T. Basyuk, A. Vasyliuk, Peculiarities of an Information System Development for Studying Ukrainian Language and Carrying out an Emotional and Content Analysis // CEUR Workshop Proceedings. – 2023. – Vol. 3396: Computational Linguistics and Intelligent Systems 2023: Proceedings of the 7th International Conference on Computational Linguistics and Intelligent Systems. Volume II: Computational Linguistics Workshop, Kharkiv, Ukraine, April 20-21, 2023.pp. 279–294.

[6] Kehai Chen, Rui Wang, Masao Utiyama, and Eiichiro Sumita. 2022. Integrating Prior Translation Knowledge Into Neural Machine Translation. IEEE/ACM Trans. Audio, Speech and Lang. Proc. 30 (2022), 330–339. https://doi.org/10.1109/TASLP.2021.3138714

[7] Carlos Escolano, Marta R. Costa-jussà, and José A. R. Fonollosa. 2022. Multilingual Machine Translation: Deep Analysis of Language-Specific Encoder-Decoders. J. Artif. Int. Res. 73 (May 2022). https://doi.org/10.1613/jair.1.12699

[8] G. Tambouratzis, M. Vassiliou, S. Sofianopoulos, Machine Translation with Minimal Reliance on Parallel Resources (SpringerBriefs in Statistics), Springer, 2017

[9] Shumin Shi, Xing Wu, Rihai Su, and Heyan Huang. 2022. Low-resource Neural Machine Translation: Methods and Trends. ACM Trans. Asian Low-Resour. Lang. Inf. Process. 21, 5, Article 103 (September 2022), 22 pages. https://doi.org/10.1145/3524300

[10] "Machine Translation" URL. https://www.studysmarter.co.uk/explanations/english/linguistic-terms/machine-translation/

[11] Juan Diego Zamudio Padilla and Liuqin Wang. 2023. Binary Semantic Pattern Rules for Chinese-English Machine Translation Based on Machine Learning Algorithms. ACM Trans. Asian Low-Resour. Lang. Inf. Process. Just Accepted (October 2023). https://doi.org/10.1145/3626095

[12] T. Basyuk, A. Vasyliuk, V. Lytvyn, O. Vlasenko, Features of designing and implementing an information system for studying and determining the level of foreign language proficiency// CEUR Workshop Proceedings. – 2023. – Vol. 3312: Modern Machine Learning Technologies and Data Science Workshop (MoMLeT&DS 2022): Proceedings of the Modern Machine Learning Technologies and Data Science Workshop, Leiden, The Netherlands, November 25-26, 2022. pp. 212-225.

[13] Thangkhanhau Haulai and Jamal Hussain. 2023. Construction of Mizo: English Parallel Corpus for Machine Translation. ACM Trans. Asian Low-Resour. Lang. Inf. Process. 22, 8, Article 220 (August 2023), 12 pages. https://doi.org/10.1145/3610404

[14] Yan Gong and Li Cheng. 2023. Research on the Application of Translation Parallel Corpus in Interpretation Teaching. ACM Trans. Asian Low-Resour. Lang. Inf. Process. Just Accepted (September 2023). https://doi.org/10.1145/3623270

[15] Yan Gong. 2022. Study on Machine Translation Teaching Model Based on Translation Parallel Corpus and Exploitation for Multimedia Asian Information Processing. ACM Trans. Asian Low-Resour. Lang. Inf. Process. Just Accepted (November 2022). https://doi.org/10.1145/3523282

[16] Hung Phan and Ali Jannesari. 2020. Statistical machine translation outperforms neural machine translation in software engineering: why and how. In Proceedings of the 1st ACM SIGSOFT International Workshop on Representation Learning for Software Engineering and Program Languages (RL+SE&PL 2020). Association for Computing Machinery, New York, NY, USA, 3–12. https://doi.org/10.1145/3416506.3423576

[17] Hui Fang, Hongmei Shi, and Jiuzhou Zhang. 2021. Heuristic Bilingual Graph Corpus Network to Improve English Instruction Methodology Based on Statistical Translation Approach. ACM Trans. Asian Low-Resour. Lang. Inf. Process. 20, 3, Article 41 (May 2021), 14 pages. https://doi.org/10.1145/3406205

[18] T. Poibeau, Machine Translation (The MIT Press Essential Knowledge series), The MIT Press, 2017

[19] P. Koehn, Neural Machine Translation, Cambridge University Press , 2020

[20] A. Vasyliuk, T. Basyuk, V. Lytvyn, Design and Implementation of a Ukrainian-Language Educational Platform for Learning Programming Languages// CEUR Workshop Proceedings. – 2023. – Vol. 3426: Modern Machine Learning Technologies and Data Science Workshop (MoMLeT&DS 2023): Proceedings of the Modern Machine Learning Technologies and Data Science Workshop, Lviv, Ukraine, June 3, 2023. pp. 406–420.

[21] G. Joe, Hybrid Machine Translation for Low-Resource Languages, Self Publisher, 2023

[22] Seungchul Lee, Saumay Pushp, Chulhong Min, and Junehwa Song. 2018. Exploring Relationship-aware Dynamic Message Screening for Mobile Messengers. In Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers (UbiComp '18). Association for Computing Machinery, New York, NY, USA, 134–137. https://doi.org/10.1145/3267305.3267673

[23] C. Huyen, Designing Machine Learning Systems: An Iterative Process for Production-Ready Applications, O'Reilly Media, 2022

[24] S. Sundaramoorthy, UML Diagramming: A Case Study Approach, Auerbach Publications, 2022

[25] P. Mrzyglocki, UML Summarized: Key Concepts and Diagrams for Software Engineers, Architects and Designers, Independently published, 2023

[26] B. Shamile, Software Development with UML Diagrams, Independently published, 2022

[27] A. Dennis, B. Wixom, D. Tegarden, Systems Analysis and Design: An Object-Oriented Approach with UML, Wiley, 2020

[28] V. Ovsyak, ALGORITHMS: methods of construction, optimization, probability research. – Svit, 2001.

[29] N. Smyth, Android Studio Hedgehog Essentials - Kotlin Edition: Developing Android Apps Using Android Studio 2023.1.1 and Kotlin, Payload Media, 2023

[30] D. Griffiths, D. Griffiths, Head First Android Development: A Learner's Guide to Building Android Apps with Kotlin, O'Reilly Media, 2021

[31] N. Smyth, iOS 17 App Development Essentials: Developing iOS 17 Apps with Xcode 15, Swift, and SwiftUI, Payload Media, 2023