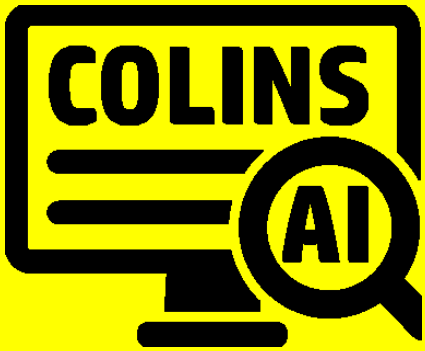
Natalia Sharonova
Vasyl Lytvyn
Olga Cherednichenko
Natalia Borysova
Yevhen Kupriianov
Olga Kanishcheva
Thierry Hamon
Natalia Grabar
Victoria Vysotska
Agnieszka Kowalska-Styczen
Izabela Jonek-Kowalska
(Eds.)



COMPUTATIONAL LINGUISTICS AND INTELLIGENT SYSTEMS

Proceedings of the 5th International Conference, COLINS-2021.
Volume II: Workshop

Kharkiv, Ukraine
April, 2021, 22-23

N. Sharonova, V. Lytvyn, O. Cherednichenko, N. Borysova, Y. Kupriianov, O. Kanishcheva, T. Hamon, N. Grabar, V. Vysotska, A. Kowalska-Styczen, I. Jonek-Kowalska (Eds.): Computational Linguistics and Intelligent Systems. Proceedings of the 5th International Conference on COLINS 2021. Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013, colins.in.ua, online [1]

The 5th International Conference on COLINS 2021 is organized by:

- National Technical University «Kharkiv Polytechnic Institute», Ukraine
- Lviv Polytechnic National University, Ukraine
- Institut Galilée of Université Paris 13, France
- Politechnika Śląska, Poland
- Ukrainian Scientific and Educational IT Society, Ukraine

This volume represents the proceedings of the Workshop Conference of the 5th International Conference on Computational Linguistics and Intelligent Systems, held in Kharkiv, Ukraine, in April 2021. It comprises 18 contributed papers. The volume is organized in two parts. Part I contains the Poster papers of the Workshop. Part II contains the contributions to the Student section of the Workshop Conference.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: nvsharonova@ukr.net (N. Sharonova), Vasyl.V.Lytvyn@lpnu.ua (V. Lytvyn), olha.cherednichenko@gmail.com (O. Cherednichenko), borysova.n.v@gmail.com (N. Borysova), eugeniokuprianov@gmail.com (Y. Kupriianov), kanichshevaolga@gmail.com (O. Kanishcheva), hamon@limsi.fr (T. Hamon), natalia.grabar@univ-lille3.fr (N. Grabar), victoria.a.vysotska@lpnu.ua (V. Vysotska), Agnieszka.Kowalska-Styczen@polsl.pl (A. Kowalska-Styczen), Izabela.Jonek-Kowalska@polsl.pl (I. Jonek-Kowalska)
ORCID: 0000-0002-7555-1507 (N. Sharonova), 0000-0002-9676-0180 (V. Lytvyn), 0000-0002-9391-5220 (O. Cherednichenko), 0000-0002-0801-1789 (Y. Kupriianov), 0000-0002-9035-1765 (O. Kanishcheva), 0000-0002-1521-4875 (T. Hamon), 0000-0002-0237-4554 (N. Grabar), 0000-0001-6417-3689 (V. Vysotska), 0000-0002-7404-9638 (A. Kowalska-Styczen), 0000-0002-4006-4362 (I. Jonek-Kowalska)

Preface

It is our pleasure to present you the proceedings of the Workshop Conference of COLINS-2021, the fifth edition of the International Conference on Computational Linguistics and Intelligent Systems, held in Kharkiv (Ukraine) on April 22-23, 2021.

The main purpose of the CoLInS conference is a discussion of the recent research results in all areas of Natural Language Processing and Intelligent Systems Development.

The conference is soliciting literature review, survey and research papers comments including, whilst not limited to, the following areas of interest:

- machine learning;
- discourse analysis;
- segmentation, tagging, and parsing;
- speech recognition;
- sentiment analysis and opinion mining;
- text categorization and topic modeling;
- text mining;
- information retrieval;
- artificial intelligent;
- information extraction;
- statistical language analysis;
- text summarization;
- data mining and data analysis;
- computer lexicography;
- social network analysis;
- question answering systems;
- web and social media;
- NLP applications;
- machine translation;
- intelligent text processing systems;
- memory systems and computer-aided translation tools;
- computer-aided language learning;
- knowledge representation;
- knowledge-oriented systems;
- natural language processing;
- ontologies and ontology-based systems;
- corpus linguistics.

The language of COLINS Conference is English.

The conference took the form of oral presentation by invited keynote speakers plus presentations of peer-reviewed individual papers. The papers were distributed among 90 external reviewers from France, Germany, India, Moldova, Poland, Serbia, Slovenia and Ukraine. The total number of reviews is 645. To take more correct decision regarding the acceptance or rejection the papers got 3-6 reviews. The peer review statistics is as follows: 110 papers (3 reviews), 46 papers (4 reviews), 19 papers (5 reviews), and 6 papers (6 reviews).

The conference gathered participants from different countries including Azerbaijan, Germany, India, Kazakhstan, Kuwait, Morocco, Poland, Russia, Serbia, Slovenia, Turkey, Uganda, Ukraine and United Kingdom.

There was also an exhibition area for poster and demo sessions. A Student section of the conference for students and PhD students ran in parallel to the main conference. These papers and extended abstracts were published in this Volume II of COLINS 2021 proceedings.

The conference would not have been possible without the support of many people. First of all, we would like to thank all the authors who submitted papers to COLINS 2021 and thus demonstrated their interest in the research problems within our scope. We are very grateful to the members of our Program Committee for providing timely and thorough reviews and, also, for being cooperative in doing additional review work. We would like to thank the Organizing Committee of the conference whose devotion and efficiency made this instance of COLINS a very interesting and effective scientific forum.

April, 2021

Natalia Sharonova
Vasyl Lytvyn
Olga Cherednichenko
Natalia Borysova
Yevhen Kupriianov
Olga Kanishcheva
Thierry Hamon
Natalia Grabar
Victoria Vysotska
Agnieszka Kowalska-Styczen
Izabela Jonek-Kowalska

Committees

General Chair

Natalia Sharonova, National Technical University "KhPI", Ukraine

Co-Chair

Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine
Mykhailo Godlevskyi, National Technical University "KhPI", Ukraine

Steering Committee

Natalia Sharonova, National Technical University "KhPI", Ukraine
Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine
Olga Cherednichenko, National Technical University "KhPI", Ukraine
Natalia Borysova, National Technical University "KhPI", Ukraine
Yevhen Kupriianov, National Technical University "KhPI", Ukraine
Olga Kanishcheva, National Technical University "KhPI", Ukraine
Thierry Hamon, LIMSI-CNRS & Université Paris 13, France
Natalia Grabar, CNRS UMR 8163 STL, France
Victoria Vysotska, Lviv Polytechnic National University, Ukraine
Agnieszka Kowalska-Styczen, Silesian University of Technology, Poland
Izabela Jonek-Kowalska, Silesian University of Technology, Poland

Program Chairs

Yevhen Kupriianov, Lviv Polytechnic National University, Ukraine
Olga Cherednichenko, National Technical University "KhPI", Ukraine
Victoria Vysotska, Lviv Polytechnic National University, Ukraine

Proceedings Chair

Victoria Vysotska, Lviv Polytechnic National University, Ukraine
Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine

Presentations Chair

Natalia Sharonova, National Technical University "KhPI", Ukraine
Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine
Nina Khairova, National Technical University "KhPI", Ukraine
Olena Levchenko, Lviv Polytechnic National University, Ukraine
Olga Cherednichenko, National Technical University "KhPI", Ukraine
Yevhen Kupriianov, National Technical University "KhPI", Ukraine
Olga Kanishcheva, National Technical University "KhPI", Ukraine
Dmytro Dosyn, Lviv Polytechnic National University, Ukraine
Tetiana Shestakevych, Lviv Polytechnic National University, Ukraine
Zoia Kochuieva, National Technical University "KhPI", Ukraine
Svitlana Petrasova, National Technical University "KhPI", Ukraine
Oksana Ivashchenko, National Technical University "KhPI", Ukraine
Anastasiia Kolesnyk, National Technical University "KhPI", Ukraine
Natalia Borysova, National Technical University "KhPI", Ukraine
Karina Melnyk, National Technical University "KhPI", Ukraine
Nataliia Hrytsiv, Lviv Polytechnic National University, Ukraine

Poster and Demo Chairs

Yevhen Kupriianov, National Technical University "KhPI", Ukraine
Natalia Borysova, National Technical University "KhPI", Ukraine
Svitlana Petrasova, National Technical University "KhPI", Ukraine

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

PhD Symposium Chairs

Olga Cherednichenko, National Technical University "KhPI", Ukraine
Yevhen Kupriianov, National Technical University "KhPI", Ukraine
Svitlana Petrasova, National Technical University "KhPI", Ukraine

IT Talks Chairs

Olga Cherednichenko, National Technical University "KhPI", Ukraine
Olga Kanishcheva, National Technical University "KhPI", Ukraine

Local Organization Chairs

Olga Cherednichenko, National Technical University "KhPI", Ukraine
Olga Kanishcheva, National Technical University "KhPI", Ukraine
Yevhen Kupriianov, National Technical University "KhPI", Ukraine
Natalia Borysova, National Technical University "KhPI", Ukraine
Svitlana Petrasova, National Technical University "KhPI", Ukraine

Publicity Chair

Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine
Victoria Vysotska, Lviv Polytechnic National University, Ukraine
Khrystyna Mykich, Lviv Polytechnic National University, Ukraine

Web Chair

Olga Kanishcheva, National Technical University "KhPI", Ukraine
Victoria Vysotska, Lviv Polytechnic National University, Ukraine
Yevhen Kupriianov, National Technical University "KhPI", Ukraine

Program Committees

MAIN Conference COLINS 2021

Agnieszka Kowalska-Styczen, Silesian University of Technology, Poland
Aleksandr Gozhyi, Petro Mohyla Black Sea National University, Ukraine
Anatoly Sachenko, Ternopil National Economic University, Ukraine
Andrii Berko, Lviv Polytechnic National University, Ukraine
Ankur Singh Bist, Towards Blockchain, India
Bohdan Rusyn, Humanities of Radom, Radom, Poland
Borut Werber, University of Maribor, Slovenia
Chang Shu, NorthWest university of political science and law, China
Dmitry Lande, Institut for Information Recording of NAS of Ukraine, Ukraine
Dmytro Peleshko, GeoGuard, Vancouver, British Columbia
Evelin Krmac, University of Ljubljana, Slovenia
Fadila Bentayeb, ERIC Laboratory, University of Lyon 2, France
Galia Angelova, Bulgarian Academy of Sciences, Bulgaria
Irina Ivasenko, Karpenko Physico-Mechanical Institute of the NAS of Ukraine, Ukraine
Iryna Yevseyeva, Newcastle University, England)
Izabela Jonek-Kowalska, Silesian University of Technology, Poland
Jean-Hugues Chauchat, Université Lumière Lyon 2, France
Jun Su, Hubei University of Technology, China
Jörg Rainer Noennig, Technische Universität Dresden, Germany
Klaus ten Hagen, University of Applied Science Zittau/Goerlitz, Germany
Lidia Pivovarova, University of Helsinki, Finland
Lyubomyr Chyrun, Ivan Franko National University of Lviv, Ukraine
Maksym Korobchynskyi, Military-Diplomatic Academy named after Eugene Bereznyak, Ukraine
Manik Sharma, DAV University, India

Posters and Demonstrations Track

Keynote Speakers

Olena Levchenko, Lviv Polytechnic National University, Ukraine
Oleh Tyshchenko, Lviv Polytechnic National University, Ukraine
Marianna Dilai, Lviv Polytechnic National University, Ukraine
Maria Shvedova, Kyiv National University, Ukraine
Ruprecht von Waldenfels, Friedrich-Schiller-Universität Jena, Germany
Lubomyr Demkiv, Lviv Polytechnic National University, Ukraine
Dmytro Chumachenko, National Aerospace University "Kharkiv Aviation Institute", Ukraine
Sergey Yakovlev, National Aerospace University "Kharkiv Aviation Institute", Ukraine
Abdullah Al Mutawa, Kuwait University, Kuwait

Additional Reviewers

Alexander Shportko, Academician Stepan Demianchuk International University of Economics and Humanitie, Ukraine
Nataliya Ryabova, Kharkiv National University of Radio Electronics, Ukraine
Grygoriy Zholtkevych, V.N. Karazin Kharkiv National University, Ukraine
Vyacheslav Kharchenko, National Aerospace University -'Kharkiv Aviation Institute', Ukraine
Petro Pukach, Lviv Polytechnic National University, Ukraine
Nataliia Kunanets, Lviv Polytechnic National University, Ukraine
Petro Kravets, Lviv Polytechnic National University, Ukraine
Anatolii Vysotskyi, Anat Company, Ukraine
Oleh Veres, Lviv Polytechnic National University, Ukraine
Solomiia Albota, Lviv Polytechnic National University, Ukraine
Marianna Dilai, Lviv Polytechnic National University, Ukraine
Iryna Khomytska, Lviv Polytechnic National University, Ukraine
Andrii Vasyliuk, Lviv Polytechnic National University, Ukraine
Rostyslav Yurynets, Lviv Polytechnic National University, Ukraine
Taras Basyuk, Lviv Polytechnic National University, Ukraine
Tetiana Shestakevych, Lviv Polytechnic National University, Ukraine
Vasyl Andrunyk, Lviv Polytechnic National University, Ukraine
Vasyl Lenko, Lviv Polytechnic National University, Ukraine
Viktor Grigorovich, Ivan Franko Drohobych State Pedagogical University, Ukraine
Alina Petrushka, Lviv Polytechnic National University, Ukraine
Achyut Shankar, Amity University, Uttar Pradesh, India
Gennady Shepelev, Federal Research Center 'Computer Science and Control' of Russian Academy of Sciences, Russian Federation
Oleksandr Berezko, Lviv Polytechnic National University, Ukraine
Anastasiia Chuchvara, Centre of Mathematical Modelling of Pidstryhach Institute of Applied Problems of Mechanics and Mathematics of National Academy of Sciences of Ukraine
Yurii Bilushchak, Lviv Polytechnic National University, Ukraine
Zoia Kochuieva, National Technical University "Kharkiv Polytechnic Institute" , Ukraine
Mariia Voronenko, Kherson National Technical University, Ukraine
Maryna Vovk, National Technical University "Kharkiv Polytechnic Institute, Ukraine"
Natalia Borysova, National Technical University "Kharkiv Polytechnic Institute" , Ukraine
Mykhaylo Andriychuk, Lviv Polytechnic National University, Ukraine
Nataliia Romanyshyn, Lviv Polytechnic National University, Ukraine
Nataliya Boyko, Lviv Polytechnic National University, Ukraine
Oleksandr Markovets, Lviv Polytechnic National University, Ukraine
Olga Lozynska, Lviv Polytechnic National University, Ukraine
Olha Kots, Lviv Polytechnic National University, Ukraine
Olha Trach, Lviv Polytechnic National University, Ukraine
Solomia Fedushko, Lviv Polytechnic National University, Ukraine
Tetiana Bilushchak, Lviv Polytechnic National University, Ukraine
Roman Holoshchuk, Lviv Polytechnic National University, Ukraine
Vadym Ptashnyk, Lviv National Agrarian University, Ukraine

Victor Chumakevych, Lviv Polytechnic National University, Ukraine

Local Organization Committee

Natalia Sharonova, National Technical University "KhPI", Ukraine
Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine
Nina Khairova, National Technical University "KhPI", Ukraine
Olena Levchenko, Lviv Polytechnic National University, Ukraine
Olga Cherednichenko, National Technical University "KhPI", Ukraine
Yevhen Kupriianov, National Technical University "KhPI", Ukraine
Olga Kanishcheva, National Technical University "KhPI", Ukraine
Yuliia Hlavcheva, National Technical University "KhPI", Ukraine
Victor Grigorovich, Lviv Polytechnic National University, Ukraine
Victoriia Ryzhkova, National Aerospace University, Ukraine
Dmytro Dosyn, Lviv Polytechnic National University, Ukraine
Tetiana Shestakevych, Lviv Polytechnic National University, Ukraine
Zoia Kochuieva, National Technical University "KhPI", Ukraine
Svitlana Petrasova, National Technical University "KhPI", Ukraine
Oksana Ivashchenko, National Technical University "KhPI", Ukraine
Anastasiia Kolesnyk, National Technical University "KhPI", Ukraine
Natalia Borysova, National Technical University "KhPI", Ukraine
Karina Melnyk, National Technical University "KhPI", Ukraine
Nataliia Hrytsiv, Lviv Polytechnic National University, Ukraine
Victoria Vysotska, Lviv Polytechnic National University, Ukraine
Khrystyna Mykich, Lviv Polytechnic National University, Ukraine

Sponsors

National Technical University "Kharkiv Polytechnic Institute"
http://www.kpi.kharkov.ua/eng/

Lviv Polytechnic National University, Ukraine
http://www.lp.edu.ua/en

Institut Galilée of Université Paris 13
https://galilee.univ-paris13.fr

Politechnika Śląska
https://www.polsl.pl/Strony/Witamy.aspx

Ukrainian Scientific and Educational IT Society
https://usit.eu.org

Informational Partners

Kharkiv IT Cluster
http://it-kharkiv.com/en/

MANNING
https://www.manning.com/

Lviv IT Cluster
https://itcluster.lviv.ua/en/

PI-MINDS
http://pi-minds.com/en/

SYTOSS
https://sytoss.com

SSA Group
https://www.ssa.group

Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur
https://www.limsi.fr/fr/

Global Work
http://globalwork.top/

Zone3000
https://zone3000.net/

SoftServe
https://www.softserveinc.com/

Fortifier
www.4tifier.com

DATAART
https://www.dataart.com.ua

Ukrainian Lingua-Information Fund
https://www.ulif.org.ua/

Content

PART I.
POSTER PAPERS

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

Applying Recurrence Plots to Classify Time Series

Lyudmyla Kirichenko, Tamara Radivilova and Juliia Stepanenko

Kharkiv National University of Radio Electronics, 14 Nauky ave., Kharkiv, 61166, Ukraine

Abstract

The article describes a new approach to the classification of time series based on the construction of their recurrence plots. After transforming the time series into recurrence plots, two approaches are applied for classification. In the first case, quantitative recurrence characteristics are used for classification as features. In the second case, the time series is presented in the form of a black and white image of its recurrence plot. A convolutional neural network is used as an image classifier. The data for the classification are the electrocardiograms realizations of 100 values, which contained records of healthy people and patients with a diagnosis of ischemia. Research results showed the advantages of classifying images of recurrence plots, indicate a good classification accuracy in comparison with other methods and the potential capabilities of this approach.

Keywords 1

Time series classification, machine learning classification, recurrence plot, ECG time series, quantitative recurrence characteristic

1. Introduction

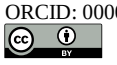
Analysis and classification of time series plays an important role in many areas of science and technology: in biology, seismology, physics, economics, in particular, in solving problems of diagnostics and forecasting. When time series classifying using machine learning, most often a set of some statistical features is extracted from the time series, which is input of the classifier. Various methods can be used as classifiers, including widespread neural networks [1,2].

A new and non-trivial approach to the classification of time series is the transformation of a series into another structure, for example a graph, a surface or a table, and the classification of features obtained on the basis of this structure [3-5]. If the structure obtained from the time series can be visualized, that is, represented as an image, then the resulting images can be classified by computer vision methods [3, 6-8].

One of the methods that allows visualizing the time series dynamics is the method of recurrence plots, proposed for the analysis of nonlinear dissipative systems and widely used in other areas of research [9-12]. In recent years, visualization of recurrence plots has been used to analyze and classify time series of various nature.

In this paper, the classification of electrocardiograms (ECG) is considered. Cardiac diseases are referred to those diseases that respond well enough to treatment if they are at an early stage. The main diagnostic method in cardiology for a long time has been ECG, which is widely used for the functional study of the cardiovascular system.

The purpose of the presented work is to classify ECG time series based on the construction of recurrence plots. After transforming the time series into recurrence plots, two approaches are applied for classification: the use of quantitative recurrence characteristics as classifier features and the recognition of recurrence plot images using a convolutional neural network.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: lyudmyla.kirichenko@nure.ua (L. Kirichenko); tamara.radivilova@nure.ua (T. Radivilova); yuliia.stepanenko@nure.ua (J. Stepanenko)
ORCID: 0000-0002-2780-7993 (L. Kirichenko); 0000-0001-5975-0269 (T. Radivilova)


2. Recurrence plots

Recurrence analysis is one of the nonlinear dynamic's methods used for time series analysis and is designed to identify non-obvious dependencies in the time series dynamics. Recurrence analysis of time realizations is based on the fundamental property of the trajectories of dissipative systems: repeat of states (recurrence).

This property was formalized in the "recurrence theorem", which says that if a system reduces its dynamics to limited subset of a n- dimensional space, then the system with a probability almost equal to 1, returns arbitrarily close to some initially specified state [11].

Let's consider some time realization, represented by its values $\{x_1, x_2, ..., x_i, ..., x_N\}$. Recurrence states of a point x_i are states x_j that fall into n-dimensional neighborhood of x_i with a given radius ε .

The recurrence of states x_i, where i is a time moment, is reproduced using a two-dimensional square matrix (recurrence plot) with black and white dots, where both coordinate axes i and j are discrete time axes, black dots with coordinates (i, j) indicate the presence of recurrence between points x_i and x_j . Thus, the recurrence plot is a black and white image.

For clarity, Fig. 1 shows a recurrence plot of a sinusoidal periodic trajectory (left), and a plot of the stochastic realization of encephalogram [13] (right).

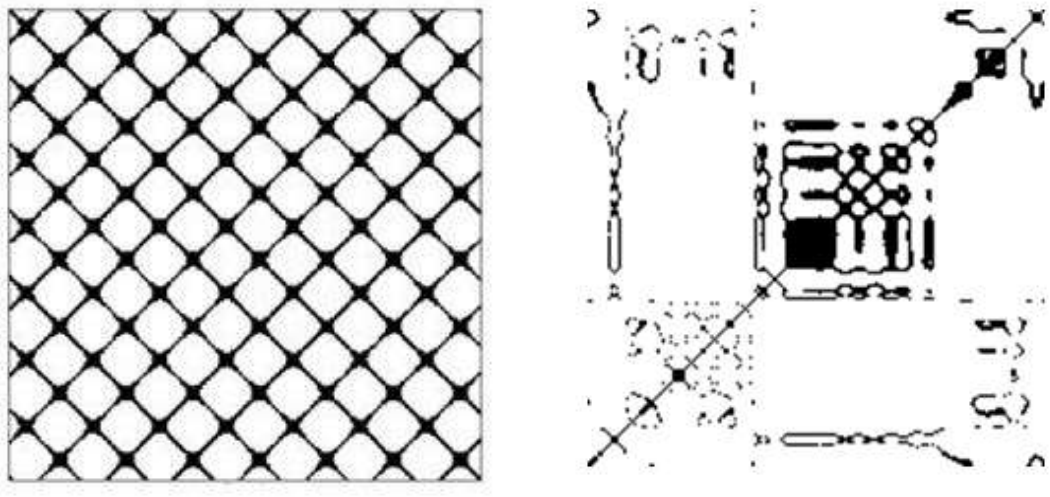


Figure 1: Recurrence plots: sinusoid (left) and encephalogram realization (right)

In [9-12], an approach was proposed for the numerical analysis of recurrence plots, which allows to obtain quantitative recurrence characteristics of a time series. Let us consider some quantitative characteristics that can be used as features for time series classification.

The most obvious characteristic is the recurrence rate (*RR*), which shows the density of points in a recurrence plot that corresponds to the probability of states repeating.

A number of characteristics are based on the calculation of the diagonal lines lengths L in the recurrence plot. The presence of a diagonal line corresponds to a situation when a some part of the phase trajectory repeats itself (within the specified accuracy ε), passing through the same region of the phase space at different time intervals. The average length of the diagonal lines *Lavg* corresponds to the average time during which the dynamics of the trajectory is repeated. Usually, random time series have a small length of diagonal lines, and a large number of separate recurrence points. Regular, in particular periodic time series, correspond to recurrence plots with long diagonal lines and a small number of separate recurrence points.

Shown in fig. 1 the recurrence plot of sine wave actually contains only diagonal lines that indicate the periodic nature of the trajectory of the system.

The derived characteristic from the lengths and number of diagonal lines is a measure of the system predictability (determinism, *DET*). It is based on the fact that the average length of the diagonal lines corresponds to the average predictability time of the system behavior. Entropy of the diagonal lines (*L_ENTR*) is calculated based on the frequency distribution of L and shows the complexity of the trajectory deterministic component.

Some of the quantitative characteristics are calculated on the basis of the vertical lines lengths *V* in the recurrence plot, which correspond to the trajectory being in the same state (laminarity, *LAM*) The value *LAM* indicates the presence of system conditions when the system movement stops or moves very slowly.

The average length of vertical structures (trapping time, *TT*) indicates the time that a trajectory can spend in the neighborhood ε of a certain state. Entropy of the vertical lines (*V_ENTR*) is calculated based on the frequency distribution of the vertical lines lengths and indicates the complexity of the laminar component of the trajectory.

Fig. 1 shows recurrence plot of the realization of an encephalogram, where both diagonal and vertical structures are present. Visually, you can determine that the lengths of the diagonal lines are small, and there are also a significant number of separated recurrence points in the structure.

Table 1 shows the values of the above-described quantitative characteristics corresponding to the recurrence plots shown in Fig. 1. Obviously, there is a significant difference between the characteristics of the deterministic and stochastic trajectories.

Table 1

Quantitative Recurrence Characteristics

	RR	Lavg	Det	L_ENTR	LAM	TT	V_ENTR
Sinusoid	0.12	39.76	0.998	0.03	0.67	0.92	0.024
Encephalogram	0.045	4.58	0.247	1.51	1.32	7.83	2.51

Thus, a time series can be represented by recurrence plot that reflects its dynamics. The classification of recurrence plots can be carried out on the basis of their quantitative characteristics, which act as features for the classifier. Another classification method could be to classify recurrence plot images using a convolutional neural network.

2. Convolutional Neural Networks

Convolutional neural network is a special architecture of artificial neural networks aimed at efficient image recognition. The idea behind convolutional neural networks is to alternate between convolutional layers, sub-sampling and regular layers of a neural network.

The structure of the network is unidirectional, in principle multilayer. For training, standard methods are used, most often the method of back propagation of an error. The function of activating neurons can be different, depending on the task solved by the neural network. Each fragment of the image is multiplied by the matrix (kernel) of the convolution element by the element, and the result is summed and written to the same position in the original image.

The investigated image is passed through a series of convolutional nonlinear, merge, and fully connected layers and an output is generated. The output can be a label or a probability of the class that best describes the image.

The first layer in the network is always convolutional. This is a set of functional cards of the same size. Each map has a synaptic core, which is a window that slides over the entire area of the preliminary map and finds certain features of objects. The neurons on the first convolutional layer are not associated with each pixel of the input image, but only with pixels in their own receptor fields. Further, the neuron of the second convolutional layer is connected only to the neurons inside the rectangular region of the first layer. The considered architecture of the neural network makes it possible to focus on low-level objects in the first hidden layer in order to further combine them into high-level objects.

The first level output is the second level input value. After applying a set of filters after the first layer, filters will be activated, which represent the properties of the highest level. The more

convolutional layers an image passes and the further it travels through the network, the more complex the characteristics are reflected in the object maps.

After the convolutional layers comes the pooling layer, the main task of which is to thin out the input image to reduce the computational load, memory consumption and the number of parameters, reducing the risk of overfitting.

The last type of layer is a regular multi-layer perceptron layer. The purpose of the layer is classification, it models a complex non-linear function, the optimization of which increases the quality of recognition. The output layer is connected with all neurons in the preceding layer. The number of neurons on the output layer is equal to the number of classification classes. [14, 15].

3. Description of the Experiment

3.1. Input Data

The input data for research in this work were data obtained from the repository "UEA & UCR Time Series Classification Repository" [16]. The dataset name is "ECG200".

It contains medical time series that are ECG realizations: 200 samples, of which 100 are intended for training the classifier and 100 for testing. Each series corresponds electrical activity recorded during the heartbeat and contains 100 values. The ECG realizations are divided into two classes: "norm" (class 0) and "ischemia" (class 1).

Fig. 2 shows schematic images of the ECG realizations for a healthy person and a patient diagnosed with ischemia [16].

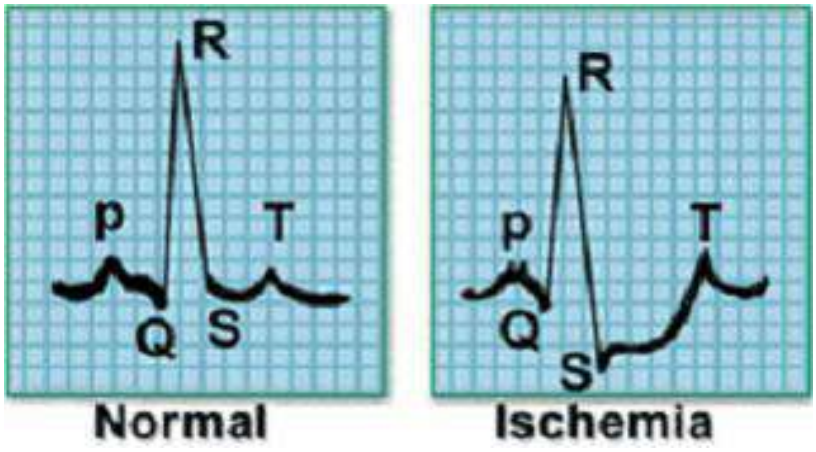


Figure 2: Schematic ECG, "Normal" and "Ischemia"

Table 2 shows the number of time series for classes “0” and “1”.

Table 2
Number of Time Series for Experiment

Dataset	Class 0	Class 1	Total
Train	31	69	100
Test	36	64	100
Total	67	133	200

Fig. 3 shows examples of ECG time series from the dataset, which are typical for a healthy person (class “0”) and person with ischemia disease (class “1”).

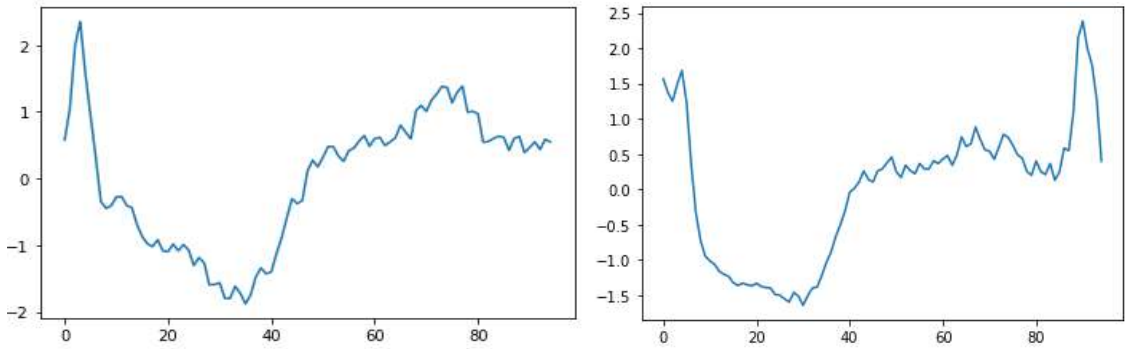


Figure 3: Realizations of ECG; Left - Class “0”, Right - Class “1”

Fig. 4 presents examples of the recurrence plots which correspond ECG time series of Fig. 3. It should be noted that, in contrast to the schematic image, the difference between the ECG time series of class "0" and class "1" is not visually observed, while the recurrence plots have visual differences.

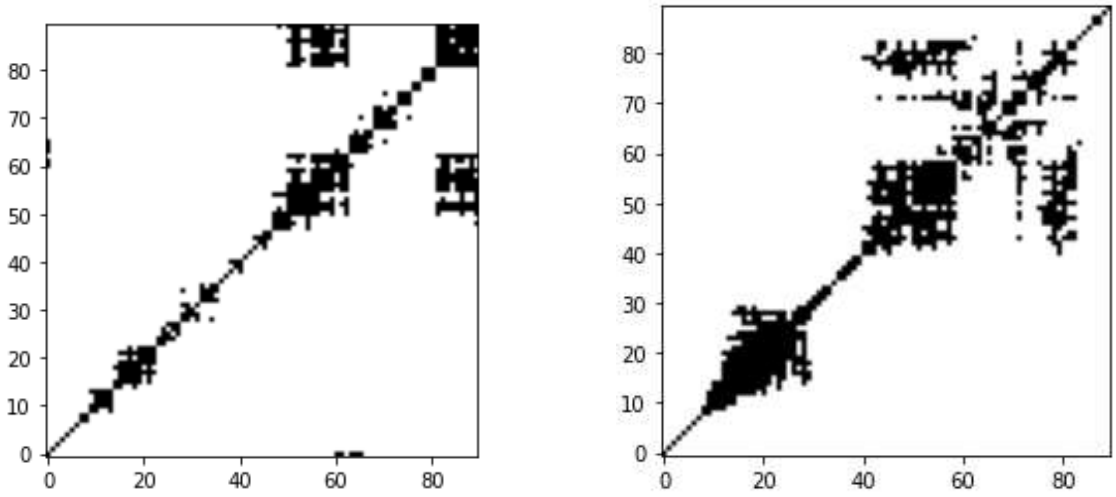


Figure 4: Recurrence Plots; Left - Class “0”, Right - Class “1”

3.2. Experiment

We need to solve the problem of binary classification of medical time series. For the classification, the Python language was chosen. Python is a high-level object-oriented programming language with strong dynamic typing. It is an open source language containing many libraries for processing and graphing data. Python is one of the most demanded and popular program language, as evidenced by numerous ratings and analysis of proposals on the software development market [17].

Time series classification was carried out by two methods. In the first case, the time series were transformed into recurrence plots, from which the quantitative characteristics were calculated. The obtained characteristics were the input features for the classifier.

The following sample recurrence measures were used as features: the recurrence rate *RR*, the predictability *DET*, the laminarity of the time series *LAM*, the maximum length of diagonal lines *Lmax*, the maximum length of vertical lines *Vmax*, inverse value of the maximum diagonal line length *DIV*, average length of a diagonal line *Lavg*, trapping time *TT*, entropy of diagonal lines *L_ENTR*, - entropy of vertical lines *V_ENTR*.

Fig. 5 shows several values of recurrence characteristics obtained from ECG realizations belonging to class "0".

	is_train	RR	DET	LAM	Lmax	Vmax	DIV	Lavg	TT	L_ENTR	V_ENTR	class
0	1.0	0.12889	0.91824	0.95115	26.0	25.0	0.03846	4.70968	6.32484	1.83398	2.07083	0.0
2	1.0	0.13630	0.89349	0.95924	28.0	18.0	0.03571	4.31429	5.16585	1.78724	2.10995	0.0
3	1.0	0.09333	0.93393	0.94180	30.0	27.0	0.03333	7.06818	7.82418	2.05041	2.44319	0.0
6	1.0	0.09432	0.78932	0.90969	25.0	21.0	0.04000	3.59459	4.04070	1.49819	1.79750	0.0
7	1.0	0.11531	0.83412	0.88330	39.0	20.0	0.02564	3.66667	4.91071	1.57012	2.06482	0.0

Figure 5: Recurrence Characteristics of Class “0”

Fig. 6 shows the values of the same recurrence characteristics obtained for class "1". As is clear from the above examples, the characteristic values of classes “0” and “1” differ from each other. For example, the average value *Lavg* for class "0" in the given five inputs is 4.67, which is higher than for class "1", where the average is correspondingly 3.15. Similar differences can be seen if we carry out calculations for all the given characteristics.

	is_train	RR	DET	LAM	Lmax	Vmax	DIV	Lavg	TT	L_ENTR	V_ENTR	class
1	1.0	0.03531	0.10204	0.31818	3.0	4.0	0.33333	2.50000	2.45946	0.69315	0.63964	1.0
4	1.0	0.06889	0.73077	0.83333	24.0	13.0	0.04167	3.71739	4.42857	1.32311	1.93830	1.0
5	1.0	0.06790	0.43913	0.55455	4.0	7.0	0.25000	2.58974	3.01980	0.96288	1.29269	1.0
8	1.0	0.07679	0.49248	0.66238	6.0	13.0	0.16667	2.56863	3.16923	0.95351	1.38169	1.0
9	1.0	0.09506	0.85588	0.92597	27.0	16.0	0.03704	4.40909	5.32090	1.85633	1.88740	1.0

Figure 6: Recurrence Characteristics of Class “1”

To carry out the classification in the first case, a fully connected multilayer perceptron with an activation function of the ReLU type was chosen [18]. This neural network is a versatile approximator and is capable of detecting hidden patterns in data. To prevent overfitting of the model and to increase the classification accuracy, several layers of batch normalization were included in the structure of the neural network [19].

In the second case, the classification was based on the recognition of images of recurrence plots. Input ECG time series for training and test samples were transformed into recurrence plots images. Some of the resulting images for both classes are shown in Fig. 7.

To create a neural network, the Keras library was used, which is the most popular for creating neural networks. The developed convolutional neural network contains five layers; the first two ones are convolutional. The output of the last layer is fed to a 2-sided softmax, which produces a distribution over 2 classes. Neurons in fully connected layers are connected to all neurons in the previous layer. The non-linear ReLU function is applied to the output of each convolutional and fully connected layer. The Adam stochastic optimization method was chosen as the training method [20].

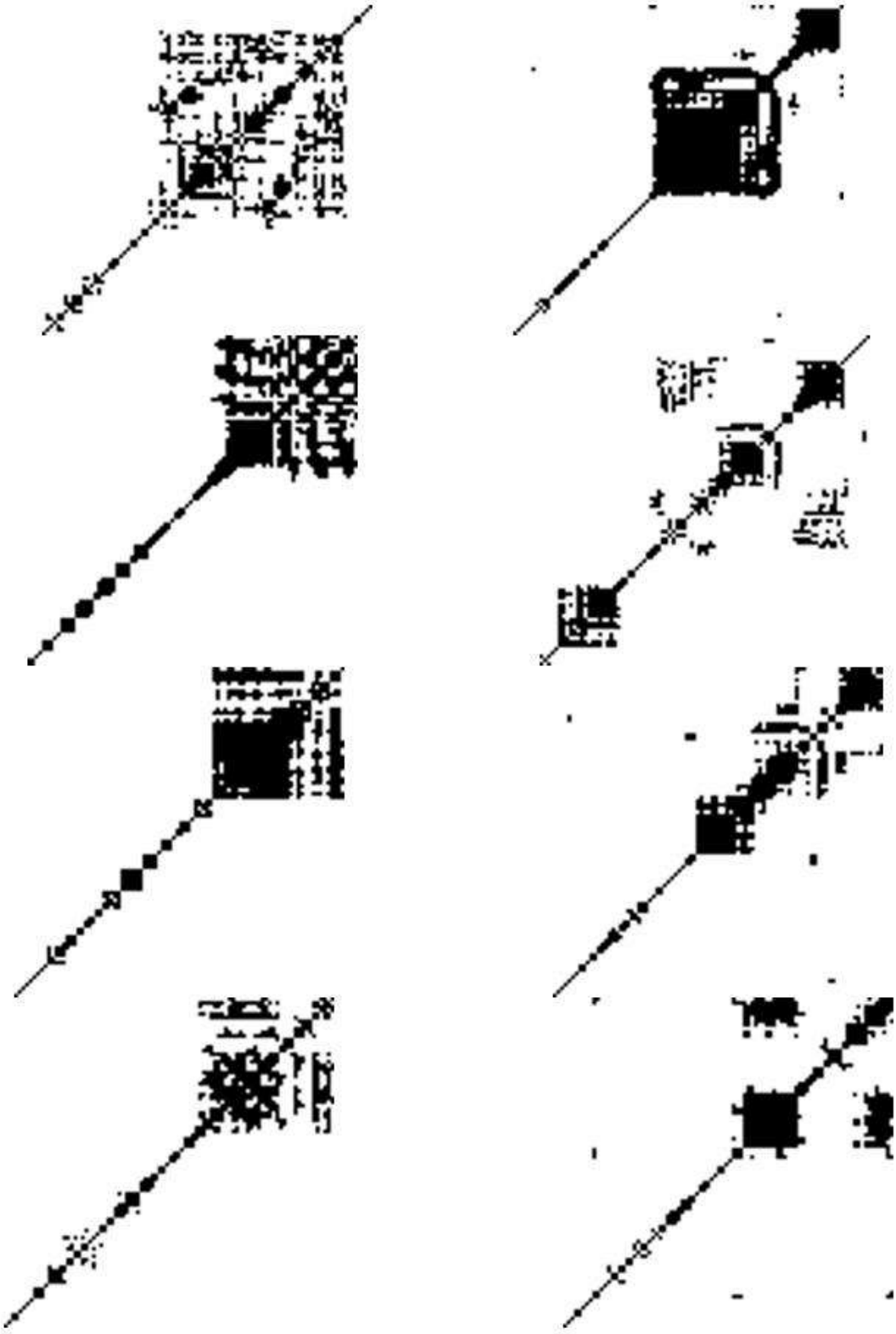


Figure 7: Recurrence Plots of Class "0" (Left) and Class "1" (Right)

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

3.3. Classification quality metrics

To determine the classification accuracy, metrics were used that are determined by the number of correctly and falsely detected cases presented in the confusion matrix, namely: true positive (*TP*) - when the ECG of a healthy person was correctly identified; true negative (*TN*) - when the disease was correctly recognized; false positive (*FP*) - when the ECG was healthy, but was classified as a disease; and false negative (*FN*) - when the disease ECG was taken for the healthy ECG. The classification metrics are calculated as a function of these four values.

Accuracy is the proportion of correctly defined ECGs for healthy and diseased person:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

The Precision metric can be interpreted as the proportion of objects called positive by the classifier and, at the same time, are really positive, and the Recall metric shows what proportion of objects of a positive class from all objects of a positive class was found by the algorithm:

$$Precision = \frac{TP}{TP + FP}, \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

F -metric is the harmonic mean of *Precision* and *Recall*:

$$F = 2\frac{Precision * Recall}{Precision + Recall}. \quad (4)$$

The ROC curve was also plotted. The Area Under Curve - Receiver Operating Characteristic curve (AUC-ROC) is a way to evaluate the model as a whole. The ROC curve is a curve from point (0,0) to point (1,1) in the coordinates True Positive Rate (TPR) and False Positive Rate (FPR), where

$$TPR = \frac{TP}{TP + FN}, \quad (5)$$

$$FPR = \frac{FP}{FP + TN}. \quad (6)$$

Ideally, when the classifier makes no mistakes ($FPR = 0, TPR = 1$), the area under the curve is equal to 1; when the classifier determines the probabilities of the classes at random, the AUC-ROC will approach 0.5, since the classifier will issue the same number of *TP* and *FP*; it is obvious that the value of the area under the curve evaluates the quality of the algorithm.

4. Research results and discussion

Consider the results of the classification carried out by the two methods described above and compare them using classification quality metrics.

The results of the classification on the basis of numerical recurrence characteristics are presented in Table 3. It is clear that the ECG time series related to class "1", i.e. electrocardiograms of patients with ischemic disease are recognized much more accurately than normal ECG records.

Table 3

Classification Evaluation Metrics

	Precision	Recall	F -metric
Class 0	0.80	0.64	0.71
Class 1	0.75	0.94	0.83
Accuracy			0.81

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

ROC-curve is a reliable method for assessing accuracy. In fig. 8 the ROC-curve for classification based on quantitative characteristics is presented, the value of the area under the ROC-curve is 0.76.

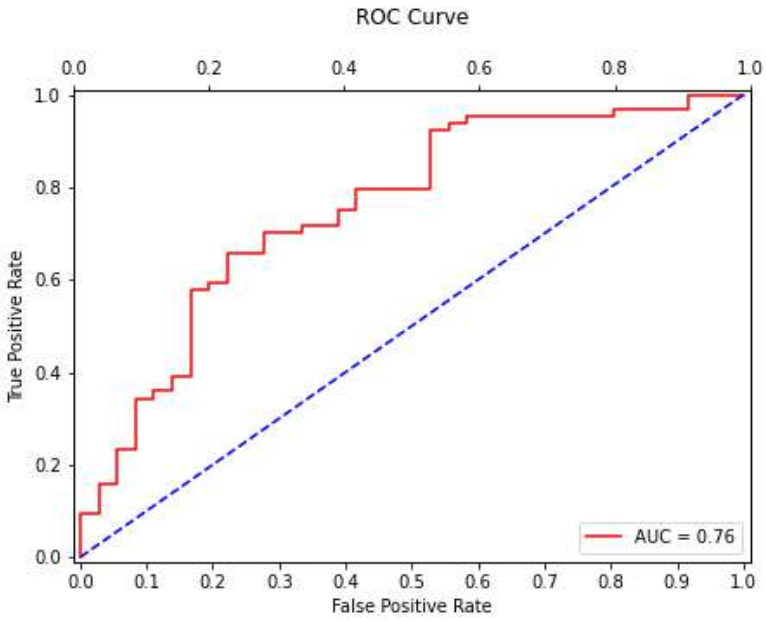


Figure 8: ROC Curve for Classification Based on Quantitative Characteristics

The evaluation metrics for classification of recurrence plot images using developed convolutional neural network were calculated and presented in Table 4.

Table 4
Classification Evaluation Metrics

	Precision	Recall	F -metric
Class 0	0.93	0.72	0.81
Class 1	0.86	0.97	0.91
Accuracy			0.89

From the obtained values of the metrics it follows that as well as in the first case the ECG of patients with ischemia is determined more accurately than the ECG of patients without heart disease. Perhaps this is due to the greater number of realizations in the sampled data or some characteristic features of the ECG.

In fig. 9 the ROC curve is presented, the value of the area under the ROC-curve is 0.92.

The results showed that the classification of recurrence plot images using a convolutional neural network gave significantly higher accuracy for all metrics than classification based on quantitative characteristics using a fully connected multilayer perceptron.

It should be noted that in the dataset description it was indicated that the best classification accuracy of these data was obtained using the Bag-of-SFA-Symbols (BOSS) classifier and equals 89% [16]. Although we have achieved the same precision, we used the simple neural network. Obviously, when using a deep neural network aimed at recognizing black and white images, the classification accuracy will be higher.

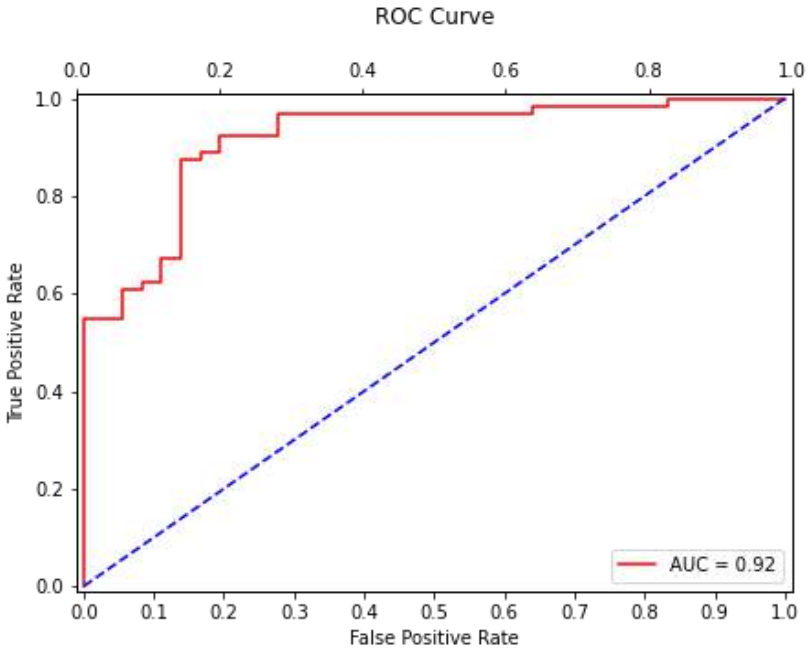


Figure 2: ROC Curve for Classification of Recurrence Plot Images

5. Conclusion

The article discussed comparative analysis of the time series classification based on the application of recurrence plot method. Two approaches were applied for classification: the use of quantitative recurrence characteristics as features and the recognition of recurrence plot images

The input data for the experiment were ECG time series containing 100 values, which have been divided into two classes: "normal" and "ischemia". Research results have shown the advantages of classifying images of recurrence plots. With this approach the classification accuracy has been 89%, while the accuracy of classification based on numerical characteristics has been 81%. Image classification have been carried out using a simple convolutional network, however, the accuracy value was equal to the best accuracy obtained by classifying this dataset using was equal methods.

The considered approach of image recognition has great potential for other applications related to the analysis and classification of time series. Our future research will focus on improving the neural network architecture in order to better recognize black and white images of typical recurrence plots.

6. References

[1] J. C. B. Gamboa, Deep learning for time-series analysis, 2017. URL: https://arxiv.org/pdf/1701.01887.pdf.

[2] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P. A. Muller, Deep learning for time series classification: a review. Data Mining and Knowledge Discovery 33.4 (2019): 917-963. doi: 10.1007/s10618-019-00619-1.

[3] Marisa Faraggi, Time series features extraction using Fourier and Wavelet transforms on ECG data. URL: https://slacker.ro/2019/11/23/time-series-features-extraction-using-fourier-and-wavelet-transforms-on-ecg-dat.

[4] L. Kirichenko, T. Radivilova, V. Bulakh. Binary Classification of Fractal Time Series by Machine Learning Methods, in: V. Lytvynenko, S. Babichev, W. Wójcik, O. Vynokurova, S. Vyshemyrskaya, S. Radetskaya (Eds.), Lecture Notes in Computational Intelligence and Decision Making, volume 1020 of Advances in Intelligent Systems and Computing, Springer, Cham, 2020, pp. 701-711. doi: 10.1007/978-3-030-26474-1_49

[5] T. Radivilova, L. Kirichenko, V. Bulakh, Comparative analysis of machine learning classification of time series with fractal properties, in: Proceedings of 8th International Conference on Advanced Optoelectronics and Lasers, CAOL 2019, IEEE, Sozopol, Bulgaria, 2019, pp. 557-560. doi: 10.1109/CAOL46282.2019.9019416

[6] L. Kirichenko, P. Zinchenko, T. Radivilova, M. Tavalbeh, Machine Learning Detection of DDoS Attacks Based on Visualization of Recurrence Plots, in: Proceedings of the International Workshop of Conflict Management in Global Information Networks, CMiGIN 2019, Ceur, Kyiv, Ukraine, 2019, pp. 23–34.

[7] L. Kirichenko, P. Zinchenko, T. Radivilova, Classification of Time Realizations Using Machine Learning Recognition of Recurrence Plots, in: S. Babichev, V. Lytvynenko, W. Wójcik, S. Vyshemyrskaya (Eds.), Lecture Notes in Computational Intelligence and Decision Making, volume 1246, of Advances in Intelligent Systems and Computing, Springer, Cham, 2021, pp. 687-696. doi: 10.1007/978-3-030-54215-3_44.

[8] N. Hatami, Y Gavet, J. Debayle, Classification of time-series images using deep convolutional neural networks, in: Proceedings of Tenth International Conference on Machine Vision, ICMV 2017, 10696, 106960Y, 2018.

[9] J. P. Eckmann, S. O. Kamphorst, D. Ruelle, Recurrence plots of dynamical systems. Europhysics Letters 4.9, (1987): 973-977.

[10] N. Marwan, N. Wessel, U. Meyerfeldt, A. Schirdewan, J. Kurths, Recurrence-plots-based measures of complexity and application to heart-rate-variability data. Physical Review E 66.2 (2002): 026702-1-026702-6. doi: 10.1103/PhysRevE.66.026702

[11] N. Marwan, M. C. Romano, M. Thiel, J. Kurths, Recurrence plots for the analysis of complex systems. Physics reports 438.5-6 (2007): 237-329.

[12] L. Kirichenko, T. Radivilova, V. Bulakh, Classification of fractal time series using recurrence plots, in: Proceedings of 2018 International Scientific-Practical Conference Problems of Infocommunications. Science and Technology, PIC S&T 2018, IEEE, Kharkiv, Ukraine, 2018, pp.719-724. doi: 10.1109/INFOCOMMST.2018.8632010.

[13] L. Kirichenko, T. Radivilova, V. Bulakh, P. Zinchenko and A. Saif Alghawli, Two Approaches to Machine Learning Classification of Time Series Based on Recurrence Plots, 2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP), Lviv, Ukraine, 2020, pp. 84-89, doi: 10.1109/DSMP47368.2020.9204021.

[14] Y. LeCun and Y. Bengio, Convolutional Networks for Images, Speech, and Time-Series, in M. A. Arbib (Eds.), The Handbook of Brain Theory and Neural Networks, MIT Press, 1995.

[15] C. Dan, U. Meier, J. Masci, L. M. Gambardella, J. Schmidhuber, Flexible, High Performance Convolutional Neural Networks for Image Classification, in: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, volume 2, 2013, pp.1237–1242. URL: http://people.idsia.ch/~juergen/ijcai2011.pdf.

[16] Time series classification. URL: http://www.timeseriesclassification.com

[17] D. Cielen, A. Meysman, M. Ali, Introducing Data Science: Big Data, Machine Learning, and more, using Python tools, Manning Publications, 2016.

[18] J. Brownlee, A Gentle Introduction to the Rectified Linear Unit (ReLU), Machine learning mastery, January 2019. URL: https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks

[19] S. Ioffe and C. Szegedy, Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, in: Proceedings of the 32nd International Conference on Machine Learning, volume 37, 2015, pp. 448-456.

[20] D. P. Kingma and J. Ba Adam, A Method for Stochastic Optimization, in: Proceedings of the 3rd International Conference on Learning Representations (ICLR), 2015.

Analysis of Vulnerabilities of IoT-Devices and Methods of Their Elimination

Oleksii Liashenko, Darina Kazmina, Dmytro Rosinskiy and Yana Dukh

Kharkiv National University of Radio-Electronics, 14 Nauky Ave, Kharkiv-UA-61166, Ukraine

Abstract

Relevance and the problem setting: at present, vulnerabilities in the firmware of IoT-devices pose a serious threat, as attackers, who at first have exploited the vulnerabilities, gain remote access to devices which allows them to form botnets that are then used to capture new devices or organize serious DDos attacks. Therefore, currently, there is an urgent need to increase the effectiveness of vulnerability detection methods in the firmware. The purpose of this work is to analyze and define the term "vulnerability", to provide the classification of vulnerabilities of IoT-devices, the causes of vulnerabilities of IoT-devices, to analyze the stages of vulnerability detection, and to present the example of a search algorithm for vulnerable IoT-devices.

Keywords 1

IoT, vulnerability, IoT-device

1. Introduction

The Internet of Things is the concept of a computer network of physical objects ("things"), equipped with embedded technologies to interact with each other or with the external environment, which considers the organization of such networks as a phenomenon capable of restructuring economic and social processes, eliminating the need of human participation for activities and operations.

The concept was formulated in 1999 as an understanding of the prospects for the widespread use of radio frequency identification for the interaction of physical objects with each other and with the external environment. Filling the concept with a variety of technological content and implementation of practical solutions for its realization since 2010 is considered a steady trend in information technologies, primarily due to the widespread use of wireless networks, cloud computing, development of technologies of machine-to-machine interaction, the beginning of the active transition to IPv6 and development of software-defined networks.

Cisco analysts consider the period from 2008 to 2009 to be the "real birth of the Internet of Things" because, according to their estimates, it was during this period that the number of devices connected to the global network exceeded the population of the Earth, thus "the Internet of Men" became "the Internet of Things".

Since 2009, "The Internet of Things" Conference has been held annually in Brussels with the support of the European Commission where European Commissioners and MEPs, government officials from the European countries, heads of companies such as SAP, SAS Institute, Telefónica, leading scientists of large universities and research laboratories present their reports.

Since the early 2010s, the Internet of Things has become the driving force for the Fog computing paradigm, which extends cloud computing principles from data centers to a vast number of interacting geographically distributed devices seen as the Internet of Things platform.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: oleksii.liashenko@nure.ua (O. Liashenko); daryna.kazmina@nure.ua (D. Kazmina); dmytro.rosinskyi@nure.ua (D. Rosinskiy); yana.dukh@nure.ua (Yana Dukh)
ORCID: 0000-0002-0146-3934 (O. Liashenko); 0000-0002-0725-392X (D. Rosinskiy)

Since 2011, Gartner has been placing the Internet of Things in the general hype of new technologies at the stage of "technological trigger" indicating more than 10-year period of time, and since 2012 a specialized "hype cycle of the Internet of Things" has periodically been issued.

The Internet of Things is used in many areas of life, for example:

• the Internet of Things in healthcare - Healthcare deserves a special place in the list of the best areas of application of the IoT. In it, the Internet of Things directly affects people's lives and illustrates the importance of connected healthcare as an area;

• smart city - the city's IoT technologies include smart parking, noise maps, smart lighting, and roads. Although this group of devices is currently under development, it has great prospects. With its help it is possible to increase safety in any city, it is better to control traffic and pollution;

• Internet of Things in energetics - a smart grid that can automatically collect the necessary data and instantly analyze the current circulation. As a result, both customers and suppliers will be able to optimize the use of electricity;

• smart devices - these include devices that are controlled by an application on a smartphone and worn on the body. These gadgets intersect with medical IoT-devices because they can be used to test basic health and improve treatment. The key players in this market are the Apple, Samsung, and Motorola companies - they develop fitness bracelets, GPS straps, smart implants, and other IoT-devices.

The IoT-devices often flash in news headlines concerning the topics of information security. Malicious routers, pet-assisted voice assistants, and electric car charger hacks are just a small part of things that creates an aura of vulnerability around the Internet of Things as a technology[1-3].

Security issues have haunted the Internet of Things since the invention. Everyone, from suppliers to corporate users and consumers, is concerned that their fashionable new devices and IoT systems may be compromised. The problem is actually even worse because vulnerable IoT-devices can be hacked and used in giant botnets that threaten even properly secured networks.

According to the study by Palo Alto Networks, 57% of smart devices are prone to attacks of medium and high danger. According to Gartner, about 20% of organizations have been attacked by the Internet of Things in the last three years. Statista predicts that by 2025, the number of connected smart devices worldwide will reach 75 billion, and the number of attacks on them will increase by about five times.

All harmful components in IoT-devices can be identified as the term "vulnerability". Thus, a vulnerability is a flaw in the system, using which, one can intentionally violate its integrity and lead to malfunction.

Usually, a vulnerability allows an attacker to "trick" an application, i.e. to perform actions unpredictable by the creator or to force the application to perform an action to which it should not be entitled. This is done by introducing data or code into the program in any way in such places that the program will perceive them as "its own". Some vulnerabilities appear due to insufficient verification of the data entered by the user and allow inserting the arbitrary commands (SQL-injection, XSS, SiXSS) in the interpreted code. Other vulnerabilities arise due to more complex problems, such as writing data to the buffer without checking its boundaries (buffer overflow). Searching for vulnerabilities is sometimes called probing, for example, when talking about probing a remote computer - they mean finding open network ports and the presence of vulnerabilities related to applications that use these ports.

The purpose of this work is to define the term "IoT", the term "vulnerability" and to identify the main methods of finding and eliminating vulnerabilities from IoT-devices.

2. Vulnerabilities of IoT-Devices

Security is the main criterion of any IoT device. This applies to both software and hardware. The proliferation of IoT-devices has caused an increase in demand for them, and this has required an acceleration of the production process. And as companies began to turn to various third parties, the development of systems began to take place in geographically distant places. Such changes entail a serious security problem - even if the company purchases the necessary equipment from trusted

suppliers, there is no guarantee that the device has not been modified somewhere in the supply chain, and that the hardware is not harmful.

As of early 2021, the IoT market was almost $ 742 billion. The total number of devices connected to the Internet is currently estimated at more than 30 billion. After that, the analytical agency IHS Markit predicts nonlinear growth to 125 billion devices by 2030. Such a volume of production is quite possible, but already now the shocking pace of production of IoT-devices is achieved mainly due to the cheapest "Chinese" devices, in the development of which security was thought of last.

2.1. Types of Vulnerabilities in IoT-Devices

The international non-profit organization OWASP (Open Web Application Security Project) took over the security of the Internet of Things as early as in 2014, releasing the first version of "OWASP Top 10 IoT". The updated version of the "TOP 10 vulnerabilities of the Internet of Things" with updated threats was released in 2018. This project aims to help manufacturers, developers and consumers understand the IoT security issues and make more informed IS decisions when creating IoT-ecosystems.

1. Insufficient physical security.

One of the security challenges of the IoT ecosystem is that its components are distributed in space and are often installed in public or unprotected places. This allows attackers to access the device and take control of it locally or use it to access another network.

The attacker can copy the settings (IP network, MAC address, etc.) and put his device instead of the original to eavesdrop or reduce network performance. He can hack an RFID reader, install hardware, infect malware, steal data, or simply physically disable an IoT device.

The solution to this problem is one - to complicate the physical access to devices. They can be installed on protected areas, at height, or use anti-vandal protected cabinets.

2. Dangerous default settings.

Devices or systems come with unsafe default settings or do not have the ability to make the system more secure by restricting users from changing the configuration.

Every manufacturer wants to earn more and spend less. The device can implement many smart features, but cannot provide the ability to change security.

For example, password validation is not supported, there is no possibility to create accounts with different rights – the administrator's and users', there is no setting of encryption, logging, and notification of users about security events.

3. Lack of ability to manage and control the device.

There is a lack of security support on devices deployed in production, including asset management, update management, safe decommissioning, system monitoring, and response.

IoT-devices are often a "black box". They do not have the ability to monitor the status of work, identify which services are running and what they are interacting with.

Not all manufacturers allow users of IoT-devices to control the operating system and running applications fully, as well as check the integrity and legitimacy of downloaded software or install update patches on the OS.

During attacks, the device's firmware can be reconfigured so that it can be started only by complete reflashing of the device. A similar disadvantage was taken, for example, by the malware Silex.

The solution to these problems can be using specialized software to manage Internet of Things devices, such as cloud solutions AWS, Google, IBM, etc.

4. Dangerous data transmission and storage.

There is a lack of encryption or access control to sensitive data anywhere in the ecosystem, including during storage, transmission, or processing.

IoT-devices collect and store environmental data, including various personal information. The compromised password can be replaced, and the stolen data from the biometric device - fingerprint, retina, facial biometrics - no.

At the same time, IoT-devices cannot only store data in unencrypted form but also transmit it over the network. If the transmission of data in the open form over the local network can be somehow

explained, then in the case of a wireless network or transmission over the Internet, they can become the property of anyone.

The user himself can use secure communication channels for data transmission, but the encryption of stored passwords, biometric, and other important data must be taken care of by the device manufacturer.

5. Insufficient protection of confidentiality.

There is personal information of the user stored on the device or in the ecosystem, which is used dangerously, improperly, or without permission.

This item of TOP-10 echoes the previous one: all the personal data must be stored and transmitted in a secure form. But this paragraph examines privacy in a deeper sense, namely in terms of protecting privacy/private life.

IoT-devices collect information about what and who surrounds them, including the fact that they do not suspect people. Stolen or improperly processed data concerning the user can both inadvertently discredit a person (for example, when improperly configured road cameras have exposed an unfaithful spouse) or be used in blackmail.

To solve the problem, one should know exactly what data are collected by the IoT-device, mobile application, and cloud interfaces.

One should be sure that only the data necessary for the operation of the device are collected, should check whether there is permission to store personal data and whether they are protected, as well as storage policies should be prescribed. Otherwise, if these conditions are not met, the user may have problems with the law.

6. Use of hazardous or obsolete components.

There is the use of outdated or unsafe software components or libraries that may compromise one's device. This includes dangerously configuring operating system platforms and using third-party software or hardware components from a compromised supply chain.

One vulnerable component can nullify all the established security.

The solution to this problem is to monitor the release of security patches and update the device, and if they do not work, it is necessary to change the manufacturer.

7. Lack of secure update mechanisms.

It is a lack of safe device upgrades. This includes the lack of firmware validation on the device, the lack of secure delivery (without encryption during transmission), the lack of rollback prevention mechanisms, and the lack of notifications of security changes due to updates.

The inability to update the device is in itself a weak point of security. Failure to install the update means that the devices remain vulnerable indefinitely.

But in addition, the update and firmware can also be dangerous. For example, if the software does not use encrypted channels, the update file is not encrypted or not verified for integrity before installation, there is no anti-rollback protection (protection against reverting to a previous, more vulnerable version) or no security change notifications due to updates.

The solution to this problem is also on the side of the manufacturer. But everyone can check if their device can be updated at all. Make sure that the update files are downloaded from a trusted server over an encrypted channel, and that the device uses a secure update installation architecture.

8. Dangerous ecosystem interfaces.

There is a dangerous web interface, API, cloud, or mobile interfaces in the ecosystem outside the device, which allows compromising the device or related components. Common problems include lack of authentication or authorization, lack of weak encryption, and lack of I/O filtering.

Using unsafe web interfaces, APIs, cloud, and mobile interfaces allows compromising the device or related components even without connecting to it.

For example, Barracuda Labs analyzed the mobile application and web interface of one of the "smart" cameras and found vulnerabilities that allow getting a password to the Internet of Things device:

- the mobile application ignored the validity of the server certificate;
- the web application was vulnerable to cross-site scripting;
- it was possible to bypass files on the cloud server;
- device updates were not protected;

• the device ignored the validity of the server certificate.

To protect, it is urgent to change the default username and password, make sure that the web interface is not prone to cross-site scripting, SQL injection, or CSRF attacks.

Password protection against password attacks must also be implemented. For example, after three attempts to enter the wrong password, the account should be locked and allow recovering the password only through a hardware reset.

9. Dangerous network services.

They are unnecessary or unsafe network services running on the device itself, especially open to the external network that endanger the confidentiality, integrity, authenticity, availability of information, or allow unauthorized remote management.

Unnecessary or dangerous network services endanger the security of the device, especially if they have access to the Internet.

Dangerous network services can be susceptible to buffer overflow and DDoS attacks. Open network ports can be scanned for vulnerabilities and dangerous connectivity services.

One of the most popular vectors of attacks and infection of IoT-devices is still brute-force attacks on NOT disabled Telnet services and SSH. After gaining access to these services, attackers can download malicious software to the device and gain access to valuable information.

10. Weak, guessed, or hard set password.

It is the use of easily hacked, public, or unchangeable credentials, including backdoors in firmware or client software that provides unauthorized access to deployed systems.

Surprisingly, the biggest vulnerability so far is the use of weak passwords, default passwords, or passwords leaked to the network.

Despite the obvious need to use a strong password, some users still do not change their default passwords. In June 2019, Silex malware took advantage of this, turning about 2,000 IoT-devices into a "brick" during one hour.

And before that, the well-known botnet and worm Mirai managed to infect 600,000 IoT-devices using a database of 61 standard login-password connections.

The solution is to change the password [4].

2.2. Causes of Vulnerabilities in IoT-Devices

1. Network.

What makes an IoT device easy to use – the ability to control it remotely and centrally from anywhere in the world - is the biggest security threat. According to Palo Alto Networks' 2020 "Unit 42 IoT Threat Report", 98% of IoT-device traffic is not encrypted and transmitted over the Internet. One of the most common Internet of Things protocols is MQTT. It supports users' authentication and encryption, but by default, these options are not enabled in IoT-devices. The TCQ/IP-based MQTT protocol (without encryption option) works with port 1883. The Shodan search engine finds more than 300,000 devices that transmit data to the global network without encryption. This is 10 thousand times more than the number of devices found that use it (the port for working with MQTT via TLS 88831). The MQTT protocol uses a client-server model: data is exchanged between the client and the message broker. The protocol client can be authenticated by sending a login and password to the MQTT broker in the request. In response, the MQTT broker sends a code with a specific number. Upon successful authentication and in cases when it is not used, the broker returns a response with the code "MQTT Connection Code: 0". According to the Shodan, there are about 51,000 devices on the Internet that do not support user authentication. Moreover, with the Shodan, one can find MQTT-connected medical devices, such as electrocardiographs, without the use of authentication and encryption. One of the recommended security measures is the segmentation of IoT networks. However, according to Palo Alto Networks statistics, only 3% of healthcare facilities have only the Internet of Things medical equipment in one segment of the physical or virtual network. 25% of institutions also have non-medical IoT-devices (such as IP phones and printers), and the remaining 72% have mixed IoT and IT resources on the same network. This means that malware from users' computers can spread freely to medical equipment. After all, those vulnerabilities for which long-established ways to resolve in IT devices can still be used to disable the Internet of Things.

2. Software.

Installing security patches on your Internet of Things device can cause a lot of difficulties. Not only upgrading working equipment is a risky affair, but many manufacturers do not release software upgrades at all. Only 17% of smart devices run on supported operating systems. The remaining 83% use older versions of Linux, Unix, Embedded, Windows 7, and even Windows XP (Figure 1).

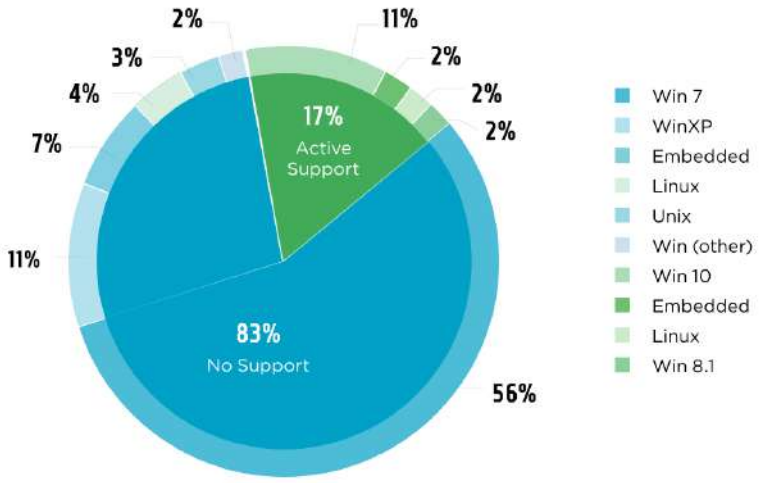


Figure 1: Versions of Operating Systems Used on IoT-Devices

3. Cloud services.

Currently, the IoT-systems are almost completely dependent on cloud services. According to the "2018 Cloud Computing Survey", 34% of IT and IS managers are concerned about the security of cloud services and data storage. The studies by Palo Alto Networks have identified 34 million vulnerabilities directly in cloud services given by such large providers as AWS, Azure, GCP. For example, the Amazon Elastic Computing Cloud (EC2) vulnerabilities have allowed hackers to gain unauthorized access to the physical machine. As soon as an attacker has the opportunity to share resources, he can try to carry out an attack on third-party channels (side-channel attack) and steal the data stored in the cloud from IoT-devices. A multi-rental environment also makes it possible to take advantage of vulnerable CPUs and shared memory with the victim. An example is the Meltdown vulnerability in the Intel and ARM chips, which exploited the ability of processors to process instructions out of turn. In this case, the code was executed, even if it contained an error (otherwise in a normal situation it could not be executed). As a result of such speculation, attackers could obtain data from memory. A hacker can use limited network bandwidth to perform a DDoS attack against other applications disclosed in it. In addition, if end-to-end encryption is not configured when transferring data between the cloud and the device, all information stored as "plain text" may be compromised. This happened in 2017 when 2.2 million voicemails left by children to their smart CloudPets toys were stolen by hackers. General equipment outages can also lead to loss of data control. For instance, in 2017, due to the failure of the Amazon S3 cloud storage, not only websites but also Internet of Things devices stopped working. The control over them was lost.

4. Applications and web services [5].

Researchers at the University of Michigan and Pernambuco Federal University analyzed the 37 most popular applications for the Internet of Things and found that:

- 31% of applications do not have encryption;
- in 19% of applications the encryption keys are hard-coded and cannot be changed by the user;
- 50% of all applications are potentially vulnerable to exploits;
- many other programs control devices via local network or broadcast messages, for example, via UDP.

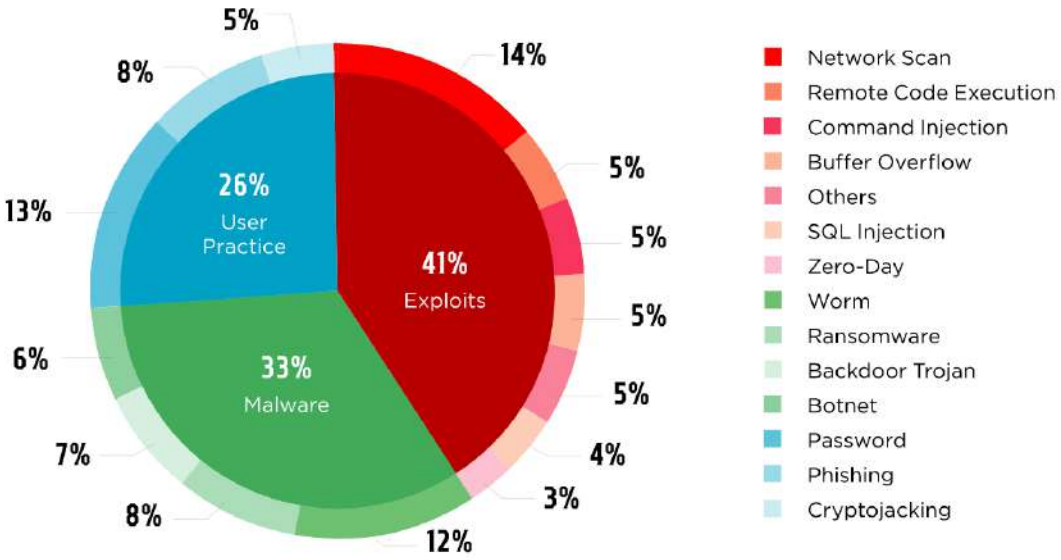


Figure 2: Vulnerability Statistics in IoT Devices According to Palo Alto Networks for 2018

5. Physical vulnerabilities.

None of the computer security measures will help if the device is physically accessible. Even if the data is encrypted when it is saved and the attacker will not have the opportunity to log into the system and infect the device with malware or set a hardware tab, the attacker can always implement a DoS attack using conventional technologies [6].

2.3. Analysis of stages of vulnerability detection

The process of finding vulnerabilities in the firmware according to the OWASP Firmware Security Testing Methodology 2019 consists of the following stages:

1. Collecting the information. At this stage the technical documentation, instructions are studied.

2. Getting the firmware. The firmware can be obtained as follows: from the development team or the client, can be assembled from scratch, using instructions from the manufacturer, can be obtained from the manufacturer's website, can be retrieved directly by hardware via UART, JTAG, PICit, etc.

3. Analyzing the firmware. After receiving the firmware image, the aspects of the file and its characteristics are studied, it is checked whether the file is unencrypted or binary, its entropy is checked.

4. Extracting the file system. Based on the data obtained at the previous stage, the file system (and bootloader) is extracted from the firmware.

5. Analyzing the contents of the file system. At this stage, the data for the stages of runtime analysis and dynamic analysis are collected.

6. Emulating the firmware. Using the data obtained at the previous stages, the firmware, as well as the encapsulated executable files, can be emulated to be checked for potential vulnerabilities.

7. Analyzing the dynamics. At this stage, dynamic testing is performed when the device is operating in a normal emulated environment. Objectives at this stage may vary depending on the project and the level of access given. Typically, this stage includes bootloader configuration analysis, web testing and APIs, fuzzing (network and application services), and active scanning.

8. Analyzing execution time. Runtime analysis involves connecting to a running process or binary file while the device is running in its normal or simulated environment.

9. Binary exploiting. After a vulnerability in a lifelong file has been identified, proper concept validation (PoC) is required to demonstrate the real impact and risk.

After analyzing the stages of identifying vulnerabilities in the firmware and currently available sets of different tools and utilities, it can be concluded that individual stages of the process can be automated in whole or in part.

The stage of the firmware analysis, which mainly uses the following tools and utilities – file, binwalk, strings, hexdump – can be automated. It is necessary to obtain automatically the architecture for which it is compiled, the kernel version, and the operating system version from the binary firmware file. In case of an error in extracting this data, it is necessary to check the entropy of the file. Then all the information obtained is generated in the form of a report and the form of an output file required for the next stages [7-10].

The following tools and utilities are used for the file system extraction stage: binwalk, dd, unsquashfs, cpio, jefferson, ubidump.py, firmware-mod-kit. The binwalk utility (with the -e switch) allows retrieving some file systems automatically, but it does not extract the following file systems: squashfs, ubifs, romfs, jffs2, yaffs2, cramfs, initramfs. For data file systems it is necessary to implement automatic calculation of the offset, using the original data from the previous stage, and then their automatic extraction. A directory with an unpacked file system is the source data for the next stages.

During the file system analysis stage, it is necessary to automate the search for obsolete dangerous services, search in CVE-databases and Exploit-databases by versions of found services, search for hard-coded credentials (usernames, passwords, API keys, SSH keys), firmware update functionality, which can be used as an entry point. At the end of the work, it is necessary to generate the initial data for the next stages and the report with the relevant information.

Due to the introduction of automation of individual stages of the process of detecting vulnerabilities in the firmware of IoT-devices, the speed of the whole process increases. Also, automation eliminates the human factor, which reduces the probability of error, i.e. increases accuracy and completeness.

Thus, the indicators of the effectiveness of the object of the study have been determined as the speed, accuracy, and completeness of the process of detecting vulnerabilities in the firmware. Then, it is necessary to determine the methods and means of measuring these parameters.

2.4. An Example of a Search Algorithm for Vulnerable IoT-Devices

Researchers offer many algorithms for finding hacker-friendly devices, and the most effective ones have already been tested by botnet creators. Using vulnerabilities in botnets is the most reliable criterion for assessing the ease of their mass exploitation in practice.

Someone comes from the firmware (more precisely, from those wild errors detected when analyzing it by the reverse engineering methods). Others take the name of the manufacturer (it can be identified by the first three octets of the MAC address) or the OS version (most devices report it in the network response, including search engine spiders). In any case, for a successful search, it is needed some distinguishing feature of a vulnerable device, and it would be good to find several such markers. Therefore, it is offered to use the following:

1. It should be turned to the database of vulnerabilities, for example, MITER or Rapid7, and interesting gaps in certain IoT-devices can be found. The most vulnerable in terms of use will be the following types of vulnerabilities:
    - those detected after the manufacturer has stopped supporting the device and releasing patches;
    - those recently discovered (for which there are no fixes yet, or most users have not had time to apply the fix);
    - architectural bugs that are poorly fixed by software patches and are rarely fixed such as the Specter vulnerability, which exists in several variants and is still relevant;
    - those that affect several models and even types of devices (for example, due to the common of the component of the web interface or vulnerabilities in the communication protocol itself).

2. Then it should be examined the details of the vulnerabilities found and the devices affected by them. It should be read all available documentation in search of unique markers and details of mistakes made by the developer. It is necessary to determine the features that distinguish the devices one is interested in from the mass of other similar ones. For example, the response from a vulnerable

device contains a line with the number of a particular version of the OS, a revision of the protocol, or it will open a custom port.

3. It should be composed advanced search queries for Google (Google Dorks) and specialized search engines on the Internet of Things:

- Shodan;
- Censys;
- ZoomEye.

Searching for IoT-devices is easy to speed up with scripts, for example, RussianOtter Mult-API Network Scanner or GasMasK. To use them (as well as to use one's own scripts) one will need to register with Shodan and Censys.

4. It should be checked the goals of the search engine, and (if necessary) it should be sifted it with additional queries. This need almost always arises, so scripts are often used to pars the results, for example, here's one from thesubtlety.

5. It should be selected the tools to connect to the found IoT-devices. In most cases, a browser will suffice. To control cameras and DVRs, one sometimes needs to install an older version of Java RE and a specific video codec. Telnet and SSH clients are often required. Less often one needs software from a developer, such as Cisco Smart Install Client.

6. Depending on how far one intends to go, it should limit oneself to collecting statistics or make a test connection and try to change the settings. The latter is not recommended, including the fact that one can easily run into a trap (honeypot). Interpol also needs to raise the rate of detection of crimes in the field of Internet security, and not very careful researcher is an ideal goal.

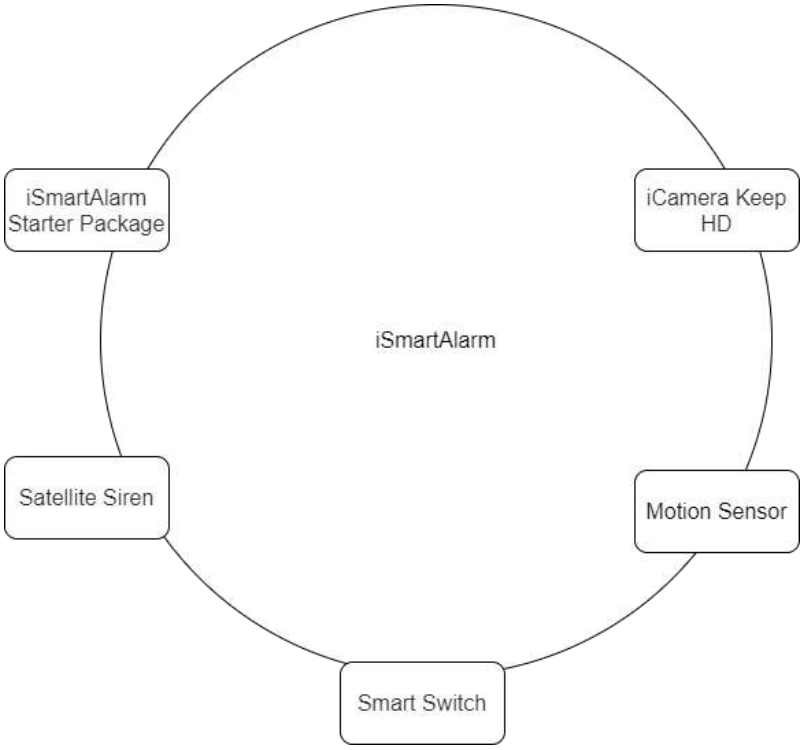


Figure 3: Disabling the Alarm iSmartAlarm

The further full automation of the search for weak or medium vulnerabilities in different types of devices can become the effect of using this algorithm.

With the help of the algorithm presented above, the alarm was turned off iSmartAlarm. Due to the modest computing resources on IoT devices, it is extremely easy to perform a DoS attack. Banal ICMP flooding paralyzes them, which in the case of security systems is no less dangerous than an unauthorized system. For example, the iSmartAlarm Cube home / office alarm contains a number of

vulnerabilities (CVE-2017-7728, CVE-2017-7729, CVE-2017-7730) that allow it to be blocked remotely with one command [7-9].

It is enough to run the hping3 utility (it is included in the new Kali Linux in the Information Gathering → Live Host Identification section) and type the command: $ hping3 --flood -S -p <port> <IP>.

Here —flood is the mode of sending packets without waiting for a response, the -S switch sets the SYN flag, and -p is the port number (by default, it is set as 12345). All! As soon as you hit Enter, ICMP packets will pour into the iSmartAlarm Cube. The alarm will be so carried away by the endless answers to them that it will not work in the event of a physical intrusion (the controller simply will not have time to process the data from the motion sensor in the allotted time). Moreover, it will not be possible to bring the alarm system back to life without rebooting and disconnecting from the Internet either remotely or locally.

If this is not enough, then CVE-2017-7728 allows you to get full remote control over the signaling, since authentication is crookedly implemented in it. The finished PoC in Python is here - thanks to Ilya Schneidman. Plus, there is a good chance on the spot to exploit another vulnerability from the above triad - CVE-2017-7729. iSmartAlarm allows you to intercept the authorization key over the local network, since it is transmitted in an open (unencrypted) form.

At the time of writing PoC, iSmartAlarm has not provided any official comments or patches [9, 10].

3. Conclusions

As the number of IoT-devices grows every year, the number of potential vulnerabilities and threats in the software and hardware of these devices also increases.

How quickly vulnerabilities are detected depends on the speed with which software or hardware is resolved, and therefore there is a need to improve the vulnerability detection process in IoT-devices.

This paper analyzes the terms “IoT”, “IoT device” and “vulnerability”, provides the classification of IoT device vulnerabilities, the causes of IoT device vulnerabilities, identifies the stages of vulnerability detection, and provides the example of a search algorithm for vulnerable IoT-devices.

There are indeed many vulnerabilities in IoT devices, but not all of them are as easy to exploit as in the examples listed above. Some require a physical connection, being nearby or on the same local network. The use of others becomes temporarily difficult after the details are published and until the official patch is released.

On the other hand, manufacturers are in no hurry to patch firmware and generally admit their mistakes. Therefore, there are always enough easy targets. Compiling an accurate list of them will take much more effort than a one-time call to specialized search engines. The lion's share of Shodan, Censys and ZoomEye search results are not related to easily hacked devices. It is simply that the network response of many nodes overlaps with the request of researchers looking for suitable targets.

The real extent of the prevalence of potential targets for botnets can be judged only after an in-depth analysis of search results and direct checks, which are usually neglected.

4. References

[1] S. Kolehmainen, Security of firmware update mechanisms within SOHO routers. University of Jyväskylä, Finland, 2019, pp. 3-97.

[2] B. Jeannotte, A. Tekeoglu, Artorias: IoT Security Testing Framework, in: 2019 26th International Conference on Telecommunications (ICT), Hanoi, Vietnam, 2019, pp. 233-237. doi: 10.1109/ICT.2019.8798846.

[3] Y. Ma, L. Han, H. Ying, S. Yang, W. Zhao and Z. Shi, SVM-based Instruction Set Identification for Grid Device Firmware, in: 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 2019, pp. 214-218. doi: 10.1109/ITAIC.2019.8785564.

[4] S. Prashast, Hui Peng, Jiahao Li, Hamed Okhravi, Howard Shrobe, and Mathias Payer, FirmFuzz: Automated IoT Firmware Introspection and Analysis, in: Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things (IoT S&P'19). 2019, ACM, New York, NY, USA, 15-21. doi: https://doi.org/10.1145/3338507.3358616.

[5] A. Markov, A. Fadin, V. Shvets, V. Tsirlov, The experience of comparison of static security code analyzers, in: International Journal of Advanced Studies. 2015. V. 5. N 3. P. 55-63.

[6] A.V. Barabanov, A.S. Markov, A.A. Fadin, V.L. Cirlov, Statistika vyyavleniya uyazvimostej programmnogo obespecheniya pri provedenii sertifikacionnyh ispytanij [Software vulnerability detection statistics for certification testing]. Voprosy kiberbezopasnosti. 2017. № 2 (20). P. 2-8. [in Russian].

[7] Z. Zhang, M. C. Y. Cho, C. Wang, C. Hsu, C. Chen and S. Shieh, IoT Security: Ongoing Challenges and Research Opportunities, in: 2014 IEEE 7th International Conference on Service-Oriented Computing and Applications, Matsue, 2014, pp. 230-234. doi: 10.1109/SOCA.2014.58.

[8] M. M. Hossain, M. Fotouhi and R. Hasan, Towards an Analysis of Security Issues, Challenges, and Open Problems in the Internet of Things, in: 2015 IEEE World Congress on Services, New York, NY, 2015, pp. 21-28. doi: 10.1109/SERVICES.2015.12.

[9] A. Riahi, Y. Challal, E. Natalizio, Z. Chtourou and A. Bouabdallah, A Systemic Approach for IoT Security, in: 2013 IEEE International Conference on Distributed Computing in Sensor Systems, Cambridge, MA, 2013, pp. 351-355. doi: 10.1109/DCOSS.2013.78.

[10] N.D. Zhou, N. Vlajic, D. Zhou, IoT as a Land of Opportunity for DDoS Hackers, in: Computer, vol. 51, no. 7, pp. 26- 34, July 2018. doi: 10.1109/MC.2018.3011046.

List of Non-Outer Projective Planar Graphs

Volodymyr Petrenjuk[a] and Dmytro Petrenjuk[b]

[a] Centralukrainian national technical university, Universitetskyj 8, Kropivnytskyj, 25006, Ukraine
[b] V.M. Glushkov Institute of Cybernetics of the NAS of Ukraine, Glushkova 40, Kyiv, Ukraine

Abstract

The graph is outer-projective-planar, if embeds on the projective-plane with all vertices on the boundary of one distinguished cell, and non-outer-projective-planar in another case. The main result: diagrams of graphs as a result of the algorithm are given and the numbers of reachability of sets of vertices of minors of the projective plane and sets with points of connection of a star to subgraphs of these minors are calculated. The list of non-outer projective-planar graphs that was declared in [6] has presented here.

Keywords 1

Graph representations, geometric and topological aspects graph theory, projective graph.

1. Introduction

The main notations and definitions are taken from [1]. The problem of search all non-outer projective planar graphs has the following two subtasks.

1. Investigate the structure of projective plane graphs, minimal concerning the operation of removal or contraction to a point of an arbitrary edge, with a given set of points, having the number of reachability t, t = 2, and is itself or has a subset projective planar graphs and give their graph diagrams indicating the specified subsets of points;

2. Investigate the structure of the glueing graph and the algorithm for constructing no projective planar or non-Klein surface graphs as φ-images of a small number of special graphs. Their special graphs are elements of the set of minimums relative to the number of reachability 2 for a given Klein surface or projective plane, having a reachability number of 2 and are minimal relative to the reachability number in the operation of removing an arbitrary point.

The solution of subtask 1 is to construct all minimal non-outer projective planar graphs solved in [2] by searching all different options for deleting one of the vertices of the projective planar minor graph and selecting no isomorphic graphs of nonorientable genus 1. The idea of construction is similar to how minimally non-planar projective graphs K_5 or $K_{3,3}$ are formed from minimal non-outer planar graphs K_4 or $K_{2,3}$ by glueing a simple star St (v) to the minimum power subsets of points of graphs K_4 or $K_{2,3}$ with the number reachability 2. According to subtask 1, the obtained theoretical results are presented in part 1, and in part 2 the algorithm and diagrams of graphs constructed by it are given.

Subtask 2 is to identify the minimum subset of points in the minimum non-projective planar or minimal non-Klein planar graphs with a given number of reachability 2 and the nature of their bonding for another construction of non-projective planar or construction of all non-Klein surface minor graphs. A similar problem was solved in [3], where the coverage of non-projective planar or non-Klein surface graphs G with the number of vertices not more than 10 as obstructions of the nonorientable genus $\gamma(G)$ by subgraphs homeomorphic to K_5 or $K_{3,3}$. Pairs of which inform subgraphs homeomorphic to obstructions of the nonorientable genus is associated in [3] for nonorientable surfaces of the genus, not more than 5, and for the torus also has the specified coating.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine

EMAIL: petrenjukvi@i.ua (V. Petrenjuk); dmytrotheukrainian@ukr.net (D. Petrenjuk)
ORCID: 0000-0001-7313-9642 (V. Petrenjuk)

However, in [4, p. 203] a counterexample is given. In [5] the solution of a similar problem of construction of non-projective plane graphs by the method of relative components is given. Some results on the analogue of this task were given in [8]. The list of non-outer projective planar graphs has presented here.

For graph $\Im$ (obtained as a φ -image $G + St_n(g_0)$ with n vertices of the star $St_n(g_0)$ amalgamate with vertices of the set X having the number of reachability $t_G(X)$ and $\theta_G(X), \partial\theta_G(X)$, the following inequality holds: $\gamma(\Im) \le \gamma(G) + t_G(X) - \theta_G(X) - \partial\theta_G(X) - 1$.

Was introduced a characteristic at $\theta_G(X)$ is a measure of the cyclic connectivity of 2-cells of set $S_G(X)$ as opposed to $\partial\theta_G(X)$ which characterizes the cyclicity of the set $S_G(X)$. They can be used in the analysis of graph models of linguistic circuits which know that vertex and vertex links have some common property-context and some pairs of vertexes may conflict or contradict each other. To resolve these conflicts, we suggest placing graph models on the surface of another kind without crossing the edges at the inner points. In order to investigate the behaviour of a mathematical model of a complex system placed on the orienting surface S, its graph model G without multiple edges and loops is considered. Then it is possible to use the transform method created for graphs to solve modelling problems by splitting into "simpler" submodels with further identification of elements made with predefined properties. So the expansion of model G can be determined by the following transformation:

$$\varphi : (G, St_n(g_0), \sum_{i=1}^{n}{}^{\bullet}(g_i + a_i)) \to (\Im, \{a_i^*\}_{i=1}^n)$$

where $\{a_i\}_{i=1}^n$ is the set X of points of graph G with the number of reachability $t_G(X)$, which is one set for identification and amalgamation, and the other $\{g_i\}_{i=1}^n$ is the set of end vertices of the star $St_n(g_0)$ with the centre g_0. Generalization of the characteristic relating to the cyclic structure of the set X points of the graph G embedded in the surface S. Introduction of a new characteristic that measures the chain structure of the set X of points of graph G on S. This result will be useful in the systematic analysis of both graph models and their topological aspect. which will have common properties at the edges and vertices of the graph model. The solution to our problem is based on the method of graph transformations [1], whose founder is M.P. Khomenko, and the concepts he introduced.

Definition 1. For a given embedding f , $f : G \to S$, a graph G in S and a given set of points X , $X \subset G^0 \cup G^1$ determine $t_G(X, S, f), t = t_G(X, S, f)$, the number of reachability of the set X relative to S , if there is a set $S_G(X), S_G(X) = S \setminus f(G)$, which satisfies the condition: $(f(X) \subseteq \bigcup_{i=1}^{t} \partial s_i \cap X) \wedge (f(X) \not\subset \bigcup_{i=1, i \ne j}^{t} \partial s_i \cap X)$, $j = 1,2,..,t$. We say that the set X has a reachability number t, $t_G(X, S) = t$, relative to S , if among all no isomorphic embedding's f , $f : G \to S$, the number t is the smallest among the numbers $t_G(X, S, f)$. We consider further the set X of points of the graph G t-non-planar concerning the surface S, or (t, S) - non-planar, if $t \ge 2$, where $t_G(X, S) = t$. If $t = 2$, S is a projective plane, and the set X is the set of vertices of the graph G, $X = G^0$, then we will call the graph G non-outer projective planar. A graph G is outer-projective-planar if embeds on the projective-plane with all vertices on the boundary of one distinguished cell.

Definition 2. Suppose the embedding f , $f : G \to S$, of the graph G in the surface S , which implements t, $t_G(X, S) = t$, where $S_G(X) = S \setminus f(G)$ $S_G(X) = \{s_i\}_1^t$. We will say that concerning a given surface S the set X will have the characteristic $\theta_G(X, S, f)$, $\theta_G(X, S, f) = \theta$, $\theta \ge 1$, if there are θ three cells $\{s_i\}_1^3$ from the set $S_G(X)$, on the boundaries of which the subsets X_i, $X_i \subseteq X$, are placed arbitrarily and satisfy the relation: $G^0 \cap \partial s_1 \cap \partial s_2 \supseteq \{a_1\} \wedge G^0 \cap \partial s_2 \cap \partial s_3 \supseteq \{a_2\} \wedge$

$G^0 \cap \partial s_1 \cap \partial s_3 \supseteq \{a_3\}$, and generates the smallest subgraph G' of the graph G, (possibly degenerate), contains the points $\{a_i\}_1^3$ of pairwise intersection of cell boundaries $\{s_i\}_1^3$. The set X will have the θ-characteristic $\theta_G(X)$ if $\theta_G(X) = \max \theta_G(X, f)$, where the maximum is taken for all embedding's $f : G \to S$, realizing $t_G(X, f) = t$ and $\theta = \theta_G(X, f)$.

2. Main Results

2.1. The Mathematical Base for the Algorithm

Theorem 2.1. The graph G is non-outer projective planar if and only if then $G = H \setminus v$, where v is a vertex of graph-obstruction H of the projective plane N_1.

Theorem 2.2.[9]. For an arbitrary graph - obstruction G of the projective plane N_1 and each of its vertices v with the set $M(v)$ of all vertices of the incident occurred the following statements:

1. For the subgraph $G \setminus v$ of the nonorientable genus, the following relations will take place:

a) If $\gamma(G \setminus v) = 1$, then we have the following relations a1) and one of a2) or a3):

a1) $t_{G\setminus v}(M(v), N_1) = 2$, wherein the set $M(v)$ belongs to the boundaries $\partial s_1, \partial s_2$ of two cells s_1, s_2 of the projective plane having at least one common vertex;

a2) each edge of the subgraph $G \setminus v$ is significant in relation to a genus $\gamma(G \setminus v)$ with respect to removing the edge or compressing it in point;

a3) each edge of a subgraph $G \setminus v$ is significant with respect to the removal or compression operations of an edge;

b) If $\gamma(G \setminus v) = 0$ then, one of the following two relationships will occur:

b1) $t_{G\setminus v}(M(v), N_1) = 3$ and the set $M(v)$ is located on the boundaries of three cells s_1, s_2, s_3 of the projective plane satisfying the relation $\partial s_3 \cap \partial s_1 \cap \partial s_2 \neq \varnothing$, each edge of the subgraph $G \setminus v$ being significant relative $t_{G\setminus v}(M(v), N_1)$ to the operations of removing the edge or compressing it to a point, and each point w of the set $M(v)$ satisfies equality $t_{G\setminus v}(M(v) \setminus \{w\}, N_1) = t_{G\setminus v}(M(v), N_1) - 1$;

b2) $t_{G\setminus v}(M(v), \sum_0) = 2$, where $t_{G\setminus v}(M(v), \sum_0)$ is the number of reachability of the set $M(v)$ relative to the Euclidean plane $\sum_0$, is realized by minimal embedding $f : (G \setminus v) \to \sum_0$ at the boundaries $\partial s_1, \partial s_2$ of the cells s_1, s_2, where $\{s_1, s_2\} \subset \sum_0 \setminus f(G \setminus v)$. Satisfies equality $\partial s_1 \cap \partial s_2 = \varnothing$, which is, separated by a ring from the cells, then relative to the projective plane. The set $M(v)$ will have a number of reachability $t_{G\setminus v}(M(v), N_1) = 2$, with each point w of the set $M(v)$ satisfies equality $t_{G\setminus v}(M(v) \setminus \{w\}, N_1) = t_{G\setminus v}(M(v), N_1)$ and the set $f(M(v) \setminus \{w\})$ by some embedding $f' : G \setminus v \to N_1$ is placed at the boundaries $\partial s_1', \partial s_2'$ of two cells s_1', s_2' having at least one common point where $\{s_1', s_2'\} \subset \sum_0 \setminus f'(G \setminus v)$, and equality $\partial s_1' \cap \partial s_2' \neq \varnothing$ is satisfied.

2. Each minor G of the nonorientable genus 2 (except G_3, E_1, G_4) is covered by a maximum of 4 (e.g., graphs A_2, G_1) subgraphs or parts homeomorphic to one of the following graphs: $K_{2,3}$, K_4, $K_5 \setminus e$, $K_{3,3} \setminus e$, K_5, $K_{3,3}$. The number of reachability 2 relatively Klein surface N_2 for the set of vertices (for $G \in \{G_3, E_1, G_4\}$ we have), and for each removed edge e the graph $G \setminus e$ will have at N_1 the number of reachability equals 2 for the set of vertices;

3. The presence of the coating specified in statement 2 is not sufficient to make the graph an obstruction of nonorientable genus 2.

4. If $\gamma(G \setminus v) = 0$ and on the Euclidean plane $\sum_0$ made up a set $M(v)$ of points of a graph G formed from the obstruction graph of a projective plane N_1 by removal of a vertex v and adjacent edges. If is given by an arbitrary minimal embedding $f : G \setminus v \to \sum_0$ on the boundaries of two cells that

have no common points and have endpoints that do not belong to their borders. Removing an arbitrary point from the set M leads to the failure of relation 4.

2.2. Algorithm

The construction of all no isomorphic non-outer projective plane graphs based on the results of the following polynomial algorithm 1:

Begin of Algorithm 1.

Input: The set P of 35 minors P_i of the projective plane N_1 with the sets of numbered vertices, which for each graph P_i is divided into equivalence classes l_{ij} with respect to the transitivity of its vertices, where $P_i^0 = \sum_{j=1}^{n_i} l_{ij}, n_i \leq |P_i^0|$.

Output: List X of all no isomorphic graphs.

$X := \varnothing$;

$v := 0$;

// where P_0 is the current graph of the order $|P_0|$ with the selected vertex, which is a representative of the transitivity class l_{0j} of its vertices.//

For i =1 step 1 to 35, do these steps:

begin //cycle action on i.

$P_0 := P_i$;

$v := v_{i1}$;

procedure A(P_0, Π_0, P_0^0, N_2);

// Procedure A(P_0, Π_0, P_0^0, N_2);; Construct the embedding of the graph G in the surface S (projective plane N_1 or Klein bottle N_2) and determine the cells of the graph at the boundaries of which is a given subset M of the set of vertices of the graph G with which the incident vertex v //;

Output ($P_0, \sum_{j=1}^{n_i} l_{ij}$) in X ;

For k = 2 step 1 to $|P_0|$, do these steps:

begin

If $v \approx v_{ik}$ then go to the end of the cycle by k;

// that is, the vertices belong to the same class of transitivity; /

else $P_0 := P_i \setminus v$; // remove the vertex V and all adjacent edges; /

$\Pi_0 := \Pi_i$;

L: = Function B (P_0 , X);

If L == true // graph $P_i \setminus v$ no isomorphic to any of the graphs in the list X //

then do:

begin;

$M := \{\forall u \mid (u,v) \in P_0^1\}$;

If K (G) == 1 // the graph $P_i \setminus v$ contains a subgraph of Kuratowsky //

then do

begin;

procedure A(P_0, Π_0,M, N_1);

output (Π_0, M) in X ;

end;

else do

begin;

procedure A(P_0 , Π_0 ,M, Σ_0);
output (Π_0 , M) in X ;
end;
end; // of cycle by k ;
end; // of cycle on i ;
End of Algorithm 1.

Procedure A(G , Π , M, S) do the following:

// Must construct the embedding Π of a graph G (without vertices of degree 2) with a given number of vertices in the surface S (Euclidean plane, projective plane or Klein surface) and determine the cells on the boundaries of which are the set of vertices M //.

If a graph G has a subgraph or part of the graph H is homeomorphic K_5 or $K_{3,3}$, then we construct embedding's of these graphs in the projective plane, otherwise, we attach a graph to the Euclidean plane Σ_0. In nested graphs K_5 or $K_{3,3}$ a projective plane, there are cells s_5 , s_{33} with the following boundaries: ∂s_5 - a cycle of length 5 and 5 triangles for K_5 , or ∂s_{33} - a cycle of length 6 and 4 quadrilaterals for $K_{3,3}$, in which we will embed stars with centres taken from a subset $G^0 \setminus H^0$.

First of all, we will put all these stars in cells with either cycle boundaries of length 5 for or length 6 for and try to use no more than one additional Mobius strip glued to the cells ∂s_5 or ∂s_{33}. The number of vertices $|G^0|$ of the obstruction graph of the projective plane is at least 12. The number of options for the location of the centres and edges of stars, not more than 7 stars, is equal r^7 because each centre of the star does not belong to two cells, where r the number of cells of the graph embedded in the projective plane $r = 6$ for K_5, $r = 5$ for $K_{3,3}$.

The time complexity of procedure A(G , Π , M, S) is proportional $O(r^7)$.

The function K(G) will determine the presence or absence of a graph G of a subgraph or part of a homeomorphic K_5 or $K_{3,3}$ and will give it out. To do this, we need to examine the complement of the $\overline{G}$ graph G for the presence of a subgraph of five isolated vertices, i.e. $\overline{K}_5$, or two triangles without common vertices, i.e. $2K_3$. If such subgraphs of the graph are detected, the function K (G) will give 1 and return to algorithm 1 the found vertices as vertices of the graph K_5 or $K_{3,3}$. In the absence case $\overline{K}_5$, $2K_3$ the function K (G) will give 0.

The function B(P_0 ,X,) checks for the presence of an isomorphism of a graph P_0 with another element of the set of graphs X and will have polynomial complexity [7].

2.3. Data analysis of Work of Algorithm

The output data of algorithm 1 is described in figures 1, 2, 3, 4, 5, 6. The analysis of output data of algorithm 1 in the next corollaries.

Corollary 2.1. The correctness of algorithm 1 will follow from Theorem 2.1 and Theorem 2.2. [9] .

Corollary 2.2. The next 78 non-outer projective planar graphs have number reachability of their set of vertices equal 2:

1. There are 16 graphs with genus 0: $E_{20}\backslash 8$, $E_{22}\backslash 5$; $F_1\backslash 1$, $F_1\backslash 2$, $F_1\backslash 3$, $E_2\backslash 6$, $D_{17}\backslash 6$, $C_4\backslash 5$ (has another set of glueing red vertexes to endpoints of $St_6(5)$ then $E_{22}\backslash 5$), $D_3\backslash 2$, $D_3\backslash 4$, $D_2\backslash 4$, $A_2\backslash 4$, $B_1\backslash 4$, $B_3\backslash 1$, $B_7\backslash 4$, $C_3\backslash 4$.

2. There are 62 graphs with nonorientable genus 1 as:

$F_6\backslash 2$, $F_6\backslash 3$, $G_1\backslash 5$, $E_{19}\backslash 2$, $E_{20}\backslash 9$, $E_{22}\backslash 1$, $E_{22}\backslash 2$, $E_{27}\backslash 3$, $E_{27}\backslash 7$, $F_1\backslash 9$, $E_2\backslash 1$, $E_2\backslash 5$, $F_1\backslash 8$, $E_2\backslash 8$, $A_1\backslash 2$, $E_5\backslash 1$, $E_5\backslash 2$, $E_6\backslash 1$, $E_6\backslash 2$, $E_6\backslash 7$, $E_{11}\backslash 1$, $E_{11}\backslash 5$, $E_{11}\backslash 6$, $E_{11}\backslash 8$, $E_{18}\backslash 2$, $E_{18}\backslash 1$ (is subgraph of $E_{18}\backslash 2$), $D_4\backslash 1$, $D_4\backslash 5$, $D_4\backslash 7$, $D_9\backslash 4$,

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

$D_{12}\backslash 6$, $D_{12}\backslash 8$, $C_4\backslash 1$, $C_4\backslash 2$, $C_7\backslash 2$, $D_3\backslash 1$, $D_3\backslash 6$, $D_3\backslash 8$, $D_2\backslash 1$, $D_2\backslash 2$, $D_2\backslash 5$, $A_2\backslash 6$, $E_1\backslash 8$, $E_1\backslash 1$, $E_2\backslash 1$, $B_1\backslash 6$, $B_3\backslash 5$, $B_7\backslash 1$, $B_7\backslash 2$, $B_7\backslash 6$, $B_7\backslash 3$, $B_7\backslash 5$, $B_7\backslash 7$, $C_1\backslash 8$, $C_1\backslash 5$, $C_1\backslash 1$, $C_2\backslash 3$, $C_3\backslash 5$, $C_3\backslash 1$, $C_3\backslash 2$, $C_3\backslash 5$, $C_3\backslash 7$.

Corollary 2.3. The next 41 non-outer projective planar graphs with genus 0 have number reachability of their set of vertices equal 3 and:

a) θ - characteristic equal 1: $F_6\backslash 1$, $F_6\backslash 4$, $G_1\backslash 3$, $E_{19}\backslash 1$, $E_{19}\backslash 5$, $E_{19}\backslash 6$, $E_{20}\backslash 7$, $E_{20}\backslash 1$, $E_{20}\backslash 8$, $E_{22}\backslash 2$ (for red vertices only), $E_{27}\backslash 2$, $E_{27}\backslash 6$, $F_1\backslash 5$, $E_2\backslash 2$, $E_2\backslash 4$, $E_2\backslash 6$, $E_2\backslash 10$, $E_3\backslash 1$, $E_5\backslash 2$, $E_6\backslash 5$, $E_{11}\backslash 2$, $E_{11}\backslash 7$, $D_4\backslash 3$, $D_4\backslash 6$, $D_9\backslash 1$, $D_9\backslash 5$, $D_{12}\backslash 2$, $D_{12}\backslash 5$, $C_7\backslash 1$, $C_7\backslash 3$, $D_4\backslash 5$, $D_2\backslash 8$, $D_{17}\backslash 6$, $E_2\backslash 4$, $E_2\backslash 2$, $B_7\backslash 8$, $C_2\backslash 2$, $C_3\backslash 9$.

b) θ - characteristic equal 0: $A_1\backslash 1$, $C_4\backslash 3$, $E_1\backslash 6$.

Corollary 2.4. The next 11 non-outer projective planar graphs with genus 1 have number reachability of their set of vertices equal 3 and θ-characteristic equal 1: $E_{27}\backslash 5$, $E_{27}\backslash 9$, $F_6\backslash 3$, $E_3\backslash 7$, $E_{11}\backslash 3$, $E_{11}\backslash 4$, $D_9\backslash 2$, $D_{12}\backslash 3$, $D_4\backslash 10$, $D_4\backslash 6$, $C_2\backslash 9$.

3. Graphs

The following figures 1, 2, 3, 4, 5, 6, 7 is presented the result of algorithm 1.

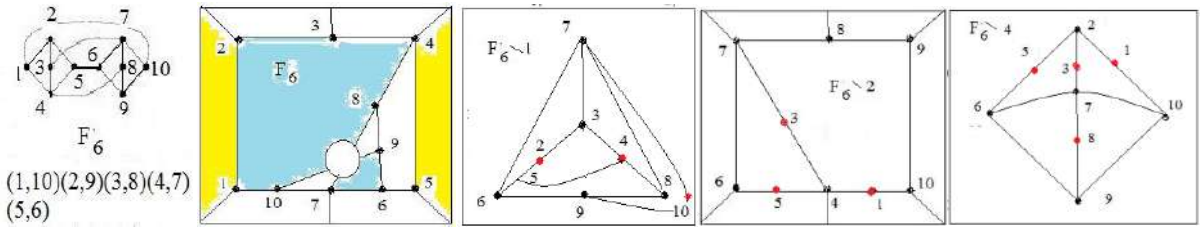
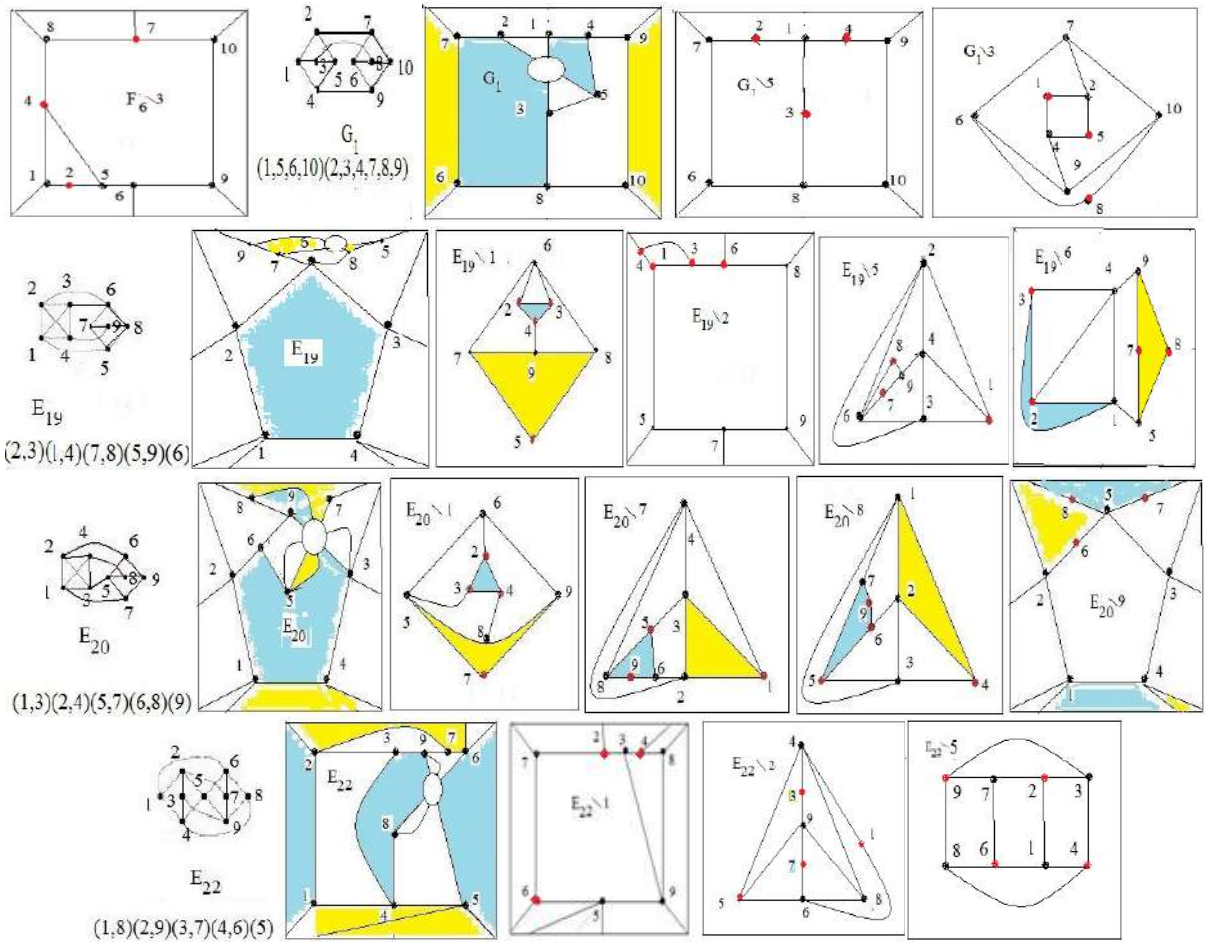


Figure 1



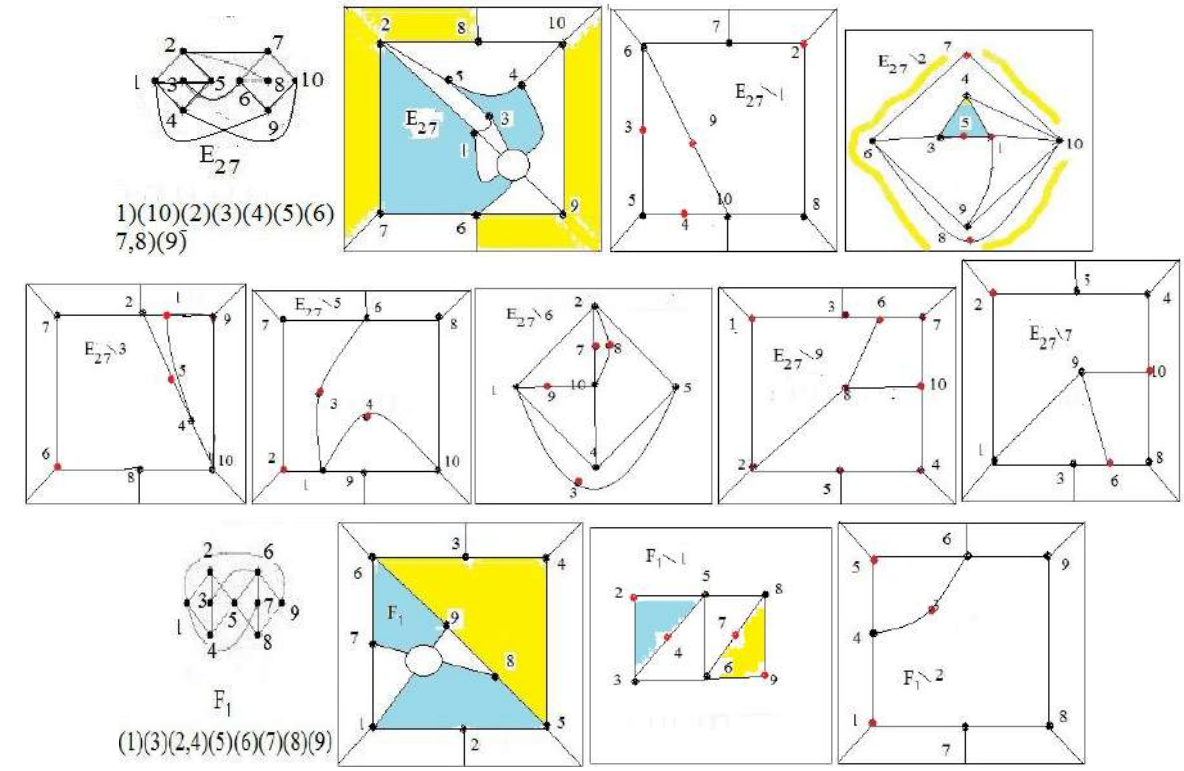
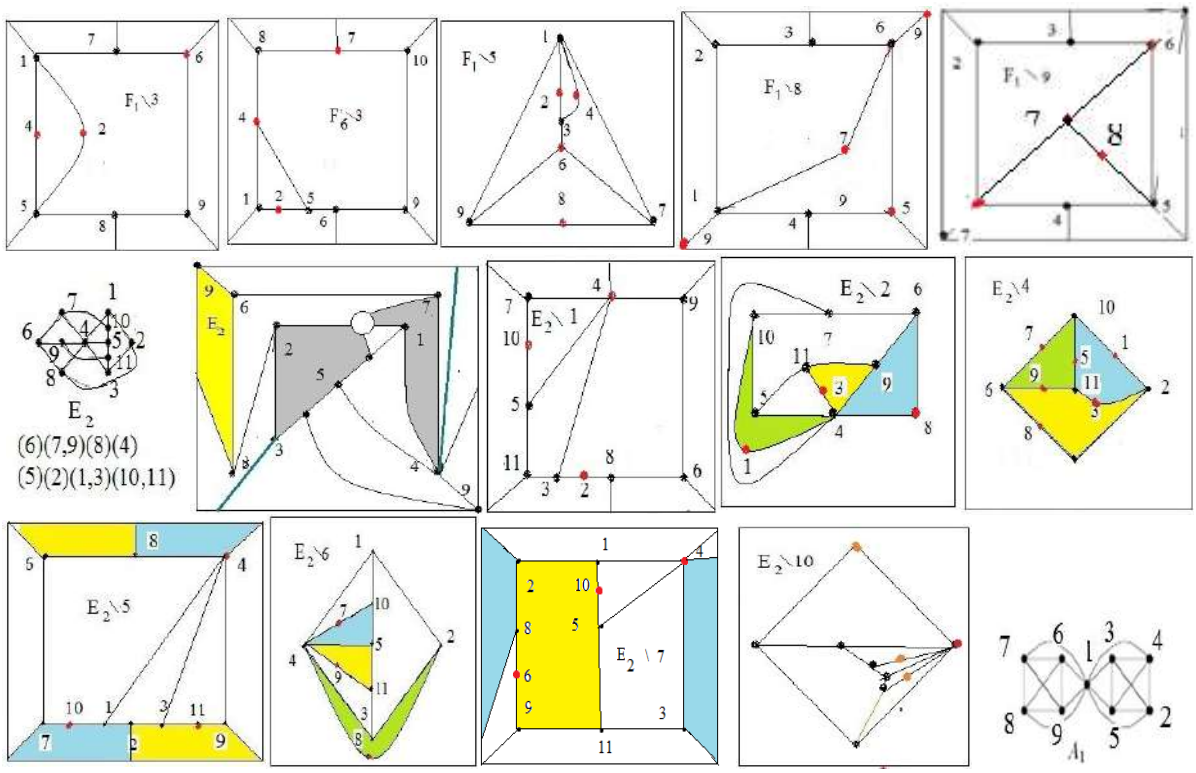


Figure 2



COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

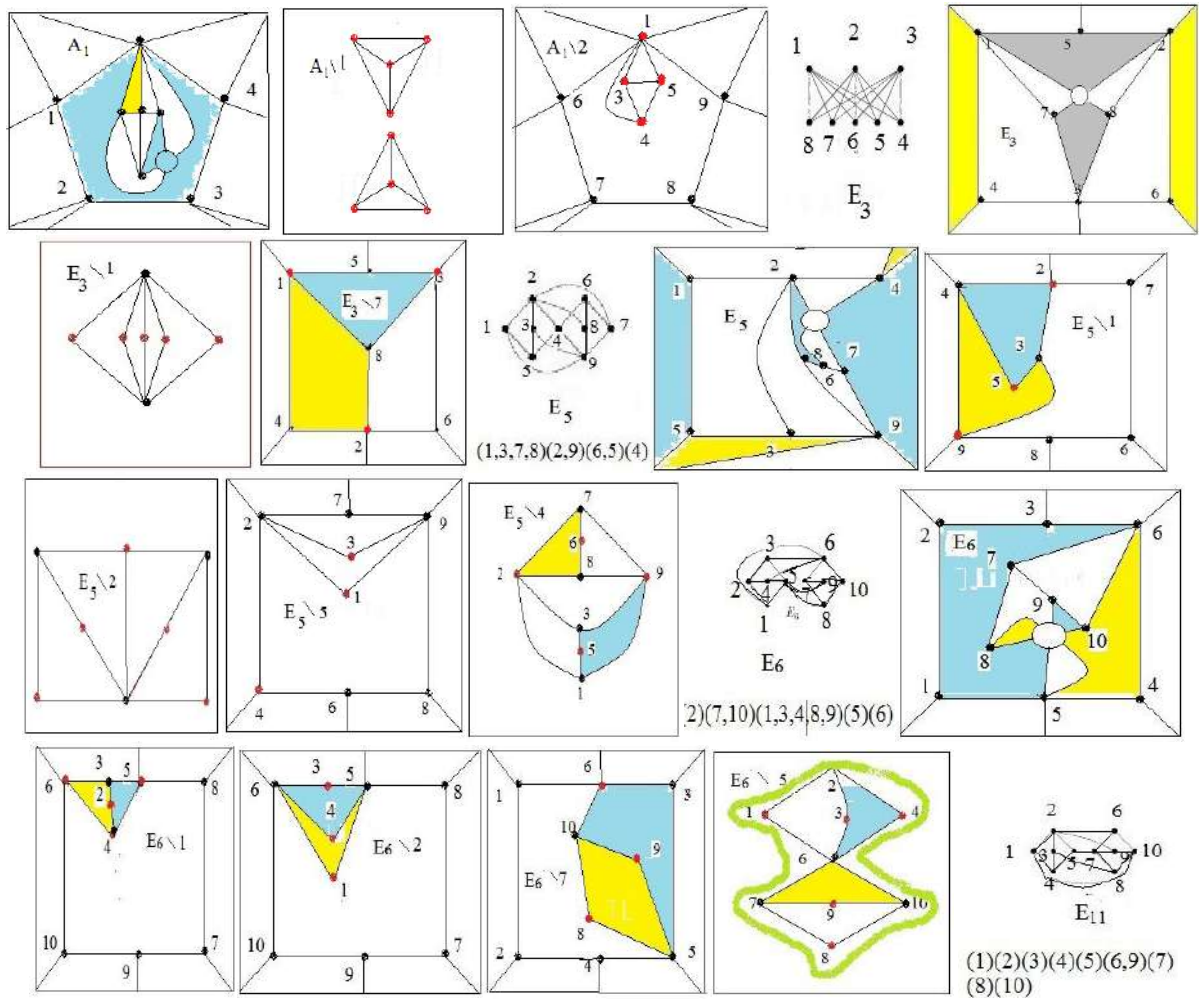
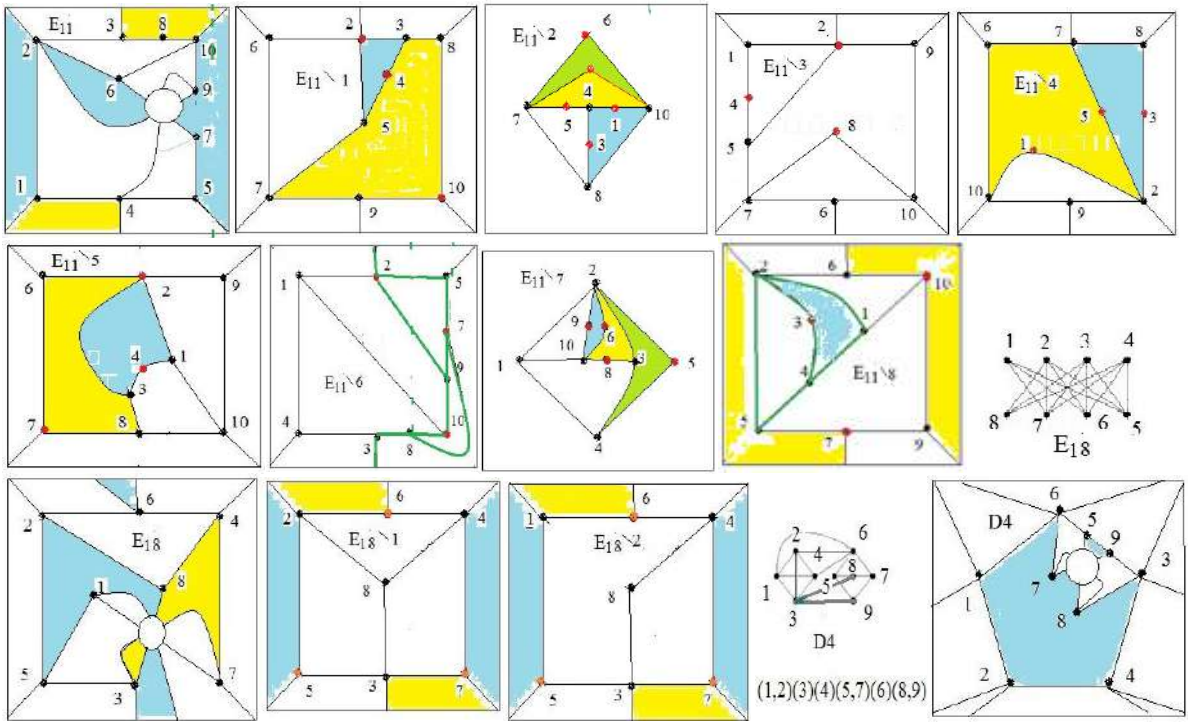

A1
A1\1
A1\2
E3
E3\1
E3\7
E5
(1,3,7,8)(2,9)(6,5)(4)
E5\1
E5\2
E5\3
E5\4
E6
2)(7,10)(1,3,4|8,9)(5)(6)
E6\1
E6\2
E6\7
E6\5
E11
(1)(2)(3)(4)(5,6,9)(7)
(8)(10)

Figure 3


E11
E11\1
E11\2
E11\3
E11\4
E11\5
E11\6
E11\7
E11\8
E18
E18\1
E18\2
D4
(1,2)(3)(4)(5,7)(6)(8,9)

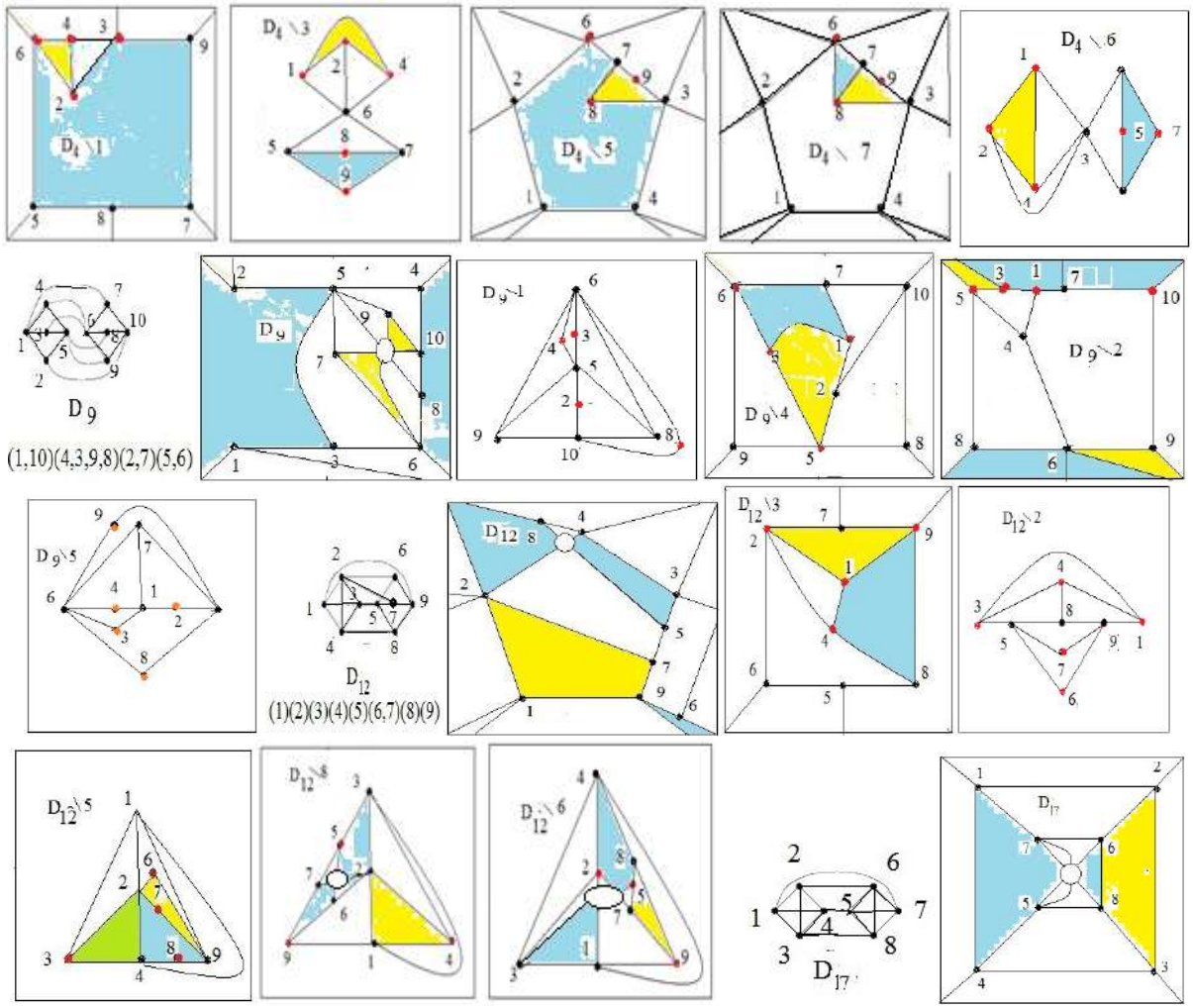
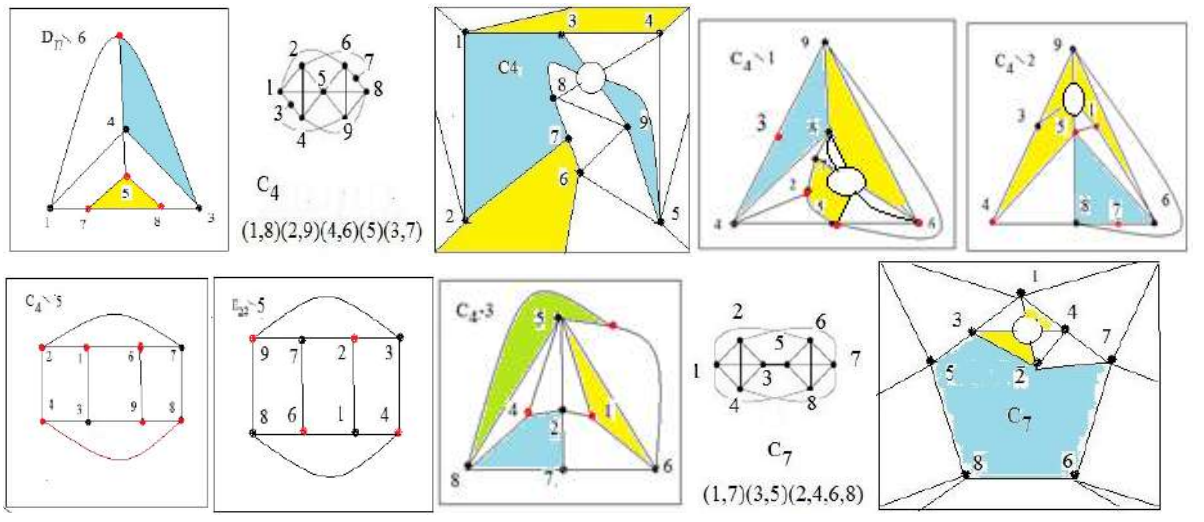


Figure 4



COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

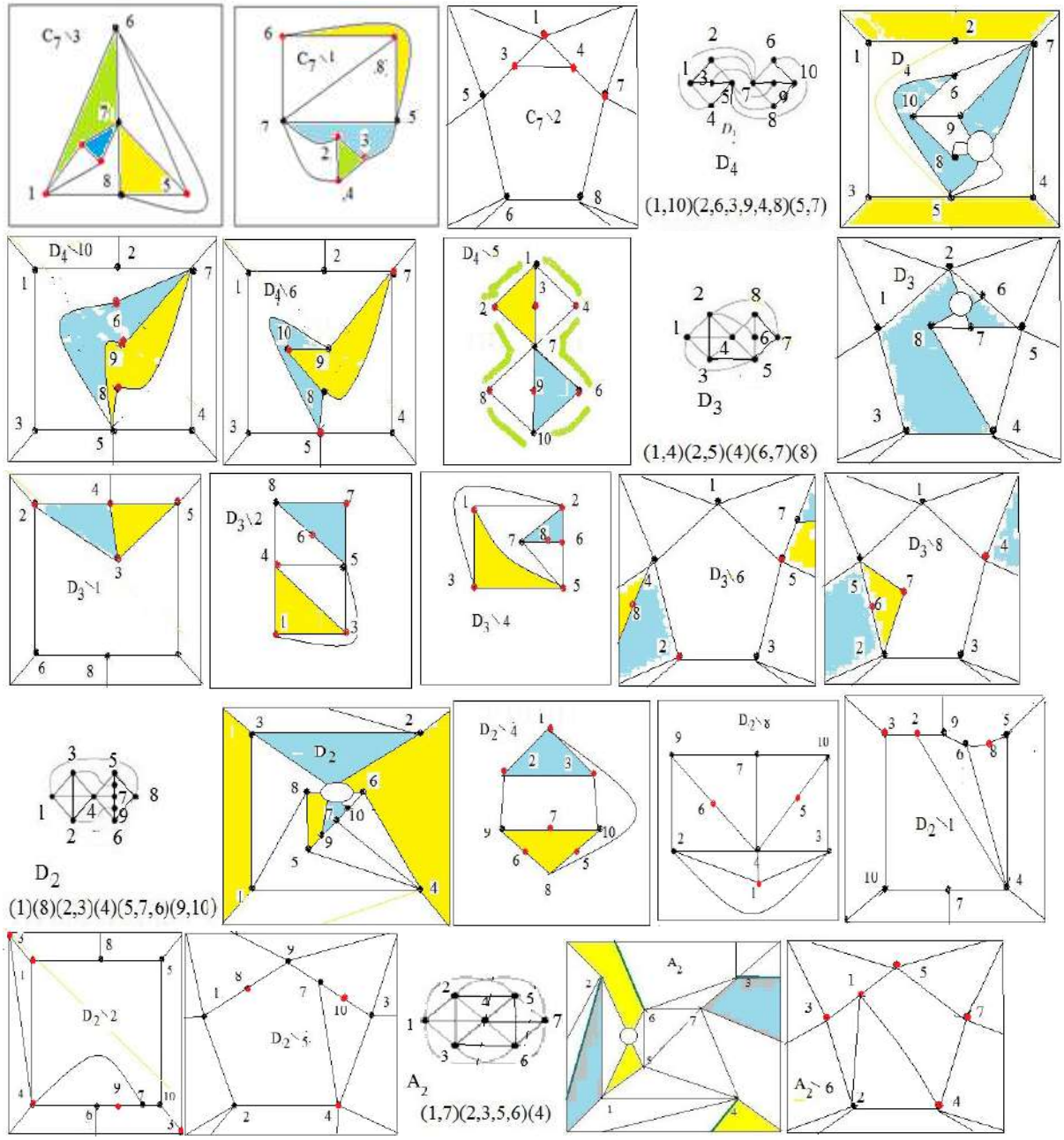
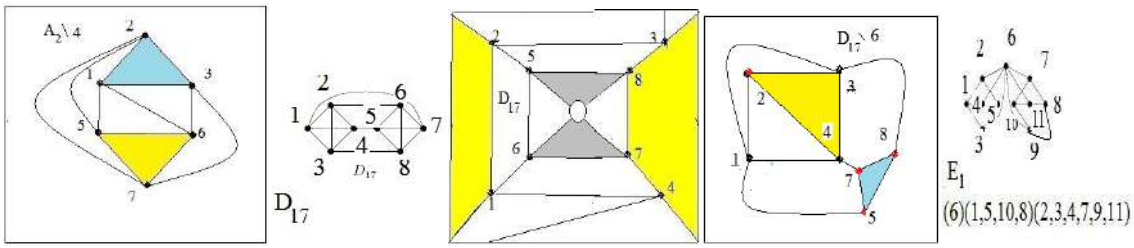


Figure 5



COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

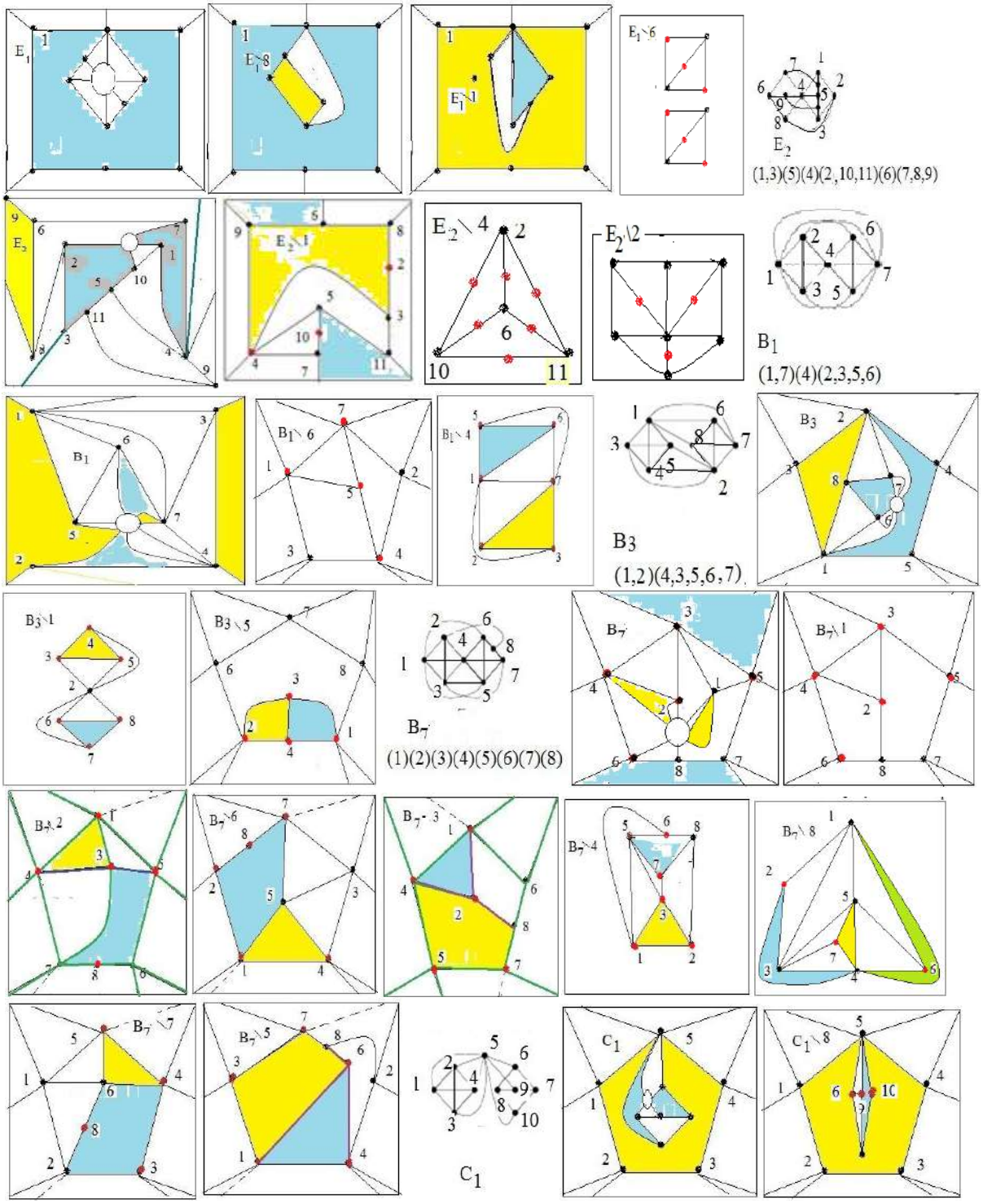

(1,3)(5)(4)(2,10,11)(6)(7,8,9)
B1
(1,7)(4)(2,3,5,6)
B3
(1,2)(4,3,5,6,7)
B7
(1)(2)(3)(4)(5)(6)(7)(8)
C1
C2
(1,3,4)(9)(5,6,8)(2,7)

Figure 6

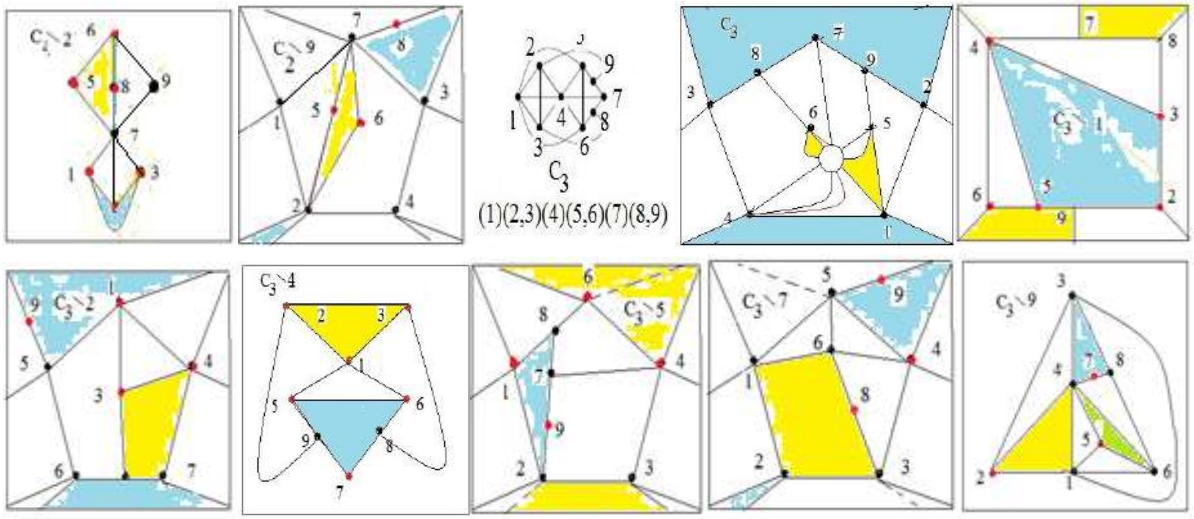


Figure 7

4. Acknowledgements

The authors gratefully acknowledge the support and help of chairs of the organizing committee of the Colins conference 2021.

5. References

[1] M.P. Khomenko, φ - transformation of graphs. Institute of Mathematics, Kyiv, 1973.
[2] D. Archdeacon, N. Hartsfield, C. H. C. Little, B. Mohar, Obstructions sets for outer-projective - planar graphs. Ars Combinatoria, 1998, 49, 113-128.
[3] Hur Surkhjin, The Kuratowski covering conjecture for graphs of the order less than 10. Dissertation, The Ohio State University, 2008. URL: https://etd.ohiolink.edu/rws_etd/send_file/send?accession=osu1209141894&disposition=inline
[4] Bojan Mohar, Carsten Thomassen, Graphs on surfaces, Johns Hopkins University Press, 2001.
[5] Anna Flötotto, Embeddability of graphs into the Klein surface. Dissertation, Universitat Bielefeldvorgeleg, 2010.
[6] V. Petrenjuk, About Transformation graphs as a tool for investigation. Proceedings of the 4th International Conference on Computational Linguistics and Intelligent Systems (COLINS 2020). Volume I: Main Conference Lviv, Ukraine, April 23-24, 2020,1309-1319. URL: http://ceur-ws.org/Vol-2604/
[7] LEDA: A library of efficient data types and algorithms, Max Planck Institute for Computer Science. URL: http://www.mpi-sb.mpg.de/LEDA/
[8] K. Scott, Outermobius and cylindrical graphs. Senior Thesis, Princeton University, 1997.

Discourse Markers as Means of Compositional Integrity in English Last Wills and Testaments

Olha Kulyna

Lviv Polytechnic National University, 12 Bandery street, Lviv, 79000, Ukraine

Abstract

A Last Will and Testament as a legal document of Inheritance Law is of particular importance for the life of modern societies of all developed and underdeveloped countries. The research focuses on the complex analysis of the study of English Last Will and Testament as a social and communicative phenomenon which is a repetitive speech act that generates a typical linguistic layout of the content to meet the communicative needs of a testator/testatrix on the issue of the inheritance of property and money after their death in the situation of bequest. The corpus of the research contains 400 wills written in England between 1837 and 2015 (525 023 characters). Attention is paid to discourse markers which provide structural integrity of the text in wills. The main aim of this article is to conduct the analysis of discourse markers found in English Last Wills and Testaments. The classification of discourse markers by B. Fraser has been used in the study. A structural method has been applied to single out groups of discourse markers. Discourse markers of sequence as a subtype of discourse activity markers, parallel discourse markers, contrastive discourse markers, elaborative discourse markers and inferential discourse markers as subtypes of message relationship markers are common in the texts of Last Wills and Testaments. These markers complement the content of a previous statement, combine parts of a sentence, introduce new information, contrast events, actions and even participants. The usage of discourse markers facilities communication and ensures the compositional integrity of the text.

Keywords 1

Last Will and Testament, discourse marker, compositional integrity, structural method.

1. Introduction

Throughout the history of England, there have been many laws on the disposal of personal property by will in ecclesiastical law, uncodified law and Anglo-Saxon law. The Wills Act (1837) provides the right to bequeath one's personal movable and immovable property to every adult resident of the United Kingdom [Wills act]. Most of the provisions of the Act of 1837 are still in force in England and Wales. According to Art. 9 of this act, the will must be composed in writing, signed by the testator (or another person on their behalf) and witnessed by at least two persons [26].

The research focuses on compositional structure of English Last Wills and Testaments which is considered to be a social and communicative phenomenon and reflects the socially determined needs of a testator in the situation of bequest (the issue of the inheritance of property and money after the testators' death) [2, p. 147-157]. Attention is paid to discourse markers which provide structural integrity of texts of Last Will and Testament.

The aim of this article is to provide the detailed analysis of discourse markers types in English Last Wills and Testaments and to show that they are means of gaining the structural unity on the level of text structure.

To accomplish the aim, the following tasks have been set:

1. to illustrate a typical structure of an English Last Will and Testament;

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: olha.v.kulyna@lpnu.ua (O. Kulyna)
ORCID: 0000-0002-2334-0660 (O. Kulyna)

2. to identify the notion of discourse marker from view point of foreign and Ukrainian linguists;
3. to provide the insights into the types of discourse markers;
4. to analyse the structural elements of textual integrity of English Last Wills and Testaments gained by discourse markers.

The **object** of the research is texts of Last Will and Testament written in England in the period between 1837 and 2015.

The **subject** of the research is discourse markers which provide structural unity and integrity for Last Wills and Testaments.

The corpus of the research contains 400 English Last Wills and Testaments written between 1837 and 2015 (525 023 characters). Last Wills and Testaments written before 1859 were obtained from National Archive of Great Britain (www.nationalarchive.gov.uk). Last Wills and Testaments composed after 1858 were gained at government website of the United Kingdom (www.gov.uk/search-will).

The **novelty** is provided by the fact that discourse markets as means of structural integrity of texts of wills have been studies for the first time in linguistics. These markers complement the content of a previous statement, combine parts of a sentence, introduce new information, contrasts events, actions and people.

2. Methodology

General scientific methods such as descriptive, idealization, modelling and contextual-interpretation analysis have been used in the research. Methods of observation, comparison, classification, generalization and interpretation which are essential for a descriptive method are used to provide structure of a will, to discuss discourse markers and classify them into relevant groups. The understanding of criteria is significant at this stage. The electronic form of the experimental array of texts enabled the usage of a method of automated searching of certain linguistic units (MATLAB) and the establishment of the frequency of their usage, the results of which, however, required further manual processing (Excel). Calculation and means of systematization played an important role to help analyse the corpus of the research.

3. Composition and title of English Last Will and Testament

All wills have a typical composition: title, introduction of a testator, testamentary part, nomination of fiduciaries (executor(trix), trustees) and self-proving affidavit are obligatory. Spiritual part and revocation of previous wills are optional [3, p. 107-113]. Figure 1 shows a composition of English Last Will and Testament.

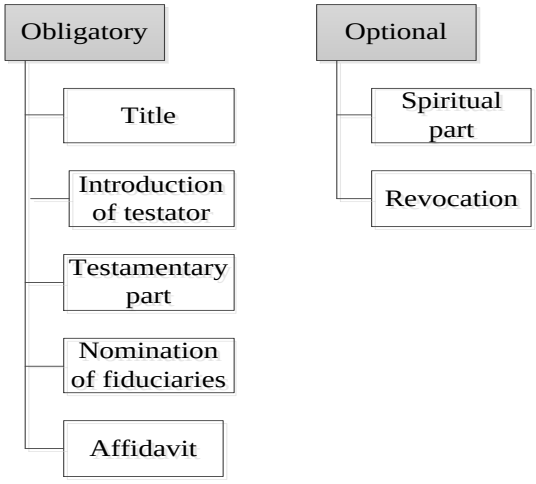


Figure 1: Composition of Last Will and Testament

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

The analysis of wills shows that they have the common title *Last Will and Testament*. There are two explanations for the origin of this term. The first theory emphasizes the meaning of the words *will* and *testament*. The name was used from the XVI century [23, p. 694]. *Will* was used to bequeath real estate while *testament* – personal property [6, p. 694]. Another theory emphasises the etymology of words and states that these words are synonyms. *Will* is derived from the Old English Word *willa* and means *desire, wish, longing, liking, inclination, disposition to do something, mind, determination, purpose, request, joy, delight, and testament* is its Latin synonym *testamentum* which means *last will, publication of a will*) [5, p. 221]. *Will* as a document which expressed a person's will to despose his/her property after death was first recorded at the end of the XIV century, and *testament* in the same meaning was first dated in XIII century [ibid]. It is worth noting that the term *testamentum* in the meaning of *treaty* and *will* was used collaterally in the Christian tradition [5, p. 221–222]. The Greek term διαθήκη, ης, ἡ denoted will (covenant) in early Christian Latin. This term means an official statement chiefly in writing expressing a person's desire to dispose the property after his/her death in the study of law. The term *testamentum* is not only a legal term. It is also used in theological texts as God's instruction and a part of Bible (*Old Testament ma New Testament*) [5, p. 222].

C. Ferland called the first theory "historical" and stated that there was no focus on the differences in terms in the modern world but a word combination is used as an official title for naming a legal document to dispose real and personal property. The author also stated that last marks the last will of the testator and revokes the previous will [15]. K. Sneddon refuted such an explanation and believed that a three-word title was formed rather for sake of harmony and is one of the features of the Last Will and Testament genre [23, p. 695–696]. The author added that these words combined in the title in different ways by 1500s: *testament and will, testament and last will and even testament and latter will*. A treatise of Testaments and Last Wills by H. Swinburne and J. Wake published in 1743 accepted that only the title *Last Will and Testament* has been preserved and is still being in use [24].

Acts of the English Parliament now and past in history use the term will. The attribute last indicates the importance of the document, but it doesn't state that the will is final. A testator can revoke or change his/her will before the death. So, it is unknown which will is considered to be the last. We cannot agree with K. Sneddon on the question of euphonious since the term is a binominal construction which is common for legal texts.

4. Discussions

Language dynamics is of great interest in linguistics and focuses on the organization of the text [1 to help readers follow and comprehend the information presented.

The compositional integrity of Last Will and Testament texts is ensured by discourse markers which acquire meaning in context. Discourse markers are rather a complex object of linguistic analysis, and there is no generally accepted definition and even the terms used to denote these special words differ. Discourse markers belong to different parts of speech and take on different meanings in context; their meanings are often different from those provided in dictionaries. The main function of these markers is to link parts of the text.

In linguistics there are several names to denote them: 1) discourse connectives [8; 20, p. 452]; 2) discourse markers [21; 22; 16, p. 932]; 3) discourse operators [16; 17; 18; 19]; 4) discourse words [8]; 5) pragmatic connectives [9; 10; 11; 12; 13]; 6) cue phrases [14].

D. Shiffrin defines discourse markers as means independent of the sentence structure, which serve for coherence of parts of the text [22, p. 35–40]. B. Fraser interprets them as a pragmatic class, lexical utterances that show the relationship between discourse segments, emphasize and mark aspects of communication which a communicator wants to convey [16, p. 940]. According to E. Traugott, discourse markers are discursive and deictic units that form utterances, but not the context itself and have a metatextual function [25, p. 6].

In this article discourse markers are considered as lexical means that belong to different parts of speech, have different meanings in context and serve to link parts of the text in English Last Wills and Testament.

D. Shifrin was the first to make a detailed analysis of the following discourse markers: *and, because, but, I mean, now, or, you know, oh, so, then, well* [22, p. 35–40].

B. Fraser claimed that now, I mean, oh, you know are not discourse markers [16, p. 933]. He proposed a classification that we take as a basis for the analysis of discourse markers in wills:

1. markers which signal aspects of topic change (***topic change markers***): *back to my original point, by the way, on a different note*;
2. markers which signals the current discourse activity (***discourse activity*** markers) which express: 1) clarifying (by way of clarification, to *clarify*); 2) conceding (*admittedly, after all, all in all, all the same, anyhow, anyway, at any rate, besides, for all that, in any case/event, of course, still and al*); 3) explaining (*by way of explanation, if I may explain, to explain*); 4) interrupting (*if I may interrupt, to interrupt, not to interrupt*); 5) repeating (*at the risk of repeating myself, once again, to repeat*); 6) sequencing (*finally, first, in the first place, lastly, next, on the one/other hand, second, to begin, to conclude, to continue, to start with*); 7) summarising (*in general, in summary, overall, so far, summarizing, summing up, thus far, to sum up, at this point*);
3. message relationship markers which signal the relationship of the basic message being conveyed by current utterance to some prior message (***message relationship marke***rs): 1) parallel (*also, alternatively, analogously, and, correspondingly, equally, likewise, or, otherwise, similarly, too*); 2) contrastive (*all the same, but, contrariwise, conversely, despite, however, I may be wrong but, in spite of, in comparison, in contrast, instead, never/nonetheless, notwithstanding, on the one/other hand, on the contrary, rather, regardless, still, that said, though, well, yet*); 3) elaborative (*above all, also, besides, better, for example, for instance, further (more), in addition, in fact, in other words, in particular, indeed, more accurately, more importantly, more precisely, more specifically, more to the point, moreover, namely, on top of it all, to cap it all off , what is more*); 4) inferential (*accordingly, as a consequence, as a result, consequently, hence, in this/that case, of course, so, then, therefore, thus*) [17, с. 27−31].

Discourse markers of the first group are not present in the texts of Last Wills and Testaments. Among the markers of the second group, sequencing discourse markers predominate: *first, in the first place, lastly, next, on the one/other hand, finally*. For example:

***First**, I will and direct the payment of all my just debts, funeral and testamentary charges…* (John Moore, 1859);

***In the first place** I direct my just debts funeral and testamentary expenses to be paid* (Mary Spragg, 1866);

***Lastly** I revoke all other Wills and declare this to be my last Will* (Alice Lewis, 1881);

In the next place to pay the following legacies to my children (Thomas Spragg, 1860);

*And **lastly** I appoint my Trustees Executors of this my Will* (Ernest William Tranter, 1915).

Message relationship markers are of most importance in English Last Wills and Testaments. The most common are parallel discourse markers, namely *also, and, or, equally, likewise*. The most numerous is *and*. For example:

***And** I direct that as soon as… **And** as to for and concerning all the residue… **And** subject to the trust aforesaid upon…**and** upon further trust that the said trustee… **And** that my trustees or trustee may…* (James Beckett, 1854);

***And** I appoint Herbert Smith Esq, Ruardean Hill Drybrook, Gloss Executors of this my Will* (Benjamin Hope, 1916).

Also occurs in different positions and in different parts of a will (most often in a testamentary part) and functions to indicate another action:

*I **also** give and bequeath to my said wife…* (James Beckeyy, 1854);

***Also** I give and devise unto my said wife and her assogns for the term of her natural life all my real estate of every description* (John Almond, 1855);

*Certain monies will **also** be due from Arthur Foxall Esq tenant of Selsley, Bushey Heath, particulars of which enclosed* (Joseph Samuel Demmery, 1915);

*Provided that my Trustees shall **also** have power to meet any expenses which they may incur in the exercise of any of their powers in respect of chattels out of the capital and income of my estate…* (Diana Proncess of Wales, 1993).

Also is often used when a testator describes actions of various gifts:

*And **also** all my household goods and furniture plate linen and China horses cows pigs implements of husbandry securities for Monday and all my real and personal estate …* (Hanry Haspell, 1848);

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

I give to my dear sister-in-law Georgina Hogarth the sum of 8, 000 free of legacy duty. I ***also*** *give the said Georgina Hogarth all my personal jewellery not hereinafter mentioned, and all the little familiar objects from my writing-table and my room, and she will also know what to do with those things* (Charles Dickens, 1870).

The usage of parallel discourse markers ***likewise, otherwise and or*** is also common. For example:

And I have ***likewise*** *advanced to my son Thomas Witter the sum of one hundred and eighty pounds for his own absolute use* (Thomas Witter, 1866);

Should any Executors and Trustees appointed above either die in my lifetime or be ***otherwise*** *unable or unwilling to act as my Executors and Trustee I appoint the following to fill any vacancy arising Richard Snith …* (Sue Smith, 2006).

Or indicates an alternative:

I may die ***or*** *seized or possessed otherwise entitled to either in remainder reversion or expectancy or otherwise howsoever I hold the same unto him his heirs and assigns for her* (Jogn Wright, 1849).

Contrastive discourse markers are represented by such units as ***but, despite, however, instead, never / nonetheless, rather***. Unlike parallel markers, they indicate an alternative instruction. Example:

But *after the death of my said wife Hannah Witter I further give and bequeath the same to my last named children* (Thomas Witter, 1866);

Nevertheless *upon trust to permit and suffer my wife Hannah Spragg or otherwise empower her to receive the rents issues profits and annual income of my said real estate and to have the full use and enjoyment of my said personal estate for and during the terms of her natural life for her own sole and absolute use and benefit and from and immediately after her decease to the use and be hoof of my two daughters Margaret Yates wife of John Gardener Yates and Ellen Warburton wife of George Warburton their heirs executors administrators or assigns respectively as tenants in common with benefit of survivorship in default of their or her dying without leaving lawful issue only and I appoint the said Dixon Gibbs and Joseph Wood joint executors of this my Will* (John Spragg, 1852).

Elaborative discourse markers include *further (more), in addition to, on account of, whatever and whenever*. For example:

My executors and trustees shall have the following powers ***in addition to*** *all other powers over any share* (Diana Princess of Wales, 1993);

… shall stand to their credit as a payment ***on account of*** *their share…* (Emily Pennell, 1905);

… to sell ***whatever and wherever*** *they decide* (Diana Princess of Wales, 1993).

In Last Wills and Testaments inferential discourse markers are represented by such language units as *accordingly, in accordance with, hence, in this/that case, so, then, therefore, thus*. For example:

Then *I direct that the balance shall be equally divided* (Emily Pennell, 1905);

…disposed of by them ***according to*** *the trusts and exigencies of the same estate* (Thomas Owen, 1859);

I declare that ***in case*** *any doubts shall arise… to bring in at any auction and so receive or vary the terms* (Charles Robert Darwin, 1882);

... when they consider it proper to invest trust monies and to vary investments ***in accordance with*** *the powers contained in the Schedule to this my Will* (Diana Princess of Wales, 1993).

Figure 2 introduces us to the usage of discourse markers in the texts of Last Wills and Testaments.

Two types of discourse markers are common for wills: discourse activity markers (1148 units which is 16,2 % of the total amount) and message relationship markers (5863 units which is 83, 8 %). Sequencing markers as a subtype of discourse activity markers indicate the chronological sequence of actions. Let's have a look at the frequency in their usage. *Finally,* (59 units, 0,8 %) in text of Last Wills and Testaments is used to introduce a final point or a reason. *First* (520 units, 7,4 %) shows foremost in position, rank or importance or anything which comes before all others in time or order. *In the first place / in the second place* (104 units, 1,5 %) used when listing the most important parts of something or the most important reason for something. *Lastly* (75 units, 1 %) used to show when something comes after all the other things listed in wills. *Next* (130 units, 1,8 %) indicated what came immediately after the present one in order or rank. *On the one / other hand* (104 units, 1,5 %) used to introduce a statement that is followed by another contrasting statements. Second (156 units, 2,2 %) subordinates or inferiors position, rank or importance.

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

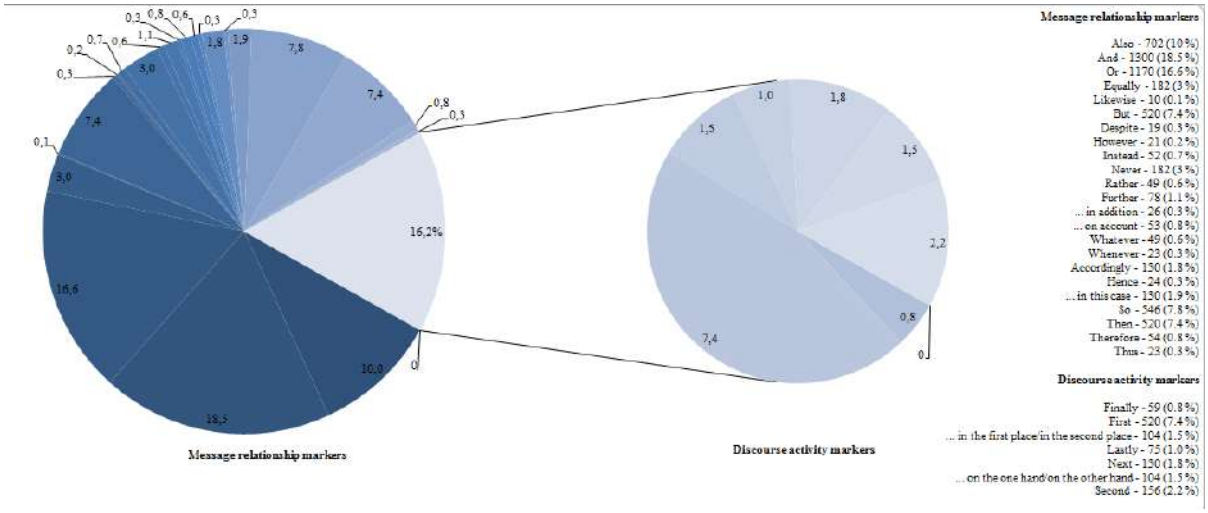


Figure 2: Usage of Discourse Markers in the Texts of English Last Wills and Testaments

In English Last Wills and Testaments sequence markers link sentences together into a larger unit of discourse. In wills message relationship markers are represented by parallel markers (*also, and, or, equally, likewise), contrastive markers (but, despite, however, instead, never/nonetheless, rather*), elaborative markers (*further (more), in addition, on account of, whatever and whenever*) and inferential markers (accordingly, in accordance with, hence, in this/that case, so, then, therefore, thus). In fact, all four subtypes of discourse message relationship markers are used in wills. Parallel markers help to identify the correct intended list in the sentence, paragraph or even the whole text. They help as well to list logically various entities. Thus, *also* (702 units, 10 %) is used to give more information about a person or a thing, to add another relevant fact or to indicate that something is true; *and* (1300 units, 18,5 %) connects words, phrases, clauses and sentences or introduces an additional comment or intention; *or* (1170 units, 16,6 %) is used to link alternatives or can be a sentential connective to form complex sentence; *equally* (182 units, 3 %) shows equal amounts, the same degree or to add patterns that are important; *likewise* (10 units, 0,1 %) is used to compare things and to show their similarity. Contrastive markers in wills express various types of contrast both at the sentence level and at the text level. *But* (520 units, 7,4 %) has two ways of usage in the texts of wills. It is used to introduce a phrase or clause contrasting with the previous utterance and to indicate the impossibility of what has been stated. *Despite* (19 units, 0,3 %) is not frequent in texts and is used to say that the action of bequest happens even though something might prevent it. *However*, (21 units, 0,2 %) introduces a statement that contrasts with what has been said previously. *Instead* (52 units, 0,7 %) indicates an alternative to something expressed in the text of a will. *Never* (182 units, 3 %) *in* studied texts refers to future and means that at no time action is possible. *Rather* (49 units, 0,6 %) used to indicate the second thing, the result of a choice or a change of behaviour. Elaborative markers provide the set of cues and create cohesiveness, coherence and meaning in the texts of Last Wills and Testaments. For example, *further (more)* (78 units, 1,1 %) indicates the extent to which a person, a thing is distant from another. *In addition* (26 units, 0,3 %) mentions another item connected with the discussion issues. *On account of* (53 units, 0,8 %) reflects the benefits of someone or something. *Whatever* (49 units, 0,6 %) emphasizes a lack of restriction in referring to things or amount mentioned in wills. *Whenever* (23 units, 0,3 %) emphasizes a lack of restriction and refers to any time that something happens. The last time of discourse markers used in wills are inferential markers. This type of markers suggests that a message is consequential to some extend to foregoing statement. Accordingly, /in accordance *with* (130 units, 1,8 %) shows suitability or rightness for a certain situation. *Hence* (24 units, 0,3 %) is used as an inference from the fact or for the reason. *In this / that case* (130 units, 1,9 %) is used to talk about things a person should do in order to become a heir. *So* (546 units, 7,8 %) mainly is referred back to something that has been mentioned or to introduce the result of the decision. *Then* (520 units, 7,4 %) indicates what follows next in order or a necessary consequence. *Therefore* (54 units, 0,8 %) introduces a logical result or conclusion of a testator.

Thus (23 units, 0,3 %) is similar in meaning to hence, then and therefore. In the texts it shows a result or consequence of the previous thought.

Discourse markers complement the content of a previous utterance, combine parts of a sentence, introduce new information, or contrast events, actions, or participants. The usage of discourse markers facilitates communication and ensures the compositional integrity of a will. Parallel discursive markers *and*, *or* and *also* are most often used in English wills. Inferential discourse markers so, contrastive marker but and the sequencing discourse marker first has the second place.

5. Conclusions

This study sets out to determine the structure of English Last Will and Testament with research focus on discourse markers. In this investigation the aim was to show that discourse markers provide integrity for the text of wills. It is substantial that all studied wills have a typical structure: title, introduction of a testator, testamentary part, nomination of fiduciaries (executor (trix), trustees) and self-proving affidavit are obligatory. Spiritual part and revocation of previous wills are optional. One of the most significant findings to emerge from this study is that discourse markers provide structural integrity of Last Will and Testament texts. In the study, the classification of discourse markers by B. Fraser was used.

The following conclusions can be drawn from the present study:

1. Discourse markers complement the content of a previous statement;
2. Discourse markers join ideas together or combine parts of a sentence;
3. Discourse markers introduce new information, contrast events, actions and people;
4. Discourse markers show attitude and to some extent control the communication;
5. Discourse markers indicate the result of an action or of what was said before.

Two types of discourse markers were found in wills: discourse activity markers (1148 units which is 16,2 % of the total amount) and message relationship markers (5863 units which is 83,8 %). Sequencing markers as subtypes of discourse activity markers indicate the chronological sequence of action: *finally, first, in the first place/in the second place, lastly, next, on the one / other hand* and *second*. In English Last Wills and Testaments sequence markers join ideas or link sentences together into a larger unit of discourse. In wills message relationship markers are represented by parallel markers (*also, and, or, equally, likewise*), contrastive markers (*but, despite, however, instead, never/nonetheless, rather*), elaborative markers (*further (more), in addition, on account of, whatever and whenever*) and inferential markers (*accordingly, in accordance with, hence, in this/that case, so, then, therefore, thus*).

In fact, all four subtypes of discourse message relationship markers are used in wills performing such functions as identifying the correct intended list in the sentence, paragraph or even the whole text; list logically various entities; give more information about a person or a thing, add another relevant fact or indicate that something is true or false; connect words, phrases, clauses and sentences or introduces an additional comment or intention; link alternatives or are a sentential connective to form complex sentence; show equal amounts, the same degree or add patterns that are of particular importance; compare things and show their similarity; express various types of contrast both at the sentence level and at the text level; introduce a phrase or clause contrasting with the previous utterance and to indicate the impossibility of what has been stated; introduce a statement that contrasts with what has been said previously; indicate an alternative to something expressed in the text of a will; have reference to future in the text; provide the set of cues and create cohesiveness, coherence and meaning in the texts of Last Wills and Testaments; indicates the extent to which a person, a thing is distant from another.

Parallel discursive markers and, or and also are most often used in English wills. Inferential discourse markers so, contrastive marker but and the sequencing discourse marker first has the second place.

Further research is required to determine the efficiency and impact of discourse markers on improving content writing based on the gender of the testator/testatrix. It would be interesting to compare the differences and similarities of the usage of discourse markers in such cases.

6. References

[1] I. A. Bekhta, T. O. Bekhta, Text in fiction literature: psycholinguistic reflection of the context, Scientific notes of National Univesity "Ostroh Academy", S. Philology, Ostroh: Publishe NaUOA, 2019. doi: 10.25264/2519-2558-2019-6(74)-11-14.

[2] O.V. Kulyna, Last Will and Testament as a genre of testamentary discourse and approaches to the study, Scientific note of Mariupol State University, S. Philology, № 17, Mariupol: MDU, 2017, pp. 147–157.

[3] O.V. Kulyna, Structural and typological parameters of the genre of Last Will and Testament: based on English wills 1937–1858, New philology: journal of scientific works, № 69, Zaporizzia: ZNU, 2017, pp. 107–113.

[4] A dictionary of American and English Law with definitions of the technical terms of the common and civil law, ed. by S. Rapalje, R. L. Lawrence, 3d ed, Union, New Jersey, 2002.

[5] A New English dictionary on historical principles: founded mainly on the materials collected by the philological society, ed. by J. A. H. Murray, Oxford: Clarendon Press, 1919, Vol. IX, pt. II: Su-Th., P. 221 (850).

[6] A new English dictionary on historical principles: founded mainly on the materials collected by the philological society, ed. by J. A. H. Murray, Oxford: Clarendon Press, 1919, Vol. IX, pt. II: V-Z. P. 131 (1228).

[7] A new English dictionary on historical principles: founded mainly on the materials collected by the philological society, ed. by J. A. H. Murray, Oxford: Clarendon Press, 1919, Vol. IX, pt. 1: Si-St., P. 118 (1228)

[8] Ch. Bauer-Ramazani, English Discourse Markers. URL: academics.smcvt.edu/cbauerramazani/ AEP/BU113/English /discmarkers.ht.

[9] T. A. van Dijk, Text and Context. Explorations in the Semantics and Pragmatics of Discourse, London: Longman, 1977.

[10] T. A. van Dijk, Discourse, context and cognition. Discourse Studies, 2006, Vol. 8 (1), P. 159–177. URL: http://dis.sagepub.com/cgi/content/abstract/8/1/159.

[11] T. F. van Dijk, Issues in functional discourse analysis. URL:www.citeseerx.ist.psu.edu/ viewdoc/download?doi=10.1.1.651.9358&rep=repl&type=pdf.

[12] T. A. van Dijk, Macrostructures: An Interdisciplinary Study of Global Structure in Discourse, Interaction, and Cognition. Hillsdale, NJ: Lawrence Erbaum, 1980.

[13] T. A. van Dijk, Pragmatic Connectives. Journal of Pragmatics, North-Holland Publishing Company, 1979, № 3, pp. 447–456.

[14] H. A. Dry, J. Lawler, Using computers in linguistics, London: Routledge, 1998.

[15] C. Ferland, What is the meaning of Last Will and Testament? URL: http://info.legalzoom.com/ meaning-last-testament-3948.html..

[16] B. Fraser, What are discourse markers? Journal of Pragmatics, 1999, Vol. 31, pp. 931–952.

[17] B. Fraser, Types of English discourse markers. Acta Linguistica Hungarica, 1988, Vol. 38 (1–4), pp. 19–33.

[18] B. Fraser, Pragmatic Markers. Pragmatics, 1996, Vol. 6, iss. 2, pp. 167–190.

[19] G. Redeker, Linguistic markers of discourse structure. Linguistics, 1991, № 29 (6), pp. 1139–1172.

[20] M. Rysova, K. Rysova, The centre and periphery of discourse connectives, Pacific Asia Conference on language, information and computation: proc. of the 28 th conf., 2014, pp. 452–459. URL: http://www.aclweb.org/anthology/Y14-1052.

[21] D. Schiffrin, Approaches to Discourse, Oxford: Cambridge, MA, 1994.

[22] D. Schiffrin, Discourse Markers. Cambridge: Cambridge University Press, 1987.

[23] K. J. Sneddon, In the name of God, Amen: language in the Last Wills and Testaments. Quinnipiac Law Review, 2011, Vol. 29, pp. 665–727.

[24] H. Swinburne, J. Wake, A treatise of Testaments and Last Wills compiled out of the laws, ecclesiastical, civil and canon. W. Clarke and Sons, 1803. URL: https://books.google.com.ua/books/about/A_Treatise_of_Testaments_and_Last_Wills.html?id=h 34zAAAAIAAJ&redir_esc=y

[25] E. Traugott, The role of the development of discourse markers in a theory of grammaticalization, Manchester, 1995. URL: http://www.wata.cc/forums/uploaded/136_1165014660.pdf.

[26] Wills Act 1837. URL:http://www.nzlii.org/nz/legis/consol_act/wa1837121.pdf

Designing Linguistic Ontologies for Training Information Systems

Olha Tkachenko[a], Kostiantyn Tkachenko[a], Oleksandr Tkachenko[b]

[a] State University of Infrastructure and Technology, I. Ogienko str., 19, Kyiv, 02000, Ukraine
[b] National Aviation University, Liubomyra Huzara ave. 1, Kyiv, 03058, Ukraine

Abstract

The article discusses the problems of designing linguistic ontologies for educational information systems. An approach to the formalized description of linguistic ontologies is considered, taking into account the concepts of subject areas of training information systems and the relationship between these concepts. The thesaurus of the training information system, built on the basis of linguistic ontologies, is considered.

Keywords 1

training information system, subject area (domain), ontology, linguistic ontology, information resource.

1. Introduction

The increase in the volume of text information (electronic documents, web content, educational and methodological material of training information systems, etc.) provides the need for processing such unstructured information, improving the quality and efficiency of existing methods of processing words and developing new ones.

Among the directions of processing unstructured text information, one can single out, for example:

- search for information;
- classification clustering of text documents,
- filtering, rubrication of text documents,
- annotation of a document (group of documents);
- search for similar documents and duplicates,
- document segmentation;
- assessment of semantic similarity and kinship;
  - extraction of information;
  - recognition of named entities;
  - extraction of relationships;
  - extraction of facts;
  - extraction of knowledge;
  - co-reference permission;
- answers to questions in natural language;
- machine translate;
- summary of the text;
- analysis of the sentiment of the test document;
- intellectual analysis;
- automatic creation of ontologies / dictionaries / thesaurus / knowledge base;
- speech recognition and speech synthesis.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: oitkachen@gmail.com (O. Tkachenko); tkachenko.kostyantyn@gmail.com (K. Tkachenko); aatokg@gmail.com (O. Tkachenko)
ORCID: 0000-0003-1800-618X (O. Tkachenko); 0000-0003-0549-3396 (K. Tkachenko); 0000-0001-6911-2770 (O. Tkachenko)


Modern educational information systems work with textual information of subject areas, which include thousands of different classes of entities that are among themselves in a huge number of different types of relationships [1, 2].

Therefore, the methods of processing textual information in such systems are often guided by the use of statistical characteristics of this information, in particular:

- frequency of occurrence of words in a sentence, text, set of documents (educational materials, test items, reference information, etc.);
- joint occurrence of words.

Such methods use minimal knowledge about the subject area (domain), language, its features and diversity.

User of training information systems (lecturer, methodologist, student), performing text information processing, primarily:

- reveals the main content of the document and the meaning of its key concepts;
- the main topic, subtopics and key concepts of the document (educational materials, test items, reference information, etc.).

For this, the user of training information systems (lecturer, methodologist, student) usually uses a large amount of knowledge about:

- language of presentation of educational materials, test items, reference information (linguistic knowledge);
- subject area (ontological knowledge);
- organization of coherent text (relations between units of knowledge).

Lack of linguistic and ontological knowledge leads to a variety of problems when, for example:

- ways of formulating queries differ from templates for describing relevant situations in documents that are supported by training information systems;
- long requests are processed (for example, when referring to help information);
- the context of the language (individual words and expressions used in the query) is not fully taken into account.

Thus, modern intellectual training systems for processing text information (or training information systems with elements of intellectualization) face the following problems [3]:

- processing of text information of online courses in the considered subject area;
- taking into account the linguistic features of the language and the structure of the corresponding training or test text.

These problems are especially acute in information retrieval systems, automatic text processing systems (including their generation) and training information systems.

Intellectual text analysis is one of the key tasks in the field of artificial intelligence associated with the problems of automatic analysis and synthesis of natural language arising from the interaction of a user (lecturer, methodologist student) with a training information system.

The solution to these problems is closely related to the use of various approaches of artificial intelligence and computational linguistics.

The development of ontological modeling and machine learning methods has made it possible to achieve the quality required for practical use in natural language processing tasks in training information systems.

The use of additional linguistic and ontological knowledge in the automatic processing of texts in training information systems is a difficult task.

This is due to the fact that such knowledge should be described in specially created resources (thesauri, ontologies), which should contain descriptions of a large number of words and phrases and be able to logically derive new knowledge.

When using such resources, it is usually necessary to solve the problem of word ambiguity, i.e. choose their correct value.

The paper considers the extraction of information from the text, which can be used to create formal models of specific areas of knowledge.

In work, this is the area of training courses in the disciplines "Informatics" and "Information systems and technologies".

A simplified approach to language modeling includes various statistical models based on

distribution semantics.

This approach determines the semantic similarity between two linguistic elements (such as words or phrases) based on their distribution properties in large fragments of educational methodological or test text without specific knowledge of the lexical or grammatical meanings of the elements.

One of the ways to represent words with this approach is to cut documents into sets and sequences of words − n-grams [4, 5], which take into account the information contained in verbose constructions of length n (bigrams for word pairs, etc.).

N-gram (for n = 1) ignores all properties of an educational-methodical or test document, except for the number of words in it.

A word set is a collection of documents in the form of a matrix, the rows of which correspond to the documents, and the columns to a specific term.

Intersection values describe the number of words in a particular document.

For $n \geq 1$, constructions of several words contain additional information (phrases, idioms, etc.) in comparison with a set of single words.

These models often include a weight for each term-document pair.

The indicator is the number of occurrences (frequency) of a term in each document or the probability of finding a word in a document.

This rates the more general words as more important, although this is not always the case.

It is more common to weigh n-grams so that the weight of a word in a certain document is proportional to its quantity in a given document and at the same time is inversely proportional to the frequency of using this word in other documents from the same collection [5].

One of the paradigms of computer resources for training information systems are formal ontologies (for example, the Semantic Web [6, 7]).

But the automatic processing of unstructured natural language texts is difficult to carry out using formal ontologies [8, 12, 13].

Therefore, for the automatic processing of texts, special ontologies (terminological, lightweight, linguistic) are developed [9, 10, 11], in which concepts are not always strictly formalized.

Linguistic ontology is an ontology, the concepts of which are largely associated with the meanings of linguistic units, terms of the subject area [12, 13, 14, 15].

Linguistic ontologies cover most of the words of a language or subject area and at the same time have an ontological structure that manifests itself in the relationship between concepts.

Therefore, linguistic ontologies can be considered as a special type of lexical database and a special type of ontology.

The paper describes a linguistic ontology designed for automatic text processing for the considered subject area, and the resources that are developed on the basis of this ontology.

2. Formalized Linguistic Ontologies

The following can serve as a formal definition of ontologies:

$$O =< C, E, At, R, A >,$$

where:

- C − concepts (classes) of ontology;
- E − instances of ontology;
- At − attributes of concepts and instances of ontology;
- R − relations between concepts;
- A − axioms of ontology.

Formalized ontologies consider various computer resources, in particular, rubricators or thesauri. Typically, rubricators do not include instances and attributes, i.e. the formal model of rubricators is a model of the form:

$$O =< C, R, A >.$$

Formalized ontologies are logical theories built on axioms. To describe them are used:

- logics: descriptive logics, modal logics, first-order predicate logic;
- ontology description languages: DAML + OIL, OWL, CycL, Ontolingua [16, 17, 18].

Ontologies (thesauri, rubricators), the concepts of which are not fully defined in terms of formal properties and axioms, are called lightweight ontologies.

There are different interpretations of the relationship between ontology and the natural language of documents of the training information system:

- ontology is a structure independent of natural language;
- ontology is a structure that is independent of a specific natural language;
- elements of the language lexicon are included in the formal definition of ontology;
- the formal definition of ontology includes the entire lexicon of the subject area (domain).

Based on the foregoing, the formal model of ontology can be described as:

$$O = < V, C, R_{VC}, R_{VR}, R, A >$$

where:

- $V = V_C \cup V_R$ – vocabulary of ontology, containing a set of lexical units for V_C concepts and V_R relations;
- C – a set of concepts of ontology;
- R_{VC} – a set of connections between lexical units $\{v_j\} \subset V$ and the corresponding concepts from $\{c_k\} \subset C$ and relations of the given ontology;
- R_{VR} – the set of links between lexical units $\{v_j\} \subset V$ and the corresponding relations $\{r_i\} \subset R$ of the given ontology;
- R is the set of relationships between concepts of ontology;
- A is a set of ontology axioms.

In the considered formal approaches, words of a natural language are one of the components of the ontological model, lexical expressions are presented only as auxiliary elements that name the concepts and relations of the ontology.

Establishing relationships between concepts, words and expressions of a natural language has many problems, in particular, the introduction of a new concept into an ontology must be associated with existing linguistic elements; definition of relations "concept − linguistic element".

Therefore, a large number of widely known medical ontological resources are thesauri that do not have a high degree of formalization of their structure.

Thesauri are linguistic ontologies, i.e. ontologies based on the meanings of real natural language expressions.

Training information system thesaurus is a normative vocabulary of terms in natural language that explicitly indicates the relationship between terms and is intended to describe the content of documents and search queries.

The basic unit of thesauri is terms, which are categorized into descriptors (= authorized terms) and non-descriptors (= ascriptors).

At their core, descriptors unambiguously correspond to the concepts of the subject area (domain). Relationships between descriptors are divided into: hierarchical and associative.

Hierarchical relationships are usually viewed as asymmetric and transitive.

Hierarchical relationships used in teaching information systems thesauri:

- class − subclass (predecessor − successor, above − below) − is installed between two descriptors, if the concept of a lower − level descriptor (successor, subclass) is included in the concept of a superior descriptor (predecessor, class);
- whole − part.

The purpose of developing training information systems thesauri is to use their units (descriptors) to describe the main topics of documents in the process of manual indexing.

Therefore, it is important that the set of thesaurus descriptors allow describing the topics of educational, methodological, test and reference documents of the subject area.

In this case, the indexing process for such a thesaurus is based on linguistic, grammatical knowledge, as well as knowledge of the subject area.

To determine the semantics of the document test, the component of the training information system – the program “Indexer” – must first read the text, understand it and then state the content of the text using the descriptors specified in the thesaurus.

The program “Indexer” should have a good understanding of all the terminology used in the text −

to describe the main topic of the text, he will need a much smaller number of terms.

The presence of the program "Indexer" testifies to the intellectualization of the training information system.

Thus, the formal model of the thesaurus (T) of the training information system can be represented as follows:

$$T = <D, C, R, A>,$$

where:

- D is a set of domain descriptors corresponding to the concepts of a given domain, which are necessary to express the main topics of documents in this domain;
- C − a set of terms (concepts) of the subject area: области: $D \subset C$;
- R – relations of the thesaurus, $R = R_I \, U \, R_A$ (R_I – hierarchical and R_A – associative relations of the thesaurus);
- A – axioms of transitivity of hierarchical relations.

The described model of the thesaurus of the training information system is intended for its use documents in the process of expert analysis of educational, methodological, test and reference documents.

A thesaurus intended for automatic text processing should contain much more information about the structure and language of the subject area.

The relationships between the terms specified in the thesaurus should be formalized for their use in the training information system.

3. Linguistic Ontologies in the Training Information Systems

Formal ontologies (with their independence from a particular language) are difficult to use in automatic text processing for information retrieval applications because:

- units of formal ontology must be associated with units of a specific natural language;
- the desire for a clear formalization of relations between concepts in a formal ontology is difficult to observe when creating super-large resources;
- leads to problems in establishing relations "concept − linguistic expression".

An training information system deals not only with general vocabulary, but also with specific subject areas and their terminologies.

The description of the terminology of the subject areas of training information systems should use:

- information retrieval context;
- resource units, which are created based on the values of terms;
- description of verbose expressions; principles of inclusion (non-inclusion) of verbose units;
- a small set of relationships between conceptual units.

The use of a linguistic resource in automatic text processing in a training information system should take into account the following provisions:

- conceptual units are created based on the meanings of real linguistic expressions;
- multi-step hierarchical construction of the lexical and terminological system of concepts;
- principles of describing the meanings of polysemous words and expressions;
- development of linguistic ontology as a hierarchical system; the use of formally defined relations with formal properties;
- the use of transitivity and inheritance of relations between concepts of domain as axioms (inference rules).

The LO linguistic ontology model for the SA subject area can be represented as follows:

$$LO = < C, E, N, R, P_{tr} \, T, S, W, L, T_W >$$

where:

- C − a set of concepts of ontology, where concept is a class of entities that have the same properties and relationships with other classes of entities;
- E − a set of instances of ontology concepts, a mapping $E: C \rightarrow E$ is given;
- N − a set of unique names of concepts and instances in the ontology;

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

- R is a set of relationships between concepts;
- P_{tr} − set of withdrawal rules;
- T − a set of linguistic expressions, the values of which are presented in the ontology;
- S − a set of relations between linguistic expressions (T) and concepts (C): $\{s\ (c_i, t_j)\}$;
- W − a set of polysemous words and expressions from T: $W \subset T$; $W = W_m \cup W_a$, where W_m are text inputs that refer to more than one concept of the ontology, and W_a are multivalued text inputs that are represented in the ontology by only one value;
- L − a set of lemmatic representations of a linguistic expression (for example, the phrase information system is presented in a lemmatic form as an INFORMATION SYSTEM);
- T_W is a mapping of the terminological composition of a given subject area to text inputs and ontology concepts.

The proposed linguistic ontology of the subject area is a knowledge base of the ontological type about the conceptual system, the lexical and terminological composition of the subject area (disciplines "Informatics" and "Information systems and technologies"), supported by the corresponding training information system.

The unit of linguistic ontology is a concept, as a unit in a system of concepts, which has its own specific properties that distinguish this unit from other units in the system of concepts.

Each entered concept must have a unique name. The name can be an unambiguous word or phrase, the meaning of which corresponds to this concept.

Each concept is supplied with a set of text inputs − language expressions, the values of which correspond to the given concept. Such linguistic expressions are ontological synonyms among themselves.

The texts may contain many variants of text inputs of a particular concept.

The developer of a training information system or a specific online training course must record these options immediately when entering a concept, or supplement it when found in a specific text.

In the texts of the subject area, a significant part is made up of words that belong not only to a specific subject area, but also to the general vocabulary of many subject areas, for example, *create*, *participate*, *accept*, *evaluate*, etc.

Therefore, the polysemantic words described in the linguistic model are divided on:

- the set W_m, which includes expressions related to two or more concepts;
- the set W_a, which includes expressions related to one concept, but these words may have a different meaning in the general lexicon, which is marked by a special mark of ambiguity.

Relationships between concepts from an ontological resource should perform the following functions:

- these relations should be used in the classic functions of information retrieval thesauri to expand a search query or display a heading of a document;
- relations should be used to resolve the ambiguity of linguistic units included in the resource;
- relations in an ontological resource can be used to identify lexical connectivity in texts and to use the revealed text structure to improve the quality of text processing.

When creating a linguistic ontology of large magnitude, for processing texts that are not limited in style, genre, size, the most stable way is to rely on relationships that do not disappear, do not change during the entire lifetime of any or the vast majority of instances of the concept: for example, software is always consists of programs.

Therefore, in linguistic ontology, relations are described only between such concepts ci and cj, which are inherent in at least one of these concepts by definition.

The properties of transitivity and inheritance are used as axioms.

For a logical conclusion when processing texts in the subject area, it is necessary to describe the relationship between concepts that retain their significance, reliability in various contexts of mentioning concepts.

The main relations in the proposed linguistic ontology are:

- class-subclass;
- whole-part;
- relation of ontological dependence (asymmetric association);

- symmetrical association.

Let the class−subclass (c_i, c_j) be the relationship between the concepts c_i and c_j (c_i is a subclass of c_j), $r(c_i, c_j)$ be an arbitrary relationship between the concepts c_i and c_j.

Class−subclass relationships have transitivity and inheritance properties.

However, the same expressions of natural language can correspond to different relationships between entities of the subject area, including those with completely different properties [13. 19].

Therefore, you should check the established class-subclass relationship. For example, to check the belonging of instances of a lower-level concept c_i to a set of instances of a higher-level concept, which implies an answer to the question:

If an object is an instance of one concept, then will it necessarily be an instance of some other concept c_j?

The feature of the whole-part relationship is one of the most famous and useful in various subject areas. *Part–whole* relationship is the variety of its manifestations. The most typical objects to which this relation applies are physical objects, entities that last in time, groups of entities, processes, etc.

When modeling this relationship in computer resources, it is important to ensure its transitivity. When describing the whole–part relationship in the proposed model of linguistic ontology, efforts were made to ensure the transitivity of this relationship. That is, it is necessary to describe the whole–part relationship as follows:

if the text (a fragment of the text) is devoted to the discussion of a part, then it can be assumed that the text (a fragment of the text) will be relevant to the discussion of the whole.

The condition for ensuring such inheritance is the ontological dependence of the existence of a part on the existence of the whole.

The part dependency can be like this:

- *in existence*, when an instance of a part cannot be separated from an instance−whole;
- *generic*, in which the existence of an instance−part requires the existence of at least one instance of the whole.

The description of hierarchical relationships should be independent of the context in which they are mentioned.

This is important in automatic text processing, since in automatic mode it is often impossible to use the context to confirm the existence of a particular relationship.

In linguistic ontologies, the following properties of the whole-part relationship are used:

- part (c_1, c_2) $\leftrightarrow$ whole (c_2, c_1);
- whole (c_1, c_2) $\wedge$ whole (c_2, c_3) $\rightarrow$ whole (c_1, c_3) – transitivity of the relation;
- class (c_1, c_2) $\wedge$ whole (c_2, c_3) $\rightarrow$ whole (c_1, c_3) – inheritance of the whole relation with respect to the class−subclass relation.

The concept c_i is externally dependent on the concept c_j if for all instances of c_i there is an instance c_j that is not part or material of the instance c_i.

For example, the concept of a son is externally dependent on the concept of a parent, since it exists only within the family in relation to its parents.

And the concept of a car is not externally dependent on any entity, since it requires the existence of a motor, which is part of the car.

The asymmetric association relation Ass represents an external ontological relationship between concepts. This relationship is established between the concepts c_1 and c_2 if the following conditions are satisfied:

- between the concepts c_1 and c_2, the class-subclass and / or whole-part relations cannot be established;
- the statement is true: the existence of c_2 means the existence of c_1.

These conditions mean that the dependent concept c_2 is externally dependent on c_1:

$$Ass_1(c_2, c_1) = Ass_2(c_1, c_2).$$

Ontological dependency relationships are applicable to different areas, so they are most often used in top-level ontologies.

For various applications of automatic word processing, some groupings of concepts and relations in linguistic ontology are used.

4. Linguistic Ontologies Based on the Described Model

The above principles were the basis for the development of an ontology for the disciplines "Information systems and technologies" and "Informatics".

The created ontological resources have the same structure. They are ontologies because they describe the concepts of the domain and the relationship between them.

These resources belong to linguistic ontologies, since the introduction of concepts is largely motivated by the meanings of linguistic units related to the subject area of the resource.

At the same time, they are thesauri, since each concept is associated with a set of linguistic expressions (words, terms, phrases) with which this concept can be expressed in a text - such a set of textual concept inputs is necessary to use ontologies for automatic text processing.

Each term is provided with a description (dictionary entry), has hierarchical links with other terms and synonyms.

Figure 1 shows a list of hyperlinks to dictionary entries of the "main root" key terms (concepts) of the subject area "Information systems and technologies" and "Informatics".

Having opened the dictionary entry of a term, we get a description of the term, a list of other related terms and lists of publications and persons related to this term. The performed layout allows you to view the thesaurus in alphabetical order of its text inputs.

The choice of a specific text input, for example, TECHNOLOGY, allows you to see the totality of concepts to which this word is attributed, namely to the concepts of INFORMATION TECHNOLOGY and INFORMATION SYSTEM.

- Алгоритм
- Архітектура обчислювальної машини
- База даних
- Дані
- Знання
- Інформатика
- Інформаційна система (ІС)
- Інформаційна технологія (ІТ)
- Інформаційні ресурси
- Інформаційний пошук
- Інформація
- ІТ-бізнес
- Кібернетика
- Комп'ютер
- Комп'ютерна мережа
- Мова програмування
- Модель
- Обчислювальна система
- Операційна система (ОС)
- Програмний код
- Програмне забезпечення
- Програмування
- Система
- Технологія

- Algorithm
- Computer architecture
- Database
- Data
- Knowledge
- Informatics
- Information system (IS)
- Information Technology (IT)
- Information resources
- Information search
- Information
- IT business
- Cybernetics
- Computer
- Computer network
- Programming language
- Model
- Computing system
- Operating system (OS)
- Program code
- Software
- Programming
- System
- Technology

Figure 1: Key Concepts of Disciplines "Informatics" and "Information Systems and Technologies"

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

For each concept, complete lists of text inputs are indicated, including words of different parts of speech, as well as phrases. So, for the concept INFORMATION TECHNOLOGY, the text inputs are words and expressions: *technology*, *information*, *software*, *information resources*, *information system* (Figure 2).

WHOLE	Інформаційна технологія	Information technology
PART	Інформатизація	Informatization
ASSOCIATION	Інформаційна система	Information system
PART	Цифровізація	Digitization
ASSOCIATION	Інформаційна революція	Information revolution
PART	Інформаційні ресурси	Information resources
ASSOCIATION	Мова програмування	Programming language
PART	Програмування	Programming
PART	Програмний код	Program code
PART	Програма	Program
ASSOCIATION	Обчислювальна система	Computing system
ASSOCIATION	Обчислювальна техніка	Computers

Figure 2: Article of the Concept INFORMATION TECHNOLOGY

For each concept, relationships with other concepts are indicated. In Figure 2, in the article the concepts of INFORMATION TECHNOLOGY are indicated:

- parts of the concept INFORMATION TECHNOLOGY (INFORMATIZATION, DIGITIZATION, INFORMATION RESOURCES, PROGRAMMING, PROGRAM CODE, PROGRAM, etc.);
- ontologically dependent concepts, i.e. concepts that could not have appeared if information technology did not exist: INFORMATION SYSTEM, INFORMATION REVOLUTION, PROGRAMMING LANGUAGE, COMPUTING SYSTEM, COMPUTERS, etc.

Figure 3 shows the concepts related to the key concept of ALGORITHM and which are in different types of relationships with it.

CLASS	Алгоритм	Algorithm
SUBCLASS	Алгоритм Дейкстри	Dijkstry's algorithm
PART	Блок-схема	Block diagram
SUBCLASS	Машина Т'юринга	Turing machine
SUBCLASS	Машина Поста	Post Machine
ASSOCIATION	Мова програмування	Programming language
SUBCLASS	Нормальній алгорифм Маркова	Normal Markov algorithm
PART	Програмний код	Program code
ASSOCIATION	Програма	Program
SUBCLASS	Примітивно-рекурсивна функція	Primitive-recursive function
PART	Складність алгоритму	Complexity of the algorithm

Figure 3: Article of the Concept ALGORITHM

For each concept, relationships with other concepts are indicated. In Figure 3 in the article of the concepts of ALGORITHM are indicated:

- types of algorithms formalization (DIJKSTRY'S ALGORITHM, TURING MACHINE POST MACHINE NORMAL MARKOV ALGORITHM PRIMITIVE-RECURSIVE FUNCTION);
- parts of the concept of ALGORITHM (BLOCK DIAGRAM, COMPLEXITY OF THE ALGORITHM, etc.);

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

- ontologically dependent concepts, i.e. concepts that could not have appeared if there were no algorithms: PROGRAM, PROGRAMMING, PROGRAM CODE, PROGRAMMING LANGUAGE, etc.

The information base supporting the proposed linguistic ontology includes:

- set of concepts for the subject area under consideration (disciplines "Informatics" and "Information systems and technologies", which are supported by the corresponding training information system):
  - concepts of general vocabulary;
  - concepts of subject areas "Informatics" and "Information systems and technologies";
- interpretation of concepts;
- set of relationships between the concepts of the considered subject area;
- many text inputs of the thesaurus; • description of text inputs:
  - lemmatical representation of text input;
  - syntactic type;
  - the main word of the noun phrase;
- set of correspondences of text inputs to the concepts of the thesaurus of the training information system.

5. Conclusion

The article presents a model of linguistic ontology for the subject area (disciplines “Informatics” and “Information systems and technologies”).

This model is used in the development of a training information system that supports online learning in these disciplines.

In the proposed model, a set of relations of a linguistic ontology is described, which is specially selected to describe the subject area under consideration.

The functions of relations of the linguistic ontology of information retrieval are possible when providing multi-step logical inference based on the properties of transitivity and inheritance of relations and their independence from the context of the concept.

To provide these properties, it was proposed to use a small set of relations.

Ontological definitions of the relations used were introduced. Such system of relations reflects the most essential relationships between entities and can be used to describe relationships between concepts in a variety of disciplines, supported by educational information systems.

The proposed linguistic ontological model was implemented in the implementation of a training information system that supports the disciplines "Informatics" and "Information systems and technologies."

6. References

[1] G.Z. Liu, A Key Step to Understanding Paradigm Shifts in E-learning: Towards Context-Aware Ubiquitous Learning, British Journal of Educational Technology, 2017. Vol. 41. № 2. pp. E1-E9.

[2] V.A. Trainev, New information communication technologies in education, Moscow, Dashkov and K, 2018. (in Russian).

[3] Matthew U. Scherer, Regulating artificial intelligence systems: risks, challenges, competencies, and strategies, Harvard Journal of Law & Technology Vol.29, № 2, Spring 2016.

[4] ISO 25964-1:2011, Thesauri and interoperability with other vocabularies. Part 1: Thesauri for information retrieval. Geneva: International Organization for Standards, 2011.

[5] G. Miller, Nouns in WordNet. WordNet – An Electronic Lexical Database. The MIT Press, 1998, pp. 23-47.

[6] O. Tkachenko, A. Tkachenko, K.Tkachenko, Ontological Modeling of Situational Management, Digital platform: information technology in the sociocultural area. Vol. 3. № 1 (2020). pp. 22-32.

[7] Kamran Munira, M. Sheraz Anjumb, The use of ontologies for effective knowledge modelling and information retrieval. 2017. URL: https://www.sciencedirect.com/science/article/pii/S2210832717300649. doi: 10.1016/j.aci.2017.07.003.
[8] S. Nirenburg, Y. Wilks, What's in a symbol: Ontology, representation, and language, Journal of Experimental and Theoretical Artificial Intelligence, 2001. Vol. 13(1). pp. 9-23.
[9] S.E. Greger, S.V. Porshnev, Building an ontology of information system architecture, Fundamental Research, № 10, 2013, pp. 2405-2409.
[10] J. Sowa, Building, Sharing and Merging Ontologies. URL: http://www.jfsowa.com/ontology/ontoshar.htm.
[11] C. List, Levels: descriptive, explanatory, and ontological, 2018. URL: http://eprints.lse.ac.uk/87591/1/List_Levels%20descriptive_2018.pdf.
[12] S. Nirenburg, V. Raskin, Ontological Semantics. MIT Press, 2004. 420 p.
[13] Kudashev, Quality Assurance in Terminology Management: Recommendations from the TermFactory Project. – Helsinki: Unigrafia, 2013. 216 p. URL: http://www.projectglossary.eu/download/QA_in_TM_Kudashev.pdf.
[14] B. Magnini, M. Speranza, Merging Global and Specialized Linguistic Ontologies, Proceedings of OntoLex. (2002). pp. 43-48.
[15] T. Veale, Y. Hao, A context-sensitive framework for lexical ontologies, Knowledge Engineering Review, 2007. Vol. 23(1). pp. 101-115.
[16] I. Horrocks, Reviewing the design of DAML+OIL: An ontology language for the Semantic Web. URL: https://www.researchgate.net/publication/2477217_Reviewing_the_design_of_DAMLOIL_An_ontology_language_for_the_Semantic_Web
[17] Web Ontology Language (OWL). URL: https://www.w3.org/OWL/
[18] Ontolingua. URL: http://www.ksl.stanford.edu/software/ontolingua/
[19] N. Guarino, Some Ontological Principles for Designing Upper Level Lexical Resources, Proceedings of First International Conference on Language Resources and Evaluation. Granada, Spain, 1998.

Theoretical Basics of Creating an Electronic Corpus-Based Dictionary of Legal Terminology

Ihor Vozniak

Lviv Polytechnic National University, Lviv, Ukraine

Abstract

Development of professional and scholar terminology as well as its lexicographical inventorization remain one of the biggest challenges for modern Ukrainian linguistics. This study aims therefore to investigate theoretical basics of creating specialized terminological electronic dictionaries since electronic dictionaries are the most convenient way for both professionals and linguists to learn, study, and use the language. The notion and main characteristics of electronic dictionaries were analyzed. Specifics of terminological dictionaries and their differences from learner's dictionaries were described. Corpus-based approach was investigated as basis for the highest level of dictionary's objectivity and representativeness. The results of this research attest a huge potential for practical application of the described methodology.

Keywords 1

Computer lexicography, corpus linguistics, electronic dictionary, terminological dictionary.

1. Introduction

The branch of linguistics that deals with compiling dictionaries using computer technologies is called computer lexicography. It combines methods of computer sciences with linguistics and is also part of applied linguistics. Computer lexicography has long been an object of researches as well as set of rules and practical tools for linguists. The ubiquity of Internet technologies makes the need in electronic dictionaries more actual. Electronic lexicographic resources are widely used by specialists, translators, lecturers, students, and even laypeople who are interested in a specific scientific or scholar sphere. This makes the problems of computer lexicography topical in linguistic researches which seek to use increasingly more computer technologies.

The problem of terminological lexicography in Ukraine was object in the researches of S. Vyskushenko, S. Holovashchuk, I. Kudashev, S. Kuleshov, S. Radziievska, K. Silevestrova, B. Shunevych and others. Electronic dictionaries were studied by O. Chernysh, O. Syvak, V. Shadura, O. Hordiienko, N. Sinkevych, O. Pliushchai. The most noticeable lexicographic project regarding legal terminology is Ukrainian-German terminological database that is available at rechtsdialog.org/de/. Nevertheless, the problem of electronic terminological dictionaries remains for the greater part not enough studied in Ukrainian theoretical lexicography. We can suppose that this is caused by the fact that general dictionaries remain the most popular product for going online, while terminological dictionaries which are primarily intended for professional use remain in paper form or can be just adapted for online search (e.g., you can find the English-Ukrainian-English dictionary of science language (physics and allied sciences) at the free online source e2u.org.ua but this dictionary is not "fully" electronic).

The object of this research is methodology of compiling dictionaries. The subject of the research is methodological approaches to creating an electronic corpus-based terminological dictionary. The topicality of this research is based on the need of compiling terminological dictionaries in different areas which can help develop terminological completeness and diversity of Ukrainian language.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: ihor.z.vozniak@lpnu.ua (I. Vozniak)
ORCID: 0000-0002-3224-9121 (I. Vozniak)

The aim of this research is to develop theoretical basis for a specialized electronic corpus-based dictionary with potential for continuous development and enrichment. The following tasks were carried out:

- defining the concept of electronic dictionary, its advantages and perspectives;
- describing the ways of introducing corpus technologies in lexicographic work;
- analyzing the characteristics of terminological dictionaries on example of Ukrainian law terminology;
- preparing a set of rules for a computer linguist working on an electronic terminological dictionary;
- presenting an example of possible terminological entry;
- outlining possible problems and perspectives for further researches.

This research has practical value since it can be used as basis for creation of a dictionary. Its theoretical value lies in the possibility of using its postulates for other branches of computer lexicography.

2. Notion of Electronic Dictionaries

The notion of electronic dictionary is not unambiguous as it may initially seem. According to the classification of de Schryver, electronic dictionary in their technical evaluation can be divided into two groups: offline dictionaries (nowadays they exist in the form of either desktop programs or apps for smartphones) and online dictionaries. In metalexicographic evaluation both offline and online electronic dictionaries may have print appearance or some sort of innovative appearance [1, p. 148]. Ukrainian researchers V. Perebyinis and V. Sorokin proposed the umbrella term "computer dictionary" as opposed to paper dictionaries. In their opinion, computer dictionaries can be divided into computerized paper dictionaries and electronic dictionaries [2, p. 16]. Their classification can be graphically represented the following way:

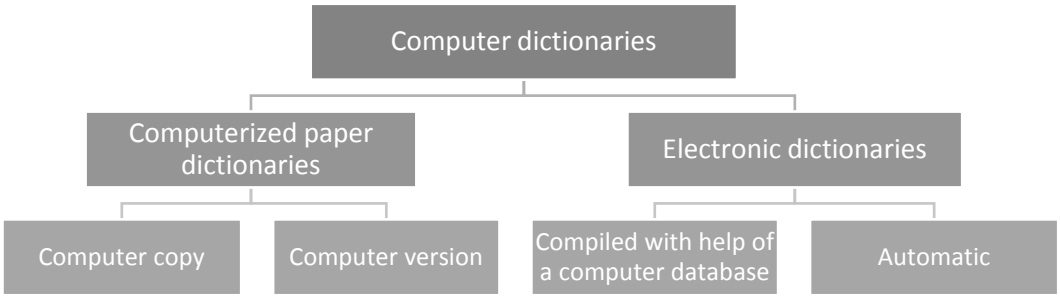


Figure 1: Classification of Computer Dictionaries According to V. Perebyinis and V. Sorokin [2, p. 16]

To our opinion, an electronic dictionary is a specific lexicographic tool which is available either online or in a form of application and gives its users additional possibilities paper dictionaries cannot provide (such as graphics, comparative tables, audios with pronunciation, corpus search etc.). Such kind of dictionaries is quite rare in Ukrainian lexicography and probably not available at all in terminology.

Online dictionary gives much more opportunities for its user thanks to the technologies which can be applied not only in the process of its creation but also long after its presentation. The biggest advantage of electronic dictionaries is that they can be revised and corrected all the time and users' feedback can be much more productive. Application of algorithms capable of automatic collection of language material can ensure that the examples in dictionary will represent the actual state of the language. These aspects also have potential for theoretical elaboration and practical realization in computer lexicography.

In this respect we can also speak about the application of artificial intelligence for enrichment the dictionary's base. Though compiling dictionaries still remains a labor-intensive task, a lot of processes have been handled over to computer over the last decades and this tendency has, of course,

no signs of slowing down [3, p. 116]. Basic lexicographic processes, e.g. searching for collocations, definitions, examples, and translation equivalents, were step by step moving from humans to machines [3, p. 117]. The quality of automatically acquired data has been also improving and is expected to reach the level of human work. Therefore, one of the most promising fields in modern computer lexicography is automated acquisition of lexicographic knowledge as well as automatic term extraction.

According to a research by C. Müller-Spitzer, A. Koplenig and A. Töpel which they finished in 2011, the most important demand to an online dictionary is reliability of content [4, pp. 204-205]. Clarity, up-to-date content, speed and accessibility have all also shown high importance among the users' expectations. To our opinion, an electronic dictionary which meets these criteria cannot be built without a corpus, though the criteria "links to the corpus" received just a medium value of importance in electronic dictionary in the above-mentioned survey.

3. Importance of Corpus in Computer Lexicography

Electronic corpora have been used in lexicography for at least five decades, and nowadays no serious dictionary project would be undertook without at least one text corpus [1, p. 167]. The idea to give any user the whole power of information which can be found in corpora was born in the early 1990s and has since then been realized in many different lexicographical projects [1, p. 167]. P. Hanks argues that if lexicography is to play a role in helping people to understand the secrets of meaning in language, it seems certain it will be corpus-driven [5, p. 432]. He also adds that lexicography of the future will surely try to come up with electronic tools that can relate actual uses of words in texts or speech to the patterns of use those words are typically associated with [5, p. 432]. So, technologies of linguistic corpora have always been expected to fulfill a number of different tasks which are actually far beyond lexicography. Nevertheless, their use in lexicography seems to be the most productive at this moment.

Corpus is usually defined as electronic collection of texts chosen according to some extralingual criteria for optimal representation of a language or its variants which can be used for further linguistic enquires [6, p. 47]. Since corpora were not aimed for lexicography at the time of their inception, this definition is not suitable for lexicography and its tasks, so there was a need to distinguish a special corpus. On this basis we may speak of a lexicographic corpus that can be defined as a formed according to specific rules text corpus to be used by lexicographers with aim of creating an electronic card-catalogue or accumulating the required stances using a concordance [2, p. 18]. A concordance allows the researcher or a specialized lexicographic tool to investigate the context in which a lemma can be found [2, p. 51]. The data from concordance is what the lexicographer relies on while elaborating the dictionary entry, so a special attention should be paid to concordance's design.

Corpus-based dictionaries, or text-oriented, reflect the regularities of the speech and therefore are widely used for compiling frequency dictionaries and concordances [2, p. 14]. Their advantage in comparison with traditional dictionaries, which are sometimes referred to as system-oriented, is that the lexicographers have a possibility to observe real cases of a lemma's usage and they are not restricted by their subjectivity in defining the lexical units. While designing a dictionary, the researcher must take into account the shortcomings of existing dictionaries and make a decision about the main concerns and orientation of future dictionary [7, p. 567]. According to F. Čermák, three general aspects should be stressed: syntagmatics, usage and context — this implies paying attention to all relevant variants of words and phrases in a language [7, p. 560]. This idea is often neglected by the adopters of prescriptive methods in lexicography and codifiers of the language. In fact, while speaking about terminological dictionaries, prescriptive methods should prevail since terminology should be homogeneous and built a system with strict relations. A corpus-based terminological dictionary can show some deviations in terms' usage with a remark that such usage is usually considered among specialists to be incorrect.

4. Terminological Dictionary as a Specific Kind of Dictionaries

Terminological dictionary is usually defined as a book or electronic resource that presents terminology of one or more fields of science, technology or areas of human activity, compiled in accordance with specific criteria [8, p. 99]. Terminological dictionary is specific lexicographic product where lexemes are conveyed in ideographic manner and according to a conceptual interpretation [9, p. 60]. Meanings of scientific concepts come from centuries and even millennia of scientific research, which is why they are usually represented by a single noun phrase or, less commonly, a verb [5, p. 420]. For this reason, lexicographers should take into account not only linguistic factors but also pay attention to the specifics of philosophy in the sphere the dictionary should be used in as well.

The most important requirement for modern terminological dictionaries is conveying specialist knowledge in the most effective manner [8, p. 100]. Such a dictionary must meet the needs of its users which can be quite different: from laypersons to students, researchers and scientists/scholars. This factor directly influences the dictionary's structure on all its levels.

Terminological dictionaries can be classified as follows:

- according to number of languages:
    - one language;
    - two languages;
    - many languages;
- according to the way of term defining:
    - encyclopedic dictionaries;
    - learner's dictionaries;
    - translation dictionaries;
    - learner's dictionaries with etymological references;
    - other mixed types;
- according to the entirety of terminological entries:
    - full;
    - short [10, p. 80].

V. Dubchynskyi noted in one of his conference speeches that modern terminological lexicography must fulfill following tasks:

- create relevant dictionary typology;
- outline main features of terminological dictionaries;
- develop methodology of compiling terminological dictionaries;
- prepare basic requirements to terminological dictionaries;
- study micro- and macrostructure of dictionaries (i.e. how to organize and structure the dictionary entries so that they completely meet the expectations of intended users);
- analyze the ways and means of lexical acquisition for dictionaries (i.e. which lexemes should be included in the dictionary);
- define the basic rules for terms description (i.e. how to write terminological dictionary entries correctly and comprehensively);
- define ways and methods of using computer technologies in lexicography (this task is quite wide and includes the technologies which are used both for preparation of paper dictionaries and for creating electronic dictionaries) [11, p. 148].

This research aims at fulfilling some of the above-mentioned tasks, namely elaborating the structure of electronic terminological dictionary and collecting lexical materials for such a dictionary.

Terminological dictionaries are subject to more strict requirements than learner's dictionaries. The following demands should be met:

- dictionary must adequately cover the chosen subject area;
- information about lexical units should be available and complete for preparing a dictionary entry;
- worthless and redundant data should be excluded;

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

- composition of dictionaries in any sphere should be similar so that parallel use of different lexicographic sources at the same time is possible (we can also add that subsequent dictionaries should take into consideration materials from previous ones which is the best way to unification and elimination of discrepancies in some terminological field);
- harmony (coherence) between all elements in the methodology of dictionary composition should be achieved [12, p. 5].

Electronic form gives the possibility to combine advantages of different dictionaries' types in one lexicographic work. To our opinion, an electronic dictionary of legal terminology should include definitions of terms, examples of their usage on basis of a corpus, etymological information, tags for specific fields of legal studies, and translation equivalents.

5. Process of Compiling an Electronic Terminological Dictionary

Any lexicographic work begins with defining the principles for future dictionary. A dictionary must be based on a lexicological material; hence the first step in dictionary compiling is determining the type of the dictionary. Researcher must decide how many and which languages it will include, its volume, choose between synchronic or diachronic character of the dictionary, its grade of descriptiveness and/or prescriptiveness (these both aspects can be combined in a dictionary or just one of them can be selected). The next is to decide who the target users of the dictionary will be. Those may be students, linguists, specialists in some area (especially when speaking about a terminological dictionary), or wide public.

The third step is to define the sources of language material. Researcher must clearly define which sources may be trustworthy enough for a reliable dictionary. The range of these sources depends on the aim of the dictionary and its potential users. In order to ensure a representativeness of dictionary entries and their informativeness, the researcher should not rely on a single source and avoid sticking to one text genre only [7, p. 568]. In legal discourse, lexicographer can rely on following types of sources according to the classification by S. Šarčević:

- predominantly prescriptive texts (for instance, constitutions, international treaties, codes, laws etc.);
- predominantly descriptive texts (those are judgments, verdicts, court acts, petitions, appeals, letters, orders and so on);
- purely descriptive texts (researches, articles, monographs, studying books etc.) [13, pp. 127-128].

These sources of information have the highest level of validity for lexicographic work since they display correct and objective usage of terminological apparatus in jurisprudence. Corpus built on such linguistic materials will give lexicographer the best materials for preparing dictionary entries.

In the scope of creating an electronic corpus-based terminological dictionary we should also pay attention to terminology which is a substantial branch of linguistics.

5.1. Stages of Creating a Corpus-Based Dictionary

Usually compiling a dictionary consists of the following steps:

1. dictionary's designing (at this stage, its type is defined, user's needs are analyzed and main characteristics are drawn);
2. collecting lexical materials and building a thesaurus;
3. analyzing and describing the lexicon;
4. editing and final proofreading of dictionary before publishing [14, p. 26]

The typical model of compiling a corpus-based dictionary includes the following steps: corpus → concordance → typical model of a lemma's usage → dictionary [15, p. 94]. The first two steps can be roughly united in a stage of lexicographic corpus creation, and the last two — in the stage of index and dictionary entry forming [15, p. 94]. While the former stage is fully computerized, the latter requires a lot of manual work.

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

The first step requires accomplishing the following tasks: general planning, collecting text materials, coding, defining the relevant metadata and linguistic annotation of the corpus [16, p. 127-128]. Studying available dictionaries with aim to determine their disadvantages is also mandatory at this stage.

Deviation in corpus poses a big problem for researchers [17, p. 219]. It can be associated with such issues like size of a corpus, choice of corpus materials, their collection (e.g. from books, journals, newspapers, magazines, periodicals etc.), sorting of corpus materials, manner of material selection (e.g. random, regular, selective etc.), aim of corpus, methods of corpus cleaning, management of corpus files and so on [18, p. 1]. Creating a terminological dictionary makes some of these problems less likely. For example, corpus materials can be found in different instructions, journals, textbooks, articles, and documents which are highly-respected or widely used in the respective sphere.

Linguistic tagging of the corpus requires prescribing grammatical, lexical and other characteristics to the elements of the text [16, p. 128]. It should provide the user with all the information about the language of the text [15, p. 99]. The type of tagging is determined by the corpus application [15, p. 99]. The main tagging type is morphological since it is basis for syntactic and semantic tagging and is considered to be obligatory for modern text corpora [15, p. 99]. Semantic tagging is prescribing semantic categories and subcategories to any given lexical unit or phrase [19, p. 7]. Semantic tagging provides cross-lingual description for lexical semantics of all sorts of word tokens [20, p. 3]. For example, L. Abzianidze and J. Bos from University Groningen used for modality tags NOT (negation), NEC (necessity) and POS (possibility) [20, p. 4]. To our opinion, semantic tagging would not bring a lot for compiling a terminological dictionary because terminology is marked by universality and neutrality.

Colocations are typical syntagmatic lexical sequences that regularly appear in a text [15, p. 96]. Terminology tends to include a lot of composite terms so it is reasonable to apply association measures. Researchers argue that the MI-score is the most suitable for terminological collocations [21]. This method uses a logarithmic scale to determine the ratio between the collocation's frequency and the frequency of random co-occurrence of the respective words [21]. The advantage of this method is that low-frequency collocations will not be overlooked during the study which is of paramount importance in terminological lexicography.

Final stages of creating a lexicographic text corpus include designing a corpus manager and making the corpus available to users [15, p. 101]. In case of a terminological dictionary which should be continuously updated and serve the aim of developing Ukrainian terminology, such a dictionary must be made available online in form of a web site which is undoubtedly the most popular form of dictionaries nowadays.

Corpus is a perfect tool to create a frequency list of lemmas which should include all its paradigmatic forms. In this way, a lexicographer can define the likely complexity of a language unit before starting analyzing the corpus material and preparing the dictionary entries [7, p. 568]. It can also give information in which spheres a term is more or less used and help to determine the preferable term in case of synonymy (which is, of course, not desirable in any terminology).

A dictionary may include different types of entries which must be defined before the start of corpus analysis. The most common entry types are single-word lemmas, multi-word lemmas (problems of their identification and classification may arise there), technical apparatus (like abbreviations) and specialized types of entries (suppletive forms, synonymic lemmas or phrases) [7, p. 569]. This entry types would be common in all languages.

Apart from delivering contexts where a term can be found, a corpus also gives a possibility to display the changes in the frequency of term's usage. This can be visualized in form of graph reflecting how often a term was used in different periods of time.

So, corpus is quite flexible linguistic tool and electronic dictionary allows revealing its full potential.

5.2. Specifics of Electronic Dictionary's Structure

A common view in lexicography presupposes that a dictionary has macrostructure and microstructure. There are slight differences between macro- and microstructures in paper and electronic dictionaries which are to be considered in lexicographic work.

Macrostructure is usually defined as ordered set of lemmas in a dictionary [22, p. 79]. German researchers C. Kunze and L. Lemnitzer state that a lemma is an epiphenomenon between macro- and microstructure of a dictionary [22, p. 79]. Mostly dictionaries use the alphabetical order of lemmas which is the most suitable for Ukrainian, but other languages (e.g. German, where niche and nest alphabetical structures may be simultaneously applied, especially for terminological dictionaries [22, p. 80]) may require some modifications of lemmas organization.

Microstructure denotes the hierarchical internal structure of a dictionary entry for a given lemma as a concrete analysis of a given lexical entry [22, p. 80]. Each dictionary entry consists of two main elements — form and semantic comment. The former includes the information about phonological, orthographical, morphological and syntactic specifics of the lemma. The latter consists of semantic information (meaning, lexic-semantic correlations with other words or phrases, examples, translation equivalents), its pragmatics and etymology [22, p. 80]. In electronic dictionaries these elements and its components can be organized in blocks using various fonts and colors, which is the most user-friendly way of presenting information and one of the advantages of electronic dictionaries compared to paper ones.

Here we must admit that this division is mostly acceptable for paper dictionaries. When speaking about electronic dictionaries, we should, of course, chiefly discuss the problems of its microstructure because it defines the principles of composing a dictionary entry. Electronic dictionaries often copy the structure of entries in paper dictionaries. Therefore, the main difference between composing a paper and an electronic dictionary is its macrostructure.

Electronic dictionary may be regarded as lexical hypertext system due to its following features:

- modular and dynamic microstructure;
- linear structure of a traditional dictionary is replaced by a hypertextual, linked structure. This means more initiative and selection for information research;
- no concrete orientation that is essential for traditional dictionaries;
- traditional terms like lemma, lexeme and lexicon entry are abolished and lexical unit is used instead [22, pp. 89-90].

Electronic dictionary dictionaries are often called lexical informational systems [22, p. 88] because they also allow for a wider range of additional features, such as information about frequency, synonymy, antonymy, meronymy, hyponymy, hypernymy, etymology, special usage notes, relation to other terms.

Since it is not always possible or not always required to include all the collected terms, lexicographer must decide which terms to include into future dictionary [23, p. 13]. Prioritized must be terms which are important to the audience the dictionary must serve [23, p. 13]. Lexicographer can also use statistical data about term's occurrence in materials used in corpus the dictionary is based on.

In context of electronic dictionary, these rules may be applied to creating tags for specific legal branches. For example, a general legal terminological dictionary can contain tags as "constitutional law", "criminal law", "international law", "law theory" etc. This will allow users to compile lists of terms for different purposes, including study and researches.

5.3. Dictionary Entry vs. Terminological Entry

Dictionary entry is usually defined as an independent dictionary unit of some lexicographical work which corresponds to the dictionary's structure and is characterized by relevant information and lexical presentation alongside other types of lexical description (such as grammatical, stylistic, syntactic usage specifics [24, p. 198]). Dictionary entry is the most important part of any dictionary since it is "working environment" for dictionary's user, consequently lexicographer should take particular notice of it.

As to the dictionary's aim, three basic types of dictionary entries are distinguished:

- one-component entries (used in orthographical, morphemic, derivational etc. dictionaries);
- two-component entries (used in two-language translation dictionaries);
- detailed informational entries (used in learner's and encyclopedic dictionaries) [24, p. 199].

Most terminological dictionaries are more knowledge-oriented than usage-oriented [25, p. 334]. Researchers W. Martin and H. van der Vliet argue that terminological dictionaries rather aim to describe some sphere of knowledge whereas general dictionaries present the meaning and usage of lexemes [25, p. 334]. This fact which was observed since the beginning of relevant lexicography researches led scholars to coining the term "terminological entry" which should not be confused with the traditional lexicographic concept of "lexical entry" (as their names suggest, the former is used in regard to terminological dictionaries and the latter — to general dictionaries). Wright singled out following differences between these two types of dictionary entries:

- lexical entry represents a linguistic unit from general language, whereas terminological entry treats a systematically-defined subset of domain-specific special language;
- lexical entries represent all part of speeches, but terminological entries are mostly nouns and often verbs;
- lexical entry is defined by a lexeme and terminological entry — by a concept;
- each terminological entry treats just one concept, whereas lexical entry can contain multiple polysemic meanings;
- lexical entry provides all necessary grammatical information, while terminological entry may generally describe important grammatical differences;
- terminological entries are often organized in different logical groups and not just strictly alphabetically;
- lexical entry describes usage of a word and terminological entry may show recommended usage [25, p. 334].

Speaking about systematicity of terminology, we should mention that these relations are both logical and lingual [26, pp. 4-5]. Logical relations presuppose that terms are interconnected since they refer to extralingual phenomena of a particular science which are in systematic relations [27, p. 45]. Linguistic relations between terms mean that usual lexic-semantic (synonymy, antonymy, hypero-hyponymy, meronymy etc.) and derivational relations, which can be observed in general language, are also characteristic of terminological systems [26, p. 5]. These relations can be shown in electronic dictionary with tags and macrostructure elements in form of lists of previous and next terms in alphabetical order which are usually situated under the dictionary entry or on the left/right side of screen.

5.4. Principles of Preparing Correct Terminological Definitions

Legal as well as any other type of terminology is characterized by the following features:

- direct correlation with the denoted concept;
- fixed and usually monosemantic meaning;
- unambiguity and concision;
- contextual independence;
- stylistic neutrality;
- systematic relations with other terms;
- functional stability;
- minimum synonymy;
- derivational motivation for secondary terms;
- wide use and stability;
- tendency to short and easy for usage forms [28, с. 8].

The most important part of a dictionary is definitions that must follow special rules:

- meaning and use are inseparable because meaning can only be deduced from examined samples;

- meaning can be deduced only from real and sufficient contexts;
- each definition should be self-sufficient and do not depend on outside information;
- each definition should be worded sufficiently, so that it does not repeat other definitions, i.e. it should be unique;
- definitions should be based on real data;
- most terms must be defined by applying the genus proximum + differentia specifica dichotomy which presupposes that definition of a term must consist of an existing definition of its closest hypernym as base and adapted by the features which are specific to the phenomenon (thing) described [7, p. 570].

United States Environmental Protection Agency recommends adhering to following rules for definitions in terminological dictionaries:

- definition should not contain the defined word;
- definition must name the conceptual group to which the term belongs and provide its distinguishing characteristics;
- the same grammar and structure should be applied to all terminological entries;
- definitions should be succinct and easily understood by the target audience and focus on distinguishing characteristics rather than contain an encyclopedic description;
- jargon should be avoided or at least clearly marked [23, pp. 20-21].

When preparing a definition for a terminological entry, there may be some ambiguities caused by a number of reasons some of which are personal experience, believes and views of the lexicographers, their knowledge, aim of the dictionary, time of dictionary's compiling, political circumstances etc. [26, p. 12]. Unfortunately, not all of them can be completely eradicated during lexicographic work, but in our opinion, the influence of some of them on the dictionary's quality can be radically reduced thanks to the application of corpus technologies. For instance, using a vast array of documents will allow the lexicographers to prepare a very precisely and correct terminological entry and their knowledge or personal experience will not play here any huge role since they will be guided by real-world documents exclusively.

Terminology consists not only of words having just one lexical meaning. Scholars speak about the "degree of technicality" [29, p. 95]. A lot of such terms are found in legal discourse which means that lexicographers should not be blended by the primary meaning of the lexeme and its definitions in other spheres. So, for example, "freedom of speech" can be differently defined in law and in political sciences or philosophy. Lexicographer working in legal terminological field must therefore rely on purely legal sources.

6. Conclusions

Due to wide utilization of Internet technologies in different spheres, including lexicography and language studies, modern computer lexicography focuses on problems of electronic dictionaries which occupy a significant share on modern dictionary market. This topic is not enough researched in Ukrainian theoretical lexicography which was one of the main aims of this research.

One of the biggest advantages of electronic dictionaries is their potential multifunctionality. A paper dictionary is only then easy to use when it belongs just to one type, e.g. terminological dictionary. If many types are combined in one paper dictionary, it becomes too bulky and complicated for use. In addition to this, paper dictionaries become outdated after some period of time since their publishing. An electronic dictionary can instead be always up-to-date if tools for automatic upgrading are applied.

While diverse electronic learner's dictionaries are nowadays available online, terminological electronic dictionaries (or at least full-fledged electronic terminological databases) still remain individual instances. Modern Ukrainian terminological linguistics struggles to solve different problems, including proposing Ukrainian terms instead of transliterating English ones, substituting terms of Soviet period with modern Ukrainian terms, sustaining systematic relations in terminology etc. To our opinion, an electronic corpus-based dictionary is the best solution to many terminological problems.

The core technology for compiling a dictionary is text corpus which can provide an invaluable source of linguistic information. Since legal terminology is quite dynamic, a corpus can help to continuously adapt terminological entries and reflect changes in terminology. Such corpus will include all the legislation which can be accompanied by texts of different official documents, books and researches.

In the research were thus outlined the basic theoretical approaches to development of an electronic corpus-based terminological dictionary on all stages of its creation. Further elaboration of this problem can include principles of building terminological databases to be used in the electronic dictionary, its potential for machine translation as well as applying artificial intelligence for lexicographic tasks.

7. References

[1] G.-M. de Schryver, Lexicographer's Dreams in the Electronic-Dictionary Age. International Journal of Lexicography, Vol. 16 No. 2 (2003): 143-199.

[2] В. І. Перебийніс, В. М. Сорокін, Традиційна та комп'ютерна лексикографія. Навч. посібник. Вид. центр КНЛУ, Київ, 2009.

[3] Simon Krek, Natural Language Processing and Automatic Knowledge Extraction for Lexicography. International Journal of Lexicography, Volume 32, Issue 2, June 2019, Pages 115–118, https://doi.org/10.1093/ijl/ecz013.

[4] Carolin Müller-Spitzer, Alexander Koplenig, Antje Töpel, What Makes a Good Online Dictionary? - Empirical Insights front an Interdisciplinary Research Project. In: Kosem, Iztok/Kosem, Karmen (ed.): Electronic lexicography in the 21st century: New applications for new users. Proceedings of eLex 2011, Bled, Slowenien, 10.-12. November 2011. - Ljubljana: Trojina, Institute for Applied Slovene Studies, 2011, pp. 203-208.

[5] Patrick Hanks, The Corpus Revolution in Lexicography. International Journal of Lexicography, Volume 25, Issue 4, December 2012: 398–436.

[6] Vincent B.Y. Ooi, Computer corpus lexicography. Edinburgh University Press, 1998.

[7] František Čermák, Notes on Compiling a Corpus-Based Dictionary. Lexikos 20 (AFRILEX-reeks/series 20: 2010): 559-579.

[8] M. Łukasik, Terminological dictionary as a comprehensive cognitive and linguistic tool. In: Language in Different Contexts: Research papers = Kalba ir kontekstai, Volume 5 (1), 2012: 98-108.

[9] М. П. Дужа-Задорожна, Особливості укладання галузевих словників (на прикладі німецько-українського словника соціальної педагогіки/соціальної роботи). Актуальні проблеми філології та перекладознавства. Випуск 16, 2019: 59-64.

[10] О. В. Іванова, Сучасна термінографія, її основні завдання і проблематика. Науковий вісник Національного університету біоресурсів і природокористування України. Серія : Філологічні науки, Вип. 272, 2017: 76-83.

[11] В.В. Дубічинський, Терминографическая проблематика. Сучасні проблеми термінології та термінографії: Тези доп. міжнар. наук. конф., Київ, 2000.

[12] Y. I. Verbinenko, Multilingual terminological dictionary in the context of a national terminology system formation. URL: https://periodicals.karazin.ua/philology/article/download/5709/5267/.

[13] S. Šarčević, Translation of cultural-bound terms in laws. Multilingua. Volume 4, 1985: 127–133.

[14] Л. Конопляник, Етапи укладання термінологічних словників. Наукові записки Вінницького державного педагогічного університету імені Михайла Коцюбинського. Серія : Філологія (мовознавство), Вип. 21, 2015: 24-29.

[15] Т. В. Бобкова, Етапи розробки лексикографічного корпусу українських законодавчих документів. Мовні і концептуальні картини світу. Вип. 48, 2014: 89-104.

[16] T. McEnery, A. Hardie, Corpus Linguistics: Method, Theory and Practice. Cambridge University Press, 2012.

[17] D. Biber, Using Register-diversified Corpora for General Language Studies. Computational Linguistics, Vol. 19 (2), 1993: 219–241.

[18] Niladri Sekhar Dash, Issues in Text Corpus Generation. URL: https://www.researchgate.net/publication/327021521_Issues_in_Text_Corpus_Generation. DOI: 10.1007/978-981-13-1801-6_1.
[19] Н.П. Дарчук, Дослідницький корпус української мови: основні засади і перспективи. Вісник Київського нац. ун-ту ім.Т. Шевченка. Серія: Літературознавство. Мовознавство. Фольклористика. ВПЦ "Київський університет", Київ, № 21, 2010: 45–49.
[20] Lasha Abzianidze, Johan Bos, Towards Universal Semantic Tagging. URL: http://www.let.rug.nl/bos/pubs/AbzianidzeBos2017IWCS.pdf.
[21] Dana Gablasova, Vaclav Brezina, Tony McEnery, Collocations in Corpus-Based Language Learning Research: Identifying, Comparing, and Interpreting the Evidence. URL: https://onlinelibrary.wiley.com/doi/10.1111/lang.12225.
[22] Claudia Kunze, Lothar Lemnitzer, Computerlexikographie: Eine Einführung. Gunter Narr Verlag Tübingen, 2007.
[23] Best Practices in Terminology Development and Management: a Guide for EPA Editors and Stewards. URL: https://sor.epa.gov/sor_internet/registry/termreg/outreachandeducation/educationalresources/referencematerials/Terminology_Development_and_Governance_(Best%20Practices)_April_2014_Final.pdf.
[24] О. Черниш, Мікрорівнева організація електронного багатомовного термінологічного словника. Львівський філологічний часопис, (8): 197-203. https://doi.org/https://doi.org/10.32447/2663-340X-2020-8.31.
[25] Willy Martin, Hennie van der Vliet, Design and Production of terminological dictionaries. In: Piet van Sterkerburg (ed.). A Practical Guide to Lexicography. John Benjamins Publishing Company, 2003: 334-387.
[26] І. О. Голубовська, В. Я. Жалай, Н. М. Биховець, Т. Г. Линник, А. Ф. Пархоменко, І. І. Рахманова, Л. М. Рубашова, Укладання термінологічних словників: концептуальність реєстрових слів-термінів, дискурс словникової статті та напрямки майбутніх досліджень. Лінгвістика XXI століття: нові дослідження і перспективи. Логос, Київ, 2012: 3-20.
[27] Олена Медведь, До уточнення характеру та рівневої типології термінологічної системності. URL: http://ena.lp.edu.ua:8080/xmlui/bitstream/handle/ntb/1018/08.pdf?sequence=1&isAllowed=y.
[28] Н. В. Артикуца, Термінологія законодавства і проблеми законодавчих дефініцій. Актуальні проблеми юридичної науки. МОН України, Хмельн. ун-т управління та права, Хмельницький, Вид-во Хмельн. ун-ту упр. та права, 2007: 6-13.
[29] Gabriela Dima, A Terminological Approach to Dictionary Entries. A Case Study. Procedia - Social and Behavioral Sciences, Vol. 63, 2012: 93-98. doi: 10.1016/j.sbspro.2012.10.016.

PART II.
STUDENT SECTION

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

An Algorithm of Automated Identification of the Noun "*думка*" + Verb and Verb + Noun "*думка*" Metaphorical Model Meaning (Based on the Novel *Музей покинутих секретів* Written by Oksana Zabuzhko)

Vitalii Karasov, Olena Levchenko

Lviv Polytechnic National University, S.Bandera str., 12, Lviv, 79000, Ukraine

Abstract

The principles of automated identification of metaphoric combination of Noun "*думка*" + Verb and vice versa have been described in the thesis. The research has been carried out on the novel *Музей покинутих секретів* written by Oksana Zabuzhko and Grac v.10 corpus that served as the source of the research material. According to the search results in the text corpora, it was found that with the help of the models of two groups, both figurative and direct meaning can be realized. It should be noted that basically 71% of cases has figurative meaning and only rarely we come across a direct one. Such a feature of the considered metaphor's models is due to the fact that the concept of "*думка*" is an abstract in itself. Due to this, the search field is significantly expanded and does not require preliminary screening of direct meanings of the lexeme "*думка*". It should also be noted that the noun "*думка*" has such a semantic category t:ment r:abstr.

Keywords 1

Metaphor, automated identification of metaphor, semantically marked up corpus of texts.

1. Introduction

The study of automatic identification of metaphor has been traced in the works of such scientists as Lisa Gandy, Nadji Allan, Mark Atallah, Ophir Frieder, Newton Howard, Sergey Kanareykin, Moshe Koppel, Mark Last, Yair Neuman, Shlomo Argamon, Chang Su, Shuman Huang, Yijiang Chen, Tomek Strzalkowski, George Aaron Broadwell, Sarah Taylor, Laurie Beth Feldman. In particular, in the article *Robust Extraction of Metaphors from Novel Data* such scientists as Tomek Strzalkowski, George Aaron Broadwell, Sarah Taylor, Laurie Feldman, Boris Yamrom, Samira Shaikh, Ting Liu, Kit Cho, Umit Boz, Ignacio Cases, and Kyle Elliott walk through the steps of metaphor identification in detail. Their overall algorithm consists of five main steps from obtaining textual input to classification of input as metaphorical or literal [5].

In science at the turn of the 19th and 20th centuries, a complex approach to the study of metaphor is formed. This is due to the formation of cognitive linguistics as a science. In cognitive linguistics, "those aspects, structures and functioning of language that are associated with the acquisition, processing, organization, storage and use of human knowledge about the world around" [1].

J. Lakoff and M. Johnson are considered to be the founders of the cognitive approach to the study of metaphors. They argue that metaphor is "ordinary" and not some "extraordinary" part of the language; "Metaphor is common in everyday life, not only in language, but also in thinking and acting", and "the conceptual system in which we think and act is fundamentally metaphorical" [3].

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: vitalikkarasov99@gmail.com (V. Karasov); levchenko.olena@gmail.com (O. Levchenko)
ORCID: 0000-0002-8039-2811 (V. Karasov); 0000-0002-7395-3772 (O. Levchenko)


L. Cameron and A. Dagan propose the idea of using corpora in their research, emphasizing the need to focus on the analysis of discourse and corpora in the study of linguistic actualizations of conceptual metaphors [2].

2. Method for Identifying the Metaphor of the Model Noun "Думка" + Verb and the Model of Verb + Noun "Думка"

As a part of the study, an analysis was carried out not only of the **noun** "*думка*" + **verb** (see group I) model, but also of the **verb** + **noun** "*думка*" model (see group II). This approach is due to the aspiration to reveal as fully as possible the author's intentions which constitute the basis of the metaphor, as well as a natural feature of the Ukrainian language.

Using the tag **[lemma="думка"] [tag="verb.*"]** in the text of the novel, 20 contexts were found, two of which are phraseological units *дурень думкою багатіє* (used 2 times), *спадати на думку*. 11 are used in a figurative meaning (the metaphor *думка думку доганяє* is undertaken three times). Within this group, there is a subgroup of metaphors constructed according to the **noun** "*думка*" + **verb** "*бути*" model. Peculiarity of this subgroup is that this model is not always used by the author in a figurative meaning (перша *думка була*: щось із Катруською !), however, when combined with other lexemes, it acquires a figurative meaning (*думка була явно чужа*, *думка була бликнула*). That is, a model **noun** + **verb** "*бути*" for the implementation of a figurative meaning implies the presence of other elements.

The **noun** + **verb** formal model implements various metaphorical models. So, for example, *спадати на думку* – metaphorical model ДУМКА – ЦЕ ЛОКУС, *думка думку доганяє* – metaphorical model ДУМКА – ЦЕ ІСТОТА. The most common metaphorical model occurs in such cases: *думка кольнул*а, *думка працювала*, *підказувала думка* etc. In addition, there are several more unusual metaphorical models. One such example is the context *думка була бликнула*, where ДУМКА – ЦЕ СВІТЛО.

Using the tag **[tag="verb.*"][lemma="думка"]**, 11 contexts were found, all of which are used figuratively. Within this group, two subgroups of metaphors can be distinguished: 1) metaphors, where the lexeme "*думка*" acts as a subject, thus performing an action (*глухо скидається думка, блимає думка*); 2) metaphors, where the lexeme "*думка*" acts as an object (*ткнешся думкою, розігнатися думкою*). In direct meaning, the verb *розігнатися* can be used with an actant subject (for example, *автомобіль*). Quantitatively, these groups are approximately equal (5 of the first subgroup and 6 of the second one).

It is also notable that in this group, in contrast to the previous one, the author used samples of expressively labeled vocabulary *шибає думка*, *ткнешся думкою*. Here we also have a metaphor built on the model of a **verb**, **verb** + **noun** "*думка*" – *обморочує*, *блимає думка*. The accumulation of verbs, as well as their selection, expands the metaphoricity and semantics of the entire construction. One of the meanings of the lexeme "*обморочувати*" according to СУМ-11 is *збивати з пантелику*, and one of the meanings of the lexeme "*блимати*" is *світити уривчасто*. Combining opposite connotations of verbs (ДУМКА – ЦЕ СВІТЛО, а «НЕДУМКА» − МОРОК), the author assigns completely new properties to thought. By creating metaphors using such combinations, the author gets the opportunity to broaden his own idea. Using this example, the paradox of the metaphoricity of a literary text can be better realized: the author, while creating the language constructions that seem difficult to understand, in fact, becomes more understandable for the reader with their help.

The **verb** + **noun** "*думка*" model is inversion, which affects its role in the text as a literary device. Such metaphor has a peculiar affinity with the poetic text. Despite the fact that quantitatively this particular group is smaller (36%), its semantics and the author's methods of implementing metaphorical content are much more diverse, as it was substantiated by the analysis performed.

The metaphors of groups I and II revealed in the research process also differ significantly in their semantic content. So, the lexeme "*думка*" in metaphors and groups has human properties. The personalized properties of "*думка*" are most often associated with a swift movement: *думка набігає*, *думка доганяє*, *думка догнала*, *думка петляла*, *думка звернула*, which can be simultaneously

interpreted as a special feature of O. Zabuzhko's idiostyle and as the author's rethinking of the idea of "*думка*" as process, movement, action.

To automatically identify the metaphor, we should pay attention to the semantic categories of the verbs. For example, *петлями* – **ca:noncaus t:move**, *доганяти* – **ca:noncaus t:move d:pref**, *згубити* – **ca:noncaus d:pref t:poss**, *багатіти* – **dt:qual t:changest der:a ca:noncaus**, *бути* – **t:be:exist ca:noncaus d:root**, *є* – **ca:caus t:physiol d:root**, *робитися* – **der:v ca:noncaus**, *доганяти* – **der:v ca:noncaus t:move d:pref**, *працювати* – **der:s**, *багатіти* – **dt:qual t:changest der:a ca:noncaus**, *бачити* – **ca:noncaus t:perc d:root**, *звернути* – **der:v d:pref**, *набігати* – **der:v ca:noncaus t:move d:pref**. This will be a vital stage in the further identification of metaphors.

Hence, we see multiple repetition of the metaphor *думка думку доганяє*. The second group of metaphors is formed using verbs with an ambiguous connotation: *шибати*, *кольнути*, *обморочувати*, *блимати* etc.

3. Conclusions

The process of identifying a metaphor will have its own characteristics and specific stages, depending on the model of the metaphor under consideration. So, while identifying metaphors of the **noun** + **verb** model, a prerequisite is to take into account the potential of inversion as a literary device when creating a metaphor, that is, the analysis must be carried out in both directions: **noun** + **verb** and vice versa. It is also important that the model under consideration assumes the presence of other elements, without which its metaphoricity cannot be fully disclosed, or cannot exist at all. The verbs used in the creation of metaphors in direct meaning in Ukrainian language are used to denote the actions of creatures, certain tools, light sources, etc., and figuratively with the names of mental processes. Automated identification of a phrase as a metaphor involves several stages of analysis, among which the main ones are working with a semantically marked up corpus of texts, determining the semantic categories of lexemes, and working with dictionaries. While developing an algorithm for automatic identification of metaphors, we should pay attention to how collocations are combined: whether it is abstract with concrete or vice versa.

Attempts to concretize abstract concepts with the help of metaphors are a natural, although complex process, and involve a significant number of results. Therefore, further research on the verbalization of mental processes on extended material will form a holistic view of the speech features of their course in the linguistic picture of the world of Ukrainians, and, in particular, specific writers.

4. References

[1] T. G. Skrebtsova, Amerikanskaja shkola kognitivnoj lingvistiki [The American school of cognitive linguistics]. Saint-Petersburg., 2000. p. 6. (In Russian)

[2] L. Cameron, A. Deignan, The emergence of metaphor in discourse. Applied Linguistics. 2006. №27/4. P. 671—690.

[3] G. Lakoff, M. Johnson Metaphors we live by. Chicago: University of Chicago Press, 2003

[4] S. L. Mishlanova, M. V. Suvorova, Otsenka sootvetstviya protsedury identifikatsii metafory MIPVU kriteriyam podlinnoy nauchnosti metoda [Evaluation of Metaphor Identification Procedure VU (MIPVU) by the Criteria of a Truly Scientific Method]. Vestnik Permskogo universiteta. Rossiyskaya i zarubezhnaya filologiya [Perm University Herald. Russian and Foreign Philology], 2017, vol. 9, issue 1, pp. 46–52. doi 10.17072/2037-6681-2017-1-46-52 (In Russian)

[5] T. Strzalkowski, G. A. Broadwell, S. Taylor et al. Robust Extraction of Metaphors from Novel Data.Conference: In Proceedings of the First Workshop on Metaphor in NLP at the North American Association of Computational Linguistics Conference (NAACL-2013) Atlanta, USA.At: Atlanta USA. 2013. URL: https://www.researchgate.net/publication/267928109_ Robust_Extraction_of_Metaphors_from_Novel_Data.

Media Discourse Analysis Based on Ukrainian Spoken and Written Corpus

Maria Razno, Nina Khairova

National Technical University "Kharkiv Polytechnic Institute", Pushkinska str., 79/2, Kharkiv, 61024, Ukraine

Abstract

This article describes Ukrainian media discourse corpus creation and the relevance of the discourse analysis task applied on this corpus, using NLP methods, that can be implemented on Python programming language. It also includes the concept of different NLP methods and algorithms, its main varieties and the most popular Python packages and libraries for working with text data. Extraction of media discourse markers analysis algorithms based on the text processing is introduced in this study. It shows how to use NLP methods in practice.

Keywords 1

Media discourse analysis, Python, discourse markers, text processing, NLP, NLTK, corpus.

1. Introduction

Corpus linguistics is a branch of linguistics, the subject of which is the study of the principles and methods of forming text corpora, as well as the development of computer systems for their processing. Modern linguistic research actively uses the corpus method, and for all European languages, including the Slavic ones. Corpora construction is very important for maintaining various language researches. That's why, the creation of a corpus is a duty to the native language.

One of the main problems in NLP is the processing of discourse - creation of theories and models of how statements stick together, forming a coherent discourse. Due to this fact, for the discourse parsing it is necessary to extract the coherent segments which could be determined by discourse markers. To achieve coherent discourse, it is necessary focus on connectivity of relations. We take the two terms S0 and S1 to represent the meaning of two related sentences, where the term S0 can cause a state approved by S1.

Discourse markers are language units, used to establish a certain explicit connection between a given segment of the discourse and the previous utterance. Also, according to traditional division in terms of grammar, discursive markers can be divided into four classes - insert words (however, thus, by the way), conjunctions (and, but, and, or), subjunctive (at that time, because, except that) and prepositional expressions (as a result, because of, despite, compared with). Moreover, it is more thorough to distinguish discursive markers by semantic features. Discursive markers are considered to have a basic semantic meaning, which indicates the type of relationship between utterances or segments of discourse, and, accordingly, they can be divided into additional, causal, conditional, temporal and opposite.

2. Suggested Method

In this study we use discourse analysis for the identification of formal discourse characteristics (markers of correlations that link the arguments of discourse and form a complete opinion). Critical discourse analysis will be used to compare the frequency of use of co-reference markers in written and spoken media discourse. Moreover, as part of this work, a corpus of Ukrainian spoken and written

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: mari.razno@gmail.com (M. Razno); nina_khajrova@yahoo.com (N. Khairova)
ORCID: 0000-0003-3356-5027 (M. Razno); 0000-0002-9826-0286 (N. Khairova)

media discourse was created. The corpus includes texts from 2000 to 2020. The corpus is divided into two parts: spoken media discourse and written media discourse. The part with spoken media discourse has more than four hundred thousand words. It includes texts that are divided into two types: interviews and speeches. Texts from the interviews were collected from online versions of Ukrainian newspapers, magazines and other online publications. Texts with speeches consist of Ukrainian politicians speeches (New Year congratulations, plans for the future, etc). The part with written media discourse has more than a million words. The texts in this section are divided into discourse from the social network "Facebook" and newspaper discourse. It includes texts from the Ukrainian newspapers “Krymska Svitlytsia”, “Galnet” and “Ukrainska Pravda”. [1] A significant part of this section is texts from Facebook posts of Ukrainian politicians, activists, artists and scientists.

Some files for the corpus of Ukrainian media discourse creation were obtained from the corpus GRAK. [2] Also, in order to collect texts of articles from various Internet resources, the method of scraping web pages with the help of the “BeautifulSoup” library and was used. [3] All texts are stored in a .txt format with utf-8 encoding.

As for the analysis of media discourse in this work, we parse an xml-document with Ukrainian discourse markers. [4] For collecting markers the “xml.dom.minidom” module is used. The root element is an element named <entry>, in which a discursive marker is passed as a parameter "word". It also has <english_equivalent> element. The grammatical part of the discursive marker is passed to the <syn> element as a child of the <cat> element. The child element <sem> contains all the semantic features of the token. This document preserves all discursive markers of the Ukrainian language, which will be used for segmentation of texts in the Ukrainian media discourse corpus and search for discursive expressions: coherence and anaphora.

To collect discourse markers from the xml document. Some markers from the xml document have a colon in the middle or at the end. We use regular expressions to edit markers for more effective search. After that we count how many times each marker appears in the written and spoken parts of corpus. Also, we find the relative frequency of a certain marker per 1000 words in the corpus, as it is divided into two unequal parts. Then we use modules “pandas”, "matplotlib" and "seaborn" to visualize the results of the analysis.

3. Conclusions and Directions for Future Work

Analyzing the obtained results in the form of a data frame, we can conclude that discursive markers are more common in spoken media discourse than in written. Markers that have the highest relative frequency of appearance in the text are: "and", "also", "as", "in addition", "but". Most often, these markers connect the arguments of the discourse to indicate the "connection" between them. Their semantic feature is additionality, so they are used to continue the same thought.

To summarize, this research proved that discursive markers are characteristic of spoken discourse rather than written. It happens due to the fact, that human’s thought in oral communication is uncontrolled and, more often, spontaneous. In writing, markers appear less often, due to the fact that the author of the written message formulates the statement in advance.

In the future, it is planned to expand the corpus of Ukrainian media discourse on the basis of this corpus and develop software for parsing the media explicit discourse of the Ukrainian language.

4. References

[1] Newspaper "Krymska svitlytsya". URL: http://svitlytsia.crimea.ua

[2] General regionally annotated corpus of the Ukrainian language (GRAK) by M. Shvedova, R. von Waldenfels, S. Yarigin, M. Kruk, A. Rysin, V. Starko, M. Wozniak. Kyiv, Oslo, Jena, 2017-2019. URL: http://www.parasolcorpus.org/bonito/run.cgi/first_form

[3] Beautiful Soup Documentation. URL: https://www.crummy.com/software/BeautifulSoup/bs4/doc/

[4] Lexicon of Ukrainian discursive markers. URL: https://github.com/TScheffler/UK_DiMLex/blob/master/UK_DiMLex.xml

An Approach to Extraction of Verb-Noun Patterns from News Data Stream

Uliana Romanova, Svitlana Petrasova

National Technical University "Kharkiv Polytechnic Institute", 2, Kyrpychova str., Kharkiv, 61002, Ukraine

Abstract

The paper describes an approach to extraction of Verb-Noun patterns from news data stream. The linguistic tagging, namely algorithms for parsing, and methods for extracting collocations are analyzed. The algorithm for the automatic extraction of Verb collocations from the designed corpus of news texts is proposed. The Stanford Universal Dependencies parser is applied to identify Verb-Noun patterns. Then t-score is implemented for extracting collocations.

Keywords 1

Collocation, Verb-Noun pattern, Stanford UD parser, t-score, data stream, corpus of news texts.

1. Introduction

Automatic data processing tasks are becoming popular in today's world. Most of these data are stored in text corpora and processed by computer programs (concordancers). One of the stages of text corpus processing is the syntactic analysis of texts and, further, identification of syntactic units, i.e. collocations.

The relevance of using programs designed for syntactic processing of texts and identification of collocations is that these programs are in great demand in the tasks of computational linguistics, since collocations play the role of the basic units of meaning, translation, etc. Therefore, identifying patterns of word co-occurrence in text data is an important aspect for compiling dictionaries, learning languages, and natural language processing.

In general, automatic processing of text corpora involves tagging, i.e. formalized text and linguistic information. In particular, parsing mean some formalized representation of the structure of a sentence or phrase (collocation). The syntactic structure is represented in the form of a tree, the nodes of which are word forms, and links express the connection or relation between these words. There are two main approaches to the formal description of syntactic structures - the grammar of dependencies and the grammar of components.

Analyzing algorithms of parsing such as Earley algorithm, CKY (Cocke-Kasami-Younger), MaltParser and Stanford Universal Dependencies Parser [1], we suggest utilizing the latter that provides universal dependencies and Stanford dependencies, as well as phrase structure trees. As the result, coherent patterns could be identified using syntactic dependency relations.

To extract statistically significant patterns from data stream, association measures are considered to be applied. One of the measures is Pointwise Mutual Information (PMI) that quantifies the discrepancy between the probability of likelihood of collocates in joint distribution and their independent individual distributions. Mutual Information (MI) refers to comparison of the dependent context-related frequencies with independent ones (when words randomly appear in the context). Pearson's chi-squared test checks whether the words in the collocation are independent of each other.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: ulyaromashka5@gmail.com (U. Romanova); svetapetrasova@gmail.com (S. Petrasova)
ORCID: 0000-0001-6011-135X (S. Petrasova)

Measure Log-Likelihood is a logarithmic likelihood function. T-score measure estimates the strength of the association between collocates [2].

2. The Approach to Extraction of Verb + Noun Collocations

The combination of verb and noun is widely used for expressing the action and the object over which the action is done, for examples, "to commit a crime", "to drive a bargain". This type of collocations is of interest to reflect news events represented by Verb + Noun patterns.

The methods for the automated extraction of collocations generally perform the following tasks: finding potential candidates – providing a measure of the statistical significance of collocation in the text corpus; establishing the linguistic correctness of candidates; measuring the semantic non-composition of candidates and developing objective criteria for identifying an expression as a collocation.

In our study, we propose the following algorithm for extraction of Verb-Noun patterns from news data stream:

1. Applying the Stanford Universal Dependencies Parser to determine the syntactic relations of words in the designed corpus news texts. Verb-Noun patterns are represented by the dobj relation that means a direct object of a verb denoted the entity acted upon.
2. Forming a list of Verb-Noun patterns based on stage 1.
3. Calculating t-score to determine the strength of the association (connection) between collocates identified on stage 2.

T-score method [3] takes into account the frequency of co-occurrence of the keyword and its collocate, answering the question of how non-random is the strength of the association between collocations:

$$t - score = \frac{f(n,c) - \frac{f(n) \times f(c)}{N}}{\sqrt{f(n,c)}},$$

where n is a keyword; c is a collocate; f (n, c) means the frequency of occurrence of the keyword n in pair with the collocate c; f (n), f (c) is absolute (independent) frequencies of the keyword n and collocate c in the text corpus; N is the total number of word forms in the text corpus.

4. Forming the final list of collocations that represent the topic of current news.

3. Conclusions

Using syntactic parsing and statistical measure t-score, the algorithm for automatic extraction of Verb collocations from the corpus of news texts has been developed. The further work will be directed at the implementation of the designed algorithm to extract lists of Verb-Noun patterns from news data stream. To compare the precision of our approach with existing applications used for retrieving collocations in corpora, XAIRA corpus manager [4] is chosen.

The results of the study can be used in the fields of machine translation, language teaching, information retrieval, information extraction, etc.

4. References

[1] M.-C. Marnee, Ch. D. Manning, Stanford typed dependencies manual, 2010, URL: https://www.semanticscholar.org/paper/Stanford-typed-dependencies-manual-Marnee-Manning/

[2] S. Petrasova, N. Khairova, W. Lewoniewski, O. Mamyrbayev, K. Mukhsina, Similar Text Fragments Extraction for Identifying Common Wikipedia Communities, in: Data. MDPI AG, Basel, Switzerland, 2018. 3(4), 66. DOI: https://doi.org/10.3390/data3040066.

[3] S. Evert, B. Krenn, Methods for the qualitative evaluation of lexical association measures, in: Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics, Toulouse, France, 2001. pp. 188–195.

[4] XAIRA corpus manager, URL: http://xaira.sourceforge.net/

Linguistic Features of Designing Open-Ended Test Systems

Anastasiia Shapovalova, Svitlana Petrasova

National Technical University "Kharkiv Polytechnic Institute", 2, Kyrpychova str., Kharkiv, 61002, Ukraine

Abstract

The paper provides an algorithm of designing a test system for automated knowledge assessment through open-ended questions. The relevance of the use of open-ended tasks and problems of processing natural language answers are analyzed. The application of WordNet and regular expressions is proposed for designing samples of correct answers in the questionnaire.

Keywords 1

Knowledge assessment, test system, open-ended questions, natural language processing, WordNet.

1. Introduction

Knowledge assessment is an integral part of the learning process. It provides feedback in the student-teacher system. The use of test tasks in closed- or open-ended forms, in combination with new educational technologies, can significantly improve the quality of the educational process due to the implementation of teaching, monitoring, organizing, diagnosing and motivating functions [1].

Nowadays, test systems are actively applied to assess the level of knowledge, in particular, through open-ended test tasks that have significant advantages, e.g. objectivity of assessment, impossibility to guess the correct answer. However, using open-ended tests, problems such as ambiguity and synonymy occur. Therefore, there are significant limitations on the use of Natural Language Processing means for their automatic verification. In turn, the use of open-ended tasks in the pedagogical questionnaires and testing practice is being reduced.

Thus, the purpose of the study is to analyze the approaches of automatic processing of natural language answers for designing an open-ended test system.

2. The Analysis of Recent Researches

Natural Language Processing (NLP) combines linguistics and computer science to study the rules and structure of language and create intelligent systems capable of understanding, analyzing and extracting meaning from texts and language. It includes various tasks that belong to the unstructured natural language, e.g. part-of-speech tagging, dependency grammar, data mining, classification, sentiment analysis, question-answering (QA) systems and others. The development of the latter is the immediate task of this research.

The tasks presented in QA and test systems are usually divided into 3 types (Table 1). Test tasks are not similar in structure. However, they always set out the task to be performed through the constituent and subject parts in a particular form.

The constituent part includes:

- designation (written or oral instructions for the respondent on the task performance);
- instructions ("complete", "enter the missing word", etc.).

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: randomnick675@gmail.com; svetapetrasova@gmail.com
ORCID: 0000-0001-6011-135X

The subject part consists of:

- invariant components (constant content for the respondent, initial data on the basis of which the task is performed);
- variable components (parts that are directly selected, grouped or added).

The main principles of construction of the subject part are following:

1. Variability (task construction on the principle of interchangeability of elements).
2. Implication (logical certainty of the content of the problem with "If…, then…" construction).
3. Reversibility (the ability to swap invariant and variable components of the subject part of the task).

Table 1

Types of Test Tasks

Type of Questions	Definition
Closed-ended questions	Tasks that have one or several suggested answers.
Semi-open questions	Tasks that are aimed at arranging the suggested answers in the right ratio and entering into the distribution matrix of answers.
Open-ended questions	Tasks that provide free answers without any suggestions offered for a respondent to choose.

Nowadays, there are many interactive test systems. For example, the portal Lingva.Skills [2] suggests processing of grammatical patterns. One of the tasks is to write a sentence in English given in Ukrainian according to the template given above the exercise. The disadvantage of this test portal is a clear answer. Often the correct answer, but not completely coinciding with the given parameters can be perceived by the program as wrong.

Another example of the online test system with closed-ended and semi-open tasks is Osvita.ua [3]. The main disadvantage is that open-ended tasks are checked manually, not automatically.

A system with the ability to create open-ended tests is Puzzle English online platform [4] for self-study of English. A sentence entered by the user is divided into separate tokens each time the space key is pressed, analyzed and compared with the correct answer template. In this case, the disadvantage is that in open-ended tasks the use of a synonym or paraphrase is clearly perceived by the algorithm as an error.

In general, NLP systems almost never give 100% correct and expected results. The main challenge is ambiguity (or so-called redundancy) of different levels that cannot always be recognized by the machine [5].

There are different types of ambiguity:

1. Semantic ambiguity: syntactic, declensional, referential, literalness.
2. Synonymy.
3. A variety of grammatical constructions.
4. Disclosure of anaphors.
5. Free word order.
6. Homonymy.
7. Neologisms.
8. Linguistic variability.

Various linguistic resources are used to solve listed problems. Dictionaries of synonyms can be efficient in eliminating the problem of ambiguity and synonymy.

Other linguistic resources are thesauri. One of the most famous is WordNet [6] proposed to use in our algorithm. WordNet and similar thesauri describe the relation between the lexical meanings as a hierarchical system of groups of synonyms, i.e. synsets.

The third type of resources for solving NLP problems is ontologies. Most often they are focused on a specific subject area (e.g. UNSPSC deals with goods and services).

3. An Approach to Design an Open-Ended Test System

The algorithm of designing a test system for automated knowledge assessment through open-ended questions is as follows:

1. Background:
- collecting textbooks to create a theoretical block on grammar and vocabulary;
- designing constituent and subject parts of tasks (the development of a test structure, number of tasks, formulation of instructions);
- preparing samples of correct answers and rules for automated processing of free answers (applying WordNet and regular expressions);
- defining evaluation criteria.

2. Interface development includes:
- user authorization;
- theoretical block on each topic;
- variability of tests on each topic with a random order of tasks to exclude the possibility of cheating;
- evaluation of answers;
- highlighting errors with the correct answer after passing the test.

In our study, we propose the following types of open-ended tasks used in the test system:
- tasks with the addition of the correct grammar form of a word to a sentence;
- tasks with the addition of a synonym to a synonymous line (set);
- tasks with the addition of a keyword and phrase to a sentence;
- tasks with a detailed free answer - translation of a phrase.

4. Conclusions

The natural language processing problems within open-ended test tasks are analyzed. The overview of up-to-date test systems is provided. As a result of our research the algorithm of the automated knowledge assessment by means of open-ended questions is offered. The further work is the implementation of the open-ended test system in the educational process.

5. References

[1] E.N. Balykina, V.D. Skakovsky, Questions of the construction of test items, in: Fundamentals of Pedagogical Measurements. Minsk, 2009, Vol. 7. pp. 128-155.
[2] Lingva.Skills: Lingva.Skills Methodology, URL: https://lingva.ua/methodology.html
[3] ZNO Online, URL: https://zno.osvita.ua/
[4] Puzzle English Understand English by ear!, URL: https://puzzle-english.com/aboutus
[5] S.V. Petrasova, N.F. Khairova, Using a Technology for Identification of Semantically Connected Text Elements to Determine a Common Information Space, in: Cybernetics and Systems Analysis, 2017, 53(1), pp. 115–124. doi: https://doi.org/10.1007/s10559-017-9912-z
[6] WordNet: WordNet Search - 3.1, URL: http://wordnetweb.princeton.edu/perl/webwn

An Overview of Existing Machine Learning Methods for Gender Classification of Names

Anna Shleiko, Natalia Borysova, Zoia Kochuieva and Karina Melnyk

National Technical University "Kharkiv Polytechnic Institute", Pushkinska str., 79/2, Kharkiv, Ukraine

Abstract

The paper presents an overview of the existing machine learning methods for solving the problem of gender classification of the authors of the written texts by names: substantiates the relevance of the research topic, analyzes the existing methods of solving the task and selects the direction of further research.

Keywords [1]

Gender classification, supervised machine learning, methods of classification

1. Introduction

The problem of determining the gender of the authors of the Ukrainian corpora texts is of remarkable importance, since it enables increased functionality and improved performance, when using the corpora utilizing all of its functions. Due to the fact that the considered problem belongs to the area of classification problems, it can be solved using machine learning methods.

2. Machine learning methods for classification

A classification is the process of predicting the class of given data points. Classification requires machine learning algorithms in order to learn how to assign a class label to examples from the problem domain. In other words, during classification, a prediction is made for a class label for a certain example of the input data. A typical classifier uses a set of training data in order to understand how given input variables are related to a certain class [1]. Classification falls into the category of supervised learning, according to machine learning terminology, meaning that the learning occurs where a training set of correctly identified observations is available [2]. The analysis of works [1-3] showed that there are many classification methods for solving the task of determining (Fig. 1).

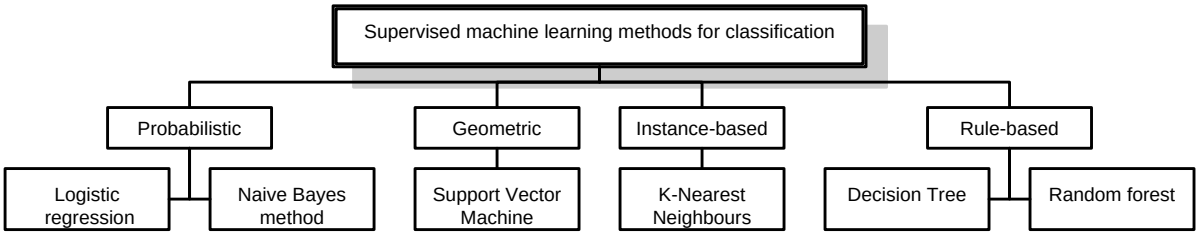


Figure 1: Supervised Machine Learning Methods for Classification

Comparison of the supervised machine learning methods for classification is presented in a more detailed way in Table 1 [1-3].

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: shleykoa@gmail.com (A. Shleiko); borysova.n.v@gmail.com (N. Borysova); aliseiko@gmail.com (Z. Kochuieva); karina.v.melnyk@gmail.com (K. Melnyk)
ORCID: 0000-0002-8834-2536 (N. Borysova); 0000-0002-4300-3370 (Z. Kochuieva); 0000-0001-9642-5414 (K. Melnyk)

Table 1
Comparison of the Machine Learning Methods for Classification

Method	Advantages	Disadvantages
Logistic regression	method is designed for this purpose (classification), and is most useful for understanding the influence of several independent variables on a single outcome variable	works only when the predicted variable is binary, assumes all predictors are independent of each other and assumes data is free of missing values
Naive Bayes method	method requires a small amount of training data to estimate the necessary parameters, it's extremely fast compared to more sophisticated methods	if we have the combination of features sometimes we can't explain the dependence of the classification result on them
K-Nearest Neighbours	method is simple to implement, robust to noisy training data, and effective if training data is large	it needs to determine the value of K and the computation cost is high as it needs to compute the distance of each instance to all the training samples
Decision Tree	method is simple to understand and visualize, requires little data preparation, and can handle both numerical and categorical data	it can create complex trees that do not generalize well, and decision trees can be unstable because small variations in the data might result in a completely different tree being generated
Random forest	reduction in over-fitting and random forest classifier is more accurate than decision trees in most cases	slow real time prediction, difficult to implement, and complex algorithm
Support Vector Machine	effective in high dimensional spaces and uses a subset of training points in the decision function so it is also memory efficient	method does not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation

As we can see, all the methods have both advantages and disadvantages. This must be taken into account when choosing the appropriate one or more methods.

3. Conclusions

After analyzing various methods of classification, it has been decided to use several supervised machine learning methods to solve the task of determining. This will provide an opportunity to compare the results of their work to choose the most fitting one.

4. References

[1] A. Sidath, Machine Learning Classifiers, 2018. URL: https://towardsdatascience.com/ machine-learning-classifiers-a5cc4e1b0623
[2] E. Alpaydin, Introduction to Machine Learning, third, 3rd. ed., MIT Press, Cambridge, MA, 2015.
[3] R. Garg, 7 Types of Classification Algorithms. URL: https://analyticsindiamag.com/7-types-classification-algorithms/

Unsupervised Open Relation Extraction

Yaroslav Tarasenko, Svitlana Petrasova

National Technical University "Kharkiv Polytechnic Institute", 2, Kyrpychova str., Kharkiv, 61002, Ukraine

Abstract

The paper describes an approach to open relation extraction based on unsupervised machine learning. The state-of-the-art methods for extracting semantic relations are analyzed. The algorithm of automatic open relation extraction using statistical, syntactic and contextual information is proposed. The results of the study can be used in information retrieval, summarization, machine translation, question-answering systems, etc.

Keywords 1

Information Extraction, Open Relation Extraction, semantic relation, TF-IDF, parsing, cluster analysis.

1. Introduction

Information Extraction is the task of collecting structured information automatically from large size of unstructured data by learning an extractor from labeled training examples for each target relation. The problem of extracting information from natural language texts remains highly relevant today. There are a large number of traditional information extraction methods based on rules, thesauri, and supervised machine learning methods. Such methods require a sufficiently large amount of a priori knowledge about the subject area, as well as linguistic resources: annotated corpora, glossaries, thesauri, systems of rules and grammar [1].

However, this approach cannot be applied for processing corpora with a large number of target relations or with no prespecified target relations. In response, a new paradigm of open relation extraction (Fig. 1) was proposed to enable the extraction of random relations from sentences by automatic identification of relation phrases, without a prespecified vocabulary [2].

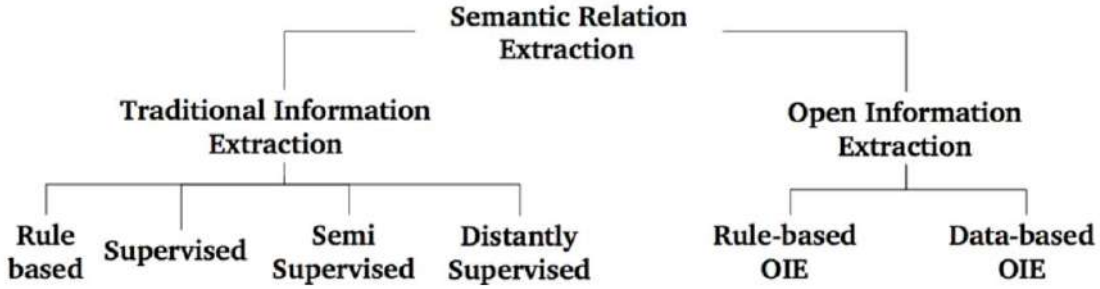


Figure 1: Approaches to Semantic Relation Extraction

Open relation extraction is the task of extracting new facts for a potentially unbounded set of relations from various sources such as knowledge bases or natural language texts [3]. The approaches to solving this problem can be divided into three groups: distant supervision, bootstrapping, and

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: yar.tarasenko99@gmail.com (Ya. Tarasenko); svetapetrasova@gmail.com (S. Petrasova)
ORCID: 0000-0001-6011-135X (S. Petrasova)

clustering [4]. The first two approaches are subtypes of semi-supervised machine learning, and the last one is a subtype of unsupervised machine learning.

Thus, due to the rapid growth of text information, the requirements for information extraction systems in new subject areas are being tightened. The methods of open retrieval of information can help to partially solve these problems without or with minimal manual work.

2. The Approach to Open Relation Extraction

The extraction of semantic relations is proposed to carry out on the basis of the unsupervised machine learning method, namely cluster analysis.

The task of open relation extraction can be divided into three stages:

1. Extraction of named entities and terms from texts using TF-IDF method.
2. Extraction of relations (triplets), i.e. pairs of entities a relation of which is potentially present, and text fragments (context) indicating the presence of this relation using Stanza dependency parser.
3. Clustering extracted relations with common semantics, from which a semantic relation can be formed based on contextual information.

As a result, linking semantically close triplets, semantic relations can be extracted from open data.

3. Conclusions and Further Work

Using syntactic parsing and unsupervised machine learning method, the algorithm for unsupervised open relation extraction from a text corpus has been developed.

The further work will be directed at the implementation of the designed algorithm to extract semantic relations from data stream. To compare the precision of our approach to retrieval of semantic relations in corpora with existing ones, the semi-supervised method [5] is chosen. The results of the study can be used in the fields of information retrieval, question-answering systems, automatic summarization, knowledge bases construction, machine translation, etc.

4. References

[1] O. Shanidze, S. Petrasova, Extraction of Semantic Relations from Wikipedia Text Corpus, in: Proceedings of 3rd International Conference: Computational Linguistics and Intelligent Systems (CoLInS 2019), Kharkiv, Ukraine, 2019, pp. P. 74–75.

[2] Peiqian Liu, Xiaojie Wang, A Semieager Classifier for Open Relation Extraction, in: Mathematical Problems in Engineering, 2018. doi: https://doi.org/10.1155/2018/4929674.

[3] F. Petroni, L.D. Corro, R. Gemulla, CORE: Context-Aware Open Relation Extraction with Factorization Machines, in: Association for Computational Linguistics, 2015. doi: 10.18653/v1/d15-1204

[4] A.O. Shelmanov, V.A. Isakov, M.A. Stankevich, I.V. Smirnov, Open information extraction from texts. Part I. Statement of the problem and overview of methods, in: Artificial Intelligence And Decision Making, 2018, pp. 47-61. URL: http://www.isa.ru/aidt/images/documents/2018-02/47-61.pdf

[5] D.S. Batista, B. Martins, M. J. Silva, Semi-supervised bootstrapping of relationship extractors with distributional semantics, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 499–504.

Categories of Information Retrieval Methods in Commercial Electronic Resources

Daria Budko, Natalia Sharonova

National Technical University "Kharkiv Polytechnic Institute", Pushkinska str., 79/2, Kharkiv, Ukraine

Abstract

The article provides an overview of existing methods of processing textual data in order to improve the site's performance and quickly find information about the product that the user needs: substantiated the relevance of the research topic, analyze the classification of problems, search and processing of textual information

Keywords 1

Information retrieval, methods of classification

1. Introduction

Recently, text analysis has attracted more and more attention in various fields. The continuous accumulation of textual data has led to the need to develop methods for carrying out information retrieval to ensure effective work with large corpora of texts.

2. Categories of Information Retrieval Techniques

One of the tasks of the Internet user is to find the necessary information upon request. In this regard, the problem of information retrieval arises, which is the process of identifying in a set of documents (texts) all those that are devoted to a specified topic (subject) and satisfy a predetermined search (query) condition or contain necessary (corresponding to information needs) facts, information ,data. The search process includes a sequence of operations aimed at collecting, processing and providing the necessary information to interested parties [2].

To simplify the search for information among a large array of information resources and make it relevant, categories of information search methods are used. (Fig. 1) [1].

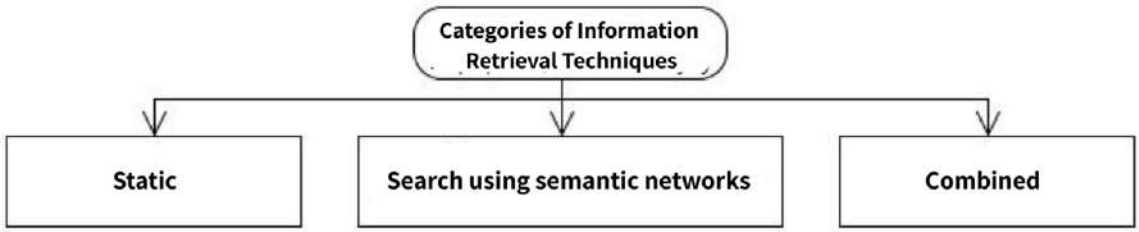


Figure 1: Categories of Information Retrieval Techniques

Comparison of categories of Information Retrieval techniques is presented in a more detailed way in Table 1.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: dasha.budko99@gmail.com (D. Budko); nvsharonova@ukr.net (N. Sharonova)
ORCID: 0000-0002-8161-552X (N. Sharonova)

Table 1
Advantages and disadvantages of using categories of information retrieval methods

Method	Advantages	Disadvantages
Static	Determines the weight of each word in the document. The advantage of such methods is that they have a high-quality mathematical model, with the help of which it is possible to obtain correct estimates of the relevance of documents.	They do not take into account the semantic load of the texts and the text of the request.
Search using semantic networks	Use data presented in the form of ontologies. In order to perform a search using this method, it is necessary to set the properties of the object. The advantage can be considered the fact that they take into account the semantic load.	They can only be used when electronic documents contain a semantic description of the content.
Combined methods	In addition to static methods, methods of semantic text analysis are used.	When you use combined categories of information retrieval techniques, errors are possible.

In conclusion, as we can see, all the methods have both advantages and disadvantages. This must be taken into account when choosing the appropriate one or more methods. After analyzing the various methods of information retrieval, you can see that each method can be applied depending on the task

3. References

[1] J.A. Richards, Jia Xiuping, Remote Sensing Digital Image Analysis: An Introduction. Berlin: Springer, 1999. 363 p.
[2] J.M. Stanton, Introduction to Data Science, Third Edition. iTunes Open Source eBook. 2012. URL: ttps://itunes.apple.com/us/book/introduction-to-data-science/id529088127?mt=11.

Problems of Evaluating Frameworks for Web Applications

Oleksandr Bieliaiev, Yuliia Selivorstova and Irina Liutenko

National Technical University "Kharkiv Polytechnic Institute", Kharkiv, 61002, Ukraine

Abstract

In the process of creating software, the task of choosing the best framework in this case often arises, for which it is necessary to evaluate the frameworks. The work considered the features of the frameworks that are used in web development. Criteria for the rationality of using frameworks for developing web applications are given. The components of front-end development are considered and the classification of frameworks is given. It is proposed to evaluate frameworks using the ISO / IEC 25010 quality model. The main functionality of front-end frameworks, which are used for evaluation, are formulated. Such methods as the synthesis of a formal decision-making model, qualimetry, SACS, analysis of variance, and factor analysis can be used to evaluate frameworks. Subsequently, to evaluate the frameworks, the SACS method was chosen as the best methodology for multi-criteria selection.

Keywords 1

Evaluation, quality, software, criterion, assessment methods, front-end, ISO 25010

1. Introduction

In recent years there has been a rapid development of Internet technologies. Sites that used to be a platform for hosting static content have now become multifunctional, interactive systems for providing a variety of information.

The development of the Internet is inextricably linked with the design of sites. The mass appearance of sites has provoked a problem of their quality.

The popularity of creating web resources has contributed to the development of various systems and programs that simplify the process of writing a site. These include frameworks which are the structure of a software system as well as software that simplifies the development and integration of various components of a large programming project. The correct choice of technology at the first stage of design guarantees 90% success in the promotion and operation of the resource on the Internet.

The research aims to improve the quality of software development by choosing the right framework for web application development based on the functionality provided by the framework and compliance with modern quality assessments.

2. Framework as a technology for web application development

A framework is a special-purpose software environment that is used to facilitate the process of combining certain components when creating programs allows you to add components as needed. Unlike a dynamic link library (DLL), the framework provides great functionality [1].

Besides, most frameworks are designed using the MVC (model-view-controller) pattern. This makes it possible to change any element of the system with minimal impact on other elements.

Web applications that use frameworks to simplify system development are divided into the following types: back-end (running on a remote computer), front-end (running in the user's browser),

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: albell.916@gmail.com (O. Bieliaiev); julia.selivorstova@gmail.com (Yu. Selivorstova); cherliv68@gmail.com (I. Liutenko)
ORCID: 0000-0003-0212-8895 (O. Bieliaiev); 0000-0002-5242-015X (Yu. Selivorstova); 0000-0003-4357-1826 (I. Liutenko)

and single-page application (a mix of back-end and front-end) [2]. Framework as a technology for web application development

Consider why frameworks simplify web application development [1]:

- support simultaneous use by multiple developers;
- provide clear documentation;
- productive in any system;
- help to develop a product in a short time;
- provide templates that simplify development;
- most frameworks have a low threshold of development.

3. Front-End Development

Front-end development is the part of creating a public segment of the web application with which the user is in direct contact and functionality which is usually performed on the client-side.

Components of frontend development [3]:

1. HTML (HyperText Markup Language) is the markup language of all elements and documents on a page and their interaction in the page structure.
2. CSS (Cascading Style Sheets) is a language for characterizing and stylizing the appearance of a document. Due to CSS, the browser understands exactly how to display items.
3. JavaScript is a language created to enliven web pages. The task of JavaScript is to respond to user actions, process keystrokes, move the cursor, mouse clicks. JavaScript also allows you to enter messages, send requests to the server, and loads data without reloading the page, and so on.

The classification of frameworks is shown in Figure 1 [4].

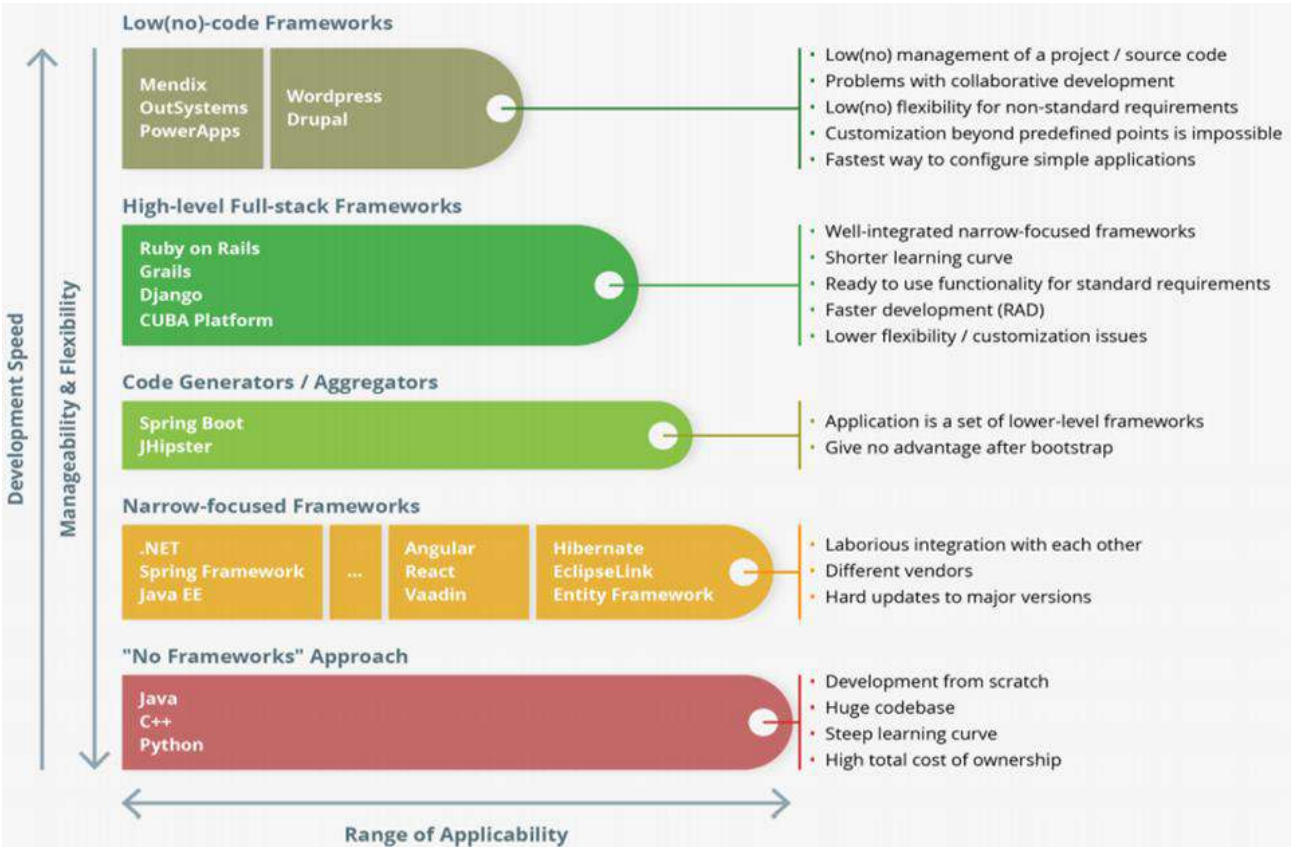


Figure 1: Classification of Frameworks

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

The following CSS frameworks will be considered for evaluation: Bootstrap, KUBE, Foundation, Skeleton, 960 Grid System, Kickstart, Yaml, Amazium, ConciseCSS. All of these frameworks give the developer the most needed styles to design a Front-end part.

4. ISO/IEC 25010 Quality Model

To more accurately select the framework for application development it is necessary to select quality criteria and their impact on the efficiency of development and implementation of the requirements. As a basis for the selection of quality criteria, it is proposed to use the ISO 25010 standard and the above frameworks will be considered as standalone software. According to ISO 25010, quality software meets the following criteria [5]:

- functional suitability;
- work efficiency;
- compatibility;
- usability;
- reliability;
- security;
- maintainability;
- portability.

5. The Main Functionalities of the Frontend Frameworks Used for Evaluation

To evaluate the frameworks, the parameters are shown in Figure 2. These are the same functionalities of frameworks that need when developing a web application.

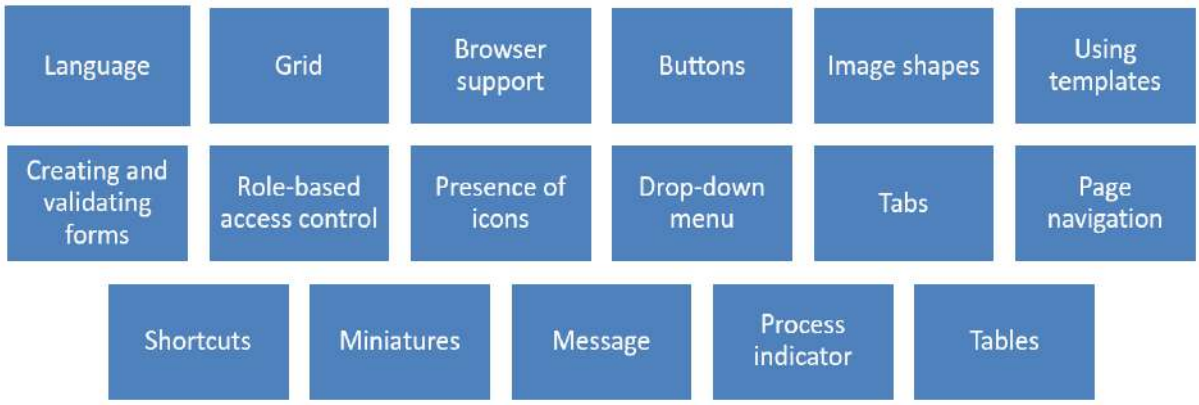


Figure 2: Functional Capabilities of Frameworks for Evaluation

6. Methods for evaluating frameworks

For ISO 25010 assessment it should be selected and considered assessment methods benefits. Several expert methods (synthesis of a formal decision-making model, qualimetry, SACS) and statistical methods (analysis of variance, factor analysis) were chosen.

The synthesis of a formal decision-making model is a method that fully meets our needs for calculating the evaluation of frameworks. This technique assumes that the conditions for decision-making are sufficiently defined, but the decision-maker must have complete information about the decision-making situation, all possible alternatives, and the consequences of their implementation and have a rational system of prioritization according to their importance [6].

Qualimetry is a universal method of quality assessment [7]. It provides qualimetric scales (order, intervals, and relationships). By choosing the most appropriate scale for the task, you can calculate the

most accurate assessment of technology. If you choose an ordered scale, you can only rank objects. You can also use the interval scale to determine how different one object is from another. And the scale of relations additionally provides information on how many times one object differs from another.

Analysis of variance is based on the typing of objects by functional feature, classification by criteria, formulation of estimates of the degree of similarity, and correction of results [8]. Thus, as a result, we obtain estimates for different objects, taking into account their characteristics.

Factor analysis is a section of multidimensional statistical analysis that combines methods for estimating the dimensionality of many observed variables by studying the structure of covariance or correlation matrices [9]. The task of the method of factor analysis is the transition from a really large number of features or causes that determine the observed variability to a small number of the most important variables with minimal loss of information.

Also, we proposed the SACS (Sequential Aggregation of Classified States) method for assessment, which relies on the use of verbal analysis methods in order to reduce the number of measurements of the criterion space [10]. This method allows, when solving practical problems, to choose both the best set of compiled criteria and an approach or a set of approaches to their creation. The proposed technology for solving multicriteria choice problems, in particular, through the formation of integral indicators, provides a systematization of available information, facilitates the choice of the final solution, makes it possible to analyze and substantiate the final results. Based on many factors, this method is the best choice for evaluating a suitable framework for developing a web application.

7. Conclusions

Existing frameworks for the development of software systems are actively used by developers in the development of web applications with different functionality and level of complexity. The analysis of the characteristics and capabilities provided by the framework allows you to make the right choice and choose the best option for a particular software, taking into account all the tasks, requirements, and capabilities of the development team. The right choice and use of the framework play one of the most important roles in the development of complex and simple web applications.

It is easy to make a mistake when choosing a framework, so it is important to first analyze all existing solutions, make sure you have the necessary functionality, good support, and compliance with this quality. When choosing be sure that the development team knows all the intricacies of working with the selected framework.

Thus, using selected frameworks, assessment methods, and functionality, you can evaluate each of the technologies, and the results of the article can be used by developers in the future to select a framework that meets the functional requirements for a particular software system. This will avoid making the wrong choice for future projects. Also, this article will allow beginners to navigate a large number of frameworks.

8. References

[1] D. Moseley, V. Baumfield, J. Elliott, M. Gregson, S. Higgins, J. Miller, D. P. Newton, Frameworks for Thinking: A Handbook for Teaching and Learning, Cambridge, Cambridge University Press, 2006, pp. 378. doi: 10.1017/CBO9780511489914.

[2] C. Saternos, S. St. Laurent, Client-Server Web Apps with JavaScript and Java: Rich, Scalable, and RESTful, 1st ed, O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, 2014, pp. 260.

[3] Front end development, 2020. URL: https://dan-it.com.ua/razrabotka-so-storony-front-end-chto-jeto-takoe-i-chem-otlichaetsja-ot-back-end/.

[4] Classification of frameworks, 2017. URL: https://www.cuba-platform.ru/blog/classification-of-development-frameworks-for-enterprise-applications/.

[5] ISO/IEC 25010, 2014. URL: http://iso25000.com/index.php/en/iso-25000-standards/iso-25010.

[6] Moskvin B.V., Decision-making theory: Textbook, Russia, Saint Petersburg, 2005, pp. 383.

[7] G. G. Azgaldov, L. A. Azgaldova, Quantitative assessment of the quality, Russia, Moskow, 1971, pp. 176.
[8] V. A. Yudenkov, Analysis of variance, Minsk, Biznesofset, 2013, pp. 75.
[9] J. O. Kim, C. U. Mueller, U. R. Klekka, I. S. Enyukov, Factorial, discriminant and cluster analysis, Moscow: Finance and Statistics, 1989, pp. 215.
[10] A. B. Petrovsky, G. V. Roizenzon. Multi-criteria choice with decreasing the dimension of the feature space: multi-stage SACS technology, Moscow: Academy of Sciences (ISA RAS), 2012, № 4, pp. 88-103.

The Approach to Creating the Recommendation System of Piano Pieces

Maiia Holshtein, Nadiia Babkova

National Technical University "Kharkiv Polytechnic Institute", Pushkinska str., 79/2, Kharkiv, 61024, Ukraine

Abstract

Nowadays a lot of descriptions of pieces of musical art can be found in Internet or in specialized collections. There is no recommendation system that offers certain composition for performance according to its difficulty level. This paper suggests the approach to creating the recommendation system of piano pieces. The approach is based on checking for collocations in descriptions of each composition. This paper shows the statistical method PMI used for searching the collocations indicating on certain difficulty level. In addition it also discusses the main problems during creating own recommendation system.

Keywords 1

Recommendation systems, machine learning, PMI, collocations, musical art, piano pieces, classification.

1. Introduction

Since there is no special recommendation system for determining the difficulty level of pieces of music at the moment, the objective of this study was to get acquainted with definition of conception of recommendation systems, features and problems of recommendation systems for creating own recommendation system.

Recommender systems is one of the classes of machine learning algorithms, that offers a user «relevant» samples and subclass of information filtration system, that builds a rating list of objects (films, music, books, news, web-sites), which can be preferred by a user. Information from user profile is used for these purposes [1].

Modern recommendation systems have a number of standard problems and disadvantages, investigation of which and development of methods for them overcoming are relevant and scientific-practical objective. The main problems of recommendation systems are Cold-start Problem, CSP, filter bubbles [2], problem of sparse data, fraud, synonymy [4], “Outliers” [2], The other problems are absence of personalization, maintaining confidentiality, absence of novelty and adaptability of user preferences [5].

2. Suggested Method

Recommendation system was developed on the base of corpus of descriptions of piano pieces. Usually an indicator of complexity of certain composition can be presented in two or three words. So bigrams and trigrams from given corpus were separated for further work. There was used statistical method PMI to search for collocations, as it is well suited for finding for terminological combinations and nominations.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: maja.golshteyn@gmail.com (M. Holshtein); nadjenna@gmail.com (N. Babkova);
ORCID: 0000-0001-9181-9641 (M. Holshtein); 0000-0002-2200-7794 (N. Babkova);

Pointwise mutual information (PMI) is a measure of connectivity, that is used in theory of information and statistics. PMI couples of results x and y, that belong to discrete random variables X and Y, give a quantitative evaluation of divergence between probability of their coincidence using their common distribution and their own distribution provided their independence:

$$PMI = log\,2\frac{p(W,W_1)}{p(W)*P(W_1)},$$

where W is the main word, W1 is context surrounding (collocate), p(W,W1) — frequency of common encountering of 2 words (context), p (W) p (W1) – absolute (independent) frequencies of encountering of 2 words.

Thousand bigrams and thousand trigrams of resulting collocations using PMI method were taken. As a result, the following bigrams and trigrams were obtained: 'peer gynt', 'blossom tremolos', 'confident agility', 'learn johann pachelbel', 'certain satie gymnopédies', 'maintains improvisational quality', etc. Bigrams and trigrams from these collocations suited for indication the difficulty level of piano pieces were selected. For example, 'sudden outbursts', 'stylistic understanding', 'considerable talents', 'keyboardist acrobatic skill', 'extemporaneous improvisation impromptu', 'various successive parallel, obligatory' 'competent students', etc. There was evaluated the precision of PMI method according the formula:

$$precision = \frac{tp}{tp+fp},$$

where tp is the number of relevant results (in given occasion it is the number of collocations indicating on difficulty level) and fp is the number of irrelevant results (not referred to indications of difficulty level). Thus, the precision of PMI method was evaluated as 14,37%.

The limitation of PMI is that this measure is prone to frequency offset and it will weigh lower or zero frequency terms more over higher frequency terms. This may result in wrong collocation relation. One way to fix this is to apply Laplace Smoothing. It can lead to incorrect relation of collocation. To solve this problem Laplace Smoothing can be applied [6].

3. Conclusions and Directions for Future Work

At the moment have been created the corpus of 75 descriptions of piano pieces, separated bigrams and trigrams for determining difficulty level by existence of certain collocation in the description of certain composition, determined four difficulty levels for classification pieces of musical art (Elementary, Intermediate, Late Intermediate, Advanced) and referred piano pieces to the corresponding difficulty levels. For example, compositions, descriptions of which contain collocations 'ears beginner', 'total beginner', omit ornamentation', 'nice little waltz', 'beautiful twinkling melody', 'steady succession notes' were referred to Elementary level, compositions, descriptions of which contain collocations 'drowsy lullaby', 'powerful crescendo', necessary balance', 'accelerando progresses gentle', 'precise clear articulation', 'variations separated intermezzo' were referred to Intermediate level, compositions, descriptions of which contain collocations 'unsuspected heights', 'textural elaborations, 'muscular exuberance', 'require bit virtuosity', 'acrobatic triplet figures', 'jazzy syncopes time' were referred to Late Intermediate level, compositions, descriptions of which contain collocations 'incredible dexterity', 'running chromaticism', 'daredevil leaps', 'virtuosic piu mosso', 'unexpected turns harmony', 'constant relentless motions' were referred to Advanced level.

Thus, having checked all the descriptions, following results were obtained: Grieg's Morning Mood, Haydn's Sonata in G, Kabalevsky's Clowns were classified as Elementary, Bach's invention №13, Joplin's The Entertainer, Mozart's Sonata N16 were classified as Intermediate, Mendelssohn's Songs without words, Bach's Prelude and Fugue in C WTC1, Schubert's Sonata in A Op. 664 were classified as Late Intermediate and Chopin's Revolutionary Etude, Liszt's Hungarian Rhapsody N6, Rachmaninoff's Piano Concerto N2 were classified as Advanced.

The main problems during developing of the recommendation system:

1. Data set in the corpus. There is no collection of technical descriptions of piano pieces. So the texts of descriptions of compositions were collected from approximately 15 accessible sources.

2. Synonymy. Since these texts were written by different authors, some of them could be descriptions of execution technique using specialized musical terms and other ones could be simple reviews from primary or secondary school pupils or them parents, so one indicator of complexity were presented by different words (piu mosso and more agile).
3. Using collocations separately from the context. During the classification the piano pieces by existence certain criterion in certain description the problem of accuracy of determining difficulty level appeared. A collocation, which is an indicator of Elementary level could appear in a description of more complex composition and vice versa. For example, the description of F. Liszt's third etude «La Campanella», which is referred to Advanced level due to its technique, contains collocation «easier execute», which let refer this composition to Elementary class. To solve this problem it is planned to consider collocation in context and in this way evaluate them complexity more accurately.

In the future it is planned to create a test set of piano pieces and evaluate the precision of developed recommendation system work.

4. References

[1] An Easy Introduction to Machine Learning Recommendation Systems, 2021. URL: https://www.kdnuggets.com/2019/09/machine-learning-recommender-systems.html.
[2] Yak-vyrvatysia-z-informatsiinoi-bulbashky, 2020. URL: https://wz.lviv.ua/blogs/388693-yak-vyrvatysia-z-informatsiinoi-bulbashky
[3] Linden G., Smith B., York J. Amazon.com recommendations: item-to-item collaborative filtering. IEEE Internet Computing, 2003. 7 (1), 76–80. doi: https://doi.org/10.1109/mic.2003.1167344
[4] Meleshko Y. V. Problemy recomendatsiynykh system ta metody yikh rishennia, Systemy upravlinnia, navigatsii ta zvyazku. 4, 2018: 120 – 124. doi: https://doi.org/10.26906/SUNZ.2018.4.120
[5] V.V. Lytvyn, V. A. Vysotska, V. V. Shatskykh, I. V. Kogut, O. S. Petruchenko, L.V. Dziubyk, V. V. Bobrivets, V. M. Panasiuk, S. I. Sachenko and M. P. Komar. Design of a recommendation system based on collaborative filtering and machine learning considering personal needs of the user. Eastern-European Journal of Enterprise Technologies, 2019. 4(2 (100), 6–28. https://doi.org/10.15587/1729-4061.2019.175507
[6] Collocation discovery with PMI, 2020. URL: https://py.plainenglish.io/collocation-discovery-with-pmi-3bde8f351833

Implementation of the Removing Homonymy by Collocation System

Anastasiia Khluieva, Zoia Kochuieva, Natalia Borysova

National Technical University "Kharkiv Polytechnic Institute"2, Kyrpychova str., 61002, Kharkiv, Ukraine

Abstract

This article describes implementation of the removing homonymy by collocation system method in Ukrainian language, which can be implemented on Python programming language. It also includes the relevance of removing homonymy phenomenon and difficulties associated with it. A study of various methods of removing homonymy was carried out and conclusions are mentioned in this work. In article there is an algorithm of the removing of homonymy, particularly homoforms, by collocation system method.

Keywords 1

Homonymy, methods of removing homonymy, collocations, corpora

1. Introduction

Modern intellectual systems do not process texts in natural languages with sufficient quality due to the presence of such a linguistic phenomenon as homonymy. This concept is marked by ambiguity of approaches to its study and, in fact, interpretations. This has caused a relentless interest on the part of linguists, because the phenomenon still remains relevant to his research. Homonymy is a difficult phenomenon, it has many aspects that require comprehensive analysis.

The main purpose of its research is usually classification; sources of origin of homonyms and their distinction from polysemous words; issues of interlingual homonymy, as well as the removal of homonymy in the automation of translations, etc.

Thus, the purpose of this work is to develop a system for removing homonymy using the method of collocations.

To achieve this goal, we have the concept of homonymy as a linguistic phenomenon, its types and problems of origin, as well as possible methods of removing homonymy. The subject of the study is the removal of homonymy by using the method of collocations. To solve this problem, the following tasks were formulated:

1. Performing an analytical review of the literature on the topic of homonymy, its types, problems that arise for its removal, and methods for eliminating this problem.
2. Development of an algorithm for removing homonymy by collocation.

2. Homonymy Phenomenon

Homonymy is a synchronous phenomenon in terminology, which is based on the absence of common sem in the meanings of the same terms of expression of terms and commonly used words, terms of one or more related or unrelated areas of knowledge and human activity. Depending on the formal, more precisely, on the grammatical (morphological) and semantic, relations between homonymous words, there are several types of homonymy: lexical (sound coincidence of different in meaning linguistic units belonging to the same part of speech), grammatical (sound coincidence in

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: nastiahlueva@gmail.com (A. Khluieva) ; kochueva@kochuev.com (Z. Kochuieva) ; borysova.n.v@gmail.com (N. Borysova)
ORCID: 0000-0002-4300-3370 (Z. Kochuieva); 0000-0002-8834-2536 (N. Borysova)

separate grammatical forms of language units of different meanings), word-forming (sound coincidence of morphemes of different word-formative meanings), syntactic (sound coincidence of different syntactic constructions), phonetic (a sound coincidence of language units of different meanings with different spellings), graphic (a graphic coincidence of language units with different pronunciations). Among the grammatical homonymy there are homoforms, homographs, homophones. Homoforms are morphological homonyms that are distinguished on the basis of sound coincidence and the same spelling of word forms belonging to different lexical and grammatical classes or different forms of the same word. Precisely this type is an object of our study.

3. Removing Homonymy Methods

Historically, almost all methods of removing homonymy are divided into two groups:

1. Methods based on rules. In turn, are divided into:

- Methods with manual entry of rules.

An illustrative example of a method with automatic rule generation is the method of the American linguist Eric Brill. Transformation rules are a set of "old tag, new tag, condition", and the application of the rule is to replace the old tag with a new one when the specified condition. The disadvantage of this method is the decrease in the increase in accuracy with increasing number of rules, which, however, is fully consistent with the Pareto principle: "80% of the effort provides 20% of the result." At the same time, the principle works in the opposite direction: performing only one initialization step is enough to achieve high accuracy of homonymy removal.

2. Methods with automatic rule generation.

- Methods based on statistics.

Statistical methods of removing homonymy allow us to calculate the probability of each possible variant that occurs on the basis of statistics: if in any context the noun occurs more often than the union, the homonym found in the same context will be more likely to be a noun than a union (if these options allowed by the dictionary). The disadvantages of probabilistic methods are the duration of formation and marking of the body of texts and low accuracy of analysis, caused by the free order of words in inflected languages. Thus, there is a need to create a method of removing ambiguity for the homoform of different parts of speech, which does not require a large number of rules or a body of manually marked texts.

Therefore, there is a need to develop a hybrid method that uses both rules and information from texts published on the Internet, which does not require repetition of the parsing procedure in the case of homonymy.

4. Collocation System Method

To get started, we need to address the issue of collocations. As it's known, collocations are compounds of words, the probability of using which together is greater than the probability of using these words separately from each other. This issue of collocation research is one of the leading ones in applied linguistics. Precisely because these compounds are usually stable, they need a certain form of words to match each other, which can help with the removal of homonymy. Based on the results of the analysis of sources on the problem of homonymy, namely the emergence of incomplete grammatical homonyms, among homonymous pairs of words homonymy with the noun most often occurs, so we take collocations with nouns to remove homonymy. From the works of Ukrainian scientist Bobkova T.V. [1], we found that collocations with nouns in the Ukrainian language are characterized by the following part-of-speech combinations:

1. adjective + noun,
2. preposition + noun,
3. verb + noun,
4. noun + noun.

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

Among the above types of collocations in the language, collocations such as “adjective + noun” (1) (ukr. “широке поле”) and “preposition + noun” (ukr. “після дати”) (2) are most often used, and we will take such collocations for our research.

To use type 1 collocations, it is necessary to find homonymous pairs, to investigate the context of these words. To get the most commonly used context with these words, we will use the Sketch Engine [2] platform, where you can find the context for a word from a text in the Ukrainian language with a volume of more than 2 billion words.

To use type 2 collocations, we will refer to the frequency dictionary of collocations and delete collocations with a frequency of more than 30 cases per 1 million words.

The base of collocation is 821 collocations (615 collocations of preposition and noun and 206 collocations of adjective and noun). In our study, we will not reduce the word to the lemma, because the study of homoforms requires exactly the form in which the problem of homonymy arises. Therefore, the collocation database will use the form of the word in which difficulties arise, with the appropriate context for comparison directly with the form of the word used in the sentence.

Hence, on the basis of the revealed sequences we created particular algorithm of work:

1. The user enters text in the interface window, then text goes through the tagging process.
2. The program checks whether the preposition with context is contained in our database. Its tag changes to the one specified in the database as long as the match is found.
3. The program checks whether the noun is contained in the context of the adjective from the database. If the match is found, its tag changes to the one specified in the database.
4. As a result, the program produces text with assigned parts of speech to each word.
5. The user receives standard tagged text provided there is no matches are found in the databases.

Given method and algorithm can be easily extended for usage of noun and not only, and continue to increase the base of collocations for Ukrainian language.

5. References

[1] Sketch Engine concordance for Ukrainian language based on Ukrainian Web 2014 corpus. URL: https://auth.sketchengine.eu/#login?next=https%3A2%2Fapp.sketchengine.eu%2F%23dashboard%3Fcorpname%3Dpreloaded 252Fuatenten14

[2] Linguistic portal Mova.info Dictionary Of Ukrainian Prepositional Collocations. URL: http://www.mova.info/Page.aspx?l1=6

Special Aspects of Translation of Medical Instructions

Vladyslav Khramtsov, Olena Orobinska

National Technical University "Kharkiv Polytechnic Institute", Pushkinska str., 79/2, Kharkiv, 61024, Ukraine

Abstract

The subject of the research "Special aspects of translation of medical instructions" is due to the fact that in our time the assortment of new types of technologies in medicine which fulfils the market of Europe and Ukraine is growing. The aim of the study is to convey the content accurately, as much as possible to preserve the features of the style. In order to achieve it, we need to know the subject and the related terminology (in our case, the medical one) and to achieve the adequacy of the translation of the text of this industry.

Keywords 1

Medical terminology, terminological word combinations, single-component terms (two-component, etc.), terms-eponyms, thesaurus in the field of medicine MeSH.

1. Introduction

Medical corpus is characterized by a significant set of terms, abbreviations, and special words. At the same time, these same concepts have different names depending on the language of the region. The relevance of the topic is due to the fact that an increasing range of drugs of chemical, plant and animal origin with a wide range of therapeutic properties, today flood the pharmaceutical market in Europe and Ukraine [1]. A decade ago in our country, medical professionals practically did not use terms of English origin, but now there are more and more such terms [3]. Therefore, it is necessary to approach the analysis of the text thoroughly, select terminological units, and use special dictionaries.

Many of the special vocabulary is based on the traditional "language of medicine" of Latin [2]. At the same time, it gives rise to certain difficulties, since some terms of Latin origin function in parallel with the corresponding English words. For this purpose, we see it more effective to use the thesaurus while translating the medical corpus [5].

2. Suggested Method

Structural analysis of the studied terminological units in the field of medicine and health care showed that according to the ratio of the number of components among them stand out:

1. single-component terms (rectouterina, bioplasm, biofilm, biodeterioration)
2. two-component (biorthogonal decomposition, excavatio rectouterina, HeLa cells, Moor's clamp, Mulligan's forceps)
3. three-component (biopolimer drilling mud, biodegradable dosage form).

Even a superficial analysis reveals a significant dominance of the two-component terms in medicine. Therefore, the preference of two-component over one-component terms can be explained by the fact that two-component terms better meet one of the most important requirements for a term - its brevity. The preference of two-component terms over three-component or four-component terms can be explained by the fact that they save speech effort.

Another important type of word combination terms is eponym.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: vladkramtsov69@gmail.com (V. Khramtsov); orobinska@khpi.edu.ua (O. Orobinska)
ORCID: 0000-0001-8396-4136 (O. Orobinska)

An eponym is a term that contains a proper name as well as a common name for a scientific concept (Євста-хієва труба- 1) guttural duct; 2) otosalpinx; 3) syrinx). Term-eponyms can be formed in a non-fixal way from a proper one by metonymic transfer, and they can also be affixal formations from a proper one [4]. During the study of medical terms of eponyms in the scientific literature, we have often encountered the opinion that eponymic concepts are inconvenient to use and pollute medical terminology.

3. Conclusions and Directions for Future Work

The materials of the study were medical corpuses in English, the analysis of which allowed us to identify common patterns and trends in the broad English-language medical space. In the structure of modern medical terminology an important place belongs to the terms and phrases denoting individual concepts of the branch and conveying their complex internal correlation and multidimensionality. We have found that one of the most important types of terms is eponym, that is, a term that has in its composition a proper, as well as a common to denote a scientific concept. There is a group of eponymic terms that have finally entered the medical vocabulary and become fixed in it.

The main difficulties in the translation of English terminology of medicine and health care are related to extralinguistic factors, namely the low prevalence of a particular concept or phenomenon in the domestic scientific or socio-economic environment or its absence. In addition, the process of terminosystem neologization does not always coincide in its vector development with the updating of the Ukrainian terminosystem, and therefore translators are often intermediaries between the two languages in terms of entering into widespread use of certain terminological units.

So, the translated material during the processing of texts in the field of medicine and health care must contain carefully checked medical terminology, so as not to be the cause of unacceptable error in the relevant professional activity. A translator must update and enrich his/her background knowledge all the time, follow the latest scientific translation publications, in order to properly assess and take into account tendencies of development of the English- and Ukrainian-language medical terminology system. When applying both lexical and grammatical transformations, the translator must remember that distortion of information in such a text is inadmissible. Semantic rather than stylistic factor must dominate in translation of medical texts.

The prospect of research is terminological systems of individual medical branches in comparing and analyzing Ukrainian-language equivalents of medical terms in various fields in order to identify terminological homonymy and polysemy.

4. References

[1] L. W. Barsalou, Frames, concepts, and conceptual fields. Frames, fields, and contrasts: New essays in semantic and lexical organization. Hillsdale, N.Y., 1992. pp. 21-74.

[2] V. Evans, M. Green, Cognitive Linguistics. An Introduction. Edinburgh: Edinburgh University Press, 2006. 830 p.

[3] Ch. Fillmore, Frame Semantics. Linguistics in the Morning Calm. Linguistic society of Korea: Selected papers from the SICOL. Seoul, 1982. pp. 111–137.

[4] R. W. Langacker, Foundations of Cognitive Grammar. Theoretical prerequisities. Stanford, 1987. V. 1

[5] Y. V. Meleshko, Problemy recomendatsiynykh system ta metody yikh rishennia, Systemy upravlinnia, navigatsii ta zvyazku, 2018. 4: 120 – 124. doi: https://doi.org/10.26906/ SUNZ.2018.4.120

[6] A.Musolff, Metaphor and Political Discourse: Analogical Reasoning in Debates about Europe. London: Palgrave, 2004. 224 p.

Linguistic Characteristics of Combat Post-Traumatic Stress Disorder in a Trauma Related Narrative: Computational Context-Aware Approach

Valeriia Didushok, Nina Khairova

National Technical University "Kharkiv Polytechnic Institute", 2, Kyrpychova str., 61002, Kharkiv, Ukraine

Abstract

These days, an increased prevalence of post-traumatic stress disorder (PTSD) and severe depression has been reported in populations exposed to war. This paper introduces using linguistic analysis of trauma narratives in the context of the study of post-traumatic stress disorder of combatants. As a subject of the analysis, posts of people who participated in combat, obtained from topic-related discussion boards were used. The approach utilizes vocabulary adaptation in NLP using the pre-trained language BERT model in addition to descriptive statistics obtained from text. The novelty of the research lies in the use of a context-sensitive model, while most of the existing research in this area is based on statistical models that use statistical inference to discover hidden patterns.

Keywords 1

PTSD, combat, content analysis, linguistic analysis, BERT, linguistic features, anomaly detection, context-aware models

1. Introduction

Modern history has brought to mankind a lot of appalling events that could not but have an impact on human perception of the image of the world, which, as it is understood by psychologists, is a reflection of the objective world in the human mental state, mediated by objective meanings and corresponding cognitive schemes, and amenable to conscious reflection, which, in turn, has an impact on the change in both society and language. Thus, traumatic events influence not only the mental state of people affected by the traumatic situation, but also their language. Through language, which is a means of conveying the subjective characteristics of personal experience, the "traumatized" person transmits his/her the raw experience.

When referring to traumatic experience, language can act as a means of how the survivor overcomes through the events, and provides an understanding of how the person experiences the consequences of the trauma: words, seen as psychological units that form the blocks of the narrative, can indicate how the traumatic events were encoded, which can help in identifying the difference in the clinical picture of patients and, accordingly, in the approach to their therapy. Thus, linguistic analysis can help to reveal subjective differences between groups of individuals who, faced with the same traumatic event, experience PTSD in different ways and, as a result, have different effects from the applied therapy [1].

Despite the fact that the study of language from the point of view of psychology has been actively studying since the beginning of the XX century, only relatively recently, thanks to the development of applied linguistics in the field of natural language processing using computer powers, it became possible to process and study linguistic characteristics on large amounts of data.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: 380689694704@pm.me (V. Didushok), khairova.nina@gmail.com (N. Khairova)
ORCID: 0000-0002-1307-2113 (V. Didushok), 0000-0002-9826-0286 (N. Khairova)

One of the outstanding works in this direction is Pennebaker's Linguistic Inquiry and Word Count (LIWC). Linguistic characteristics, according to Pennebaker, are elements of language that relate to both the form and the content of the text. Form - or style – refers to articles, prepositions, negations, and accents used in text, as well as the structure of the text is also important. Content is represented by the use of words which describe emotions, thoughts, actions, or sensations. According to Pennebaker and Tausczik, the words we use in everyday life reflect what we pay attention to, what we think about, what we try to avoid, how we feel, and how we organize and analyze our world [2].

As for the PTSD, different linguistic characteristics of the trauma from the point of view of its description are represented by the absence of emotional color, the impossibility of an accurate and structural narration of the traumatic event. The latter is due to the fact that a person suffering from PTSD is characterized by avoidance and numbness, which does not allow him/her to fully express the experience of trauma. Hence, the study of the linguistic characteristics of the texts of people who suffer from PTSD can play a key role in understanding the cognitive biases inherent in trauma and the potential to alleviate and improve the treatment of such individuals [3].

2. Proposed Method

In our study, we created a corpora of non-trauma narratives of combatants suffered from PTSD by parsing posts from https://www.myptsd.com/ discussion board. To identify combat-related posts, the "Military" prefix was used. The data was parsed with the help of Selenium Web Driver. Each topic was parsed to separate file. Each file was then manually reviewed in order to detect whether the discussion was personal (topic starter describes own feeling calling for help and answers) or general (topic starter asks to express other on their personal experience regarding specific part of life/emotions, e.g. worst moments of anger/nightmares). As a result of manual corpus revision, only those datasets that refer to trauma-related expressions and cover most of personal experiences were left.

The next step was conducting linguistic analysis of the text in order to identify candidate linguistic markers in the trauma that was guided by the Pennebaker's work. In particular, we examined the proportion of usage of pronouns as an index of self-immersion, descriptive words of negative emotions as an index of oppressed feelings, words related to cognitive processes as an index of cognitive processing, death- and religion-related words as index of mental mayhem. Obtained results provides evidence of strong correlation between language used by individuals suffering from PTSD while description of events leaded to trauma and clinical picture of PTSD, which leads to detect severity of the disorder. As a result of the conducted linguistic analysis, we were able to determine strong linguistic predictors of the PTSD symptoms which further may be used as a benchmark to define PTSD severity and stage.

Context-dependent classification on whether an individual has the PTSD is conducted with the help of fine-tuned BERT model. As input, BERT accepts two concatenated text segments separated by special tokens, the length of which corresponds to the specified maximum. The original model was pretrained on a huge dataset of untagged text. For fine-tuning our model, we used the previously created text corpus. Taking into account findings on the previous step, we studied the problem as textual anomaly detection that can be applied in the context of PTSD. In the semantic space, anomaly may reflect violations that are intentional or arising, signaling unusual behavior or phenomena. A change in the tone and vocabulary of an individual can be considered as a risk, and described linguistic features of the trauma are treated as context-specific signals. Our goal is to detect anomalies on input vectors that represent language units of the input textual data: characters, words, sentences, etc. As a result, we may be able to detect unusual behavior in the semantic space between the base and modified vectors.

3. Conclusion

As a result of our study, we found evidences that PTSD can be detected and predicted through linguistic characteristics that emerged specifically from trauma narration. Traumatic moments are

marked by depressed emotional states, feelings of panic and helplessness, which may be mingled with feelings of grief, guilt, or shame. Because the experiences of traumatic events remain deeply imprinted, the associated emotions remain equally present and leave their imprint on how the individual expresses - or does not express - their experiences of trauma.

4. References

[1] B. Busch, T. McNamara, Language and Trauma: An Introduction, Applied Linguistics, 2020. 41, 323–333. doi.org/10.1093/applin/amaa002

[2] J.W. Pennebaker, R.E. Booth, M.E. Francis, Linguistic Inquiry and Word Count: LIWC2015 – Operator's Manual, Austin, TX, 2015. URL: http://liwc.wpengine.com/wp-content/uploads/2015/11/LIWC2015_OperatorManual.pdf

[3] B. Kleim, A. B. Horn, R. Kraehenmann, M. R. Mehl, A. Ehlers, Early Linguistic Markers of Trauma-Specific Processing Predict Post-trauma Adjustment. Frontiers in Psychiatry 9, 2018. doi.org/10.3389/fpsyt.2018.00645

Information System for Educational Resources Management

Vitalii Sokoliuk and Viktor Hryhorovych

Lviv Polytechnic National University, S. Bandera Street, 12, Lviv, 79013, Ukraine

Abstract

Education is one of the best ways to form one's personality so he is able to think and act independently. It has always been given a great importance. Education systems are constantly changing in order to adapt to new conditions and get the most out of them.

Recently, self-education aimed at gaining knowledge related to cognitive interests or professional development has become increasingly popular. For this purpose, the relevant literature is studied, different lectures and research centers are attended, somebody`s own experiments and researches in the chosen field are conducted. Various methods on the Internet are rapidly developing. One of the most popular is learning on educational platforms.

Keywords 1

Education, studying, educational platform, self-education, online learning.

1. Introduction

Nowadays, more and more industries are moving online. Many stores are launching sites where you can easily order everything from appliances and clothing to food. Some companies transfer their employees to work from home, so as not to spend money on renting more space and not to spend time on the employee to get to the office. The same situation is happening now with training. More and more students refuse to study full-time and prefer online classes. This system allows students and teachers to manage their time more rationally. Also, do not forget about the huge savings that would go to living in a foreign city, food and other additional costs. The teacher has the opportunity to show a variety of articles, presentations and videos during the lesson, which was sometimes not possible due to lack of technical equipment in the classroom. At the moment, the field of online learning in Ukrainian universities is in its infancy, however, it is already beginning to instill in many people a love for online learning because of the benefits described above.

Another branch of online learning is self-learning on various educational platforms. This approach has several key advantages, due to which its popularity is growing from year to year. The first is the ability to study anytime, anywhere. This flexibility allows you to spend more time for your own benefit, because you can learn at any time. It can be a break from work, a trip on public transport or even a queue at the store. Obviously, for high-quality assimilation of the material you need to be focused and devote enough time, but the situations described above are ideal for repeating what has already been studied. This advantage allows you to combine training with other activities or work, as there are no time limits for training.

The second significant advantage is the financial side of the issue. In the case of traditional teaching methods, you need to spend a significant amount of money to get the necessary knowledge. Both free and paid educational materials are usually available on educational platforms, but even if you limit yourself to free ones, you will gain the necessary knowledge.

The next advantage is the ability to choose only what the learner needs. There is no need to study everything taught in universities if you do not need it.

It is also worth noting the relevance of educational materials. The information being studied will almost never be outdated and will meet the modern needs of the people who study it.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: vitalii.sokoliuk.mknit.2021@lpnu.ua (V. Sokoliuk), viktor.h.hryhorovych@lpnu.ua (V. Hryhorovych)
ORCID: 0000-0002-0703-126X (V. Sokoliuk), 0000-0002-5828-067X (V. Hryhorovych)

Teachers, in turn, have the opportunity to create teaching materials, which allows them to keep their knowledge in tune and earn extra money. They may not be limited to their students, but can teach anyone from all over the country.

This approach also has several significant drawbacks. First of all, not everyone can choose without mentors and mentors in the form of teachers he needs the educational materials due to insufficient knowledge of market needs and technologies that he needs to study. Self-study will also not give you a diploma that will officially certify your competence in a particular field. For many employers, a diploma is an important indicator. In contrast to the university, self-study in most cases does not have sufficient control over students and their acquired knowledge, which can affect the quality of knowledge. Also, a negative factor is the development of a sedentary lifestyle due to the lack of need to get to school. Reduced physical activity and lack of live communication during training can cause health problems, reduce motivation and interest in learning.

The teacher, in turn, cannot be sure that the amount of time he spent on the creation of educational materials will pay for itself. There is a risk that his work will not be very popular. This will mean reduced income or, at worst, no income at all.

2. Analysis of Literary and Other Sources

2.1. Description of the subject

Education is one of the best ways to form a person as a person who is able to think and act independently. From ancient times, education has been given great importance, because it was the driving force behind the development of peoples and nations. People with knowledge in a certain field have always been valued by society, because they could help in the development of this field, teach others and be an example to follow. The speed of educational development is also ensured by the ability of people to accumulate knowledge throughout life and pass it on to future generations.

People's education systems are constantly changing in order to adapt to new conditions and get the most out of them. In antiquity, it was believed that a citizen is worthy only of those activities that are not carried out for profit. Initially, military education was the most popular, but the Sophist movement ensured the spread of intellectual education, which included various subjects, such as public speaking, ethics, literature, and the exact sciences. Among them were elementary physics and mathematics [1].

The medieval education system was similar to the ancient one, but had several changes and improvements. The subjects studied were divided into two main groups. The first included logic, rhetoric and grammar. The second group includes astronomy, geometry, music and arithmetic. With the development of education, different types of schools and universities appeared [2].

In this period of civilization, each country develops its own approach to the education of its citizens. In general, all states have a similar structure, but each has its own characteristics and differences.

The first stage is often preschool education, where the child gets acquainted with the basics, learns to communicate. Such education is provided by such types of institutions as: kindergartens, orphanages, nurseries, etc.

The second stage is general secondary education. It is at this stage that a person acquires all the necessary skills for life and gets acquainted with the world and its components in more detail. For these purposes, various schools, lyceums and gymnasiums are created. Every student of these educational institutions must study all the subjects that are included in the program, regardless of his wishes. However, in some educational institutions, specialized classes are available, with a choice of disciplines desired by the student.

The next step may be professional technical or higher education. At this stage, a person chooses the direction in which he wants to develop and learns what is related to his industry. Educational institutions such as technical schools, colleges, schools, institutes, academies, universities and others are used for education. This area of education is less popular, as statistics show. 76% of school graduates have decided to enter a higher education institution [3]. Most people are limited to this level of education, because it is enough to master a particular field of science, which allows you to work in

your chosen specialty. However, if you want to deepen your knowledge, understand the subject thoroughly and conduct a variety of research, you can choose graduate and doctoral studies.

Self-education is a condition for human development in career and personality. It is mainly aimed at gaining knowledge related to human interest and cognitive interests or professional development. For these purposes, the study of relevant literature is usually used, which can be presented in the form of paper or electronic sources. Also, very popular are visiting various lectures and research centers, conducting your own experiments and research in your chosen field.

With each passing day, a variety of Internet-related tools are gaining in popularity. This is due to the global spread of the network, easier access to it and ease of learning compared to other available methods of self-education. There are many information resources and special educational systems online that help make the process of learning new material easier than ever.

Educational platform - a system that allows access to a variety of educational materials. These platforms collect a large amount of information on various topics and provide a convenient environment for its use. There are different types of data availability systems. They can be both completely free and paid. Paid systems usually have more features and capabilities for better learning. However, the vast majority of systems include both free and paid components. These components can be additional functions or tutorials.

2.2. Analysis of existing software solutions

To best determine the requirements of the created system, a search and analysis of existing software solutions on the market was conducted. The results will help to better understand the advantages and disadvantages of the developed system, will help to identify additional requirements and will improve the system as a whole.

5 systems were selected that have a similar purpose or their functionality is similar to the developed system. The functionality of analogs should include several of the following functions: creating a course / article, the ability to view training materials, purchase training materials, check the mastered material. Below are the main competitors and their descriptions.

Highlight the main characteristics of these systems and create a comparison table.

Table 1

Comparative characteristics of analogues

Platform name	Access to courses	Ability to create courses	Issuance of certificates	Course content	Availability of a mobile application
Prometheus	registered user	-	to all	video lectures, tests	iOS, Android
EdEra	registered user	-	to all	video lectures, tests	-
Coursera	registered user	-	for paid	video lectures, homework, tests	iOS, Android
Moodle	by invitation	+	no	video lectures, articles, tests	iOS, Android
Udemy	registered user	+	for paid	video lectures	iOS, Android

According to the data in Table 1, we see that all systems have access to courses for registered users except Moodle. In this case, you need the author to add or invite you to his course. You can only create your own learning materials using platforms such as Moodle or Udemy.

Most platforms issue a certificate upon successful completion of the course. The only exception is Moodle. The rules for issuing certificates in systems are different. Coursera and Udemy issue them only after the completion of paid courses, while Prometheus and EdEra for all.

Course content is different for each system. In each of the platforms the main content of the course can be in the form of videos. Moodle allows you to place basic information in the form of articles.

Test students' knowledge in the form of tests is conducted on all platforms, except Udemy. For platforms such as Prometheus, EdEra, Moodle and Coursera, a variety of tests are an important element of course content. In addition to tests, this platform also has homework assignments, which are writing essays or creating various projects. This is an additional incentive for students to learn.

Each platform has its counterparts in the form of mobile applications for operating systems based on iOS and Android. The only exception is Moodle.

It is also worth noting that the main language of instruction is Ukrainian only on Prometheus and EdEra. In the Moodle system, everything depends on the course, and Coursera and Udemy have almost no Ukrainian-language content.

2.3. System requirements

For further development of the system, it is necessary to determine its main characteristics, functions and requirements.

The main requirement of this bachelor's thesis is to create a system for organizing work with educational materials. It must have a sufficient number of functions to implement all the tasks, be easy to use, have a nice interface and ensure the security of user data. The characteristics of the system must be competitive and not inferior to competitors.

The main functional requirements of the system are:

- the ability to create training materials;
- search for training materials by rating and other parameters;
- the ability to create a separate page with selected training materials;
- ability to use training materials;
- evaluation of training materials;
- ability to submit and test assignments;
- functionality for system administration.

Access to features such as search training materials and the use of training materials should be for all users. All other functionality except system administration must be available to registered and authorized users. The administration functionality should cover all the necessary functionality to ensure system upgrade and security, prevent problematic situations and be hidden from ordinary users.

Email the link with a unique hash and require the user to follow that link. This will confirm its uniqueness. Passwords should not be stored in the database in a pure form, but should be encrypted.
For ease of use of the system, navigation should be simple and intuitive. The main sections should be in the top menu of the site, which should be available on almost every page of the site.

Based on the described requirements, we will set the task. They can be created by any registered user. The popularity of training materials will increase the rating for their author, which will allow him to earn on this platform, as he will be able to create paid training materials. Creating paid learning materials is available only after creating free ones that have gained a certain amount of likes from readers.

Any registered user will be able to create their own training materials. They can look like a regular article or a whole course. The content can be different: from ordinary textual information to images and videos. The author will deal with a convenient editor, which will present the material in the most diverse and enjoyable way to read.

The reader will be able to view both free and paid educational materials, in which the selected topic will be disclosed in more detail and will be able to do additional tasks and answer some key questions from the author, which will assess the quality of the reader. The author will evaluate the answers of readers and provide feedback on the completed additional task.

The administrator will be able to add new sections for training materials in order to increase the number of topics presented in the system. It will also be possible to remove training materials if they violate certain rules of the system or are stolen. If the user violates the rules, it can be blocked. If the

user has purchased paid training materials and started their illegal distribution, the administrator has the right to take away access to these materials.

3. System Analysis and Justification of the Problem

During the analysis of each complex system, a lot of complex problems can arise, for the solution of which various procedures and methods are used. It is system analysis that solves these problems.

Systems analysis is used to solve problems related to the uncertainty of the choice of decision to be made. It provides an opportunity to identify and analyze all options and alternative ways to solve the problem and choose the best one [4].

The goal tree is a structured relationship of goals from a common, basic goal to small sub-goals. The main idea of building a goal tree is decomposition, which means breaking each goal into several smaller ones. The goal tree does not have a maximum number of levels. The goal of the zero level is the general goal, and the last - the smallest sub-goal. A certain goal cannot be considered fulfilled until all its sub-goals have been fulfilled [5].

When constructing a tree of goals, it is necessary to pay maximum attention to the creation of the goals themselves. Each goal must meet several criteria. The first is dimensionality. For each goal there must be a quantity or qualitative parameters that will indicate its achievement. The next criterion is achievability, which indicates the possibility of successful completion of the goal. Next comes time constraints. The goal must also be relevant and clearly defined.

Since the goal tree is a hierarchically structured system of all goals and objectives of the system, it can be considered one of the most convenient and effective methods of task planning.

To specify the operation, we choose the context diagram IDEF0, the scope of which is extremely large, as it allows you to describe any system and facilitates the construction and analysis of even the most complex systems. It can be used both for the analysis of existing systems in order to improve them and for systems that are just under development. Using the IDEF0 methodology, the system can be represented as functional blocks that are interconnected.

To build this chart, you need to create a rectangle that is a function or process and arrows to each of the sides that have different values. This is the most general description of the program, which is called a context chart. There are 4 groups of arrows:

• Input arrows to the left (Input). They are the values that will be processed in a certain way and on the basis of which the output will be obtained.

• Input arrows to the lower side (Mechanism). They are what make this function possible.

• Input arrows from the top (Control). This is what makes this process manageable. This can be a variety of instructions, instructions, documents or other control processes.

• Output arrows on the right. These arrows are the result of this process or function.

After describing the main function, we decompose it. It consists in the fact that this function is divided into several components, which are called functions or robots. Usually there is a division into two works to six. These works are depicted on a new sheet. The location and relationships between these works must conform to the logic of the function described.

Arrows are used to transfer objects or data between blocks, which can merge or branch as needed. If the arrows merge, the newly created arrow contains all the data and objects from the arrows that are included in it. If the arrow is branched, then the branched part can contain all the data and objects of the parent arrow, and only some part of them. To ensure a clear understanding of the scheme, the arrows between the blocks are signed [6].

Depending on the needs, each block is decomposed to the desired level of detail of the description, or until it loses its meaning.

Process hierarchy modeling is a powerful tool that can be used to convert software specifications into basic functions that must be in the system. This will provide better tracking of software requirements. Using the functional hierarchy, you can measure the complexity of a software product in three ways:

• number of decomposition levels;

• width of each decomposition level;

• number of basic functions from which the system structure is created [7]. The number of decomposition levels for each function may differ, depending on its complexity. A tree-like model is used to represent the hierarchy of system processes.

4. Presenting the Main Material

4.1. Structure of the database

An extremely important task about the developed systems is the design of the database. This is the basis for any system where there is data handling.

13 tables have been developed for data storage in the developed system. Each of the tables was normalized to the third normal form. This means that the tables do not contain multiple attributes and there is only one key in each of the tables (it can be compiled).

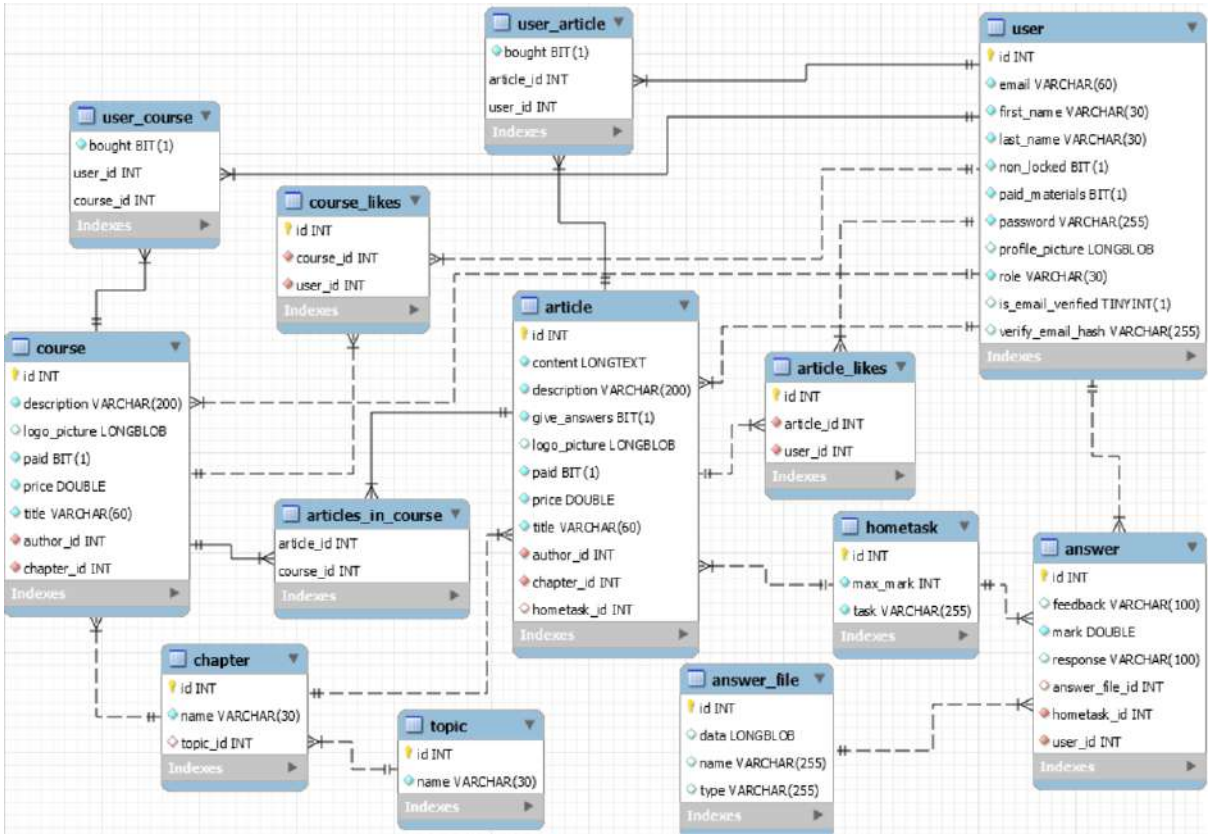


Figure 1: Structure of the database

4.2. Internal structure of the program

The program is based on DispatcherServlet, which processes all HTTP requests from the system user. Figure 2 shows the structure of the program.

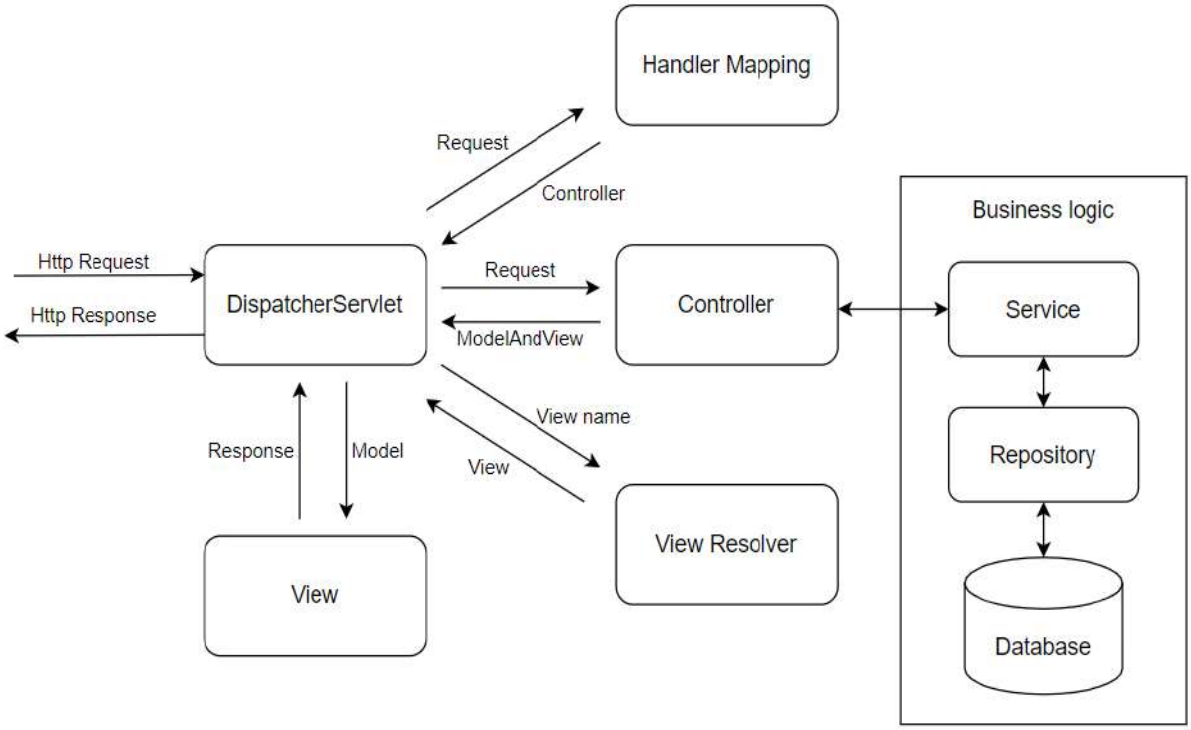


Figure 2: Internal structure of the program

After receiving an HTTP request, DispatcherServlet calls Handler Mapping, which indicates which controller should be called. The specified controller is then called, which processes the request and returns the Model and the View name. During the request processing, the logic written for the specified controller is executed. After receiving the necessary data, DispatcherServlet uses the View Resolver to determine which View to use, accesses it by passing the Model object. The response is then sent, which is processed by the browser.

5. Conclusions

The result of this bachelor's qualification work was an implemented information system for managing educational resources. Several tasks were performed for this purpose.

First of all, a detailed analysis of the subject area was conducted, after which the existing systems were identified and studied, due to which it was possible to identify and improve the requirements of the developed system.

Next, a comprehensive review of the designed system was conducted. To decompose the structure of the system, the goal tree method was used, alternative options for system development were described, and the path of system development in the form of a website was chosen. The system processes were modeled using the IDEF0 context diagram and a process hierarchy model was created. These steps helped to describe and analyze the developed system.

Next, the tools and tools for developing the created system were selected, the main ones being Java, several modules of the Spring, Maven, Bootstrap framework and the CKEditor library. The structure of the database was developed and described, after which it was implemented. The internal structure of the system, its functions were also described and the final result was presented.

To ensure the feasibility of the system, a number of calculations were made, according to which the total cost is 260455.62 UAH, profitability will be 42%, the cost of the software will be 359651.38 UAH, and net profit will be 22357, 7 UAH.

6. References

[1] Nizhnikov, S. A., Philosophy: a course of lectures: a textbook for universities, Kharkiv, Publishing House "Exam", 2007.

[2] Zemerova Tetyana Yuriyivna, World History. A practical guide, Kharkiv: Spivak T. K., 2009.

[3] pon.org.ua, 76% of school graduates applied to the university, 2018. URL: https://pon.org.ua/novyny/6485-76-vipusknikv-shkl-podali-zayavi-do-vnz.html

[4] Arshinova O. I., System analysis, Kharkiv, National Aviation University, 2008.

[5] Varenko V. M., System analysis of information processes, Kyiv, University "Ukraine", 2013.

[6] Soroka K. O., Fundamentals of systems theory and systems analysis, Kharkiv National University of Municipal Economy, 2004.

[7] Richard F. Schmidt, Software Engineering 1st Edition Architecture-driven Software Development, 2013.

Construction of Information System for Finding Tickets for Different Types of Transport

Vasyl Malion, Viktor Hryhorovych

Lviv Polytechnic National University, S. Bandera street, 12, Lviv, 79013, Ukraine

Abstract

The problem of choosing a way to buy a ticket is relevant in the world. Therefore, buyers often prefer unprofitable and problematic alternatives. In order to choose the best way to buy a ticket, all the alternatives to the usual booking offices are analyzed. Based on these data, a comparison of these methods is made at a price, time spent, range of tickets, convenience, reliability and provision of additional services.

Keywords 1

Transport, route, ticket search, transportation, aggregator.

1. Introduction

In today's world, passenger traffic is becoming increasingly important. This is due to the growing needs of people and the development of infrastructure. Therefore, the issue of ticket sales is relevant.

There are currently several ways to purchase a ticket. First of all, it is a traditional way to buy a ticket at the booking office or airport terminals. However, over time, it becomes less relevant. It is being replaced by new ways to purchase a ticket: buying a ticket on the carrier's website and using special aggregators that specialize in finding tickets. To buy a plane, bus or train ticket, it is not necessary to leave the house, look for a ticket office and stand in line. Just go to the website, choose the best offer and place an order.

First of all, this is a wide range of offers for different modes of transport. The client can get acquainted with the details of each of the proposals in seconds. You can also get acquainted with the companies that provide transportation. It is possible to change the conditions of the trip (for example, increase the number of luggage places).

As mentioned earlier, ticket search aggregators provide a wide range of offers, which allows you to compare their costs and save money. Also often in such services, the user can receive special promotional offers and the opportunity to use the bonus system.

This article compares the main methods of buying a ticket and describes the information system for selling tickets for different modes of transport. The presented system will allow the user to buy a ticket for the following types of transport: bus, train and plane. The user will be able to select the route, flight, date of departure and arrival, and purchase the appropriate ticket if any. In this case, he will be able to get the details and conditions of the trip or flight. It will also be possible to add luggage to your flight, which will be reflected in the price. The user will also be able to return the ticket and refund it.

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: vasyl.malon.mitup.2021@lpnu.ua (V. Malion); viktor.h.hryhorovych@lpnu.ua (V. Hryhorovych)
ORCID: 0000-0003-0798-3313 (V. Malion); 0000-0002-5828-067X (V. Hryhorovych)


2. Analysis of Literary and Other Sources

Before buying a ticket, a person pursues certain interests: someone wants to buy a cheap ticket, someone wants to make the purchase quick and comfortable, and someone wants to get a ticket for the same money, which has better conditions of transportation. He has several ways to buy a ticket:

- at the train station or the airport terminal;
- on the Internet on the carrier's website;
- using third-party ticket search and purchase services [1].

2.1. Description of the subject

It remains traditional to go to the train station or airport terminal and buy a ticket there. However, this method does not stand up to criticism when you see how you can buy a ticket using the Internet. Consider the main disadvantages of this method: waste of time.

Opening a browser web page is faster than arriving at a train station or airport terminal. In addition, the customer can spend a certain amount of money to arrive at the specified address of the station or airport terminal.

This happens when the user lives far from the specified address or is in a village where they do not exist:

- limited supply. The transportation database, which is available for sale at the station or the airport terminal, is limited to the offers of several carriers. This does not completely satisfy the customer who wants to have a large selection of tickets and choose the best option;
- prices. As mentioned above, limited offers may result in the purchase of a more expensive ticket. Also, on the services that are located on the Internet, you can at least every minute, without leaving home, check the price changes of a ticket. You can also automatically track price changes with one of these services without any effort [2];
- convenience. In addition to the fact that we need to spend time travelling to the ticket office or airport terminal, there is a limitation in the choice of tickets at the box office. Using specially designed services, the user can set the parameters of their trip. Therefore, he will not have to review flights that do not suit or interest him. Also, the sorting function will simplify the search for the desired ticket, giving the search result in the desired order;
- additional Services. When we travel, we have the following essentials: insurance, finding a hotel, car rental. Using the traditional method of buying a ticket, we need to take care of all these things ourselves. Preferably, you need to contact other companies that do this. However, using special services, you can avoid this process because the system itself will offer you a range of offers with services that are available in the service;
- reliability. Using the traditional way of buying a ticket, the customer can forget, lose or damage it. But if you buy a ticket on the site, you only need to have a passport and a smartphone or other means to confirm the purchase of a ticket.

Once we have demonstrated that buying tickets at airport ticket offices or terminals is outdated and inefficient, we will consider the best methods to get the desired ticket.

To use the method of purchasing a ticket from the carrier, you only need to go to the website of the company that provides transportation.

Using this simple method, we get a number of advantages over the traditional way of buying a ticket:

- convenience. Purchase a ticket in a few minutes "in one click". Go to the website and choose the conditions of transportation. After that we get a list of tickets that we can buy. Choose the right ticket, buy and enjoy the trip;
- bonus system and discounts. If we use the services of one carrier company for a long time, we can get discounts on subsequent shipments. This is achieved through the accumulation of bonus points and participation in a special promotional campaign;
- additional Services. By purchasing a ticket on the company's website, we will be able to choose the conditions of transportation we need. Also, the company often offers its customers

to take out insurance and book a hotel, which makes our preparation for the trip more comfortable;

- the opportunity to buy a ticket not only for yourself. It is also a significant factor when buying tickets, because very often customers do not travel alone. To buy a ticket for someone you need to know only his passport data, and if you take a child on a trip - to have a birth certificate [3].

Although this method of buying a ticket has a number of advantages, but it also has certain disadvantages:

- limited range of tickets. It is clear that the carrier provides only its own range of tickets. This limits the client, because at the moment when the client will need to make a flight or a trip, the companies may not carry out transportation;
- prices. By buying only on the website of one of the carriers, the customer may find that other carriers sell the same ticket and with the same conditions, but cheaper;
- getting the best transportation condition for the same cost. Again, competitors may provide the customer with better conditions of carriage than those offered by the carrier. For example, you do not need to pay extra for luggage or provide free meals during transportation.

The following way to find and buy a ticket minimizes the shortcomings of previous representatives. In order to comfortably and profitably choose a ticket, you need to use aggregator sites.

An aggregator is a website or program that collects and groups information from various sources [4]. That is, in our case, a ticket search web aggregator is a website that finds tickets from different carriers that become available to users. Aggregators can find tickets using a special search algorithm on the Internet. However, in some cases, tickets are provided by carriers. After that, the aggregator database contains a large number of tickets received from various carriers.

Carrier sites provide the opportunity to purchase a ticket directly to yourself. However, they are not limited to this, but use the services of aggregator sites. From a marketing standpoint, this is a very lucrative initiative. In this way, the carrier advertises and promotes itself in the market. This leads to an increase in the number of customers and, consequently, profits. All major carriers cooperate with aggregator sites.

Being a partner of such an aggregator site is not free. The carrier company shares the profits for the sale of their tickets. Most often, she pays a percentage of the ticket price or a fixed amount for its sale. If the aggregator does not have the opportunity to buy a ticket, the aggregator leaves a link to the carrier's website, where the user can buy a ticket. In this case, the fee is charged for the number of links.

Here are the advantages of aggregators and clearly demonstrate how they solve the disadvantages of previous ways of buying tickets:

- a wide range of ticket choices. Given that the tickets come from different carriers, the range of these tickets is increasing. The user has the opportunity to make the best choice;
- price and conditions. This is a disadvantage of the option of buying a ticket on the carrier's website, and the aggregator turns this disadvantage to its advantage. The aggregator issues search results for tickets sold by different carriers. Thus, the user will be able to choose the cheapest and most profitable ticket;
- combination of transportation. If the trip includes one or more transfers, the aggregator will ask the user to select tickets so that the waiting time between shipments was minimal;
- choice of transport. Many aggregators offer the opportunity to buy tickets for several modes of transport. If the user for some reason (price or date of transportation) is not suitable for transportation by one of the modes of transport, the aggregator will offer him an alternative - transportation by another mode of transport [5].

Also, aggregators can have shortcomings, however, they are available only at some representatives of this niche:

- price mismatch. When using a ticket sales aggregator, you sometimes notice that the prices in the aggregator and on the carrier's website are different. The fact is that some aggregators increase the price, citing the fact that it is a fee for the use of the aggregator;

- problems with returning tickets. Many users say that sometimes it is difficult or impossible to return a ticket to the aggregator. Another popular option is to return to the customer only part of the money he spent on the ticket. Often this return is a very small part of the ticket price.

Ticket search aggregators can be divided into 2 types:

- those where you can buy a ticket. In this case, the user can buy a ticket on the website or in the aggregator application. Having made such a purchase, the user will be able to get all the information about the ticket and the ticket itself, without contacting the carrier;
- those in which you can not buy a ticket. In this case, the aggregator only presents the options of tickets that can be purchased and their details. However, you need to buy a ticket on the carrier's website. The aggregator redirects the user to where he will go next, he will interact with the carrier's site. Such aggregators are often called metasearch systems.

Metasearch systems are systems that do not search their databases but obtain the necessary information through a search on the Internet. However, no metasearch engine can capture all the information available on the Internet. Having received information from different sources, the system can process and group it and later returns them as a single list of results [6].

The main advantages of metasearch systems:

- you can make only one query and get results from multiple sources. This saves our time and resources;
- analysis of one list of results is faster and more efficient than gradual analysis of results from several sources.

The main disadvantages:

- lack of own database. As a result, there is no index base, so we will not be able to add our URLs to which we would like to search;
- a modest set of metasearch settings because each search engine has its own set of these settings and they will vary depending on the representative.

2.2. Analysis of existing software solutions

Another important step is to compare the future information system with analogues. This is done in order to create the best possible system. Therefore, the following factors should be considered:

- competition in the market. If competitors have significant market positions, it will be extremely difficult to compete with them. It will take a lot of effort to develop, offer new functionality and spend a significant amount of money advertising the product;
- system shortcomings. System developers need to take into account all the major shortcomings of competitors. Then you need to try to find solutions to these shortcomings and turn them into your strengths. This can be key for the customer in choosing the system he will use;
- system functionality. System developers need to analyze the basic functionality of the system. To occupy a leading position in the market it is necessary to add to the system functionality that is not yet represented in the market.

Therefore, we will analyze the ready-made solutions that are presented on the market and show that on Table 1.

2.3. System requirements

Decisions need to be made when creating an information system. This can be a choice of the structure of the system, how the system will be developed. You should also define the principles of creating system functions and take care of software development.

Decisions can be made based on variables that are already known and will not change. However, it is often necessary to choose solutions based on variables and data that change or are not known at all. In such cases, making the right choice becomes more difficult, so it is accepted based on the requirements of the information system. They need to be created in advance - this is one of the first steps in creating a system [7].

The main requirements for the system are divided into categories:

- requirements in general;
- functional requirements;
- requirements for architecture;
- requirements for security and storage of information.

Table 1
Comparative characteristics of analogues

Platform name	Mobile application	Purchase without going to the carrier's website	Insurance	Transport	Apartment
Tickets.ua	iOS, Android	+	+	Plane, bus, train, car rental	+
Kiwi	iOS, Android	+	+	Plane, bus, train, car rental	+
Aviasales	iOS, Android	–	If provided by the service of the carrier's company	Plane	–
Momondo	iOS, Android	–	If provided by the service of the carrier's company	Plane, bus	+

2.4. System requirements

Decisions need to be made when creating an information system. This can be a choice of the structure of the system, how the system will be developed. You should also define the principles of creating system functions and take care of software development.

Decisions can be made based on variables that are already known and will not change. However, it is often necessary to choose solutions based on variables and data that change or are not known at all. In such cases, making the right choice becomes more difficult, so it is accepted based on the requirements of the information system. They need to be created in advance - this is one of the first steps in creating a system [7].

The main requirements for the system are divided into categories:

- requirements in general;
- functional requirements;
- requirements for architecture;
- requirements for security and storage of information.

2.4.1. Requirements for the system as a whole

The developed information system should provide the user with the ability to search and book tickets for different modes of transport, using data received from different carriers.

The following functionality should be available when using this system:

- user registration. The user enters the data required for registration. He will then have access to his profile;
- user authorization. The user enters data, then gets access to his profile;

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

- access to the personal account. After successful registration, the user gets access to his account, where he can view his data provided during registration and the list of booked tickets if any;
- ticket search. The user will be able to enter the details of transportation and will be able to get a list of tickets provided by carriers;
- ticket booking. The user has the opportunity to book the ticket he wants to receive;
- getting help. If the user has difficulties or problems at one stage, he should be able to get help from the administrator.

The following requirements must also be met during development:

- system design must meet all standards of modern design;
- the system must provide an adaptive design, allowing the system to be used from different devices;
- writing code using new technologies, programming patterns and following the design code.

2.4.2. Functional Requirements

Functional requirements are the functionality that the system must provide so that the user can conveniently and smoothly perform tasks in it [8]. The system will consist of several subsystems, each of which will be responsible for a separate functionality. The system will contain the following subsystems:

- user login. User authorization and authentication functions will be implemented;
- ticket search. The function of searching for tickets according to the parameters specified by the user will be implemented;
- selection of ticket parameters. A function will be implemented that will allow the user to specify the conditions of transportation, which will simplify the search for the desired ticket;
- ticket booking. The function of booking and paying for a ticket directly in the system will be implemented;
- system administration. The functions of access to the system and its management will be implemented.

2.4.3. Architectural requirements

The system must have a three-tier architecture and contain the following components, which are implemented at the appropriate level:

- database. The component is designed to store information;
- web server. The component is designed to process and transmit information;
- customer part. The component is designed so that the user can interact with the system.

2.4.4. Requirements for protection and storage of information

Information protection is several methods that aim to create confidentiality, accessibility and integrity of information [9]. The system must store user data in a secure database using information encryption libraries. First of all, you need to encrypt the password and then save it to the database. The system must also perform user authorization and authentication.

3. System analysis

A system is understood as a set of objects that interact with each other. At the same time, they must form a whole, ie the system must solve a specific problem. The system must also contain the goal to which it must move in the process of solving the problem. However, sometimes designing and developing a system can be quite difficult, and the relationships between processes are difficult to

determine [10]. System analysis is used to improve system development. It helps the system to determine the relationships between the set of objects that are part of this system [11].

3.1. Goal tree

The goal tree is a graphically displayed project goal in a hierarchical sequence. It is considered to be one of the most effective methods of task planning [12]. The goal tree is implemented by breaking down some main goal into sub-goals, which helps to better evaluate and distribute the project objectives.

Figure 1 shows the goal tree of the ticket search information system for different types of transport. The main goal contains 5 sub-goals, each of which we will consider in detail.

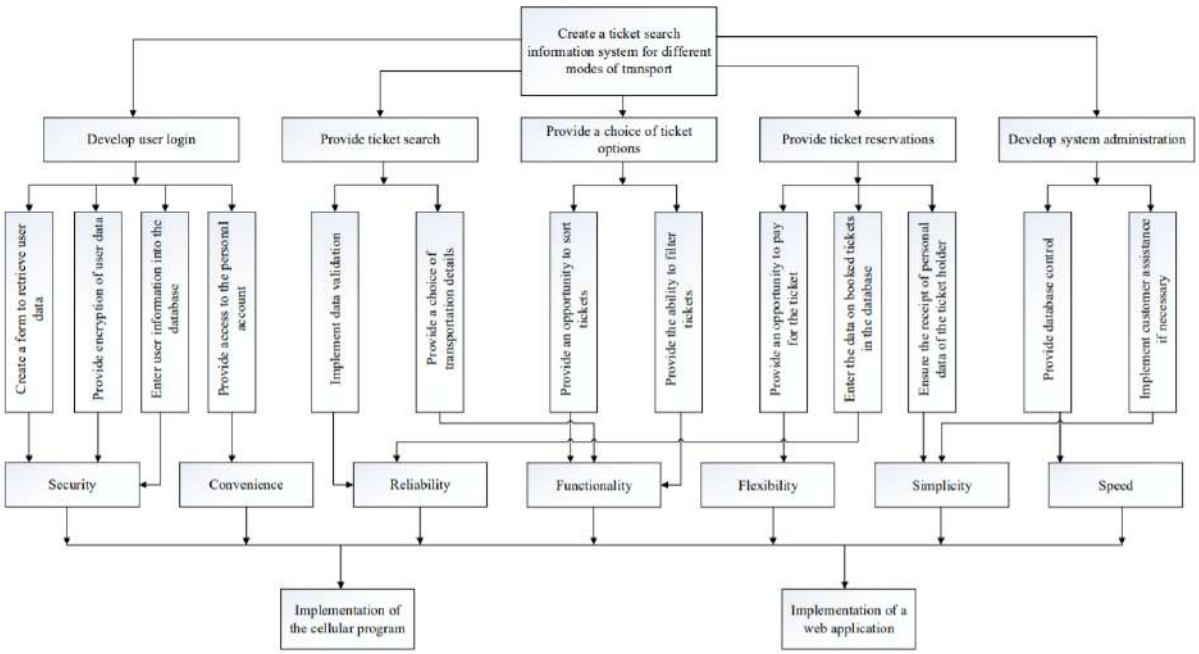


Figure 1: Goal tree of the ticket search information system for different types of transport

3.2. Data Flow Diagram

The main purpose of creating a Data Flow Diagram (DFD) is to depict the movement of information and documents. In this case, the information can be processed and modified. After implementing such a diagram, it becomes quite simple to understand all information flows, distribute them among all processes and save them in data warehouses, if necessary. Figure 2 shows the Data Flow Diagram of the ticket search information system for different types of transport.

4. Presenting the Main Material

Before developing an information system, it is important to choose the means of its creation. If you choose the wrong means of solving the problem, the development process may take longer, be inefficient and not optimized. Also, when choosing these tools, keep in mind that they are supported by different versions of different browsers.

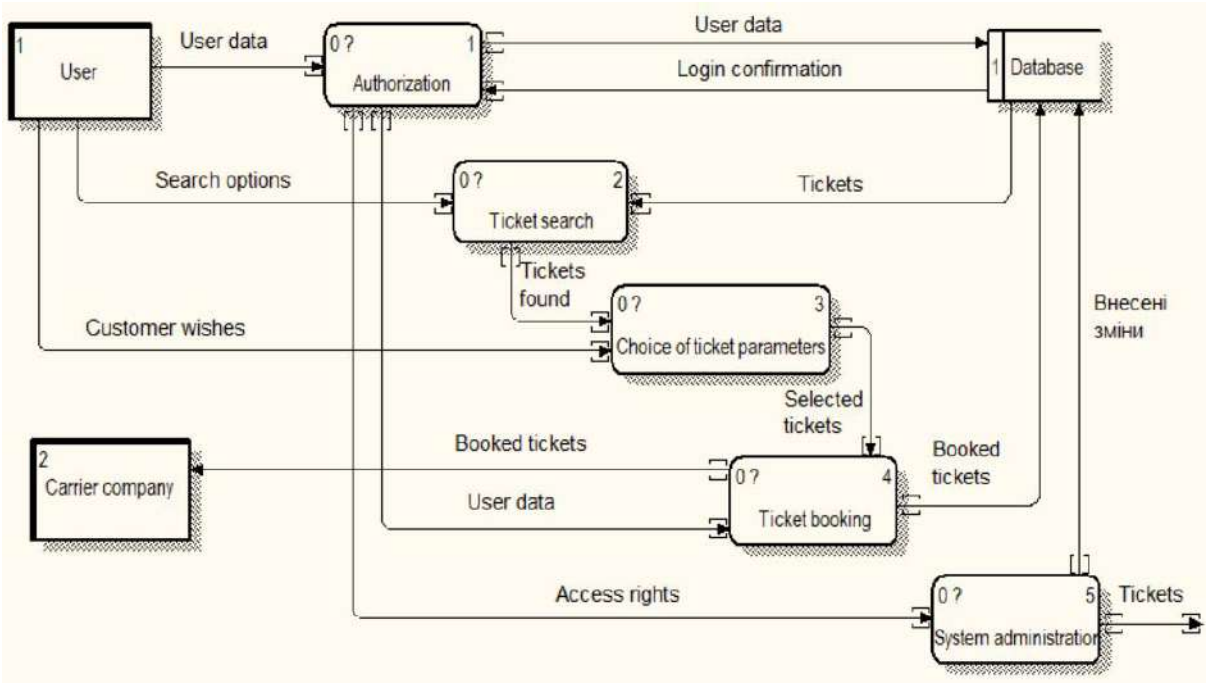


Figure 2: Data Flow Diagram of the ticket search information system for different types of transport

4.1. JavaScript

JavaScript is currently one of the most popular programming languages. It works well with HTML and CSS, complementing structural elements and design dynamics. JavaScript is often used to create websites, but it is not limited to this area but expands every year for other tasks [13]. At this point, you can use JavaScript to implement the following elements:

- development of applications for mobile devices. The React Native framework from Facebook has been created for JavaScript. You can use it to create applications based on Android and IOS;
- development of programs for personal computers. JavaScript has already developed programs such as Microsoft Office and programs created by Adobe;
- server development. Node.js is used to implement the server part, which also has many packages and libraries for better development;
- programs for maintenance of household appliances.

The main advantages of JavaScript:

- support for all new versions of browsers. That is, JavaScript will run in any popular browser;
- speed and efficiency. With Google's V8 engine, JavaScript can quickly and efficiently process all files written using this language;
- ease of use. To write and run code written in JavaScript, you do not need to download special development environments, just write the code in a notebook.

4.2. React

React is a library designed for the JavaScript language. It helps to build a user interface on a web page. The library was developed in 2013 by Facebook. Most often used to develop one-page websites [14].

React is used through its optimization. It is very fast due to the implementation of a virtual DOM (Document Object Model). A virtual DOM is a copy of a regular DOM. When a user performs certain actions on a web page, thereby changing the DOM, React first works with the virtual DOM. It reviews which items have changed and updates only them, not the entire web page. So React works pretty fast. For this, it is called jet.

Components are developed using React. The web page is conventionally divided into components. Then they are put together, which forms a full-fledged page of the website.

4.3. Node.js

Node.js was created in 2009 by Ryan Dahl. This is a platform for running programs created in the JaavScript language. Node.js works on the basis of the V8 standard used in modern web browsers [15].

Also, when working with Node.js, npm is used - a manager for managing packages that are installed to work on the project. It can install, uninstall and update modules. This package manager is used only for Windows and has an alternative to the macOS operating system in the form of the same manager - yarn.

4.4. MongoDB

MongoDB is a document-oriented database management system. It was developed in 2009. MongoDB does not require a database schema and tables for each entity. She is a representative of NoSQL systems. MongoDB presents data as JSON files. Most often used in web development. Combines well with Node.js, Express, React or Angular [16].

MongoDB can be used in two ways:

- installation on a computer. MongoDB is downloaded from the official website and then installed on a personal computer. After installing MongoDB, you need to open the mongod.exe file to run it. This can be done manually or using the command line;
- use of MongoDB Atlas. It is a database service that stores its information in the clouds. Data is hosted via AWS, Google Cloud and Azure. Also, this service allows you to automate the resources used in the database and load the database.

5. Discussion and Analysis of the Obtained Results

The created software is based on the client-server architecture and contains the relevant parts:

- server. Created for data processing and transmission. It is mainly intended for interaction with the database, receiving data from the user, their processing, the transmission of relevant data upon successful processing. Also authorizes and authenticates the user to the system, provides access to system administration;
- customer. This part of the software is responsible for the user's interaction with the system. It mainly provides tools for retrieving data from the client and mechanisms for transferring this data to the server. Also implements data reception from a server stored in the database.

5.1. Analysis of database structure

Incoming data:

- tickets from different carriers;
- user data for registration and login;
- conditions of transportation that the user wants to get, based on which the search for tickets is carried out;
- personal data of passengers for booking tickets.

Source data: this can be any information that the user received while using the developed software product. For example, a list of tickets that meet user search criteria, or information about carriers.

A non-relational database was created using MongoDB to manipulate the software data. Figure 3 shows a schematic diagram of the database for the created software product.

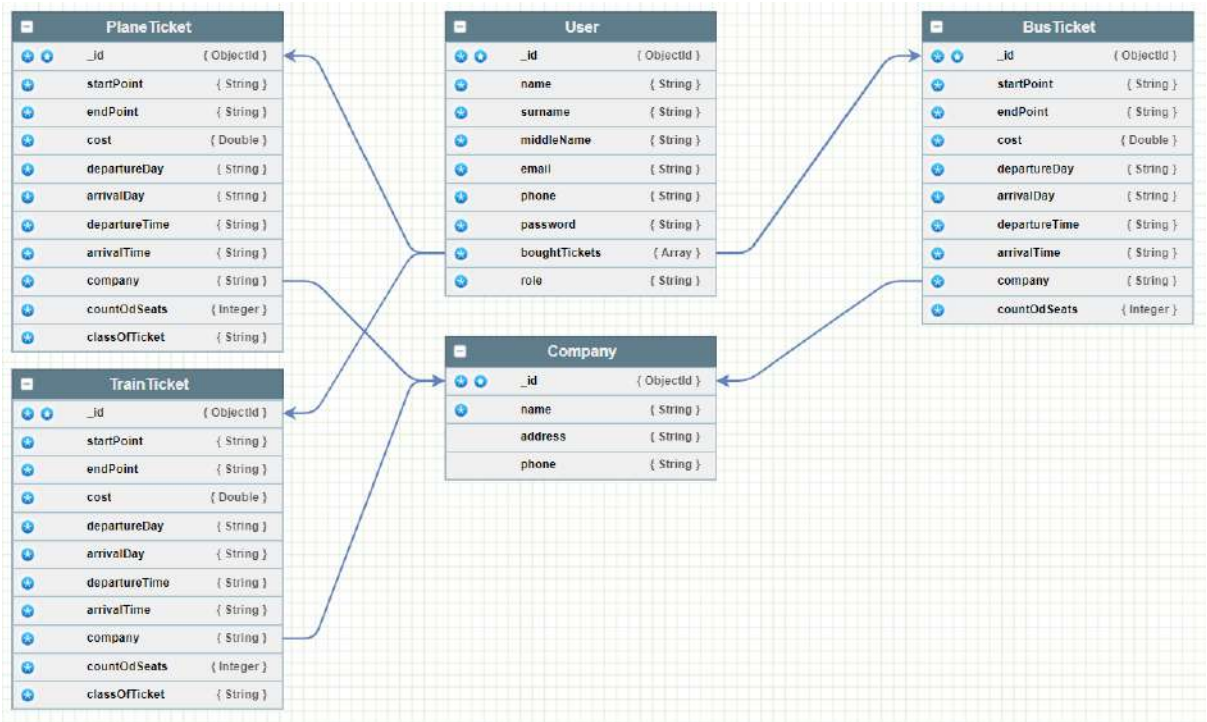


Figure 3: Schema of data base of the ticket search information system for different types of transport

5.2. Program structure

Opportunities provided by the created software product:

- user registration. The system provides a user-friendly interface for entering user data, which is necessary for successful registration. It will also notify the user if the data entered is incorrect and prompt the user to repeat the procedure;
- user authorization. The system provides a user-friendly interface through which the authorization process is carried out. It checks the entered data and, if the data is correct, it provides access to the personal account. Otherwise, the system notifies the user of the failure of the authorization;
- ticket search. The user has the opportunity to set the initial parameters of the ticket search. Based on this data, the system shows the user tickets that meet the search parameters. The system also notifies the user when no ticket is found;
- sorting and filtering tickets. After searching, the system can show many tickets. To save time and get rid of unnecessary viewing of unnecessary tickets, the created software product provides sorting and filtering functions, which facilitates and speeds up the search for the required ticket;
- ticket booking. Once the desired ticket is found, the user can make a reservation. To do this, the user is provided with forms where you need to fill in information about all passengers who will be participants in the carriage;
- ticket payment. After successfully filling in the personal data of passengers, the user can go to the ticket payment page;
- use of the personal account. After authorization, the user is given access to the personal account, where he can see his data and the list of tickets he has booked;
- system administration. The system also provides the ability to manually administer the system. To do this, you need to log in as an administrator.

COLINS'2021, Volume II: Workshop. Kharkiv, Ukraine, April 22-23, 2021, ISSN 2523-4013
http://colins.in.ua, online

6. Conclusions

An information system for finding tickets has been created, which will help solve the problem of buying a ticket on the market. To achieve the result, the following tasks are performed:

- methodological principles of research are given;
- system requirements are created;
- compared the system with analogues;
- a tree of goals and a hierarchy of tasks are built;
- the functioning of the system is specified by constructing a diagram of data flows;
- the choice of means of solving the problem was made;
- the created software is described;
- user manual created;
- the control example is analyzed;
- the economic substantiation of expediency of work is carried out.

7. References

[1] Ways to buy air tickets, 2019. URL: https://idealtrip.ru/aviabilet/turdom-birza.html.
[2] 12 tips on how to buy cheap air tickets, 2020. URL: https://travelq.ru/12-sovetov-kak-kupit-deshevye-aviabilety.
[3] The main advantages of buying a ticket on the Internet, 2019. URL: http://www.aviafaq.ru/useful/8-osnovnye-plyusy-pokupki-aviabileta-cherez-internet.html.
[4] Aggregator sites as a business, 2021. URL: https://vc.ru/life/261177-sayty-agregatory-kak-biznes-ot-a-do-ya.
[5] Aggregator: air tickets, 2020. URL: https://turproezdka.ru/prochee/agregator-aviabilety.html.
[6] Review of metasearch engines: good, bad, terrible, 2011. URL: https://nestor.minsk.by/kg/2011/28/kg12806.html.
[7] O. L. Nedashkivsky, Planning and design of information systems, Kyiv, 2014.
[8] O. L. Kozak, Reference syllabus of lectures on the course - Analysis of software requirements for students in the direction of training - Software Engineering, Ternopil, 2011.
[9] V. A. Luzhetsky, A.D. Kozhukhivsky, O.P. Voitovych, Basics of information security, Vinnytsia, 2013.
[10] S. M. Ilyashenko, Innovation management. Innovative management in the knowledge-oriented economy, Sumy, 2014.
[11] S. E. Vazhinsky, T. I. Scherbak, Methods and organization of scientific research, Sumy State Pedagogical University named after A.S. Makarenko, Sumy, 2016.
[12] Goal tree, 2018. URL: https://pidru4niki.com/18180520/ekonomika/pobudova_dereva_tsiley.
[13] Is it worth learning JavaScript: prospects, the situation in the labor market, expert opinions, 2021. URL: https://ru.hexlet.io/blog/posts/stoit-li-uchit-javascript-perspektivy-situatsiya-na-rynke-truda-mneniya-ekspertov.
[14] Angular vs react vs vue: best choice in 2021, 2021. URL: https://merehead.com/ru/blog/angular-vs-react-vs-vue-2021.
[15] How good is Node.js and why is it needed, 2019. URL: https://techrocks.ru/2019/01/20/why-do-you-need-node-js.
[16] All about MongoDB: the NoSQL database, 2020. URL: https://acodez.in/mongodb-nosql-database.