Information Technology for Foreign Languages Remote Learning with Adaptation to the User Based on Machine Learning

Taras Sopin [1], Victoria Vysotska [1,2], Oksana Markiv [1], Lyubomyr Chyrun[3], Vasyl Andrunyk[1], Sofia Chyrun[1] and Oleh Naum[4]

[1] *Lviv Polytechnic National University, S. Bandera Street, 12, Lviv, 79013, Ukraine*
[2] *Osnabrück University, Friedrich-Janssen-Str. 1, Osnabrück, 49076, Germany*
[3] *Ivan Franko National University of Lviv, University Street, 1, Lviv, 79000, Ukraine*
[4] *Ivan Franko Drohobych State Pedagogical University, I. Franko Street, 24, Drohobych, 82100, Ukraine*

Abstract

Since the goal of the work is to improve the process of remote learning of foreign languages, it was chosen to create an application for translation based on the choice of the native language and the language being studied. An educational process using a set of telecommunication technologies aimed at enabling students to learn the basic amount of information they need without direct contact between students and teachers during the learning process (which can take place both synchronously and asynchronously), and can be both an independent form of education, as well as a supplement to another more traditional form of education (full-time, part-time, extramural or externship), if necessary, giving a person the opportunity to study a foreign language training course. So, on the basis of this concept, a translation application was developed, which accurately translates both ordinary language and phraseological units, slang expressions, etc. The model is used as the basis of training, so let's analyze the model according to the main indicators. The model was pre-trained on BookCorpus, a dataset consisting of 11,038 unpublished books and the English Wikipedia (excluding lists, tables and titles). The texts are written in lowercase and tokenized using WordPiece and a dictionary size of 30,000. With probability 0.5, sentence A and sentence B match two consecutive sentences in the original corpus, and in other cases it is another random sentence in the corpus. Note that a sentence here is a continuous stretch of text, usually longer than one sentence. The only limitation is that the result with two "sentences" has a total length of less than 512 tokens. The masking procedure details for each sentence are as follows: 15% of tokens are masked; in 80% of cases masked tokens are replaced by [MASK]; 10% of the time, masked tokens are replaced by a random token from the one they replace; in the remaining 10% of cases, masked markers remain unchanged. The model was trained on 4 Cloud TPUs in a Pod configuration (16 TPUs in total) for one million steps with a batch size of 256. The sequence length was limited to 128 markers for 90% of the steps and 512 for the remaining 10%. Adam optimizer is used with learning rate: $\beta 1=0.9$, and $\beta 2=0.999$, weight decay 0.01, learning rate warm-up for 10,000 steps and learning rate linear decrease after. After training the network, the mean squared error decreased from 34.2 to 3.3. Also, training the network made it possible to reduce overtraining and improve its ability to generalize to new data. In the trained network, the number of layers and neurons was increased, which allowed it to reproduce more complex dependencies in the input data. Training the network made it possible to improve its results on test data, increase its ability to generalize, optimize its structure and parameters, choose a more effective activation function, and reduce the risk of overtraining.

Keywords

Foreign language training, machine learning, neuron network, neuron network training

MoMLeT+DS 2023: 5th International Workshop on Modern Machine Learning Technologies and Data Science, June 3, 2023, Lviv, Ukraine

EMAIL: taras.sopin.itisz.2019@lpnu.ua (T. Sopin); victoria.a.vysotska@lpnu.ua (V. Vysotska); oksana.o.markiv@lpnu.ua (O. Markiv); Lyubomyr.Chyrun@lnu.edu.ua (L. Chyrun); Vasyl.A.Andrunyk@lpnu.ua (V. Andrunyk); sofiia.chyrun.sa.2022@lpnu.ua (S. Chyrun); oleh.naum@gmail.com (O. Naum)

ORCID: 0000-0002-3114-4221 (T. Sopin); 0000-0001-6417-3689 (V. Vysotska); 0000-0002- 1691-1357 (O. Markiv); 0000-0002-9448-1751 (L. Chyrun); 0000-0003-0697-7384 (V. Andrunyk); 0000-0002-2829-0164 (S. Chyrun); 0000-0001-8700-6998 (O. Naum)

CEUR Workshop Proceedings (CEUR-WS.org)

1. Introduction

Nowadays, there is a certain class of topical tasks, the solution of which is impossible or difficult to implement without the use of artificial neural networks (ANNs) [1]. To solve such problems, human intelligence is ineffective, and traditional calculations are time-consuming or physically inadequate, because they do not reflect or poorly reflect real physical processes and objects [2]. Accordingly, it becomes necessary to use artificial neural networks to solve classification tasks. A distinctive feature of neural networks is that they are not programmed, they do not use any inference rules to make a diagnosis [3], but they learn to do this from examples diagnosis is a special case of event classification, and the greatest value is the classification of those events that are not in the training neural network set data. The use of remote techniques for foreign languages is widely used in modern institutions. Distance learning of foreign languages involves the use of modern information technologies [4]. This determined the purpose of the study, which is to improve the process of remote learning of foreign languages, with adaptation to the user's language based on machine learning by developing and implementing an application that will be used to classify operations using neural networks [5].

To achieve the goal, the following tasks were set:

1. Conduct an analysis of the literature on the use of neural network technology.
2. Analyze and describe the necessary tools for creating classification software based on neural networks.
3. Build a classification model based on neural networks.
4. Analyze materials for working with programming languages.
5. Build the software architecture.
6. Develop software.
7. Test software.
8. Analyse work results.

The object of research concerns processes for remote learning of foreign languages with adaptation to the user based on machine learning. The subject of research comprises methods and tools for developing a software product for teaching foreign languages based on neural networks. The methodological basis of the research is general scientific and special methods, which made it possible to study the subject and the object of research, to explore directions and ways of optimizing the process of remote learning of foreign languages with adaptation to the user based on machine learning. The practical significance of the obtained results is that the use of the developed software will allow to improve translation from any selected pairs of languages, which will serve as a basis for remote training of users.

2. Related works

Intelligent adaptive learning systems are emerging rapidly, but are still in the experimental stage [1-6]. The intended design of these data-responsive solutions is aimed at providing differentiated learning at a personalized level of learning [7-8]. New approaches to the development of diagnostic and formative assessment using adaptive intelligence are becoming more common [9-12]. Adaptive learning systems are designed to dynamically adjust to the level or type of course content based on an individual learner's ability or skill achievement in a way that accelerates learner performance through both automated and instructor-assisted intervention [13-18]. Adaptive systems achieve this by helping to address learning challenges such as different student learning abilities, different student backgrounds, and resource constraints. The goal of these machine learning systems is to leverage skills and determine what a student actually knows, and move students along a consistent learning path toward established learning outcomes and skill mastery in a precise and logical manner. Nowadays, many platforms use adaptive systems, in order to better understand how they differ, let's compare the 3 most popular resources used by students from all over the world: Eduflow, eloomi, ISpringLearn.

Adaptive platforms based on machine learning are the most advanced scientific method for establishing a truly adaptive state. Machine learning (ML) [2] is synonymous with pattern recognition, statistical modeling, predictive analytics, statistical regularities, and other forms of advanced adaptive capabilities. MN-based systems use programmed algorithms to create an adaptive scientific core and predict in real time about the student's mastery of the subject. Adaptive MN-based platforms use learning algorithms, also known as "learners," to create other algorithms that in turn create adaptive sequences and predictive analytics that can continuously collect data and use it to move the learner along a guided learning path.

What is unique about adaptive systems based on machine learning is their ability to determine how an individual learns and approaches a learning task within these intelligent systems, as well as provide accurate and timely feedback and improve student performance. Since MN-based systems are computationally intensive and analyze billions of bits of data in real time, system scalability can be an issue from two perspectives: how these systems are efficiently coded [7]; and the provisioning architecture used to process, load, and balance massive amounts of data.

In order to be able to guide the user through the learning and assessment process [18-21], information about the user and his/her actions must be collected and recorded in a user profile. The user profile will have to record both static and more dynamic information. The user profile as a cornerstone component of the proposed e-learning system is well studied and documented in the development process. Fig. 1 shows a schematic overview of the proposed user profile.

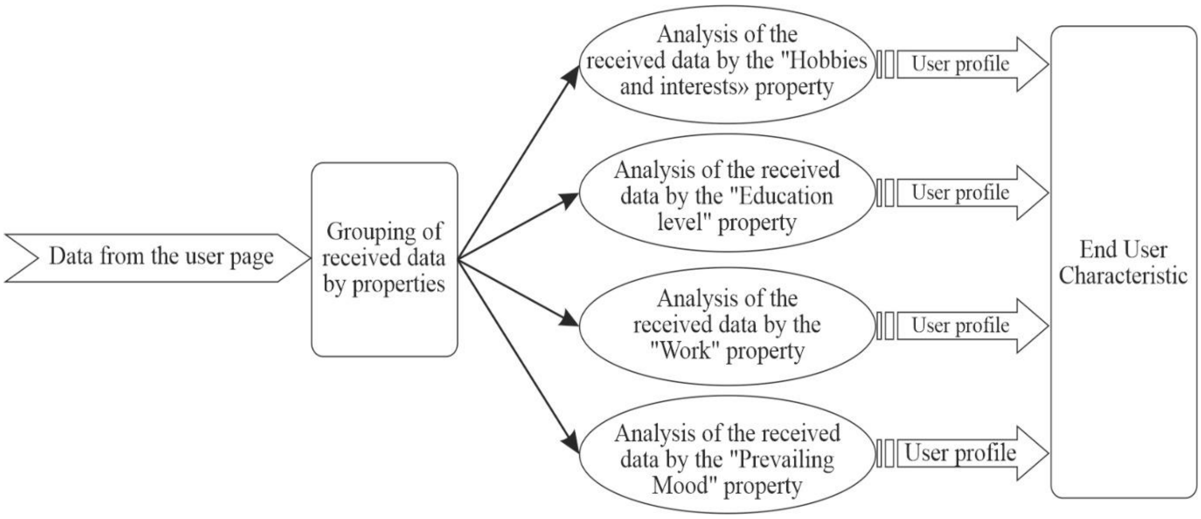

Data from the user page
Grouping of received data by properties
Analysis of the received data by the "Hobbies and interests» property
Analysis of the received data by the "Education level" property
Analysis of the received data by the "Work" property
Analysis of the received data by the "Prevailing Mood" property
User profile
User profile
User profile
User profile
End User Characteristic

Figure 1: User profile modelling

Security information relates to the user's authority to use the system and takes the form of User name; Password. The OAuth 2.0 protocol (OAuth 2.0, 2012) provides the necessary security for secure user login. Role information provides an understanding of the relationship between system users and can be described as: Administrator; Tutor and Student/Apprentice. Thus, the system can easily provide different services to different types of users. Personal information is mostly static data that records some basic information about the user: Name; Address; Telephone; School; Email; Gender; Telephone. It may also contain useful information such as postcode, which may help identify other users living in the area that an active user may connect to, as this may be useful for identification/placement within a group or beyond [21-28]. The user's interests in the system essentially represent the topics that the learner is working on (or wants to work on) and improves their performance. User interests can be collected in two ways: implicitly and explicitly [15]. Indirectly capturing user interests will mean that user behavior (topics selected for reading or specific tests selected for specific topics) will need to be observed and then mapped to the system's database (RDF database) using semantic similarity measures. Dynamic data in a user profile is essentially data created by the user while running tests. Such information as: Test ID; Overall Score; Date of shooting; Time to Completion; Qx-id, score (or just true/false).

We may wish to record the time it takes a user to answer a question because: a) this may vary from user to user; b) it can be used to distinguish between difficult and easier questions (and even use this information later to adjust the difficulty level of the question). In addition, the level of difficulty of individual questions in the test can give an idea of the overall level of difficulty of the test [20]. The data obtained from the tests will be used to capture and record the user's progress on the topic. The test can cover several topics with questions.

When a user interacts with an e-learning system, he/she does so by performing a set of actions [1]. Because a user logs in with a unique ID, their activity can be tracked. Assume that the log data is "raw". Assuming activity is recorded in sessions, the raw data would look like this: Session ID/User ID; Date/Timestamp; Duration; Action x, timestamp x. Where the action can be:

1. Test_Taken, TestID;
2. Topic_Browsed, TopicID
3. Topic_searched, TopicID
4. Talked to a Tutor, TutorID

In the educational platform, we can distinguish between self-directed and directed learning. The actions we might want to record differ slightly between them, although they share many elements. In both areas, the concept of "engagement" is very important. Engagement can be measured using a combination of the following [29-35]:

- Self-direction:

(i) How often a person logs in;
(ii) Duration of the session;
(iii) Duration of page view;
(i) Interrupted Tests;
(i) View results (has the student always viewed results?) by links;
(ii) Repetition of topics, i.e., taking another test on the same topic.

- Direction:

(i) How often the student contacted the teacher;
(ii) Teacher-provided feedback;
(iii) Additional tests assigned by the teacher;
(iv) Whether or not the user actually accepted them;
(v) References recommended by the instructor.

Evaluating the effectiveness of the use of distance education systems is impossible without studying statistical data on the organization of educational content, its quality and compliance with the educational and calendar plans of the organization of the educational process in general educational institutions [36-52]. Building a unified information system of an organization requires the integration of various information systems [23]. In the context of the tasks solved by the distance learning system, it is advisable to integrate it with the following information systems [53-62]:

- personnel management system;
- personnel evaluation system;
- knowledge management system;
- talent management system.

The integration of the systems presented above in most cases ensures the organization of information exchange between them [63-69].

3. Methods and materials

Nowadays, there are a large number of different programming languages and each of them has its own scope of application, but still, in order to conduct an analysis on the choice of the best language, it is necessary to select several of the most popular languages to conduct an analysis between them, so in this section we will turn to statistics by popularity as. The following technology stack is selected for work: JavaScript, HTML, CSS, TAILWIND, Node Js.

The following main dependencies were used for development:

- "@types/node": "18.11.3",
- "@types/react": "18.0.21",
- "@types/react-dom": "18.0.6",
- "autoprefixer": "^10.4.12",
- "eslint": "^8.30.0",
- "postcss": "^8.4.18",
- "prisma": "^4.8.0",

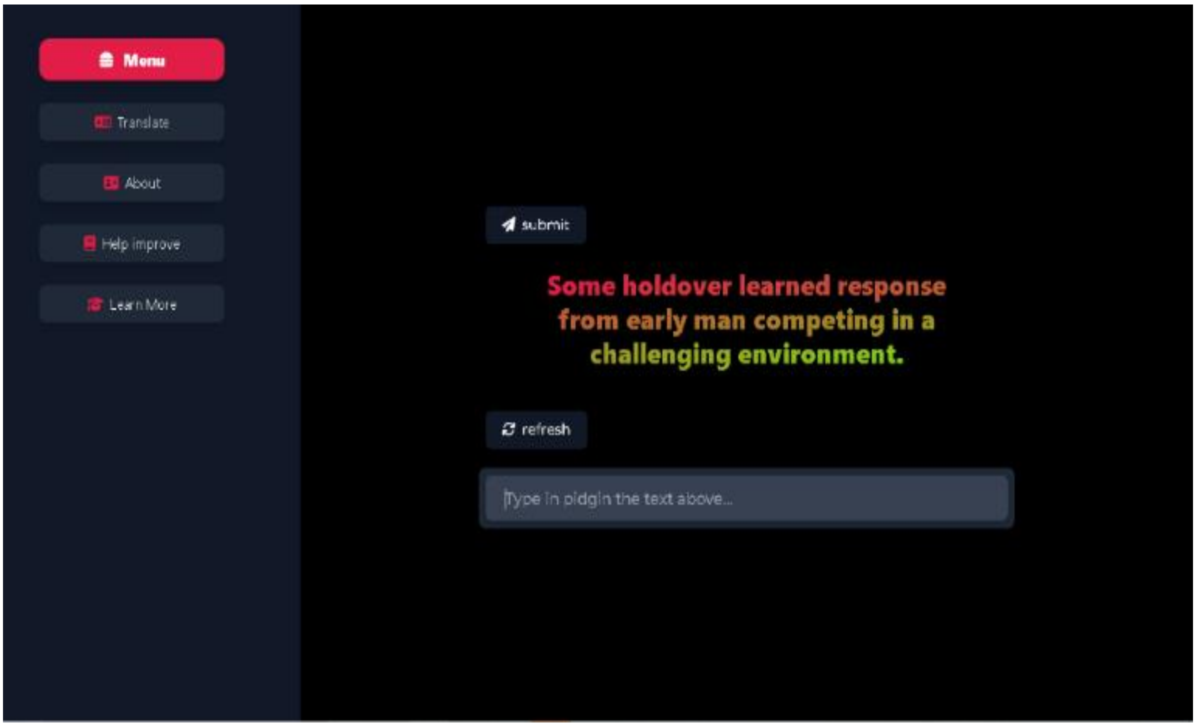
- "tailwindcss": "^3.2.4",
- "typescript": "4.9.4"

Visual Studio Code was used as the code editor. Visual Studio Code is a full-featured text editor for editing local files or the code base. It includes various features for editing the code base, which helps developers track changes. Various features supported in VSC: Syntax highlighting, Auto indent, Recognition of file types, Sidebar with files of the specified directory, macro, Plugin and packages. Visual Studio Code is used as an integrated development editor (IDE) like Sublime text and NetBeans. The current version of the VS code editor is compatible with various operating systems such as Windows, Linux and MacOS.

When searching the Internet, users often feel overwhelmed by the amount of data that comes back to them. Methods and systems are needed to help users navigate the Internet and filter information. This is especially important for distance learning sites, as it is important that every user stays on the platform. Current work has combined user profiling and a responsive user interface to help satisfy all users. User profiling [12] can be approached in one of three ways:

(1) the use of stereotypes;
(2) use of surveys/questionnaires;
(3) the use of a “learned model.

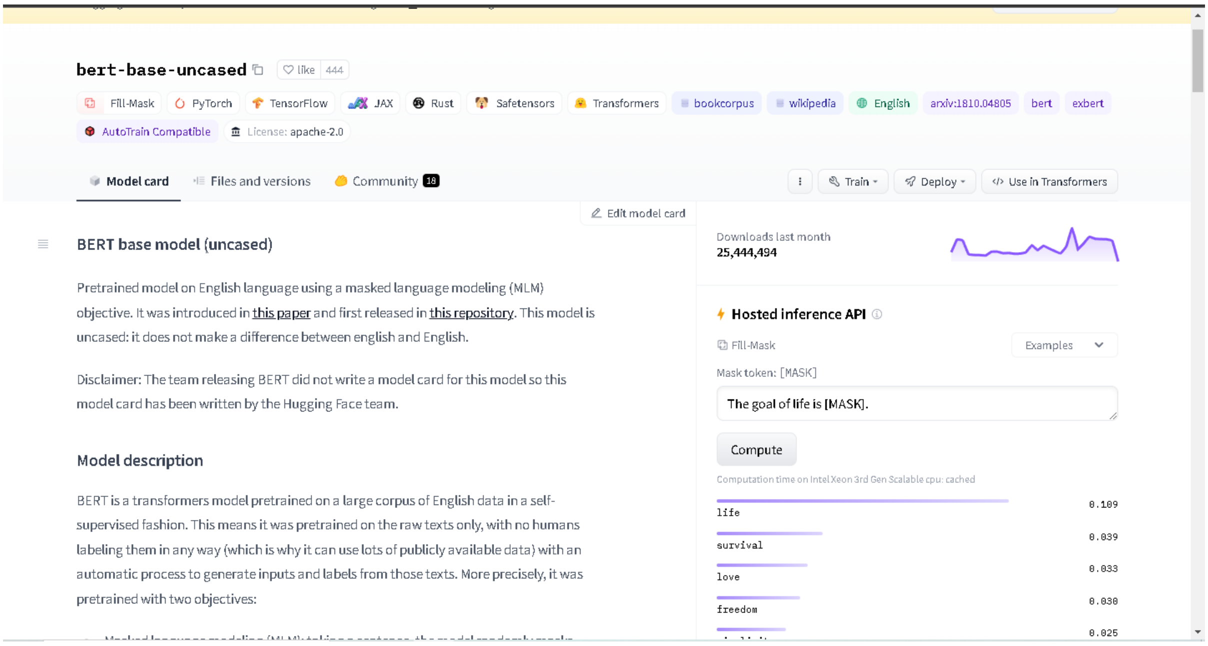
The first two approaches rely on traditional marketing techniques using known information or information gathered in person or over the phone to create relevant profiles. A third approach that uses "learned models" is our area of interest. The approach involves creating a system that initially does not know its users, but over time develops a profile model based on user interactions. Profile models can be created individually, for each user, or collectively, gathering all user data together to form an overall profile of interests and behavior.



Some holdover learned response from early man competing in a challenging environment.

Figure 2: Formation of the user profile

The developed product has a simple interface consisting of intuitive blocks. The program, the main direction of which is to help with the translation of text from different languages into a language understandable to the user, works according to the following principle: To begin with, the user is presented with a reserved set of phrases for translation, based on this neuron, the network collects data to form a user profile. Data-sets and models for training were used from the resource that collects all data-sets and models for training neural networks using languages https://huggingface.co/.



BERT base model (uncased)

Pretrained model on English language using a masked language modeling (MLM) objective. It was introduced in this paper and first released in this repository. This model is uncased: it does not make a difference between english and English.

Disclaimer: The team releasing BERT did not write a model card for this model so this model card has been written by the Hugging Face team.

Model description

BERT is a transformers model pretrained on a large corpus of English data in a self-supervised fashion. This means it was pretrained on the raw texts only, with no humans labeling them in any way (which is why it can use lots of publicly available data) with an automatic process to generate inputs and labels from those texts. More precisely, it was pretrained with two objectives:

Downloads last month
25,444,494

Mask token: [MASK]

The goal of life is [MASK].

Computation time on Intel Xeon 3rd Gen Scalable cpu: cached

```
life        0.109
survival    0.039
love        0.033
freedom     0.030
            0.025
```

Figure 3: Huggingface user interface

After training the network, it becomes possible to use the selected languages based on the translation made during initialization of the application. In order for the neural network to be able to perform the task, it must be trained. Training a neural network is a process in which the parameters of a neural network are adjusted by simulating the environment in which the network is embedded. The type of training is determined by the method of adjusting the parameters. The learning process of the neural network is shown in Fig. 4.

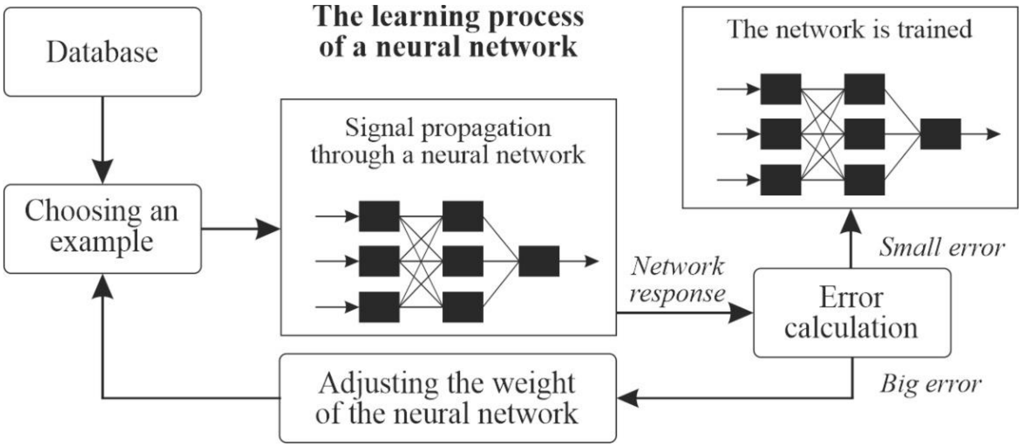


Figure 4: The learning process of a neural network

In the process of functioning, the neural network forms an output signal Y, implementing some function $Y = G(X)$. If the architecture of the network is given, then the form of the function G is determined by the values of the synaptic weights and the biased network. Let the solution of some problem be the function $Y = F(X)$, given by the input-output data parameters (X_1, Y_1), (X_2, Y_2), ..., (X_N, Y_N), for which $Y_k = F(X_k)$ $(k = 1, 2, ..., N)$. Learning consists in finding (synthesis) a function G close to F in the sense of some error function E. If a set of training examples is selected - pairs (X_N, Y_N) (where $k = 1, 2, ..., N$) and the method of calculating the error function E is selected, then the training of the neural network turns into a multidimensional optimization task that has a very large dimension, and since the function E can have an arbitrary form, learning in the general case is a multi-extremal non-convex optimization problem.

4. Experiments

According to different aspects of application and use of e-learning systems, adaptability can be defined in different ways. It is proposed to define adaptability as the property of a learning system that adapts and changes itself according to the requirements and characteristics of users before and during its use. On this basis, the following adaptive levels are distinguished: elementary adaptive level; static adaptive level; dynamic adaptive level.

Goal tree is a logical thinking tool that starts with the goal which the organization is trying to achieve and breaks down all the necessary conditions to achieve it. The goal tree is depicted in the Fig. 5.

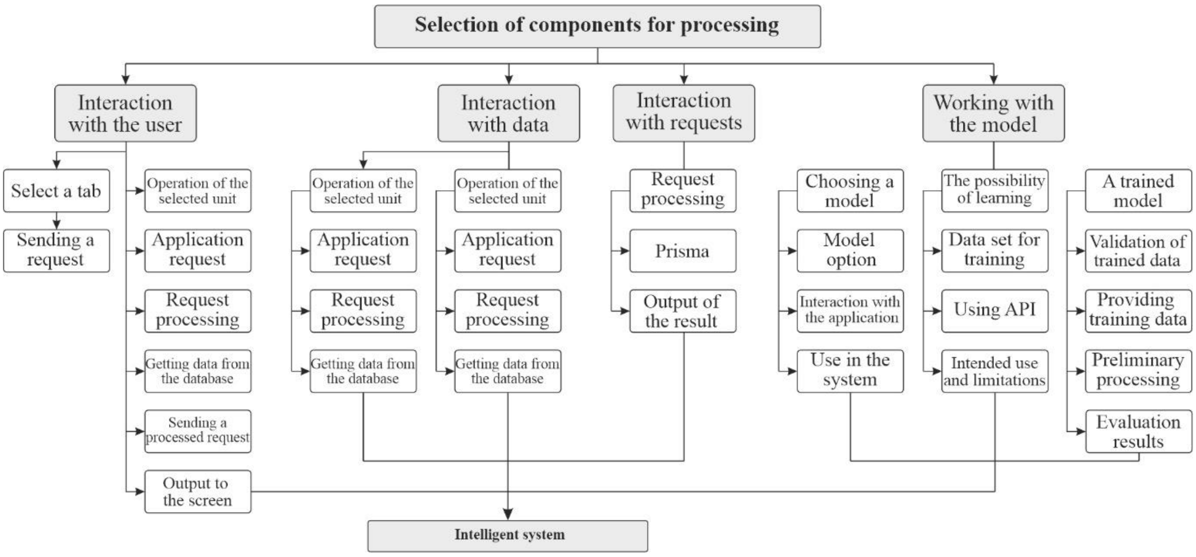


Figure 5: The goal tree

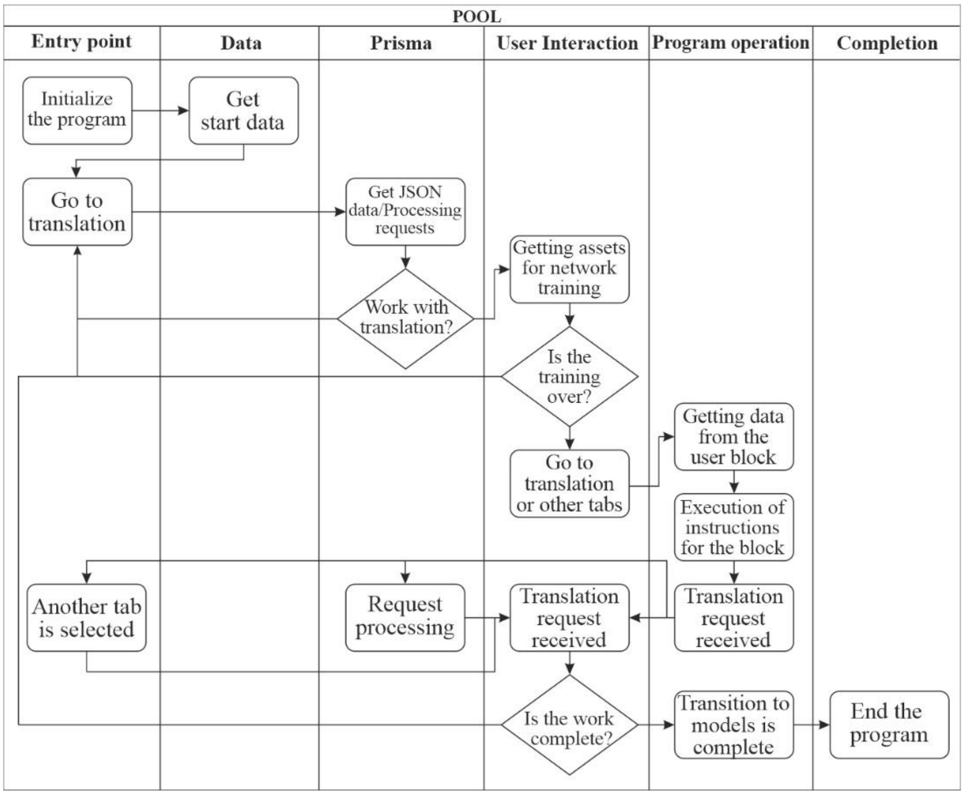

POOL
Entry point
Data
Prisma
User Interaction
Program operation
Completion
Initialize the program
Get start data
Go to translation
Get JSON data/Processing requests
Work with translation?
Getting assets for network training
Is the training over?
Go to translation or other tabs
Getting data from the user block
Execution of instructions for the block
Another tab is selected
Request processing
Translation request received
Translation request received
Is the work complete?
Transition to models is complete
End the program

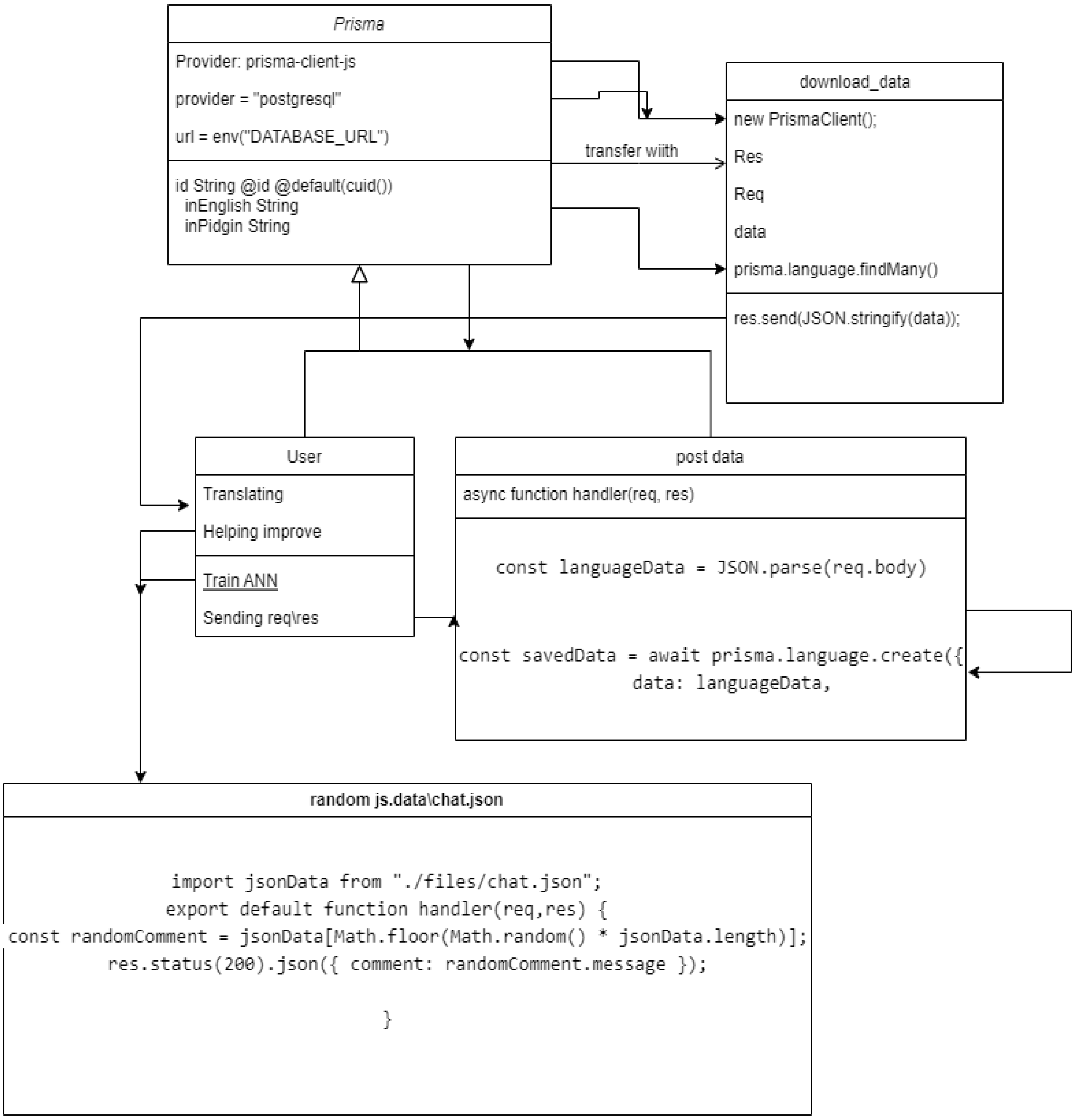
Figure 6: Activity diagram

Fig. 6 demonstrates the Activity diagram that has such main control points: Entry point (initialization); Data; Prism (orm db); Interaction with the user; Operation of the program; Termination

At the first stage - program execution, the application is initialized. The application receives initial assets, interface, data. At the second stage, after loading the main components, database data and software Json files are checked. At the third stage, a module with a prism is executed, which allows you to process requests and data sent/extracted from the database. We receive json, after processing requests and transfer the user to the main application in the translation tab.

The fourth stage is the stage of interaction with the user. If translation is selected - user data is accepted -> request processing -> providing results -> record in json -> record in db. Another option is model training, which works as follows: Training is selected -> We download json assets -> We expect a translation from the user -> We send data to the database -> We randomly generate the next translation. If the training is finished or another tab is selected, proceed to the fifth stage.

The fifth stage - The operation of the application is indicated on the activity diagram (Fig. 6). The application, at this stage, interacts with data and user requests and interacts programmatically between components and user requests. If the transition to models is selected or the application is closed, we proceed to stage six. The sixth stage – the transition to the review of models and training or closing the application is selected – we save the received data, initialize the completion of the program.



```
const languageData = JSON.parse(req.body)

const savedData = await prisma.language.create({
            data: languageData,
```

```
            import jsonData from "./files/chat.json";
            export default function handler(req,res) {
const randomComment = jsonData[Math.floor(Math.random() * jsonData.length)];
        res.status(200).json({ comment: randomComment.message });

                                }
```

Figure 7: Class diagram

The class diagram is clear and simple, when the user interacts with the application, he chooses one of the available buttons. If you choose to translate, the data is sent to prisma using download-data, which accepts the main parameters: language, request, status. If training is selected, we receive data by request in the post_data file, we have several request\status parameters, after which we interact with random_data, which, using chat.json, randomly generates and sends sentences to the user for translation from the selected asset.

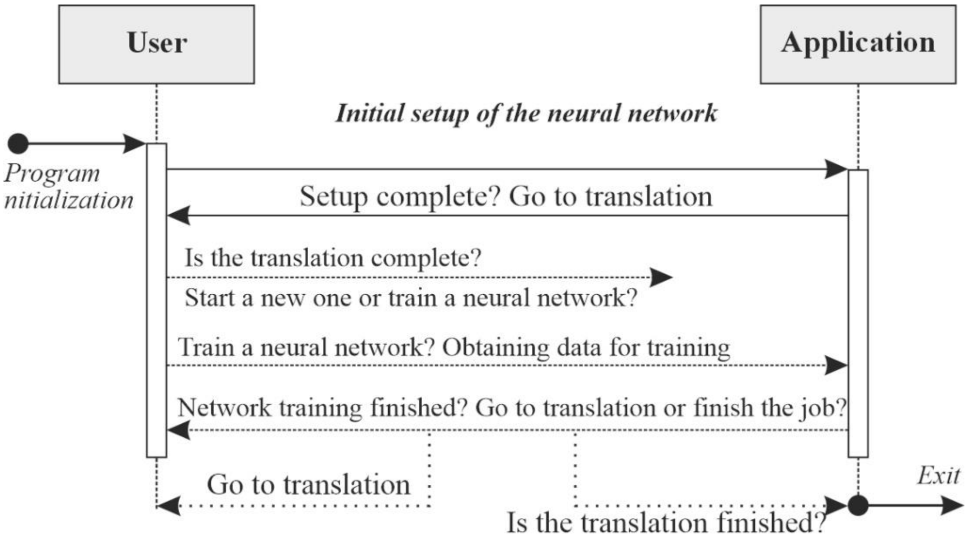

User
Application
Initial setup of the neural network
Program
nitialization
Setup complete? Go to translation
Is the translation complete?
Start a new one or train a neural network?
Train a neural network? Obtaining data for training
Network training finished? Go to translation or finish the job?
Go to translation
Exit
Is the translation finished?

Figure 8: State diagram

The state diagram is as follows: Interaction with the user starts the initialization of the program, after which the initial configuration is performed.

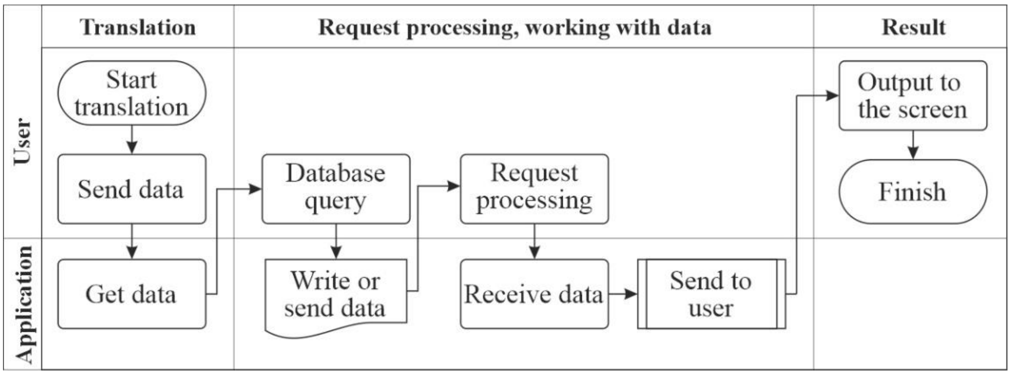


Figure 9: Cooperation diagram

System analysis relies on a number of applied logical-mathematical disciplines, technical procedures and methods that are widely used in management activities, including formalized and informal means of research, as well as on a set of principles, that is, basic rules accepted as truth, which are used as a basis for building methods of analysis. Flowchart analysis is usually applied to fairly simple systems, while fault tree synthesis is applied to more complex systems.

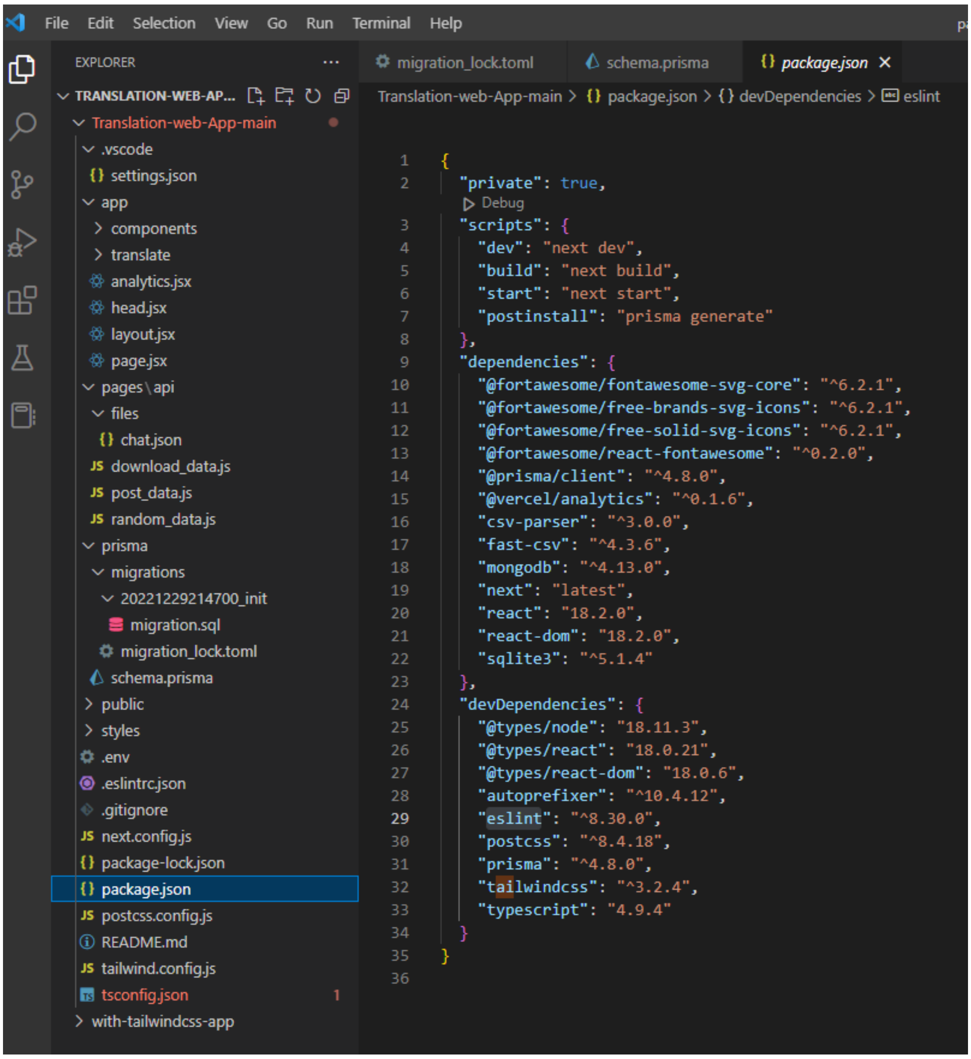
5. Results and discussions

Today, the Internet offers various ways to learn foreign languages. Thespecial variation is the study of English as the main language of communication of the global community. We will try to analyze the most effective methods of learning English using some online resources. Today there are 4 main methods of teaching English:

- Grammar Translation – a classic method of learning English (translation from the native language into a foreign language and vice versa).

- Direct Method - direct method (The main attention is paid to good pronunciation, spontaneous use of language without the use of translation, little attention is paid to grammar analysis).
- Audio-lingualism is one of the first modern methods (the method of repeating and memorizing standard phrases).
- Communicative Language Teaching – modern standard method (communicative learning method is based on the idea that successful learning of a foreign language occurs with the help of its study in real situations, which, in turn, leads to natural mastery and ability to use a foreign language).

So, for our developed application, we will use grammar translation and direct method, for use by one user, and also to increase the functionality of the application, we will use phraseological units, dialect words and slang elements for it. That will improve understanding and learning of the chosen language. To start the application, it is necessary to connect the necessary libraries and modules. First, let's install the modules and dependencies correctly:



```json
{
  "private": true,
  "scripts": {
    "dev": "next dev",
    "build": "next build",
    "start": "next start",
    "postinstall": "prisma generate"
  },
  "dependencies": {
    "@fortawesome/fontawesome-svg-core": "^6.2.1",
    "@fortawesome/free-brands-svg-icons": "^6.2.1",
    "@fortawesome/free-solid-svg-icons": "^6.2.1",
    "@fortawesome/react-fontawesome": "^0.2.0",
    "@prisma/client": "^4.8.0",
    "@vercel/analytics": "^0.1.6",
    "csv-parser": "^3.0.0",
    "fast-csv": "^4.3.6",
    "mongodb": "^4.13.0",
    "next": "latest",
    "react": "18.2.0",
    "react-dom": "18.2.0",
    "sqlite3": "^5.1.4"
  },
  "devDependencies": {
    "@types/node": "18.11.3",
    "@types/react": "18.0.21",
    "@types/react-dom": "18.0.6",
    "autoprefixer": "^10.4.12",
    "eslint": "^8.30.0",
    "postcss": "^8.4.18",
    "prisma": "^4.8.0",
    "tailwindcss": "^3.2.4",
    "typescript": "4.9.4"
  }
}
```

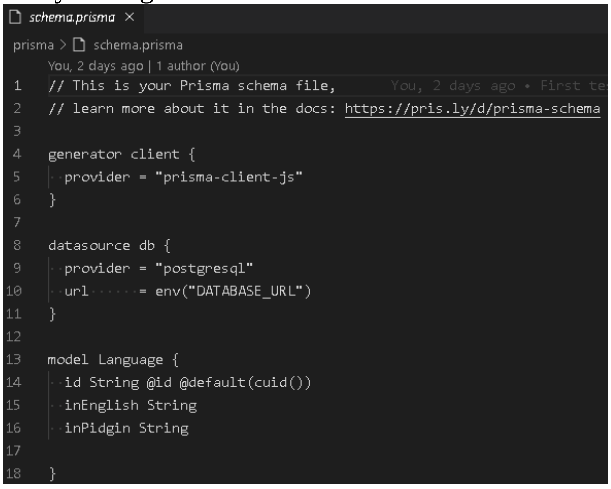
Figure 10: Connection of necessary libraries and modules

The main logic of the application is written using the Java Script language, Tailwind Css is used for styling. Prisma is used to place data obtained in the process of interaction with the user. The Prisma schema is intuitive and allows you to declare database tables in an understandable way, simplifying the data modeling process. Models are defined manually or analyzed from an existing database. The Prisma schema file is the main configuration file for setting up Prisma. It is usually called schema.prisma and consists of the following parts:

- Data Sources: Specify the details of the data sources to which Prisma should connect (for example, a PostgreSQL database);

- Generators: Defines which users should be generated based on the data model (eg Prisma Client);
- Data model definition: defines application models (data form per data source) and their relationships.

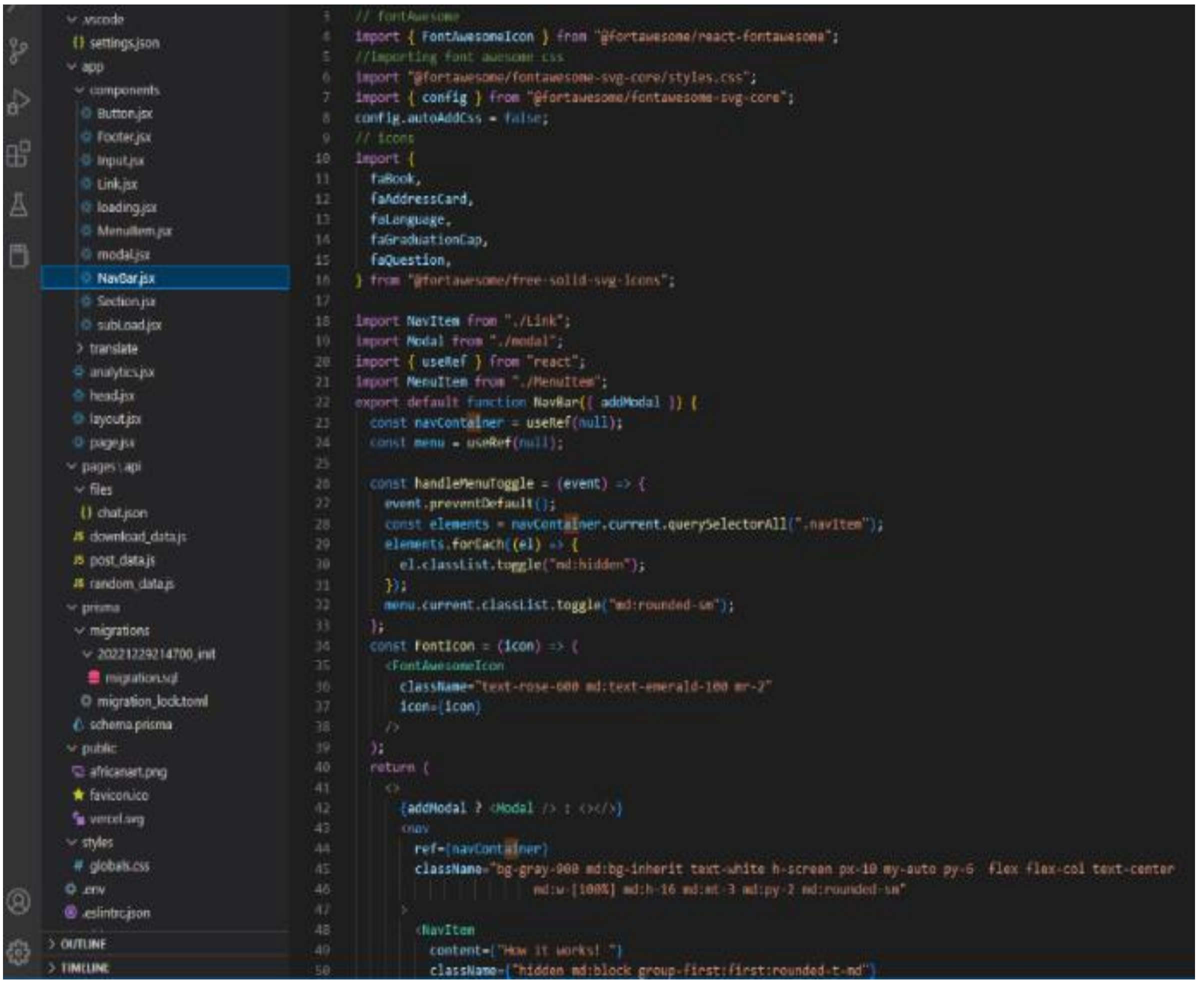
So, let's make the necessary settings:



```
// This is your Prisma schema file,
// learn more about it in the docs: https://pris.ly/d/prisma-schema

generator client {
  provider = "prisma-client-js"
}

datasource db {
  provider = "postgresql"
  url      = env("DATABASE_URL")
}

model Language {
  id String @id @default(cuid())
  inEnglish String
  inPidgin String

}
```

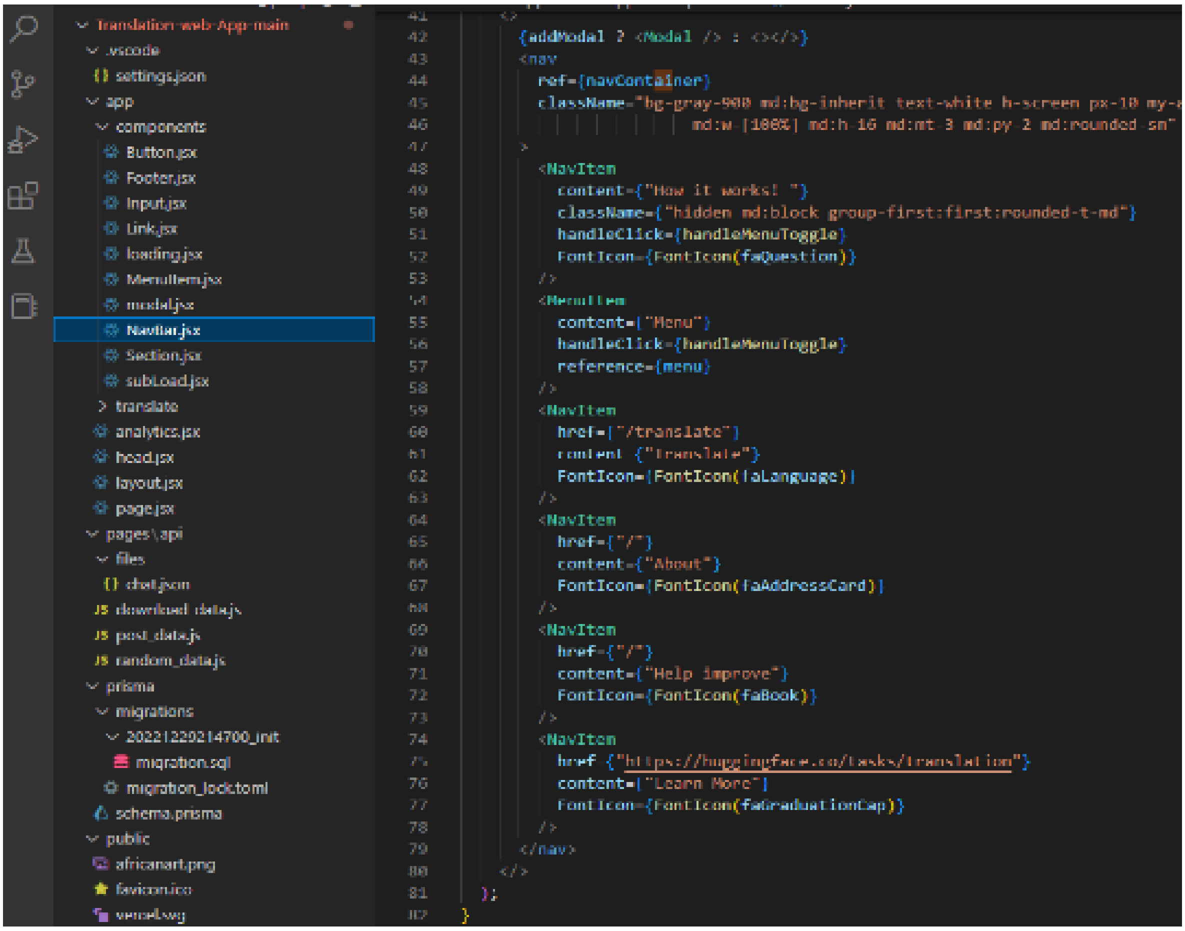
Figure 11: Prisma settings

After creating and configuring Prisma, it is necessary to create interface elements for interaction with the user:



```
// fontAwesome
import { FontAwesomeIcon } from "@fortawesome/react-fontawesome";
//importing font awesome css
import "@fortawesome/fontawesome-svg-core/styles.css";
import { config } from "@fortawesome/fontawesome-svg-core";
config.autoAddCss = false;
// icons
import {
  faBook,
  faAddressCard,
  faLanguage,
  faGraduationCap,
  faQuestion,
} from "@fortawesome/free-solid-svg-icons";

import NavItem from "./Link";
import Modal from "./modal";
import { useRef } from "react";
import MenuItem from "./MenuItem";
export default function NavBar({ addModal }) {
  const navContainer = useRef(null);
  const menu = useRef(null);

  const handleMenuToggle = (event) => {
    event.preventDefault();
    const elements = navContainer.current.querySelectorAll(".navItem");
    elements.forEach((el) => {
      el.classList.toggle("md:hidden");
    });
    menu.current.classList.toggle("md:rounded-sm");
  };
  const FontIcon = (icon) => (
    <FontAwesomeIcon
      className="text-rose-600 md:text-emerald-100 mr-2"
      icon={icon}
    />
  );
  return (
    <>
      {addModal ? <Modal /> : <></>}
      <nav
        ref={navContainer}
        className="bg-gray-900 md:bg-inherit text-white h-screen px-10 my-auto py-6  flex flex-col text-center
                  md:w-[100%] md:h-16 md:mt-3 md:py-2 md:rounded-sm"
      >
        <NavItem
          content={"How it works! "}
          className={"hidden md:block group-first:first:rounded-t-md"}
```

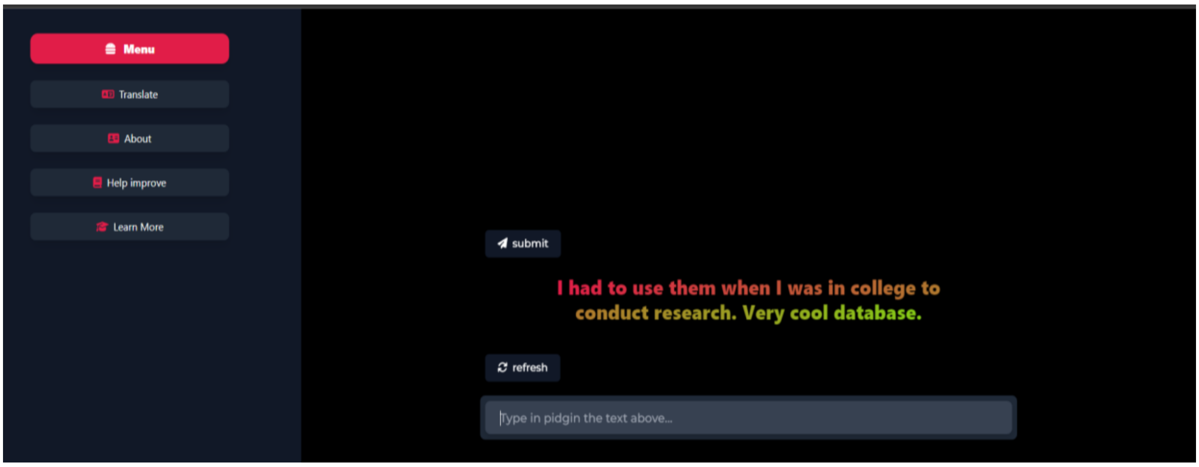
Figure 12: Creating a menu



```
    <>
      {addModal ? <Modal /> : <></>}
      <nav
        ref={navContainer}
        className="bg-gray-900 md:bg-inherit text-white h-screen px-10 my-
                    md:w-[100%] md:h-16 md:mt-3 md:py-2 md:rounded-sm"
      >
        <NavItem
          content={"How it works! "}
          className={"hidden md:block group-first:first:rounded-t-md"}
          handleClick={handleMenuToggle}
          FontIcon={FontIcon(faQuestion)}
        />
        <MenuItem
          content={"Menu"}
          handleClick={handleMenuToggle}
          reference={menu}
        />
        <NavItem
          href={"/translate"}
          content={"Translate"}
          FontIcon={FontIcon(faLanguage)}
        />
        <NavItem
          href={"/"}
          content={"About"}
          FontIcon={FontIcon(faAddressCard)}
        />
        <NavItem
          href={"/"}
          content={"Help improve"}
          FontIcon={FontIcon(faBook)}
        />
        <NavItem
          href={"https://huggingface.co/tasks/translation"}
          content={"Learn More"}
          FontIcon={FontIcon(faGraduationCap)}
        />
      </nav>
    </>
  );
}
```

Figure 13: Example figure

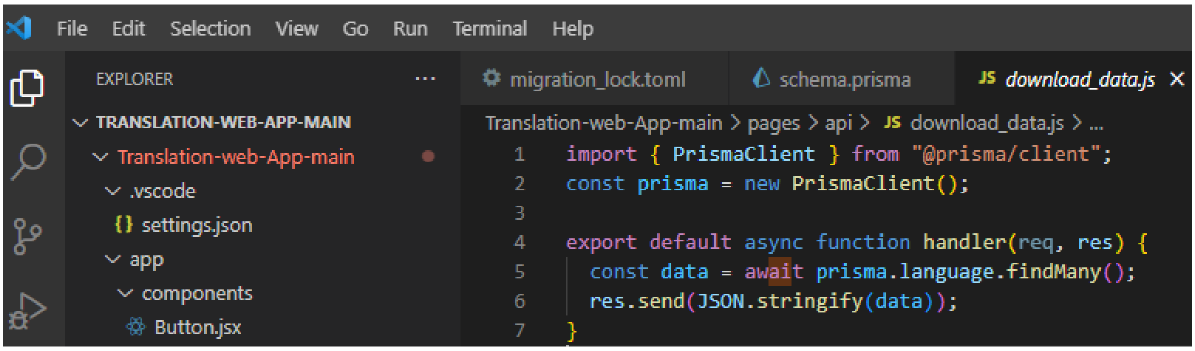
To begin with, we define what the client will use, then we import the necessary: Fonts (pictures for the menu), fonts, styles. After that, we connect the objects defined in other modules. After receiving the objects, we "lay out" them in containers and fill them with content. According to this logic, the necessary interface elements were created: Sections, buttons, electronic links, etc. After connecting the styles and the styling process, our app looks like this:



I had to use them when I was in college to conduct research. Very cool database.

Figure 14: User interface

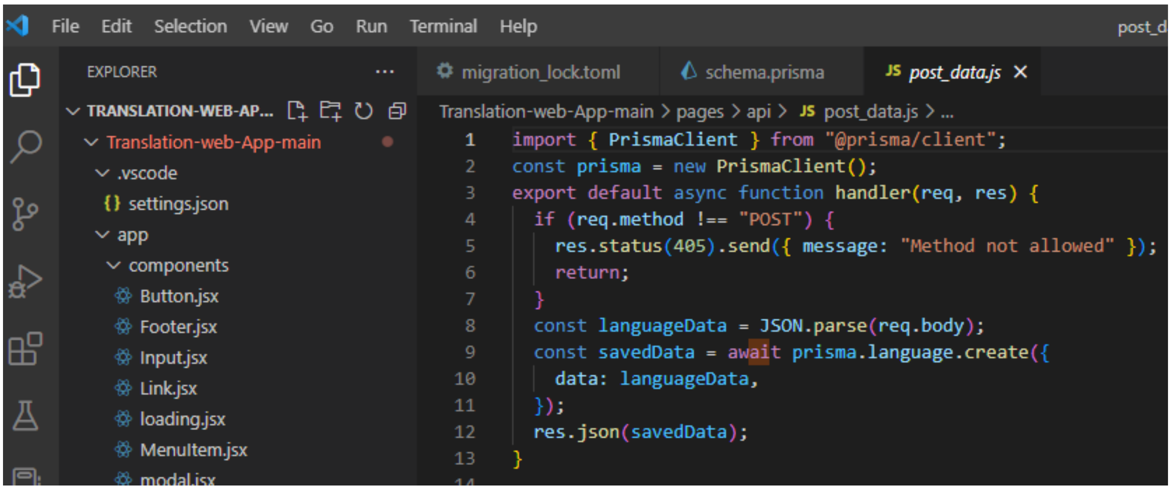
Next, we need to create some data, send some to the server, and take some from the server, the following modules perform these actions:



```
import { PrismaClient } from "@prisma/client";
const prisma = new PrismaClient();

export default async function handler(req, res) {
  const data = await prisma.language.findMany();
  res.send(JSON.stringify(data));
}
```

Figure 15: Loading data to the prism

We upload models and datasets for the neural network to the prism.

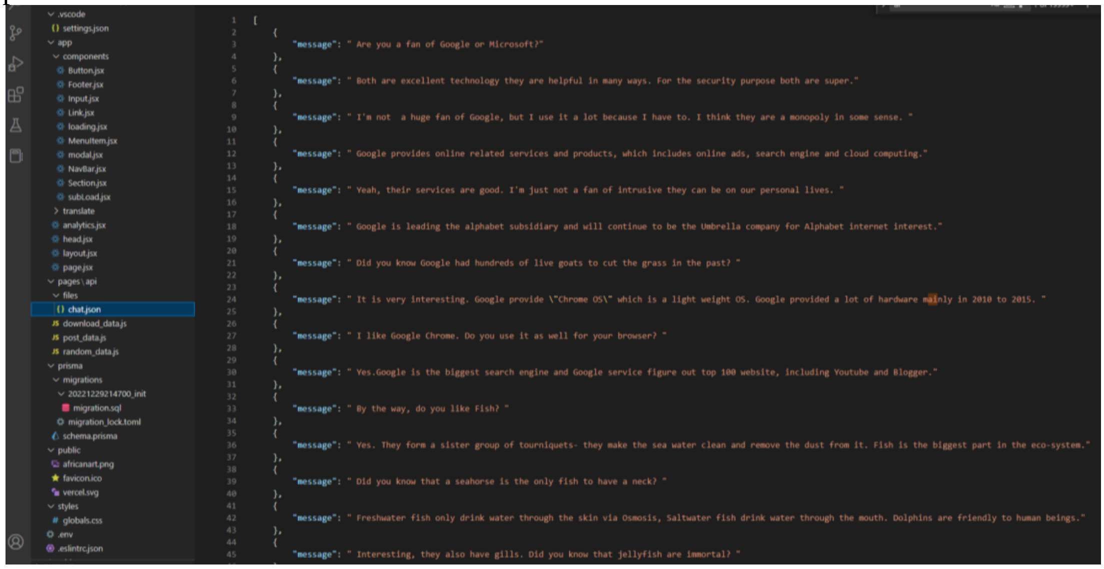


```
import { PrismaClient } from "@prisma/client";
const prisma = new PrismaClient();
export default async function handler(req, res) {
  if (req.method !== "POST") {
    res.status(405).send({ message: "Method not allowed" });
    return;
  }
  const languageData = JSON.parse(req.body);
  const savedData = await prisma.language.create({
    data: languageData,
  });
  res.json(savedData);
}
```

a

Figure 16: Loading data to the prism from the application

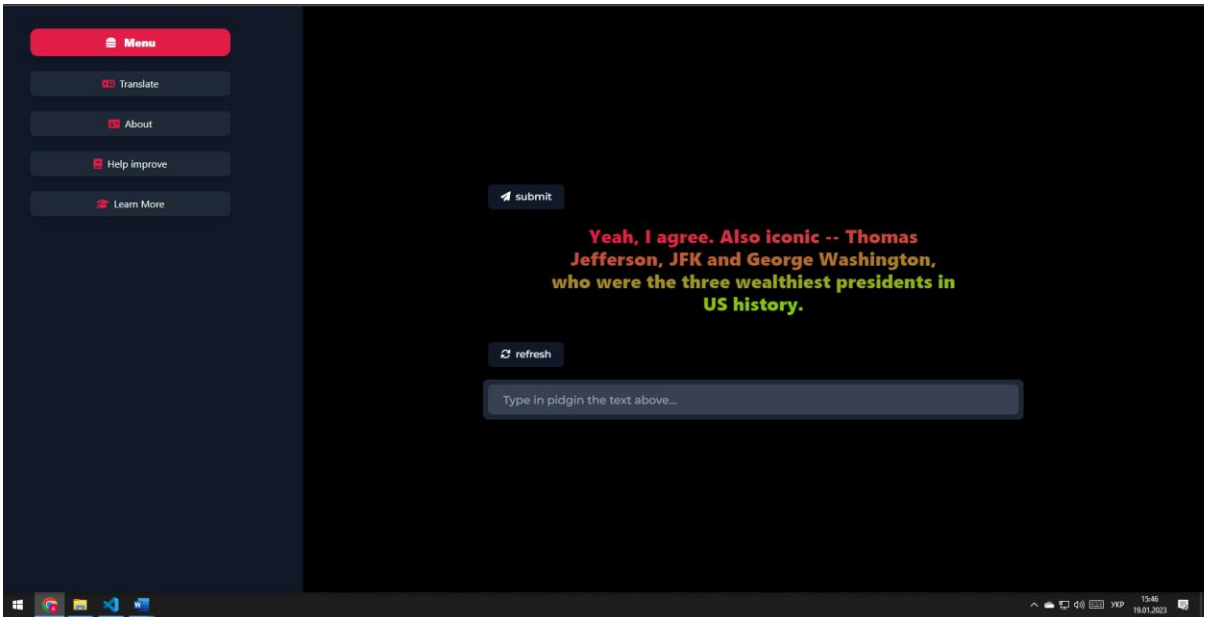
When the user interacts with the application, data is sent to the prism. POST [5] data transfer method. In programming, POST is one of many request methods supported by the HTTP protocol used on the World Wide Web. The POST request method is designed to send a request in which the web server accepts the data contained in the message body for storage. It is often used to upload a file or submit a completed web form.



```
[
    {
        "message": " Are you a fan of Google or Microsoft?"
    },
    {
        "message": " Both are excellent technology they are helpful in many ways. For the security purpose both are super."
    },
    {
        "message": " I'm not  a huge fan of Google, but I use it a lot because I have to. I think they are a monopoly in some sense. "
    },
    {
        "message": " Google provides online related services and products, which includes online ads, search engine and cloud computing."
    },
    {
        "message": " Yeah, their services are good. I'm just not a fan of intrusive they can be on our personal lives. "
    },
    {
        "message": " Google is leading the alphabet subsidiary and will continue to be the Umbrella company for Alphabet internet interest."
    },
    {
        "message": " Did you know Google had hundreds of live goats to cut the grass in the past? "
    },
    {
        "message": " It is very interesting. Google provide \"Chrome OS\" which is a light weight OS. Google provided a lot of hardware mainly in 2010 to 2015. "
    },
    {
        "message": " I like Google Chrome. Do you use it as well for your browser? "
    },
    {
        "message": " Yes.Google is the biggest search engine and Google service figure out top 100 website, including Youtube and Blogger."
    },
    {
        "message": " By the way, do you like Fish? "
    },
    {
        "message": " Yes. They form a sister group of tourniquets- they make the sea water clean and remove the dust from it. Fish is the biggest part in the eco-system."
    },
    {
        "message": " Did you know that a seahorse is the only fish to have a neck? "
    },
    {
        "message": " Freshwater fish only drink water through the skin via Osmosis, Saltwater fish drink water through the mouth. Dolphins are friendly to human beings."
    },
    {
        "message": " Interesting, they also have gills. Did you know that jellyfish are immortal? "
```

Figure 17: Assets for network training

To train the network, we use a ready-made collection of data (Assets), in the training application we have the following interface:

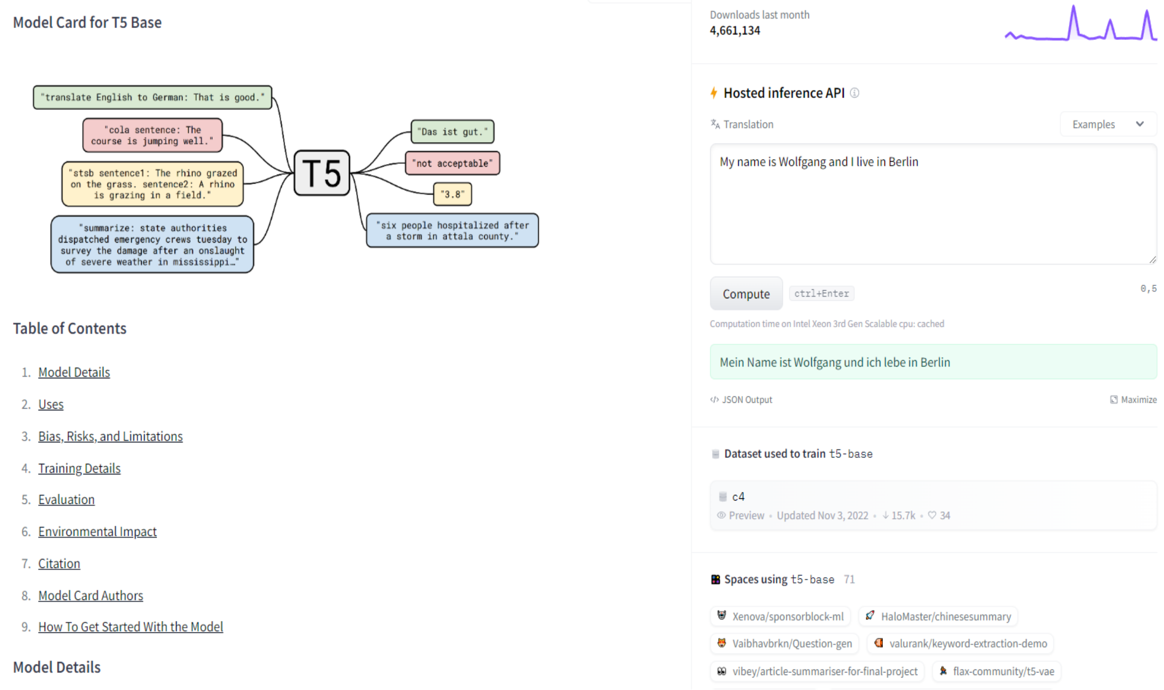


Yeah, I agree. Also iconic -- Thomas Jefferson, JFK and George Washington, who were the three wealthiest presidents in US history.

Type in pidgin the text above...

Figure 18: Interface of the developed application (Help improve)

Assets are extracted from our json file and provided to the user. After the user offers his version of the translation, the data will go to the server, after which the neural network will improve the translation based on what was sent. The main function of the developed application is translation. For the user, it is possible to configure a selected pair of languages for training. For example, you can use the pair: English-Ukrainian, after which the user will receive a correct translation. There are several important aspects here, the translator uses neural networks for training, and therefore each accurate translation contributes to the improvement of the app's translation. The neural network is trained to perform a more accurate translation by studying: phraseological units, a separate dialect, slang. The second aspect is when using models. Each model has its own initial settings, for example, one of the models we use:



Model Card for T5 Base

"translate English to German: That is good."
"cola sentence: The course is jumping well."
"stsb sentence1: The rhino grazed on the grass. sentence2: A rhino is grazing in a field."
"summarize: state authorities dispatched emergency crews tuesday to survey the damage after an onslaught of severe weather in mississippi…"
"Das ist gut."
"not acceptable"
"3.8"
"six people hospitalized after a storm in attala county."

Table of Contents

1. Model Details
2. Uses
3. Bias, Risks, and Limitations
4. Training Details
5. Evaluation
6. Environmental Impact
7. Citation
8. Model Card Authors
9. How To Get Started With the Model

Model Details

Downloads last month
4,661,134

Hosted inference API

Translation

My name is Wolfgang and I live in Berlin

0,5

Computation time on Intel Xeon 3rd Gen Scalable cpu: cached

Mein Name ist Wolfgang und ich lebe in Berlin

Dataset used to train t5-base

c4
Preview • Updated Nov 3, 2022 • 15.7k • 34

Spaces using t5-base 71

Xenova/sponsorblock-ml
HaloMaster/chinesesummary
Vaibhavbrkn/Question-gen
valurank/keyword-extraction-demo
vibey/article-summariser-for-final-project
flax-community/t5-vae

Figure 19: Model card

General information: T5-Base is a control point with 220 million parameters.

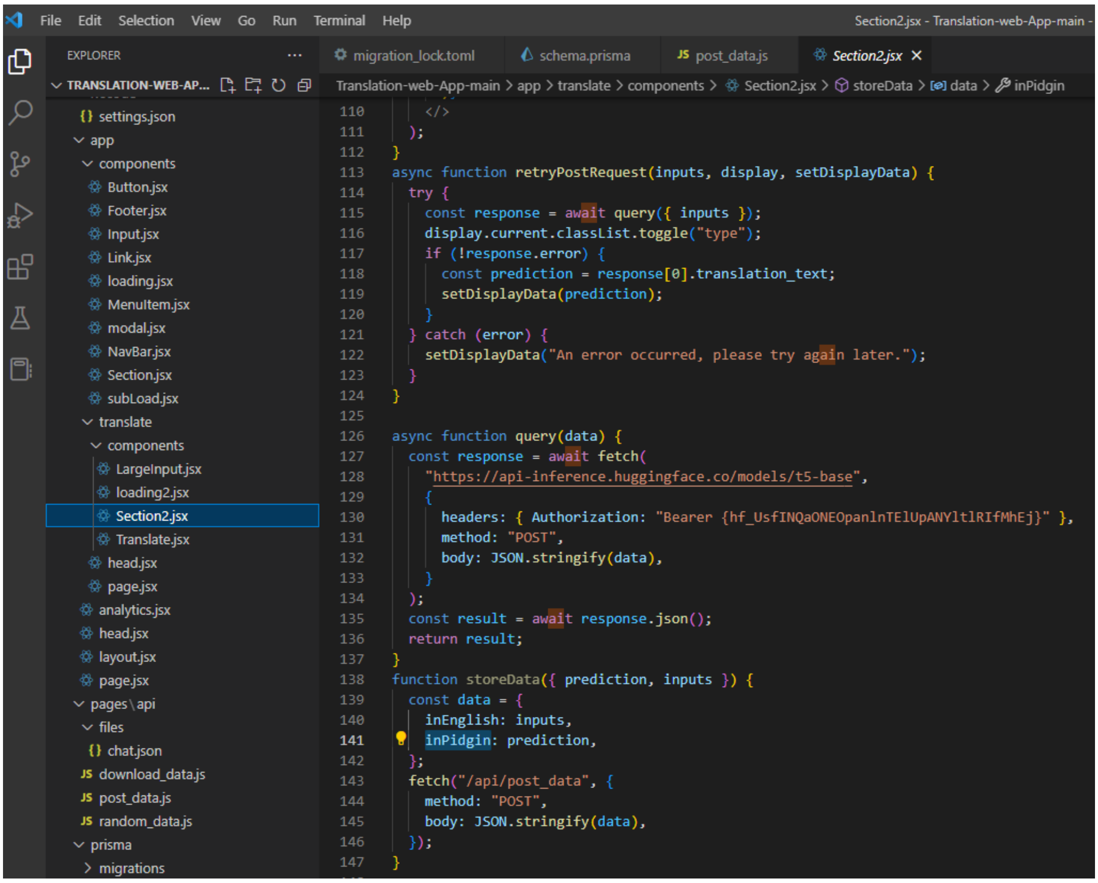
Developers: Colin Raffel, Noam Shazir, Adam Roberts, Catherine Lee, Sharan Narang, Michael Mathena, Yanqi Zhou, Wei Li, Peter J. Liu. Check out the related article and the GitHub repository

Model type: Language model. Language(s) (NLP): English, French, Romanian, German.

License: Apache 2.0. Related Models: All T5 control points

So, this is a big plus, we can choose a specific learning model for each user or group of users.

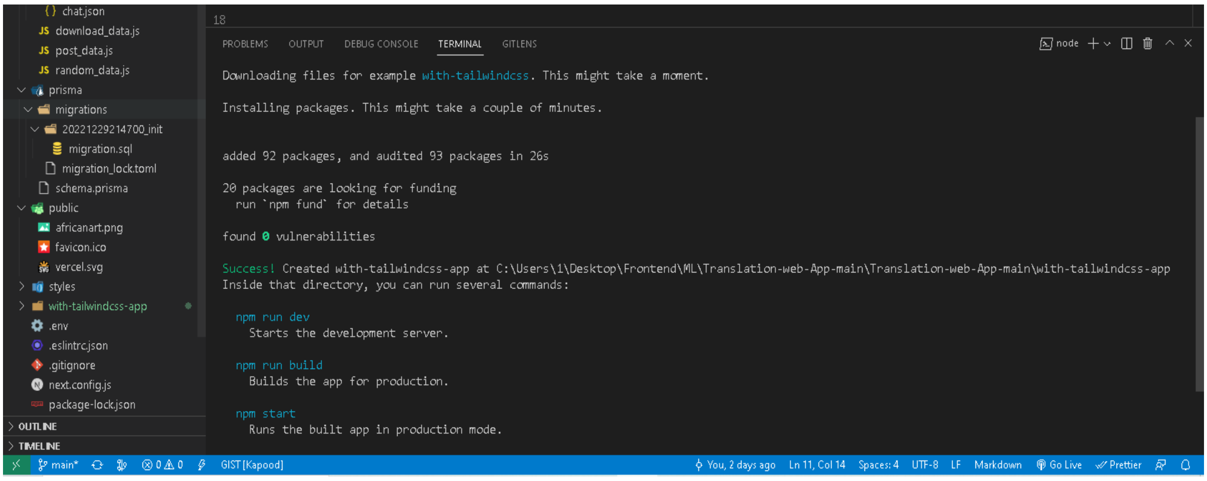
Data for translation is obtained with the help of the hugginface API using a token:



```
    </>
  );
}
async function retryPostRequest(inputs, display, setDisplayData) {
  try {
    const response = await query({ inputs });
    display.current.classList.toggle("type");
    if (!response.error) {
      const prediction = response[0].translation_text;
      setDisplayData(prediction);
    }
  } catch (error) {
    setDisplayData("An error occurred, please try again later.");
  }
}

async function query(data) {
  const response = await fetch(
    "https://api-inference.huggingface.co/models/t5-base",
    {
      headers: { Authorization: "Bearer {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}" },
      method: "POST",
      body: JSON.stringify(data),
    }
  );
  const result = await response.json();
  return result;
}
function storeData({ prediction, inputs }) {
  const data = {
    inEnglish: inputs,
    inPidgin: prediction,
  };
  fetch("/api/post_data", {
    method: "POST",
    body: JSON.stringify(data),
  });
}
```

Figure 20: Receiving data by token using an asynchronous function

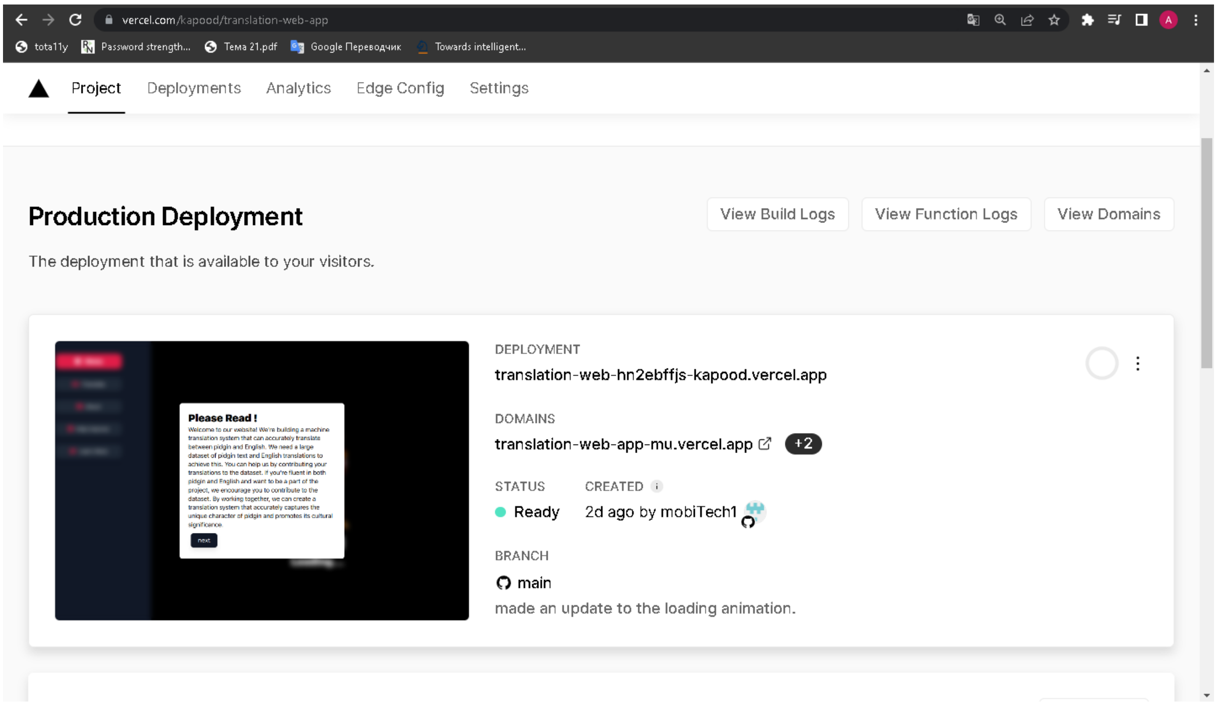
So, since the application is deployed locally, first it is necessary to unpack its styles with the following command: npx create-next-app --example with-tailwindcss with-tailwindcss-app.



```
Downloading files for example with-tailwindcss. This might take a moment.

Installing packages. This might take a couple of minutes.


added 92 packages, and audited 93 packages in 26s

20 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

Success! Created with-tailwindcss-app at C:\Users\1\Desktop\Frontend\ML\Translation-web-App-main\Translation-web-App-main\with-tailwindcss-app
Inside that directory, you can run several commands:

  npm run dev
    Starts the development server.

  npm run build
    Builds the app for production.

  npm start
    Runs the built app in production mode.
```

Figure 21: Unpacking packages

When we see that all the necessary modules are assembled, it is necessary to run the application locally, for this we use Versel. Go to https://vercel.com/dashboard, log in and add the project:



Production Deployment

The deployment that is available to your visitors.

DEPLOYMENT
translation-web-hn2ebffjs-kapood.vercel.app

DOMAINS
translation-web-app-mu.vercel.app +2

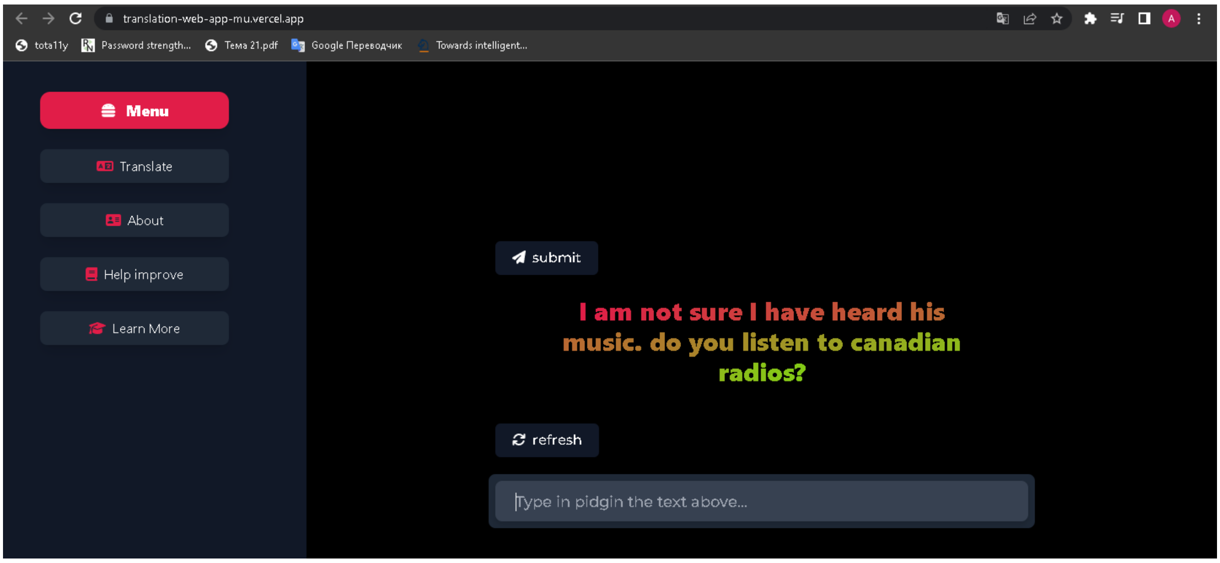
STATUS
Ready

CREATED
2d ago by mobiTech1

BRANCH
main
made an update to the loading animation.

Figure 22: Versel platform

After launching on the platform, we can get to the application itself:



I am not sure I have heard his music. do you listen to canadian radios?

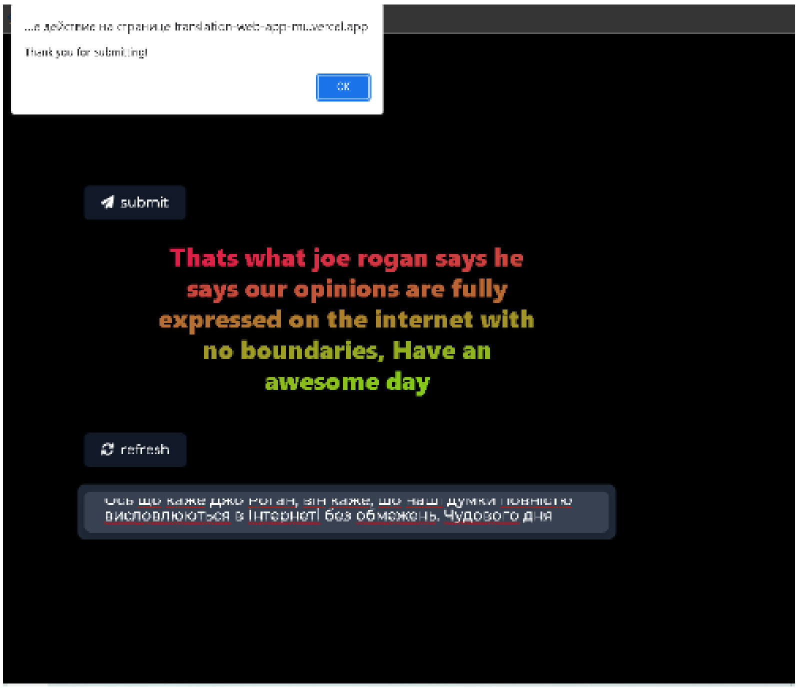
Type in pidgin the text above...

Figure 23: The developed application

On the left side, we will see a menu containing 4 buttons:

- Translate - perform a translation.
- About\Help improve - used for network training.
- Learn More – an electronic link to a web application that contains models and datasets, so we will work with each button.

In the current section, the application needs to translate the given phrase, to improve the translation and train the neural network, after we send the translation, we receive an alert that the data has been sent:



...е действие на странице translation-web-app-m...vercel.app

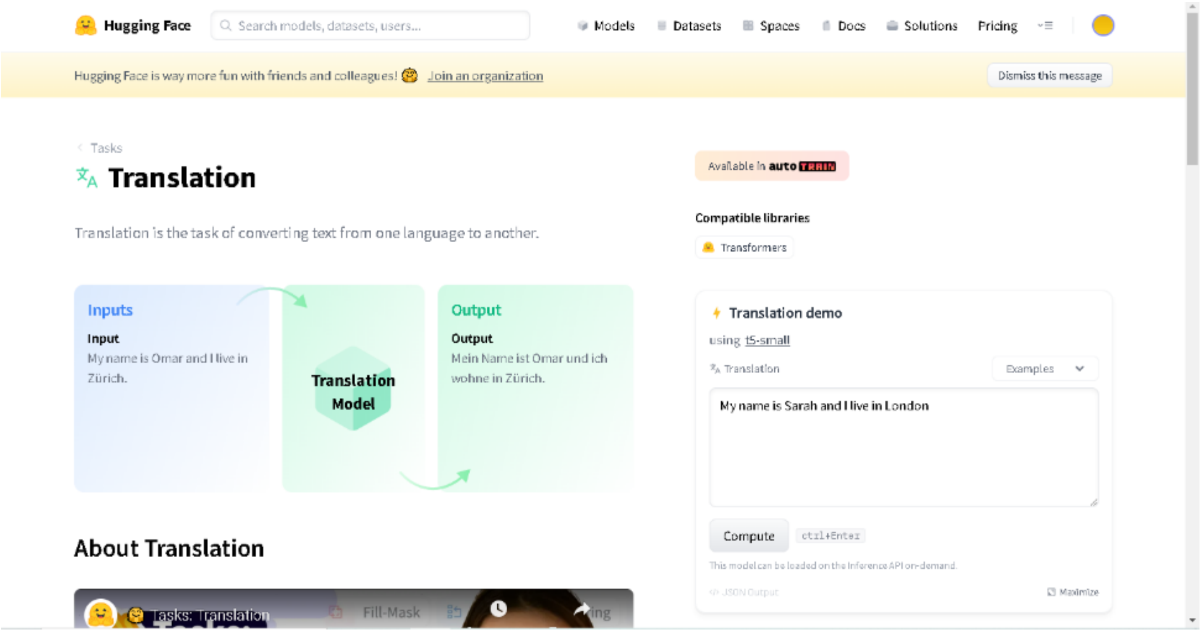
Thank you for submitting!

Thats what joe rogan says he says our opinions are fully expressed on the internet with no boundaries, Have an awesome day

Ось що каже Джо Роган, він каже, що наші думки повністю висловлюються в інтернеті без обмежень. Чудового дня

Figure 24: Alert from the application

When you click the Learn More button, you are redirected to a web resource with models and datasets:



Hugging Face is way more fun with friends and colleagues! Join an organization

Translation

Translation is the task of converting text from one language to another.

Inputs

Input

My name is Omar and I live in Zürich.

Output

Output

Mein Name ist Omar und ich wohne in Zürich.

Translation demo

using t5-small

My name is Sarah and I live in London

This model can be loaded on the Inference API on-demand.

About Translation

Figure 25: Learn More button – a redirect to a web resource

Since the goal of the work is to improve the process of remote learning of foreign languages, it was chosen to create an application for translation based on the choice of the native language and the language being studied. An educational process using a set of telecommunication technologies aimed at enabling students to learn the basic amount of information they need without direct contact between students and teachers during the learning process (which can take place both synchronously and asynchronously), and can be both an independent form of education, as well as a supplement to another more traditional form of education (full-time, part-time, extramural or externship), if necessary, giving a person the opportunity to study a foreign language training course. So, on the basis of this concept, a translation application was developed, which accurately translates both ordinary language and phraseological units, slang expressions, etc. The model is used as the basis of training, so let's analyze the model according to the main indicators. The model was pre-trained on BookCorpus, a dataset consisting of 11,038 unpublished books and the English Wikipedia (excluding lists, tables and titles).

The texts are written in lowercase and tokenized using WordPiece and a dictionary size of 30,000. Then the model input looks like this: [CLS] Sentence A [SEP] Sentence B [SEP].

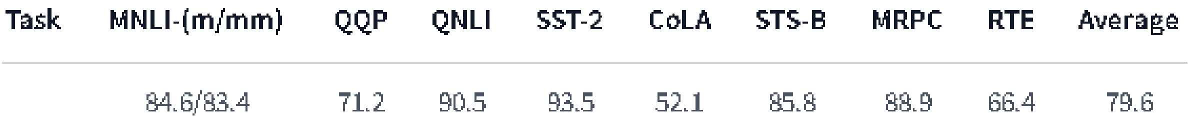
With probability 0.5, sentence A and sentence B match two consecutive sentences in the original corpus, and in other cases it is another random sentence in the corpus. Note that a sentence here is a continuous stretch of text, usually longer than one sentence. The only limitation is that the result with two "sentences" has a total length of less than 512 tokens.

The masking procedure details for each sentence are as follows:

- 15% of tokens are masked.
- In 80% of cases masked tokens are replaced by [MASK].
- 10% of the time, masked tokens are replaced by a random token (different) from the one they replace.
- In the remaining 10% of cases, masked markers remain unchanged.

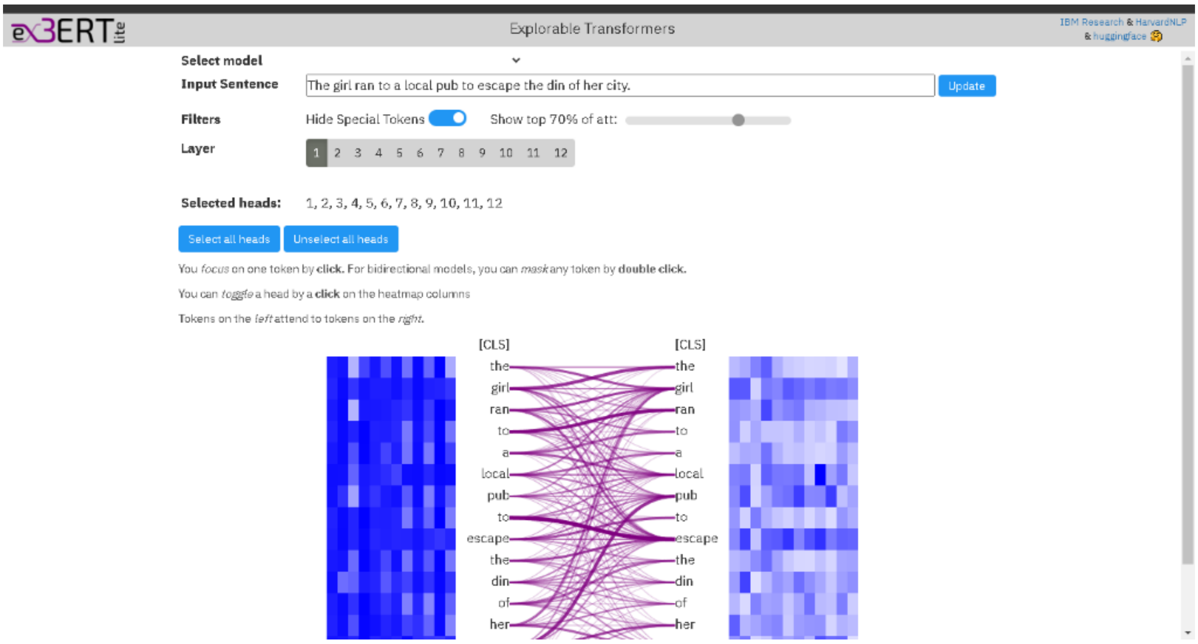
The model was trained on 4 Cloud TPUs in a Pod configuration (16 TPUs in total) for one million steps with a batch size of 256. The sequence length was limited to 128 markers for 90% of the steps and 512 for the remaining 10%. Adam optimizer is used with learning rate : 1e-4, β 1=0.9β1=0.9 and β 2=0.999β2=0.999 weight decay 0.01, learning rate warm-up for 10,000 steps and learning rate linear decrease after.

With fine-tuning for further tasks, this model achieves the following results:



Task	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	Average
	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6

Figure 26: Test results



Select model

Input Sentence: The girl ran to a local pub to escape the din of her city.

Filters: Hide Special Tokens ● Show top 70% of att:

Layer: 1 2 3 4 5 6 7 8 9 10 11 12

Selected heads: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

You *focus* on one token by **click.** For bidirectional models, you can *mask* any token by **double click.**

You can *toggle* a head by a **click** on the heatmap columns

Tokens on the *left* attend to tokens on the *right.*

Figure 27: Visualization of the model

Regarding the advantages and disadvantages of the application, the following can be highlighted:

- Advantages:

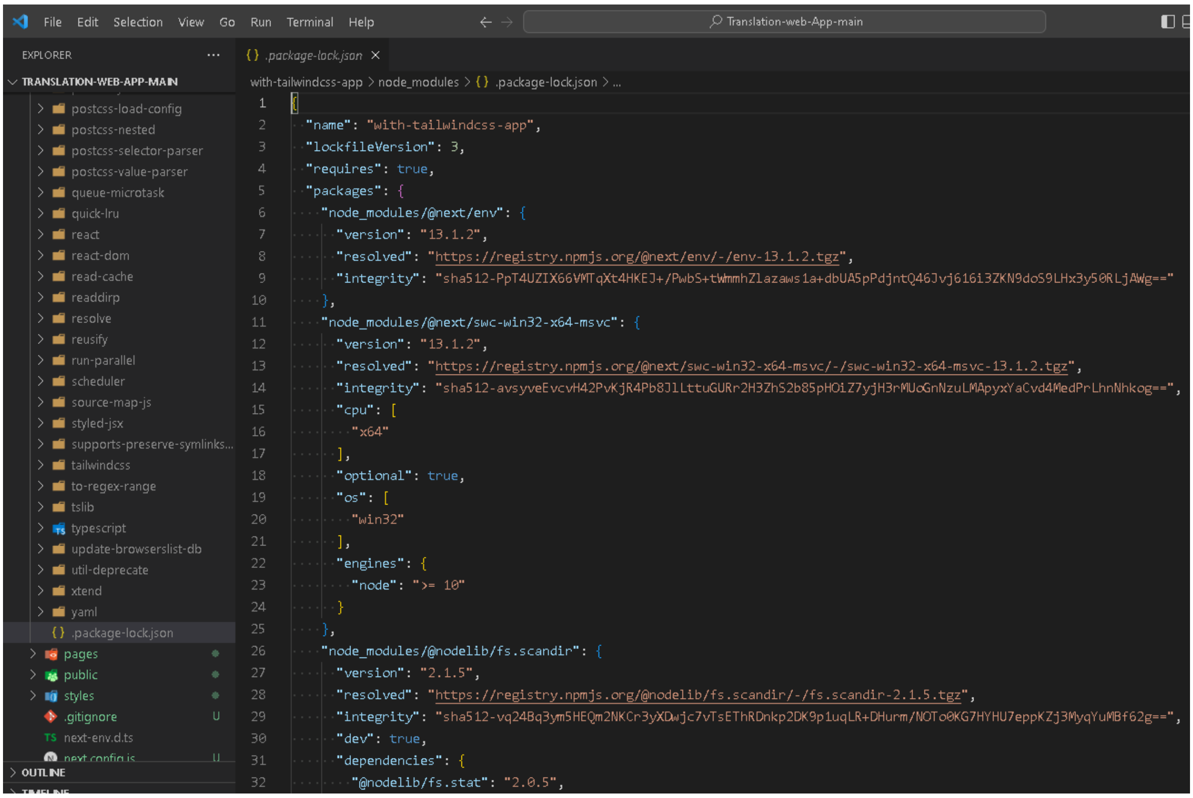
1. Simple and clear interface
2. Possibility of specific selection of models and data sets
3. Used and supported by free resources
4. Easy access to data with Prisma
5. High degree of protection

- Cons:

1. It is necessary to train the network a lot for quality learning
2. Since the application is hosted locally, it is impossible to train in large groups until there is compatible access.

So, the developed application has great advantages and few disadvantages that can be easily fixed, the application is ready to be made and tested.

Timely detection of cyber-threats is important, but in our case we have a high degree of protection of modules, files and the application, and therefore the probability of a cyber-threat is almost 0%. For each module and library, in our case, the acquisition proceeds as follows:



```
{
  "name": "with-tailwindcss-app",
  "lockfileVersion": 3,
  "requires": true,
  "packages": {
    "node_modules/@next/env": {
      "version": "13.1.2",
      "resolved": "https://registry.npmjs.org/@next/env/-/env-13.1.2.tgz",
      "integrity": "sha512-PpT4UZIX66VMTqXt4HKEJ+/PwbS+tWmmhZlazaws1a+dbUA5pPdjntQ46Jvj616i3ZKN9doS9LHx3y50RLjAWg=="
    },
    "node_modules/@next/swc-win32-x64-msvc": {
      "version": "13.1.2",
      "resolved": "https://registry.npmjs.org/@next/swc-win32-x64-msvc/-/swc-win32-x64-msvc-13.1.2.tgz",
      "integrity": "sha512-avsyveEvcvH42PvKjR4Pb8JlLttuGURr2H3ZhS2b85pHOiZ7yjH3rMUoGnNzuLMApyxYaCvd4MedPrLhnNhkog==",
      "cpu": [
        "x64"
      ],
      "optional": true,
      "os": [
        "win32"
      ],
      "engines": {
        "node": ">= 10"
      }
    },
    "node_modules/@nodelib/fs.scandir": {
      "version": "2.1.5",
      "resolved": "https://registry.npmjs.org/@nodelib/fs.scandir/-/fs.scandir-2.1.5.tgz",
      "integrity": "sha512-vq24Bq3ym5HEQm2NKCr3yXDwjc7vTsEThRDnkp2DK9p1uqLR+DHurm/NOTo0KG7HYHU7eppKZj3MyqYuMBf62g==",
      "dev": true,
      "dependencies": {
        "@nodelib/fs.stat": "2.0.5",
```

Figure 28: Receiving packets

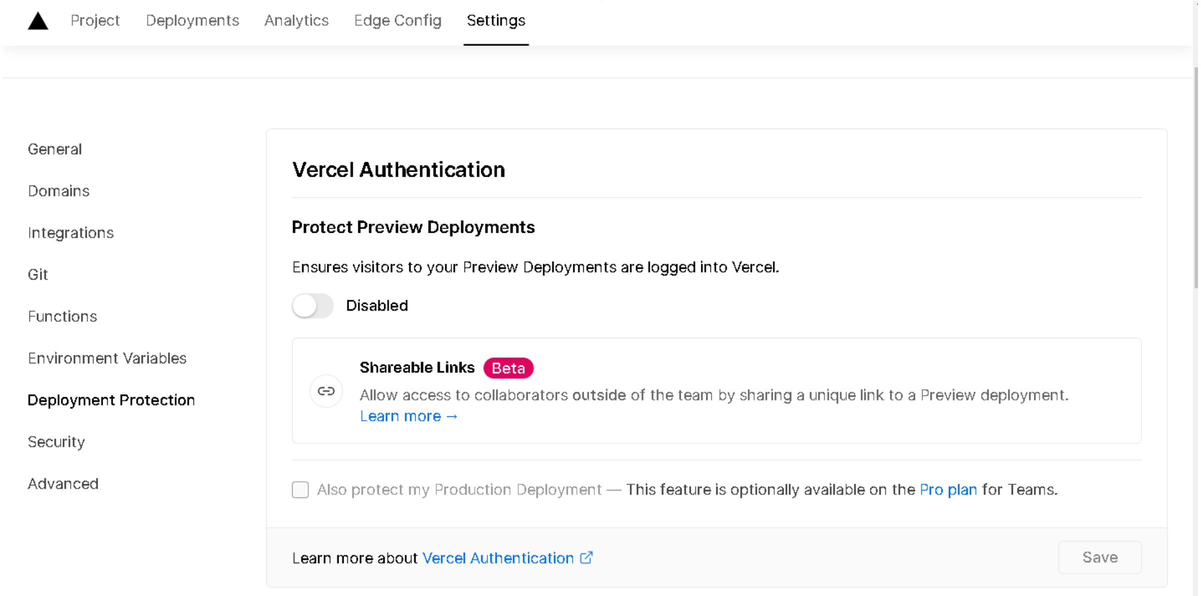
Let's look at an example:

```
    "name": "with-tailwindcss-app",
    "lockfileVersion": 3,
    "requires": true,
    "packages": {
      "node_modules/@next/env": {
        "version": "13.1.2",
        "resolved": "https://registry.npmjs.org/@next/env/-/env-
13.1.2.tgz",
        "integrity": "sha512-
PpT4UZIX66VMTqXt4HKEJ+/PwbS+tWmmhZlazaws1a+dbUA5pPdjntQ46Jvj616i3ZKN9doS9L
Hx3y50RLjAWg=="
      },
```

Figure 29: Example figure

We see: name, version of the closed file (set when starting the project and Node assembly), requires - a parameter that determines whether this module is needed, version, resolved - where we get it from, and integrity - integrity\protection of the file. All modules in the project are protected by a SHA-512 key. SHA-512 is a hashing algorithm that performs the function of hashing some data given to it.

Hashing algorithms are used in many things, such as Internet security, digital certificates, and even blockchains. It is part of a group of hashing algorithms called SHA-2, which also includes SHA-256, which is used on the Bitcoin blockchain for hashing.



Vercel Authentication

Protect Preview Deployments

Ensures visitors to your Preview Deployments are logged into Vercel.

Disabled

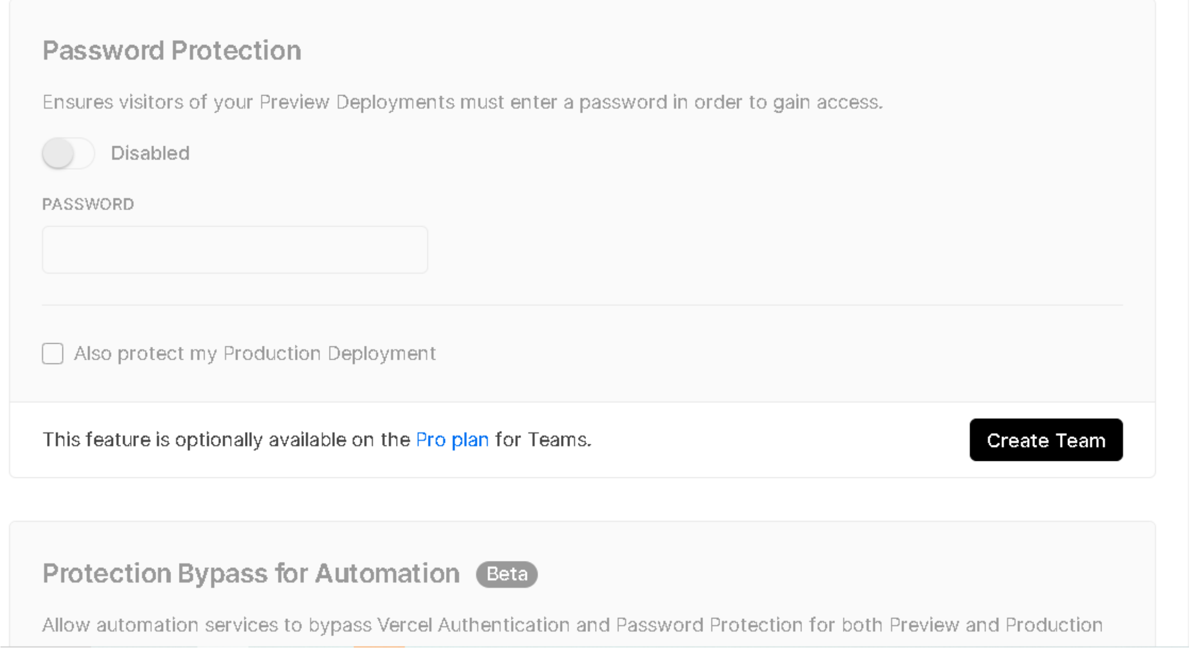
Shareable Links Beta

Allow access to collaborators outside of the team by sharing a unique link to a Preview deployment.
Learn more →

□ Also protect my Production Deployment — This feature is optionally available on the Pro plan for Teams.

Learn more about Vercel Authentication

Figure 30: Protection with Versel



Password Protection

Ensures visitors of your Preview Deployments must enter a password in order to gain access.

Disabled

PASSWORD

□ Also protect my Production Deployment

This feature is optionally available on the Pro plan for Teams.

Protection Bypass for Automation Beta

Allow automation services to bypass Vercel Authentication and Password Protection for both Preview and Production

Figure 31: Protection with Versel

The project is protected by SHA-512 keys and Versel authentication mechanisms, and in order to start working with Versel, you need to have a Git Hub account that also protects your repositories, based on this we conclude that the probability of cyber-attacks is minimal.

Training a network is the process of optimizing its weights and biases to reduce the prediction error on the training data. At the same time, the network "learns" to recognize regularities in the data and form its own representations of objects. When comparing a trained and untrained network, the following advantages of training can be noted:

- Better prediction accuracy: Training the network allows for better prediction accuracy on test data compared to an untrained network. For example, if we consider the problem of image classification, training the network can reduce the error of image classification from 30% to 5%.

- Generalization ability: Training a network allows you to increase its generalization ability. This means that the network can correctly classify new data that it has not seen before. For example, if a network has been trained to classify images from two classes (cat and dog), it can correctly classify images from three classes (cat, dog, and rabbit) if it has been trained on a large enough number of images.
- Optimization of the structure and parameters: Training the network allows you to optimize its structure and parameters to obtain better prediction accuracy. For example, you can change the number of layers and neurons in each layer, choose different activation functions and loss functions, change the batch size for training, use different optimization methods such as Adam, SGD, RMSProp, etc. Various regularization techniques can also be applied, such as Dropout, L1 and L2 regularization, as well as data augmentation to avoid overtraining.

Now let's compare our trained network and untrained network:

Table 1
Comparison of trained and untrained network

Name	An untrained network	Trained network
Activation function	Sigmoid	Relu
Number of layers	3	5
Number of neurons	50	100
Number of epochs	-	100
Loss function	root mean square	root mean square
Loss	34.2	3.3
Execution time	-	5 hours
Work time	4s	100ms
Recognition accuracy	64%	96%
Learning rate	0.01	0.001
Dropout rate	0.2	0.5

A trained network is better than an untrained network for several reasons. First, training the network allowed it to improve its performance on test data. For example, after training the network, the mean squared error decreased from 34.2 to 3.3. Also, training the network made it possible to reduce overtraining and improve its ability to generalize to new data. The second reason is that training the network allowed it to optimize its parameters and structure to enable more efficient pattern recognition in the input data. For example, the number of layers and neurons was increased in the trained network, which allowed it to reproduce more complex dependencies in the input data. The third reason is that training the network allowed us to choose a more efficient activation function (in this case, the relay), which allowed us to improve the convergence speed of the network and reduce the training time. So, training the network made it possible to improve its results on test data, increase its ability to generalize, optimize its structure and parameters, choose a more effective activation function, and also reduce the risk of overtraining.

6. Conclusions

The information system for remote learning of foreign languages with adaptation to the user based on machine learning has been created. Moreover, features of use, adaptive learning systems, existing methods, profile modelling standards and architecture have been considered and analysed. Also, the research task has been formulated. The systematic analysis of the research object has been performed, namely, the main provisions of fuzzy logic, its architecture, advantages and disadvantages have been considered. Thus, the model of the process of forming user parameters in adaptive learning systems, the tree of goals, have been analysed. The main languages and software tools used for development have been defined. So, Java Script has been chosen as the main language for development and the main software for work such as VS code (as a code editor), Versel (as a web application for testing) have been analysed. The formation of the user profile and the choice of architecture have been reviewed.

Practical implementation of the designed software product has been carried out. Also, the initialization process of the data set for neural network training has been described, the architecture and models have been explored and visualized, data preparation has been carried out, model construction has been implemented. So, the developed application has several minor disadvantages that can be easily corrected, and many advantages that are not found in analogues. A trained network is better than an untrained network for several reasons. First, training the network allowed it to improve its performance on test data. For example, after training the network, the mean squared error decreased from 34.2 to 3.3. So, training the network made it possible to improve its results on test data, increase its ability to generalize, optimize its structure and parameters, choose a more effective activation function, and also reduce the risk of overtraining.

7. References

[1] A. Esteva, et al., A guide to deep learning in healthcare, Nature medicine 25(1) (2019) 24-29.

[2] A. Nistor, E. Zadobrischi, The influence of fake news on social media: analysis and verification of web content during the COVID-19 pandemic by advanced machine learning methods and natural language processing, Sustainability 14(17) (2022) 10466.

[3] P. Li, J. Li, G. Wang, Application of convolutional neural network in natural language processing, in Proceedings of IEEE 15th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), 2018, December, pp. 120-122.

[4] L. Deng, Y. Liu, Deep learning in natural language processing, Springer, Singapore, 2018.

[5] Y. Kang, Z. Cai, C. W. Tan, Q. Huang, H. Liu, Natural language processing (NLP) in management research: A literature review, Journal of Management Analytics 7(2) (2020) 139-172.

[6] D. Rothman, A. Gulli, Transformers for Natural Language Processing: Build, train, and fine-tune deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, and GPT-3. Packt Publishing Ltd. ,2022.

[7] V. Lytvyn, P. Pukach, V. Vysotska, M. Vovk, N. Kholodna, Identification and Correction of Grammatical Errors in Ukrainian Texts Based on Machine Learning Technology, Mathematics 11(4) (2023) 904.

[8] A. Nistor, E. Zadobrischi, The influence of fake news on social media: analysis and verification of web content during the COVID-19 pandemic by advanced machine learning methods and natural language processing, Sustainability 14(17) (2022) 10466.

[9] W. E. Zhang, Q. Z. Sheng, A. Alhazmi, C. Li, Adversarial attacks on deep-learning models in natural language processing: A survey, ACM Transactions on Intelligent Systems and Technology (TIST) 11(3) (2020) 1-41.

[10] How to Digitize Texts with Open-Source Command-Line Optical Character Recognition (OCR) Software. URL: https://hdw.artsci.wustl.edu/articles/154.

[11] A. Karpathy, et al., Convolutional Neural Networks for Visual Recognition. URL: http://www.cs231n.stanford.edu.

[12] The Mnist Database of handwritten digits. URL: http://yann.lecun.com/exdb/mnist/.

[13] Kay A. Tesseract: Open-Source Optical Character Recognition Engine. URL: http://www.linuxjournal.com/article/9676.

[14] Nielsen MA Neural Networks and Deep Learning. URL: http://www.neuralnetworksanddeeplearning.com.

[15] OCRopy: Python-based tools for document analysis and OCR. https://github.com/tmbdev/ocropy

[16] S. Theodoridis, K., Koutroumbas, Pattern Recognition. New York: Elsevier Science. URL: https://books.google.com/books?id=QgD-3Tcj8DkC.

[17] U. Baumann, M. Shelley, L. Murphy, New challenges, the role of the tutor in the teaching of languages at a distance. Distances et Savoirs 6(3) (2009) 365–392.

[18] J.-C. Bertin, P. Grave, J.-P. Narcy-Combes, Second language distance learning and teaching: Theoretical perspectives and didactic ergonomics. Hershey, PA: Information Science Reference, 2010.

[19] A. Comas-Quinn, B. de los Arcos, R. Mardomingo, Virtual learning environments (VLEs) for distance language learning: Shifting tutor roles in a contested space for interaction, Computer Assisted Language Learning 25(2) (2012) 129-143.

[20] M. Grgurović, C. A. Chapelle, M. C. Shelley, A meta-analysis of effectiveness studies on computer technology-supported language learning, ReCALL 25(2) (2013) 1-32.

[21] M. S. Andrade, E. L. Bunker, A model for self-regulated distance language learning. Distance Education 30(1) (2009) 47–61.

[22] A. Kukulska-Hulme, L. Shield, An overview of mobile assisted language learning: From content delivery to supported collaboration and interaction, ReCALL 20(3) (2008) 271-289.

[23] K. Kostolányová, R. Juřičková, I. Šimonová, P. Poulová. Flexible hybrid learning: comparative study, in Proceeding of Hybrid learning. Innovation in educational practices. 8th international conference. Lecture Notes in Computer Science 9167 (2015) 70-81.

[24] G. Fulcher, Practical language testing. London: Hodder Education, 2010.

[25] C. M. Chen, C. J. Chung, Personalized mobile English vocabulary learning system based on item response theory and learning memory cycle. Computers & Education 51(2) (2008) 624-645.

[26] M. Horbova, V. Andrunyk, L. Chyrun, Virtual reality platform using ml for teaching children with special needs, CEUR Workshop Proceedings 2631 (2020) 209-220.

[27] K. Supruniuk, V. Andrunyk, L. Chyrun, AR interface for teaching students with special needs, CEUR Workshop Proceedings 2604 (2020) 1295-1308.

[28] V. Andrunyk, V. Pasichnyk, N. Antonyuk, T. Shestakevych, A Complex System for Teaching Students with Autism: The Concept of Analysis. Formation of IT Teaching Complex, Advances in Intelligent Systems and Computing 1080 (2020) 721-733.

[29] A. Badan, N. Onishchenko, O. Zeniakin, O. Yanholenko, Online Communication Simulating Spaces for Teaching Effective Foreign Language Communication, CEUR Workshop Proceedings, Vol-3387 (2023) 180-201.

[30] 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 Vol-3312 (2022) 212-225.

[31] A. Badan, N. Onishchenko, O. Zeniakin, Digital Technologies for Communication Simulation in Foreign Language Learning under Pandemic, CEUR Workshop Proceedings Vol-3171 (2022) 1160-1180.

[32] O. Krupii, K. Kasian, A Neural Network-Based Study of the Performance of a Developed Foreign Language Teaching System, CEUR Workshop Proceedings Vol-2870 (2021) 191-205.

[33] A. Badan, N. Onishchenko, Multimedia Technologies in Foreign Language Learning under Pandemic, CEUR Workshop Proceedings Vol-2870 (2021) 642-656.

[34] M. Hrendus, V. Andrunyk, M. Yavir, Y. Ryshkovets, A. Khudyi, V. Hryhorovych, M. Korobchynskyi, Developing an Intelligent Online Learning System for Foreign Language Vocabulary Training Based on Gamification. In: CEUR workshop proceedings Vol-2604 (2020) 1075-1101.

[35] V. Lytvyn, V. Danylyk, M. Bublyk, L. Chyrun, V. Panasyuk, O. Korolenko, The lexical innovations identification in English-languagee eurointegration discourse for the goods analysis by comments in e-commerce resources, in: Proceedings of IEEE 16th International conference on Computer science and information technologies, Lviv, 2021, pp. 85–97. doi: 10.1109/CSIT52700.2021.9648594.

[36] N. Shakhovska, O. Vovk, R. Hasko, Y. Kryvenchuk, The method of big data processing for distance educational system, Advances in Intelligent Systems and Computing 689 (2018) 461-473.

[37] V. Andrunyk, T. Shestakevych, M. Kryvoshyya, Choosing an Educational Application for Children with ASD, CEUR Workshop Proceedings Vol-3171 (2022) 642-652.

[38] S. Chupakhina, N. Pasieka, M. Matishak, M. Pasieka, Y. Romanyshyn, Mathematical Models of Group Dynamics When Working in Teams of Developers of Training Distance Courses, CEUR Workshop Proceedings Vol-2917 (2021) 51-61.

[39] S. Lienkov, S. Gakhovych, I. Tolok, G. Zhyrov, V. Bakhvalov, An Option of Building the Distance Learning System with Artificial Intelligence Elements, CEUR Workshop Proceedings Vol-2870 (2021) 1194-1203.

[40] D. Malikin, I. Kyrychenko, Research of Methods for Practical Educational Tasks Generation Based on Various Difficulty Levels, CEUR Workshop Proceedings Vol-3171 (2022) 1030-1042.

[41] R. Yurynets, Z. Yurynets, N. Danylevych, Innovative Methods of Assessing the Academic Success of Students in Higher Education Institutions, CEUR Workshop Proceedings Vol-3171 (2022) 1297-1307.

[42] N. Pasieka, N. Lysenko, O. Lysenko, V. Sheketa, M. Pasieka, M. Varvaruk, Activating the Process of Educational Services Using Independent Computing Resources to Manage and Monitor the Quality of Learning, CEUR Workshop Proceedings Vol-2917 (2021) 62-74.

[43] V. Lytvynenko, N. Savina, M. Voronenko, N. Doroschuk, S. Smailova, O. Boskin, T. Kravchenko. Development, Validation and Testing of the Bayesian Network of Educational Institutions Financing, in: The crossing point of Intelligent Data Acquisition & Advanced Computing Systems and East & West Scientists (IDAACS-2019), September 18-21, Metz, France, pp. 412-418.

[44] O. Pronina, O. Piatykop, The Decision Support System Education Career Choice Using Fuzzy Model, CEUR Workshop Proceedings Vol-2870 (2021) 1204-1214.

[45] R. Yurynets, Z. Yurynets, M. Denysenko, I. Myshchyshyn, A. Pekhnyk, The Influence of Educational Competencies of the Staff on the Efficiency of Hotel Companies in the Tourism Sector, CEUR Workshop Proceedings Vol-2870 (2021) 1225-1237.

[46] L. Halkiv, O. Karyy, I. Kulyniak, Y. Kis, A. Tsapulych, The National System of Higher Education and Government Procurement for Its Services as Activators of the Development of IT Entrepreneurship, CEUR Workshop Proceedings Vol-2870 (2021) 1338-1349.

[47] N. Pasieka, Y. Romanyshyn, S. Chupakhina, M. Oliinyk, M. Pasieka, Activation of the Educational Process by Changing the Curriculum in Higher School, CEUR Workshop Proceedings Vol-2870 (2021) 1350-1364.

[48] Z. Myna, T. Bilushchak, Social Networks as Tools to Promote the Majors of Higher Education Institutions During the Pandemic, CEUR Workshop Proceedings Vol-2870 (2021) 1365-1375.

[49] A. Taran, Information-retrieval System "Base of the World Slavic Linguistics (iSybislaw)" in Language Education, CEUR workshop proceedings Vol-2604 (2020) 590-599.

[50] O. Cherednichenko, O. Yanholenko, Information Technology of Web-Monitoring and Measurement of Outcomes in Higher Education Establishment, in proceedings of 7th SIGSAND/PLAIS EuroSymposium, Springer, 232 (2015) 103-116.

[51] A. Bomba, M. Nazaruk, N. Kunanets, V. Pasichnyk, Modeling the Dynamics of Knowledge Potential of Agents in the Educational Social and Communication Environment, Advances in Intelligent Systems and Computing 1080 (2020) 17-24.

[52] R. Holoshchuk, V. Pasichnyk, N. Kunanets, N. Veretennikova, Information Modeling of Dual Education in the Field of IT, Advances in Intelligent Systems and Computing 1080 (2020) 637-646.

[53] M. Konyk, V. Vysotska, S. Goloshchuk, R. Holoshchuk, S. Chyrun, I. Budz, Technology of Ukrainian-English Machine Translation Based on Recursive Neural Network as LSTM, CEUR Workshop Proceedings Vol-3387 (2023) 357-370.

[54] M. Garcarz, Legal Language Translation: Theory behind the Practice, CEUR Workshop Proceedings Vol-3171 (2022) 2-2.

[55] N. Hrytsiv, I. Bekhta, M. Tkachivska, V. Byalyk, Sylvia Plath’s I felt-Narrative Label of The Bell Jar in Ukrainian Translation: Tagging Textness Features, CEUR Workshop Proceedings Vol-3171 (2022) 240-255.

[56] M. Bekhta-Hamanchuk, H. Oleksiv, T. Shestakevych, Y. Shyika, Quantitative Parameters of J. London's Short Stories Collection “Children of the Frost” and its Translation, CEUR Workshop Proceedings Vol-3171 (2022) 697-710.

[57] K. Mandziy, U. Yurlova, M. Dilai, English-Ukrainian Parallel Corpus of IT Texts: Application in Translation Studies, CEUR Workshop Proceedings Vol-3171 (2022) 724-736.

[58] N. Hrytsiv, T. Shestakevych, J. Shyyka, Corpus Technologies in Translation Studies: Fiction as Document, CEUR Workshop Proceedings Vol-2917 (2021) 327-343.

[59] A. Lutskiv, R. Lutsyshyn, Corpus-Based Translation Automation of Adaptable Corpus Translation Module, CEUR Workshop Proceedings Vol-2870 (2021) 511-527.

[60] A. Kopp, D. Orlovskyi, S. Orekhov, An Approach and Software Prototype for Translation of Natural Language Business Rules into Database Structure, CEUR Workshop Proceedings Vol-2870 (2021) 1274-1291.

[61] T. Anokhina, I. Kobyakova, S. Shvachko, Going parallel: using earlier translations as background for facilitating re-translation technique, CEUR workshop proceedings Vol-2604 (2020) 249-258.

[62] M.-A. Lefer, N. Grabar, Super-creative and overbureaucratic: A cross-genre corpusbased study on the use and translation of evaluative prefixation in ted talks and EU parliamentary debates, Across Languages and Cultures 16(2) (2015) 187-208.

[63] S. Tetiana, The method of education format ascertaining in program system of inclusive education support, in: proceedings of International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 2017, pp. 279-283.

[64] V. Pasichnyk, T. Shestakevych, The application of multivariate data analysis technology to support inclusive education, in: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 2015, pp. 88-90.

[65] T. Shestakevych, V. Pasichnyk, N. Kunanets, Information and technology support of inclusive education in Ukraine, Advances in Intelligent Systems and Computing 754 (2019) 746-758.

[66] T. Shestakevych, V. Pasichnyk, N. Kunanets, M. Medykovskyy, N. Antonyuk, The content web-accessibility of information and technology support in a complex system of educational and social inclusion, in: International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 1, 2018, pp. 27-31.

[67] V. Pasichnyk, T. Shestakevych, N. Kunanets, V. Andrunyk, Analysis of completeness, diversity and ergonomics of information online resources of diagnostic and correction facilities in Ukraine, CEUR Workshop Proceedings 2105 (2019) 193-208.

[68] T. Shestakevych, V. Pasichnyk, M. Nazaruk, M. Medykovskiy, N. Antonyuk, Web-Products, Actual for Inclusive School Graduates: Evaluating the Accessibility, Advances in Intelligent Systems and Computing 871 (2019) 350-363.

[69] Y. Bobalo, P. Stakhiv, N. Shakhovska, The system of remote Education Resource Center elements development, in: proceedings of International Conference Computational Problems of Electrical Engineering, CPEE, 2016, 7738729.