Vasyl Lytvyn
Natalia Sharonova
Thierry Hamon
Olga Cherednichenko
Natalia Grabar
Agnieszka Kowalska-Styczen
Victoria Vysotska
(Eds.)

ISSN 2523-4013

COMPUTATIONAL LINGUISTICS AND INTELLIGENT SYSTEMS

Proceedings of the 3rd International Conference,
COLINS 2019. Volume II: Workshop

Kharkiv, Ukraine
April, 2019

Lytvyn, V., Sharonova, N., Hamon, T., Cherednichenko, O., Grabar, N., Kowalska-Styczen, A., Vysotska, V. (Eds.): Computational Linguistics and Intelligent Systems. Proc. 3rd Int. Conf. COLINS 2019. Volume I: Main Conference. Kharkiv, Ukraine, April 18-19, 2019, CEUR-WS.org, online

This volume represents the proceedings of the Workshop Conference, with Posters and Demonstrations track, of the 3rd International Conference on Computational Linguistics and Intelligent Systems, held in Kharkiv, Ukraine, in April 2019. It comprises 13 contributed papers that were carefully peer-reviewed and selected from 27 submissions. The volume opens with the abstracts of the keynote talks. The rest of the collection is organized in two parts. Parts II contain the contributions to the Main COLINS Conference tracks, structured in two topical sections: (I) Computational Linguistics; (II) Intelligent Systems.

Preface

It is our pleasure to present you the proceedings of the Workshop Conference of COLINS 2019, the second edition of the International Conference on Computational Linguistics and Intelligent Systems, held in Kharkiv (Ukraine) on April 18-19, 2019.

The main purpose of the CoLInS conference is a discussion of the recent researches 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:

- mathematical models of language;
- artificial intelligence;
- statistical language analysis;
- data mining and data analysis;
- social network analysis;
- speech recognition;
- machine translation, translation memory systems and computer-aided translation tools;
- information retrieval;
- information extraction;
- text summarization;
- computer lexicography;
- question answering systems;
- opinion mining;
- intelligent text processing systems;
- computer-aided language learning;
- 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. There was also an exhibition area for poster and demo sessions. A Student section of the conference for students and PhD students run in parallel to the main conference.

This year Organizing Committee received 78 submissions, out of which 34 were accepted for presentation as a regular papers. The papers are submitted to the following tracks: Natural Language Processing (5 papers), corpus linguistics (3 papers), computational lexicography (1 papers), intelligent computer systems building (4 papers). The papers directly deal with such languages: Ukrainian, Russian, French, English and Polish.

April, 2019

Vasyl Lytvyn
Natalia Sharonova
Thierry Hamon
Olga Cherednichenko
Natalia Grabar
Agnieszka Kowalska-Styczen
Victoria Vysotska

Committees

General Chair

Natalia Sharonova, National Technical University “KhPI”, Ukraine

Steering Committee

Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine
Mykhailo Godlevskyi, National Technical University “KhPI”, Ukraine
Thierry Hamon, LIMSI-CNRS & Université Paris 13, France
Olga Cherednichenko, National Technical University “KhPI”, Ukraine
Natalia Grabar, CNRS UMR 8163 STL, France
Agnieszka Kowalska-Styczen, Silesian University of Technology, Poland

Program Chairs

Victoria Vysotska, Lviv Polytechnic National University, Ukraine
Thierry Hamon, LIMSI-CNRS & Université Paris 13, France
Olga Kanishcheva, National Technical University “KhPI”, Ukraine

Proceedings Chair

Olga Cherednichenko, National Technical University “KhPI”, Ukraine
Victoria Vysotska, Lviv Polytechnic National University, Ukraine
Olga Kanishcheva, National Technical University “KhPI”, Ukraine

Presentations Chair

Olha Yanholenko, National Technical University “KhPI”, Ukraine
Agnieszka Kowalska-Styczen, Silesian University of Technology, Poland

Poster and Demo Chairs

Yulia Gontar, National Technical University “KhPI”, Ukraine
Dmytro Dosyn, Karpenko Physico-Mechanical Institute of the NAS of Ukraine

PhD Symposium Chairs

Maryna Vovk, National Technical University “KhPI”, Ukraine
Dmytro Dosyn, Karpenko Physico-Mechanical Institute of the NAS of Ukraine

IT Talks Chairs

Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine
Olga Kanishcheva, National Technical University “KhPI”, Ukraine

Local Organization Chairs

Olga Cherednichenko (chair), National Technical University “KhPI”, Ukraine
Olga Kanishcheva (chair), National Technical University “KhPI”, Ukraine
Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine
Victoria Vysotska, Lviv Polytechnic National University, Ukraine

Publicity Chair

Natalia Sharonova, National Technical University “KhPI”, Ukraine
Mykhailo Godlevskyi, National Technical University “KhPI”, Ukraine

Web Chair

Victoria Vysotska, Lviv Polytechnic National University, Ukraine
Olga Kanishcheva, National Technical University “KhPI”, Ukraine

Program Committees

MAIN Conference COLINS 2019

Yuriy Bobalo, Lviv Polytechnic National University, Ukraine)

Anatoliy Sachenko, Ternopil National Economic University, Ukraine

Danuta Zakrzewska, Lodz University of Technology, Poland

Dmitry Lande, Institut for Information Recording of NAS of Ukraine

Dmytro Peleshko, IT Step University, Ukraine

Fadila Bentayeb, ERIC Laboratory, University of Lyon 2, France

Galia Angelova, Bulgarian Academy of Sciences, Bulgaria

Iryna Ivasenko, Karpenko Physico-Mechanical Institute of the NAS of Ukraine

Iryna Yevseyeva, Newcastle University, England

Klaus ten Hagen, University of Applied Science Zittau/Goerlitz, Germany

Lidia Pivovarova, University of Helsinki, Finland

Manik Sharma, DAV University, India

Michael Emmerich, Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands

Michael Pokojovy, University of Memphis, TN, USA

Mykhailo Godlevskyi, National Technical University “KhPI”, Ukraine

Natalia Grabar, CNRS UMR 8163 STL, France

Natalia Sharonova, National Technical University “KhPI”, Ukraine

Nina Khairova, National Technical University “KhPI”, Ukraine

Oksana Bihun, Mathematics University of Colorado, Colorado Springs USA

Oleg Bisikalo, Vinnytsia National Technical University, Ukraine

Oleg Garasym, Volvo IT, Poland

Aleksandr Gozhyj, Petro Mohyla Black Sea National University, Ukraine

Olena Levchenko, Lviv Polytechnic National University, Ukraine

Olga Cherednichenko, National Technical University “KhPI”, Ukraine

Scheller-Boltz Dennis, Vienna University of Economics and Business, Austria

Sergii Babichev, Jan Evangelista Purkinje University in Usti nad Labem, Chech Republic

Silakari Sanjay, Rajiv Gandhi Technical University, India

Svetla Boytcheva, Sofia University, Bulgarian Academy of Sciences, Bulgaria

Thierry Hamon, LIMSI-CNRS & Université Paris 13, France

Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine

Victoria Bobicev, Technical University of Moldova, Moldova

Victoria Vysotska, Lviv Polytechnic National University, Ukraine

Viktor Mashkov, Jan Evangelista University in Ústí nad Labem, Czech Republic

Vitor Basto-Fernandes, University Institute of Lisbon, Portugal

Volodymyr Lytvynenko, Kherson National Technical University, Ukraine

Volodymyr Pasichnyk, Lviv Polytechnic National University, Ukraine

Waldemar Wojcik, Lublin University of Technology, Lublin, Poland

Wolfgang Kersten, Institut für Logistik und Unternehmensführung, Germany

Zoran Cekerevac, “Union – Nikola Tesla” University, Serbia

Olha Yanholenko, National Technical University “KhPI”, Ukraine

Borut Werber, University of Maribor, Slovenia

Posters and Demonstrations Track

Fadila Bentayeb, ERIC Laboratory, University of Lyon 2, France

Olena Levchenko, Lviv Polytechnic National University, Ukraine

Dmytro Dosyn, Karpenko Physico-Mechanical Institute of the NAS of Ukraine

Victoria Bobicev, Technical University of Moldova, Moldova

Olha Yanholenko, National Technical University “KhPI”, Ukraine

Svetla Boytcheva, Sofia University, Bulgarian Academy of Sciences, Bulgari

Keynote Speakers

Mariana Romanyshyn, Grammarly, Ukraine

Oleg Bisikalo, Vinnytsia National Technical University, Ukraine

Jean-Hugues Chauchat, Université Lumière Lyon2, France

Oleksandr Gurbych, SoftServe, Ukraine

Nataliya Ryabova, Kharkiv National University of Radio Electronics, Ukraine

Vasyl Lytvyn, Lviv Polytechnic National University, Ukraine

Thierry Hamon, Université Paris 13, France

Additional Reviewers

Ihor Kulchytskyi, Lviv Polytechnic National University, Ukraine

Dmytro Dosyn, Karpenko Physico-Mechanical Institute of the NAS of Ukraine

Marianna Dilai, Lviv Polytechnic National University, Ukraine

Maxim Davydov, Lviv Polytechnic National University, Ukraine
Yevhen Burov, Lviv Polytechnic National University, Ukraine
Oleh Veres, Lviv Polytechnic National University, Ukraine
Tetiana Shestakevych, Lviv Polytechnic National University, Ukraine
Lozynska Olga, Lviv Polytechnic National University, Ukraine
Andrii Demchuk, Lviv Polytechnic National University, Ukraine
Krzysztof Wodarski, Politechnika Śląska, Poland
Filatov Valentin, Kharkiv National University of Radio Electronics, Ukraine
Ryabova Nataliya, Kharkiv National University of Radio Electronics, Ukraine
Rizun Nina, Gdansk University of Technology, Poland
Yevgeniy Bodyanskiy, Kharkiv National University of Radio Electronics, Ukraine
Olena Orobinska, National Technical University “KhPI”, Ukraine
Lucie Martinet, University of Lyon, Lyon2, France

Local Organization Committee

Natalia Sharonova, National Technical University “KhPI”, Ukraine
Mykhailo Godlevskyi, National Technical University “KhPI”, Ukraine
Olga Cherednichenko, National Technical University “KhPI”, Ukraine
Olga Kanishcheva, National Technical University “KhPI”, Ukraine
Yulia Gontar, National Technical University “KhPI”, Ukraine
Maryna Vovk, National Technical University “KhPI”, Ukraine
Tatyana Nazirova, National Technical University “KhPI”, Ukraine
Yulia Hlavcheva, National Technical University “KhPI”, Ukraine

Sponsors

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

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

SoftServe
https://www.softserveinc.com/

ZONE3000
https://zone3000.net/

Ukrainian Lingua-Information Fund, NAS of Ukraine
https://en.ulif.org.ua/

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

SSA Group
https://www.ssa.group

SYTOSS
https://sytoss.com

Global Work
https://www.globalwork-ua.com/

Conference Program

	Day 1 (18 April 2019)	
9.00	***Registration***	
9.30	***Conference Opening***	
9.40	***Welcoming address***	
10.00	Modeling the Phenomenological Concepts for Figurative Processing of Natural-Language Constructions *Oleg Bisikalo* (Vinnytsia National Technical University)	
10.30	Smart Parking and Smart Gate *Oleksandr Gurbych* (SoftServe)	
11.00	WikiWars-UA: Ukrainian Corpus Annotated with Temporal Expressions *Natalia Grabar, Thierry Hamon*	
11.15	Developing Linguistic Research Tools for Virtual Lexicographic Laboratory of the Spanish Language Explanatory Dictionary *Yevhen Kupriianov, Nunu Akopiants*	
11.30	***Coffee-break***	
12.00	***Poster Section***	
12.30	***Stream 1 (Paper Presentations)***	
12.30	Application of the C-Means Fuzzy Clustering Method for the Patient’s State Recognition Problems in the Medicine Monitoring Systems *Nina Bakumenko, Viktoria Strilets, Mykhaylo Ugryumov*	
12.45	Mining Methods for Adaptation Metrics in E-Learning *Andrii Kozyriev, Igor Shubin, Victoria Skovorodnikova, Maria Pitiukova*	
13.00	Modeling of Decision-Making Ontology *Anna Bakurova, Elina Tereschenko, Yurii Filei, Mariia Pasichnyk, Hanna Ropalo*	Grammarly Workshop for Students “Text Classification with Using FastText”
13.15	The Aligned Kazakh- Russian Parallel Corpus Focused on the Criminal Theme *Nina Khairova, Anastasiia Kolesnyk, Orken Mamyrbayev, Kuralay Mukhsina*	
13.30	Web Content Monitoring System Development *Lyubomyr Chyrun, Iryna Yevseyeva, Dmytro Dosyn, Valentin Tyhonov, Yevhen Burov, Aleksandr Gozhyj*	
13.45	Mathematical Model of Semantic Search and Search Optimization *Taras Basyuk, Andrii Vasyliuk, Vasyl Lytvyn*	
14.00	***Lunch***	
15.00	***Stream 1 (Paper Presentations)***	***Stream 2 (Student Section)***
15.00	The Matrix-based Knapsack Cipher in the Context of Additively Homomorphic Encryption *Aleksei Vambol*	Essays Dataset Analysis with IBM Watson Personality Insights *Andrii Khomenko* (SoftServe)
15.15	Machine Learning Technique for Regular Pattern Detector Synthesis: toward Mathematical Rationale *Grygoriy Zholtkevych, Nataliya Polyakovska*	Machine Learning Text Classification Model with NLP Approach *Maria Razno*
15.30	Ontological Approach to Plot Analysis and Modeling *Yevhen Burov, Victoria Vysotska, Petro Kravets*	Extraction of Semantic Relations from Wikipedia Text Corpus *Olexandr Shanidze, Svitlana Petrasova*
15.45	Detection of the Botnets’ Low-rate DDoS Attacks Based on Self-Similarity *Sergii Lysenko, Kira Bobrovnikova, Oleh Savenko, Andrii Nicheporuk*	Consideration of the Software Tests Quality Evaluation problem *Irina Liutenko, Oleksii Kurasov*
16.00	Application of Methods of Machine Learning for the Recognition of Mathematical Expressions *Oleh Veres, Ihor Rishnyak, Halyna Rishniak*	Method for Paraphrase Extraction from the News Text Corpus *Ilya Manuylov, Svitlana Petrasova*
16.15	Semantic Segmentation of a Point Clouds of an Urban Scenes *Andrey Dashkevich*	Semantic Similarity Identification for Short Text Fragments *Viktoriia Chuiko, Nina Khairova*

16.30	The Statistical Analysis of the Short Stories by Roman Ivanychuk *Ihor Kulchytskyi*	Study of Software Systems Usability Used for Customers Loyalty Identification *Mariia Bilova*, *Oleksandr Trehubenko*
16.45	Development of Information System for Categorizing Textual Content Categorizing Based on Ontology *Victoria Vysotska*, *Vasyl Lytvyn*, *Yevhen Burov*, *Pavlo Berezin*, *Fernandes Vitor Basto*	Identify of The Substantive, Attribute, and Verb Collocations in Russian Text *Julia Lytvynenko*
	Day 2 (19 April 2019)	
09.00	Computational Intelligence Methods for Rapid Online Texts Classification *Nataliya Ryabova* (Kharkiv National University of Radio Electronics)	
09.30	Linguistics in NLP: Why So Complex? *Mariana Romanyshyn* (Grammarly)	
10.00	Method of Automated Identification of Metaphoric Meaning in Adjective + Noun Word Combinations (Based on the Ukrainian Language) *Olena Levchenko*, *Nataliia Romanyshyn*	
10.15	The Web-Application Development for Analysis of Social Graph of the User in Network *Yuliia Leshchenko*, *Alina Yelizeva*, *Yuliia Rudchenko*	
10.30	Risk Assessment Technology of Crediting with the Use of Logistic Regression Model *Yurynets R.V., Yurynets Z.V., Kunanets N., Kis Ya. P.*	
10.45	The Method of Optimization of Structure and Content of an Ontology, Provided by a Weighted Conceptual Graph *Vasyl Lytvyn* (Lviv Polytechnic National University)	
11.00	***Coffee-break***	
11.30	***Poster Section***	
12.00	Why Letter N-Grams Can Classify Texts? Search for Relevant Keywords Using the Characteristic N-Grams *Jean-Hugues Chauchat* (Université Lumière Lyon2)	
12.30	Information and Technology Support for the Training of Visually Impaired People *Nataliia Veretennikova, Oleksandr Lozytskyi*, *Nataliia Kunanets*, *Volodymyr Pasichnyk*	
13.00	Comparative Analysis of Noisy Time Series Clustering *Lyudmyla Kirichenko, Tamara Radivilova, Anastasiia Tkachenko*	
13.15	On Intelligent Decision Making in Multiagent Systems in Conditions of Uncertainty *Dmytro Chumachenko, Kseniia Bazilevych*, *Ievgen Meniailov*, *Yulia Kuznetsova*	
13.30	***Lunch***	
14.30	Computational Terminology Extraction *Thierry Hamon* (Université Paris 13)	
15.00	***Stream 1 (Paper Presentations)***	***Stream 2 (Student Section)***
15.00	(n) Assumption in Machine Learning *Dmitry Klyushin, Sergiy Lyashko, Stanislav Zub*	Use your English and join to IT world *Lyubov Kremenetskaya* (zone3000)
15.15	Spatial Interpretation of The Notion of Relation and Its Application in the System of Artificial Intelligence *Ganna Proniuk*, *Nataliia Geseleva*, *Iryna Kyrychenko*, *Glib Tereshchenko*	Presentation of Grammarly CompLing Summer School 2019 *Mariana Romanyshyn* (Grammarly)
15.30	Method of Cross-Language Aspect-Oriented Analysis of Statements Using Categorization Model of Machine Learning *Tetiana Kovaliuk, Tamara Tielyshev, Nataliya Kobets*	
15.45	Problems of Storing and Processing Data in Serverless Technologies *Tetiana Naumenko*	
16.00	Estimation of Informativeness of Recognition Signs at Extreme Information Machine Learning of Knowledge Control System *Anatoliy Dovbysh, Ihor Shelehov, Svitlana Pylypenko, Oleh Berest*	Data-To-Text Generation for Domain-Specific Purposes *Tetiana Drobot*
16.15	Semantic Analysis and Natural Language Text Search for Internet Portal *Tetiana Kovaliuk*, Nataliya Kobets	Spam Filtering Methods Based on The Neural Network *Alyona Zhernovnikova*, *Zoia Kochueva*
16.30	The Object Model of the Subject Domain with the Use of Semantic Networks *Tetiana Kovaliuk*, *Kobets Nataliya*	Collocation Extraction Based on the Semantic-Syntactic Approach *Yana Galkina*, *Svitlana Petrasova*

16.45	Quantitative Evaluation Method for Mass Media Manipulative Influence on Public Opinion *Sergiy Gnatyuk, Jamilya Akhmetova, Viktoriia Sydorenko, Yuliia Polishchuk, Valentin Petryk*	Development of Web-Service with Selection of Actual Quotes *Tetiana Trokhymenko, Zoia Kochueva*
17.00	Web Content Monitoring System Development *Lyubomyr Chyrun, Iryna Yevseyeva, Dmytro Dosyn, Valentin Tyhonov, Yevhen Burov, Aleksandr Gozhyj*	Peculiarities of Inversion in English Language *Yuliia Tyshchuk, Tatiana Bratus*
17.15	Frequency Dictionaries to the Instructions to Medical Products *Roksolana-Yustyna Perkhach, Yuliia Shyika*	Automated Building and Analysis of Twitter Corpus for Toxic Text Detection *Kateryna Bobrovnyk*
17.30	Method "Mean – Risk" for Comparing Poly-Interval Objects in Intelligent Systems *Gennady Shepelev, Nina Khairova, Zoia Kochueva*	Lingvoexpert Definition of Signs of Targeted Falsification of Written Documentation *Anna Kaplunova, Zoia Kochueva*
17.30	***Conference Closing***	

Poster Section Day 1 (18 April 2019)

- *Berko* Knowledge-based Big Data cleanup method
- *Shakhovska Nataliya, Basystiuk Oleh, Shakhovska Khrystyna, Zakharchuk* Development of the speech-to-text chatbot interface based on Google API
- *Golyan Vira, Golyan* Effective methods of intelligent analysis in business processes
- *Kazarian Artem, Kunanets Nataliia, Pasichnyk Volodymyr, Veretennikova Nataliia, Rzheuskyi Antonii, Leheza Andrii, Kunanets Oksana* Complex Information E-Science System Architecture based on Cloud Computing Model
- *Meniailov Ievgen, Krivtsov Serhii, Ugryumov Mykhaylo, Bazilevich Kseniia, Trofymova* Application of Parallel Computing in Robust Optimization Design
- *Fedushko Solomia, Syerov Yuriy, Kolos Sofiia* Hashtag as a Way of Archiving and Distributing Information on the Internet
- *Shandruk Uliana* The Quantitative Characteristics of Key Words in Texts of Scientific Genre (on the Material of the Ukrainian Scientific Journal)
- *Vladimir Golovko, Alexander Kroshchanka, Egor Mikhno, Myroslav Komar, Anatoliy Sachenko, Sergei Bezobrazov, Inna Shylinska* Deep Convolutional Neural Network for Recognizing the Images of Text Documents
- *Vladlen Shapo, Valeriy Volovshchykov* Cloud technologies application at English language studying for maritime branch specialists

Poster Section Day 2 (19 April 2019)

- *Chetverikov* Structural approach in phonetic analysis on the example of the Ukrainian language
- *Olga Malyeyeva, Natalya Nosova, Oleg Fedorovych, Victor Kosenko* The semantic network creation for an innovative project Scope as a part of project knowledge ontology
- *Kunanets Nataliya, Halyna Matsiuk* Use of the smart city ontology for relevant information retrieval
- *Romanenkov, Kosenko V., Lobach O., Grinchenko E., Grinchenko M.* The method for ranking quasi-optimal alternatives in Interval game models against nature
- *Viacheslav Frolov, Oleksandr Frolov, Vyacheslav Kharchenko* Classification of diversity for dependable and safe computing
- *Lyubomyr Chyrun, Agnieszka Kowalska-Styczen, Aleksandr Gozhyj, Andrii Berko, Andrii Vasevych, Irina Pelekh* Heterogeneous Data with Agreed Content Aggregation System Development
- *Tetiana Naumenko* Problems of storing and processing data in serverless technologies
- *Oleksii Puzik* Intelligence knowledge-based system based on multilingual dictionaries
- *Volodymyr Lytvynenko, Natalia Savina, Maria Voronenko, Maryna Yakobchuk, Olena Kryvoruchko* Bayesian Networks' Development Based on Noisy-MAX Nodes for Modeling Investment Processes in Transport

Table of Contents

PAPER PRESENTATIONS

Knowledge-based Big Data Cleanup Method

Andrii Berko[0000-0001-6756-5661]

Lviv Polytechnic National University, Lviv, Ukraine

andrii.y.berko@lpnu.ua

Abstract. Unlike traditional databases, Big Data stored as NoSQL data resources. Therefore such resources are not ready for efficient use in its original form in most cases. It is due to the availability of various kinds of data anomalies. Most of these anomalies are such as data duplication, ambiguity, inaccuracy, contradiction, absence, the incompleteness of data, etc. To eliminate such incorrectness, data source special cleanup procedures are needed. Data cleanup process requires additional information about the composition, content, meaning, and function of this Big Data resource. Using the special knowledge base can provide a resolving of such problem.

Keywords: Big Data, Ontology, Knowledge Base, Data Cleanup.

1 Introduction

Not only volume, variety, and velocity of changes are characteristic for information resources developed on the principles of Big Data. The syntax and content heterogeneity of the resources themselves, the complexity of control, influence and management of the processes of their production and development also take place [2]. These factors, often contribute to the emergence of some data item corruptions [1,2] in the information resource content.Generally, Big Data resources are presented in NoSQL database formats. It means that principal requirements for such data resources are availability and partition tolerance. At the same time, any consistency constraints are not supported for such data. Weak data consistency leads to a situation when data resources are not ready for use in the original form. Therefore some steps to prepare these resource for efficient processing are needed.It means some data values must be transformed to the form corresponded with data source purpose, meaning of the tasks, and user requirements during preparing processes. One of the principal steps of Big Data resource preparing for its use is the application of data cleanup actions. Incorrect or invalid data values have to be edited, corrected or replaced by right and valid values at the clean-up stage of Big Data resource.As a result, we can obtain the set of so-called "clean" data which are correct, valid and ready to use according to their functions.

2 Data anomalies processing in Big Data sources

Data anomalies in the Big Data resources are presented by such phenomena as absence, duplication, ambiguity, lack of meaning, inaccuracy, incompleteness, unreliability, inconsistency, etc. [3,7]. The existence of data abnormalities greatly degrades the consumer properties of information resources, makes it difficult or impossible to efficient using due to invalid data items presence. The consequence of this is the incorrect execution of operations for the search, selection or analysis of data.For example, values that are equal to each other can be processed as different due to their inaccuracy, misrepresentation, corruption or input error, when performing such operation as data mapping – Map(X) of the MapReduce method.For the same reasons, different values may be mistakenly presented as equal. The absence or inadmissibility of some values makes it impossible to use them, etc.Therefore, the correct and efficient work with the Big Data provides for the procedures of their cleanup, during which, in particular, perform the elimination of existing data anomalies. Data anomalies interpretation is one of the principal tasks for efficient Big Data resource cleanup. The interpretation of data anomalies depends on their nature and the causes of the occurrence. It allows to recognize data anomalies in Big Data resource correctly and to choose the most suitable method of this anomalies elimination. For a successful solution of data anomalies problem, these need to be classified. According to [1,5] such principal types of data anomalies are definedfor big data resources:

- value not present,
- value is unknown,
- value is invalid,
- value is duplicated,
- value is ambiguous,
- value is not accurate enough,
- value is an incomplete,
- value is unreliable etc.

Classification allows us to choose the best way to eliminate it of data anomalies.

Each data anomaly may be detected for a data item by checking its correspondence to some predefined requirements.Such requirements have to be described as conditions of data value comparison. For detection of described above types of anomalies, such conditions may be used [1].

Table 1. Correspondence between data anomalies and check conditions

Type of data anomalies	Data item check condition
value not present value is unknown	data item Is Null
value is invalid	data item Is not in <interval>
value is unreliable	data item Is not in <set>
value is duplicated	Count(data item)>1
value is ambiguous	data item != data item
value is not accurate enough	data item != value
value is an incomplete	Number(data item)<value

This list may be continued or changed according to specifics of processed data.

Usually, to eliminate data anomalies, the most commonly used techniques are the removal, ignoring or re-defining of an appropriate data element, using the average, most likely, estimated or surrogate value, etc. [1,4,5].

The process of anomalies of data eliminationin the Big Data resource performs as a replacement of the incorrect value by the new value, which define by a special procedure.In the general case, the value v_{ij} of some data unit V_i, which is formed to eliminate its anomaly, depends on the nature (category) of data anomalies – U_k and the method of its elimination – S_l. The procedure for defining a new data value can be describe as a sequence of steps of kind

$$V_i \rightarrow U_k \rightarrow S_l \rightarrow v_i. \tag{1}$$

That mean: invalid data value V_i of category$U_k$by using of method S_l have to be replaced by value v_i for elimination of one case of data anomalies in some resource.The same transformation can be presented as a mapping

$$v_i = \Phi(V_i , U_k, S_l), \tag{2}$$

where Φ is a function for define new value for invalid data item using its category and corresponding method.These actions are perform during general data cleanup process of Big Data source.

3 Using ontologies for Big Data cleanup

Because it is necessary to have exact and complete descriptions of the correspondence between anomalies in the data resource, their classification is needed. As well a formal description of the ways to eliminate such anomalies is needed. Knowledge base may be uses for such purpose in the set of tools for Big Data sources cleanup. The core of this knowledge base may be formed by an ontology of type

$$O^O = < C^O, R^O, F^O >, \tag{3}$$

where $C^O = \{C^V, C^U, C^S\}$– is the set of concepts (classes), which include such subclasses:
C^V – entity set (subclass) for a presentation of data units in the Big Data resource to be processed,
C^U –the set of entities that describe the types and nature of each of the data anomalies presented in the Big Data resource,
C^S –the set of entities for the description of data anomalies elimination methods (data anomaly problems solving);
$R^O = \{R^{VU}, R^{US}, R^{VUS}\}$ – a set of relations between the above-defined concepts that include three subsets:
R^{VU}– a subset of binary relations between data values of $C^V$and types of anomalies in Big Data resourceC^U,
R^{US} – a subset of relation between types of data anomaliesof $C^U$and methods of anomaly eliminationC^S,

R^{VUS} – a subset of ternary relations between data values and methods of data anomaly problem solving;
F^O={F^V, F^C }– is a set of axioms (rules). Two kinds of rules are needed for data cleanup process. Each one describes the steps which must be performedfor solving of data anomaly problem, The first kind of rules – F^V include rules for data validation in presented Big Data resource. The second subset – F^Cconsist of rules defines the method (from the set C^S) of data cleanup. This method directly depends on certain data values (presented by the set C^V) and a certain type of anomalies of this data item (described by the set C^U).

The resource of Big Data needs to be deeply investigated to solve the problem of data cleanup. The main goal of this investigation is to get the answer to the questions:

- what data items and values are needed to solve some defined task;
- what is the structure and content of the data source used for this purpose;
- what kinds of problems with data are possible in the presented resource;
- what factors have an influence on data quality in the resource;
- what are data quality criteria and requirements;
- what are the methods to fix corrupted or invalid data items;
- how some methods of bad data fixing would be used to correct some data items.

When all necessary information about input Big Data resource is obtained, such knowledge may be formalized as units of certain special ontology.Generalized structural model of ontology for Big Data resources cleanup presented on Fig. 1.

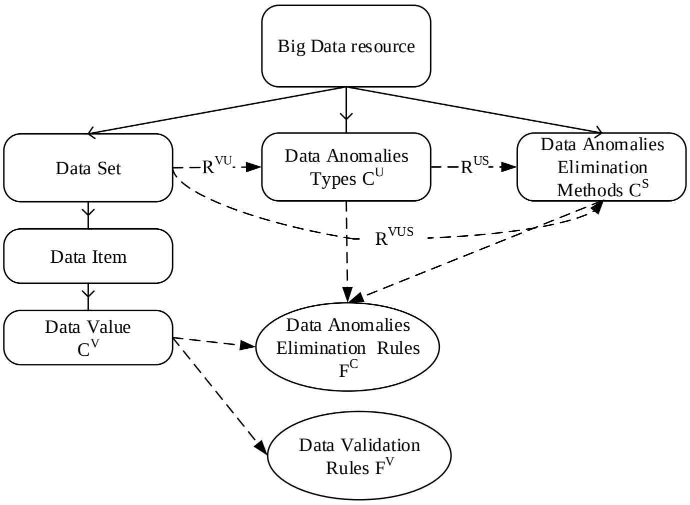

Big Data resource
Data Set
R^VU
Data Anomalies Types C^U
R^US
Data Anomalies Elimination Methods C^S
R^VUS
Data Item
Data Value C^V
Data Anomalies Elimination Rules F^C
Data Validation Rules F^V

Fig. 1. Structural model of ontology for Big Data resource cleanup

As it is shown at Fig.1, Big Data resource is described by three concepts of the ontology. The first describes certain data as a hierarchy: "Data set" (table, collection, etc) –"Data item" (column, field, element) – "Data value". The second concept describes the types of potential anomalies of data for processed data resource. And the third one - the ways to fix corrupted or invalid data in the resource.Two types of axioms describe the rules of data check to find certain anomalies and the rules of various data anomalies elimination. Possible relations between concepts and rules are described by arches. As a remark, the relation R^{VUS}seems redundant, because of its matching to R^{VU} and R^{US}. Bet such overage allows defining of unambiguous correspondence between the data set and possible methods of data anomalies elimination.

According to proposed principles of the generalized model, the ontology for any Big Data resource isdeveloped. It is necessary to determine the specific values of concepts, relationships, and rules in accordance with the content of the resource data when performing this action. As a result, the primary version of ontology for Big Data resource cleanup will be obtained.As the next, we can create an appropriate knowledge base for the data cleanup tools based on created ontology.

So, in the above-described way we can obtain the set of tools needed for efficient solving of Big Data resource anomalies elimination.Therefore, an ontology created in accordance with the principles described above can be used as the basis of the knowledge base for intelligent tools of the Big Data resources cleanup.

4 Algorithms and tools for Big Data resource cleanup

The use of ontology as the core of the knowledge base of the Big Data Cleanuptools determines the peculiarities of the process of solving data problems (Fig. 2).So, construction and tuning of basic ontology is a principal prerequisite for any Big Data source cleanup process. This stage requires a number of actions based on expert knowledge.Expert knowledge provide definition and creation of basic ontology parts just like that.

(a) Description of the set of concepts, which corresponds to data values and data unitsC^V, according to the set of requirements of ontology construction. The concept C^V for the presentation of data value may be defined as a result of hierarchic taxonomy – "DataSet ->Data Item -> Data Value" according to the proposed structural model of ontology.

(b) Construction of the set of definitions of data anomalies C^U, which are characteristic of the given Big Data source. The list of most common data anomalies is presented above.

(c) Definition of the set of methods $C^S$for data anomalies elimination. Most of well known and often used are such methods of solution of data anomaly problem are of these [1]:

- *repeat a request* to receive corrupted value,
- *recalculate* inaccurate value,
- *refine* inconsistent value,
- *replace* absent value with some aggregate value (average value, probable value, standard or default value, initial value, some calculated value, estimated value, expert value, etc.),
- *use* of artificial *surrogate marks* instead of absent or corrupted value,
- *remove* of corrupted data item from the resource,
- *ignore* the data anomaly for given data item,
- *using* of*special tools* to process uncertainties

(d) Definition of relations between concepts - data items, data anomalies and methods of data anomalies elimination – R^{VU}, R^{US}, R^{VUS}. The most suitable format for definition relations between concepts is RDF triplet [6] "Object - Predicate - Subject". So, these relations may be formed in such a way.

- For relation R^{VU} the triplets may be constructed by the scheme "*Data Value*(C^V)– Have Anomaly – *Anomaly Type* (C^U)".
- For relation R^{US} – by the scheme "*Anomaly Type*(C^U) – Eliminated by – *Method*(C^S)".
- For relation R^{VUS} the triplets have to be constructed by the scheme "*Data Value* (C^V)– *Anomaly Type*(C^U) – *Method*(C^S)". Here, the concept C^U execute a function of the predicate.

(e) Rules of data validation for its anomaly detection – F^V. This rules also may be presented as RDF triplets [6] by the scheme "Condition – Corresponds to – Anomaly Type". For example, the set of such rules may be like as the next

1. *Value is null,* Corresponds to, Value *not exist*
2. *Value is not in interval*, Corresponds to, Value *is invalid*
3. *Value is not equal to,* Corresponds to, Value *is inexact*
4. *Value is not between* (X,Y) , Corresponds to, Value *is unacceptable*and so on

(f) Rules of data anomalies elimination method using for various data values and various data anomalies types – F^C.These rules unambiguously correspond to the relation R^{VUS} between data items, data anomalies, and methods of anomalies elimination.

Ontology constructed in this way may be used as primary ontology for knowledge base of Big Data source cleanup Framework.

When the base ontology is constructedalgorithm of Big Data source cleanup can be applied for the first time. The first application may not give an effective result of data cleanup in the general case. It is because the knowledge base need the addition of new knowledge. General steps sequence of this algorithm is like the next.

1. For each data item with anomaly from classC^V, the type of anomaly has to be qualified. Qualification of anomaly means determining anomaly nature and the way of its interpretation. As a result, any concept from category "kind of anomaly"–C^Uwould correspond to any data item of C^V.If this operation can't be executed complement of ontology is necessary (see step 4 of algorithm).
2. Using data item determination and anomaly type qualification, corresponding relation, and the rule of the anomaly of certain type elimination for certain data item would be defined. Generally, data anomaly elimination rule is the expression of type

$$(V_i \wedge U_k) \rightarrow S_l, \tag{4}$$

where, V_i– data item, U_k– kind of data anomaly, S_l – method of data anomaly elimination (new data value definition).

3. At the nest step replacement if invalid data value has to be executed according to defined above relations and rules. When whole data resource is processed algorithm to be complete if not – return to step 1.
4. This step is need to recognize and fix the problem situation, appeared during the attempt of data resource cleanup. These problem situations can be categorized according to its origin:
   - no description of data item or data value in the ontology;

- no description of anomaly typein the ontology;
- no description of method to eliminate some type of data anomaly;
- no description of rule to eliminate anomaly for particular data value;
- implementation of rule to eliminate uncertainty did not effect.

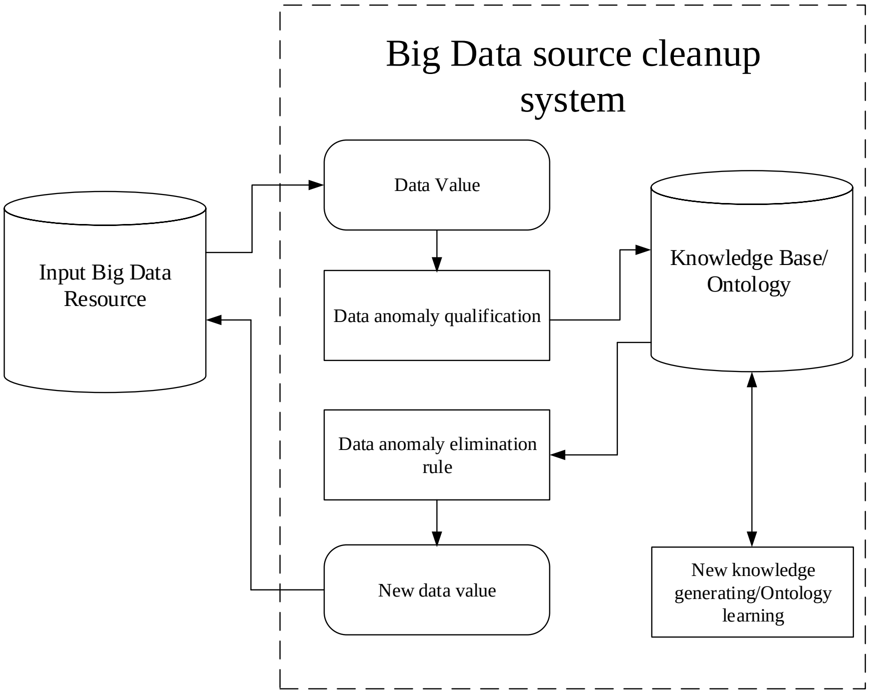

Big Data source cleanup system
Input Big Data Resource
Data Value
Data anomaly qualification
Knowledge Base/ Ontology
Data anomaly elimination rule
New data value
New knowledge generating/Ontology learning

Fig. 2. Data anomalies processing scheme for Big Data resource cleanup

5. If the situation is recognized and categorized it can be fixed by the specified way. These ways are of
   - define the new concept for the data value, for data anomaly type or for data anomaly elimination method;
   - define the new item into relation description for definition of correspondence between data value, data anomaly type, and anomaly elimination method;
   - define the new rule of data anomaly elimination.

   After execution of described over operations step 1 of algorithm may be repeated.

5 Conclusion

An approach to solve the problem of quality of Big Data sources by their cleanup was considered in the paper. The problem of such type may be solved by the development of knowledge-based intelligent tools. The peculiarity of the solution is to use an ontology as a core of knowledge base for the Big Data resource cleanup framework. It is the principal difference between the proposed approach and the traditional methods of data cleanup.The ontology can be considered as a special type of metadata which describes Big Data resource, anomalies of data and corresponding methods of data cleanup. The developed approach allows:

- to design special tools for intelligent Big Data resource cleanup;
- to make better procedures of data clearing of a Big Data resource;
- to accumulate for further use of knowledge and experience to solve data quality and data cleanup problem.

The principles and methods developed in the paper may be useful for data scientists at the processes of preparation of Big Data resources to analysis.

References

1. Alieksieiev,V., Berko, A.: A method to solve uncertainty problem for big data sources. In: Proceedings of the 2018 IEEE Second International Conference on Data Stream Mining & Processing, DSMP), 32-37(2018)
2. Aliekseyeva, K.,Berko, A.:Quality evaluation of information resources in web-projects. Actual Problems of Economics 136(10), 226-234 (2012)
3. Date, C. J.: Database in Depth: Relational Theory for Practitioners. O’Reilly, CA (2005).
4. Jaya, M. I., Sidi, F., Ishak, I., Affendey, L. S., Jabar, M. A. : A review of data quality research in achieving high data quality within organization. Journal of Theoretical and Applied Information Technology, Vol.95, No 12, 2647-2657 (2017)
5. Marz, N.,Warren, J.: Big Data: Principles and best practices of scalable realtime data systems, Manning Publications (2015)
6. RDFa Core 1.1 - Third Edition. Syntax and processing rules for embedding RDF through attributes. W3C Recommendation, https://www.w3.org/TR/2015/REC-rdfa-core-20150317(2015)
7. Rubinson, C.: Nulls, Three-Valued Logic, and Ambiguity in SQL : Critiquing Date’s Critique. In: SIGMOD Record Vol. 36, No. 4, 137-143(2007)

WikiWars-UA: Ukrainian corpus annotated with temporal expressions

Natalia Grabar[1] and Thierry Hamon[2,3]

[1]CNRS, Univ. Lille, UMR 81G3 - STL - Savoirs Textes Langage, F-59000 Lille, France

natalia.grabar@univ-lille.fr

[2]LIMSI, CNRS, Université Paris-Saclay. F-91405 Orsay, France;
[3]Université Paris 13. Sorbonne Paris Cité. F-93430 Villetaneuse. France

hamon@limsi.fr

Abstract. Reliability of tools and reproducibility of study results are important features of modern Natural Language Processing (NLP) tools and methods. The scientific research is indeed increasingly coming under criticism for the lack of reproducibility of results. First step towards the reproducibility is related to the availability of freely usable tools and corpora. In our work, we are interested in automatic processing of unstructured documents for the extraction of temporal information. Our main objective is to create reference annotated corpus with temporal information related to dates (absolute and relative), periods, time, etc. in Ukrainian, and to their normalization. The approach relies on the adaptation of existing application, automatic pre-annotation of WikiWars corpus in Ukrainian and its manual correction. The reference corpus permits to reliably evaluate the current version of the automatic temporal annotator and to prepare future work on these topics.

Keywords: Temporality, Information Extraction, Ukrainian, WikiWars, HeidelTime, Reference Corpus

1 Introduction

Unstructured documents are the most common source of information, and they may represent the majority of information available in different sources and domains. Yet, the work on unstructured narrative texts is very demanding on automatic methods for detecting, extracting, formalizing and organizing information contained in these documents. If information extraction (IE), which is part of Natural Language Processing (NLP), proposes such methods and aims at detecting and extracting relevant pieces of information from textual data, the question on availability of corpora, resources and reference data is very important. Indeed, such data are crucial for designing, testing and evaluating the automatic methods. Another important issue is related to the reliability of tools and to the reproducibility' of study results across similar data from different sources. The scientific research is indeed increasingly coming under criticism for the lack of reproducibility' of results [5,7,6]. First step towards the reproducibility of results is the availability of freely usable tools and corpora.

In our work, we focus on detection and extraction of temporal information, such as it occurs in these sentences:

- *Корабель Аполлон-11 стартував 16 липня 1969 о* 13 годині 32 хвилини *за Грінвічем. (The Apollo-11 ship took off at 1:32 pm GMT on 7/16/1969.)*
- *Протягом трьох годин, поки налагоджували зв'язок із Москвою, Гагарін давав інтерв'ю і фотографувався. (During three hours, while establishing communication with Moscow, Gagarin was interviewed and photographed.)*
- *Корейська війна – збройний конфлікт між Корейською Народно-Демократичною Республікою та Південною Кореєю, який тривав з 25 червня 1950 року до 27 липня 1953 р. (Korean war is an armed conflict between Democratic People's Republic of Korea and South Korea, which lasted from 25th of June 1950 up to 27th of July 1953.)*
- *В екваторіальному та тропічному поясі припливи і відпливи здебільшого повторюються двічі на добу. (In the equatorial and tropical areas, high and low tides mostly occur twice a day.)*
- *Тривали 118 років, з примиренням. (Lasted for 118 years, including armistices.)*
- *До середини 260-х до н.е. Римська республіка остаточно підпорядкувала собі Апеннінський півострів. (By the mid of 260 BC, the Roman Republic had gained control of the Italian peninsula.)*
- *Основним джерелом з історії греко-перських воєн є «Історія» Геродота, що містить опис подій до 478 до н.е. включно. ("The Histories" by Herodotus, which contains description of events up to 478 BC, is the main source on history of the Greco-Persian Wars.)*

Temporal information is important for several tasks and areas, as it allows to structure the entities and events according to their chronological occurrence. This is important in several situations. For instance, in historical studies, the events are usually ordered and then taught and studied in this order. Temporality has become an important research field and several challenges addressed this task up to now: ACE [1], SemEval [22, 23, 21], I2B2 2012 [20]. Yet, the main work is done on texts written in English. We propose to work with texts written in Ukrainian.

In what follows, we first present some related work (Sec. 2). We then precise our objectives (Sec. 3), introduce the material used (Sec. 4) and the proposed method (Sec. 5). Our results and their discussion are presented in Section 6. Finally, we conclude with some directions for future work (Sec. 7).

2 Related Work

Work on temporal information relies on three important steps when processing unstructured narrative documents: identification of linguistic expressions that are indicative of the temporality and their normalization [22, 4, 17, 11], and modeling and chaining of temporal information [2, 13, 14, 20, 9]. Identification of temporal expressions, which corresponds to the first step, provides basic knowledge for further tasks aiming at the processing of the temporality. The existing available automatic systems, such as HeidelTime [17] or SUTIME [4], exploit rule-based approaches, which makes them adaptable to new data, areas, and languages. Such tools usually encode temporal information with the TimeML standard.

TimeML[1] [14] is an annotation standard for temporal expressions proposed in 2010. Since then, it has became the reference for encoding temporal information in different languages. For instance, it has been used in several contexts: for encoding temporal data in challenge corpora such as TempEval [23,21,3] and I2B2 [20], for preparing corpora[2] annotated with temporal expressions such as TimcBank, TempEval, I2B2 and Clinical TempEval corpora,

TimeML offers the possibility to encode several types of temporal information and expressions (i.e. TIMEX3 tags):

1. Expressions of dates, time, durations or sets (attribute types). Dates and time are represented according to the ISO-8601 norm;
2. ISO-normalized forms of the expressions (attribute value), such as in (from examples above):
   – *16 липня 1969 о 13 годині 32 хвилини* => 1969-07-16T13:32:00
   – *трьох годин* => P3H
   – *двічі на добу* => P1D
3. Quantity and frequency of the set expressions (attributes quant or freq), such as in this expression of frequency:
   – *двічі на добу* => 2X
4. Begin and end anchors for durations (beginpoint and endpoint attributes). For instance, in example *Корейська війна – збройний конфлікт між Корейською Народно-Демократичною Республікою та Південною Кореєю, який тривав з 25 червня 1950 року до 27 липня 1953 р.* (*Korean war is an armed conflict between Democratic People's Republic of Korea and South Korea, which lasted from 25th of June 1950 up to 27th of July 1953.*), the begin anchor is *25th of June 1950* and the end anchor is *27th of July 1953*. The implicit duration is 3 years, 1 month and 2 days, which is normalized in P3Y1M2D.
5. Temporal modifiers, which have been introduced in order to annotate changed or clarified temporal expressions. For instance, in example *До середини 260-х до н.е. Римська республіка остаточно підпорядкувала собі Апеннінський півострів.* (*By the mid of 260 BC, the Roman Republic had gained control of the Italian peninsula.*), the date *260-х до н.е.* is changed by *середини*, which is the date modifier attribute MID.

In addition to the annotation of temporal expressions, TimeML also allows to describe events as well as relations between temporal expressions and/or events. In this paper, we only focus on the annotation of temporal expressions (TIMEX3) related to dates, durations and time.

There is quite few available corpora with temporal annotations. In addition to corpora mentioned above and created as part of challenges, there is the WikiWars

[1] *http://www.timeml.org*

[2] *http://timexportal.wikidot.com/*

corpus[3] [12] which provides a collection of texts issued from Wikipedia articles. These texts describe the course of the most famous wars in history, including the biggest wars that happened in the 20th century. The corpus contains 22 articles (such as WW1, WW2, Vietnamese war, Russo-Japanese war, or Punic wars). The main interest in working with these Wikipedia articles is that they contain several dates, as they are typically associated with battles, meetings, armistices, etc. The initial project contains articles in English. It has been extended to three other languages (German, Vietnamese and Croatian) [16, l0]. Hence, another interest in working with this corpus is that is contains comparable information and data in several languages.

3 Objectives

The purpose of our work is to build reference corpus in Ukrainian language annotated with temporal information. The corpus is freely available for the research. Temporal information is detected and normalized in the ISO format with respect of the TIMEX3 norm.

4 Material

Encyclopedic articles are obtained from the Wikipedia resource in Ukrainian[4], which is a free and collaborative resource. This encyclopedia contains information on a great variety of topics. For our work, we created the WikiWars corpus [12] in Ukrainian, which contains Wikipedia articles describing the most famous wars in history, including the biggest wars of the 20th century. Overall, the corpus contains 22 articles (such as WWl, WW2, Vietnamese war, Russo-Japanese war, or Punic wars), and 66,479 word occurrences. The articles have been collected similarly to the building of the original WikiWars corpus [12].

5 Methods

The methods are composed of three main steps: pre-annotation of texts with *HeidelTime* adapted to Ukrainian, manual correction, and evaluation of the current version of automatic annotations obtained with *HeidelTime*. We also propose a comparison with the English version of the annotations.

5.1 Pre-annotation

For the pre-annotation, we use the *HeidelTime* application. During a preliminary study, *HeidelTime* [17] has been extended to over 200 other languages [18] using

[3] *http://timexportal.wikidot.com/wikiwars*

[4] *https://uk.wikipedia.org*

existing multilingual resources such as Wiktionary[5], which provides data for 170 languages. The test of this version provided no results for Ukrainian (lexical ambiguity and polysemy, missing translations, Wiktionary resources not suitable for the purpose...). Hence, *HeidelTime* was first adapted to the Ukrainian language [8].

HeidelTime is a cross-domain temporal tagger that extracts temporal expressions from documents and normalizes them according to the TIMEX3 annotation standard, which is part of the markup language TimeML [14]. This is a rule-based system. Because the source code and the resources (patterns, normalization information, and rules) are strictly separated, it is possible to develop and implement resources for additional languages and areas using HeidelTime rule syntax. Three kinds of resources have been improved:

- linguistic patterns, which describe linguistic elements of the temporality (days of the week, months, numbers, etc.). This type of resources is used for the detection of temporality in texts;
- normalization resources, which are created to permit the normalization of the detected elements. In this way, all the detected units are normalized. Thanks to these resources, normalization can be performed for absolute (Example (1)) and relative (Example (2)) dates, durations and sets. Thus, the normalized values of Examples (1) and (2) are 2015-05-07 and 2017-05-09, respectively if we consider that these two dates are related;
- rules for composing more sophisticated detection of temporality, such as periods, intervals and specific expressions.

(1) *7 травня 2015 року. (May 7th, 2015.)*

(2) Через два дні. (Two days later.)

HeidelTime was adapted to Ukrainian language on newspaper articles from Українська правда[6]. Then, it was used for the automatic annotation of the WikiWars corpus, which provided 2,226 annotations. Because of the diversity of Wikipedia articles and topics, the detection of temporal information must cover a great variety of dates and formats, several of which do not occur in modern newspaper articles, on which the system in Ukrainian has been developed.

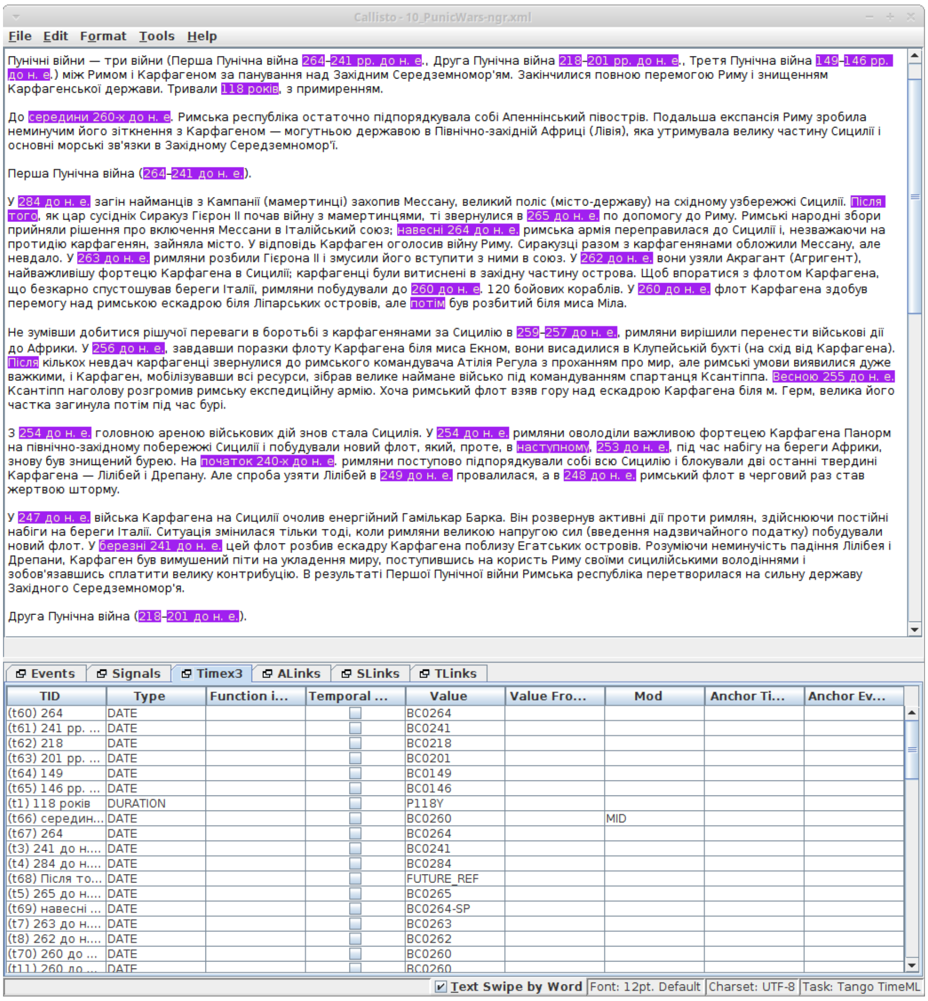
5.2 Manual correction of annotations

The results provided by the automatic annotation of temporality are then corrected and completed manually. We use the Callisto application developed by the MITRE Corp.[7]. As can be seen in Figure 1, this application permits to visualize the existing annotations in the text (superior part) together with the detected linguistic units and their normalizations (inferior part). Then, each annotation can be modified or deleted, and new annotations can be created. The same operations are available for the normalization information. Modifiers can also be added to the annotations, such as *mid*, *end*, *after*.

[5] *https://www.wiktionary.org/*

[6] *https://www.pravda.com.ua/news/*

[7] *https://mitre.github.io/callisto/manual/use.html*



Пунічні війни — три війни (Перша Пунічна війна 264–241 рр. до н. е., Друга Пунічна війна 218–201 рр. до н. е., Третя Пунічна війна 149–146 рр. до н. е.) між Римом і Карфагеном за панування над Західним Середземномор'ям. Закінчилися повною перемогою Риму і знищенням Карфагенської держави. Тривали 118 років, з примиренням.

До середини 260-х до н. е. Римська республіка остаточно підпорядкувала собі Апеннінський півострів. Подальша експансія Риму зробила неминучим його зіткнення з Карфагеном — могутньою державою в Північно-західній Африці (Лівія), яка утримувала велику частину Сицилії і основні морські зв'язки в Західному Середземномор'ї.

Перша Пунічна війна (264–241 до н. е.).

У 284 до н. е. загін найманців з Кампанії (мамертинці) захопив Мессану, великий поліс (місто-державу) на східному узбережжі Сицилії. Після того, як цар сусідніх Сиракуз Гієрон II почав війну з мамертинцями, ті звернулися в 265 до н. е. по допомогу до Риму. Римські народні збори прийняли рішення про включення Мессани в Італійський союз; навесні 264 до н. е. римська армія переправилася до Сицилії і, незважаючи на протидію карфагенян, зайняла місто. У відповідь Карфаген оголосив війну Риму. Сиракузці разом з карфагенянами обложили Мессану, але невдало. У 263 до н. е. римляни розбили Гієрона II і змусили його вступити з ними в союз. У 262 до н. е. вони узяли Акрагант (Агригент), найважливішу фортецю Карфагена в Сицилії; карфагенці були витиснені в західну частину острова. Щоб впоратися з флотом Карфагена, що безкарно спустошував береги Італії, римляни побудували до 260 до н. е. 120 бойових кораблів. У 260 до н. е. флот Карфагена здобув перемогу над римською ескадрою біля Ліпарських островів, але потім був розбитий біля миса Міла.

Не зумівши добитися рішучої переваги в боротьбі з карфагенянами за Сицилію в 259–257 до н. е., римляни вирішили перенести військові дії до Африки. У 256 до н. е., завдавши поразки флоту Карфагена біля миса Екном, вони висадилися в Клупейській бухті (на схід від Карфагена). Після кількох невдач карфагенці звернулися до римського командувача Атілія Регула з проханням про мир, але римські умови виявилися дуже важкими, і Карфаген, мобілізувавши всі ресурси, зібрав велике наймане військо під командуванням спартанця Ксантіппа. Весною 255 до н. е. Ксантіпп наголову розгромив римську експедиційну армію. Хоча римський флот взяв гору над ескадрою Карфагена біля м. Герм, велика його частка загинула потім під час бурі.

З 254 до н. е. головною ареною військових дій знов стала Сицилія. У 254 до н. е. римляни оволоділи важливою фортецею Карфагена Панорм на північно-західному побережжі Сицилії і побудували новий флот, який, проте, в наступному, 253 до н. е., під час набігу на береги Африки, знову був знищений бурею. На початок 240-х до н. е. римляни поступово підпорядкували собі всю Сицилію і блокували дві останні твердині Карфагена — Лілібей і Дрепану. Але спроба узяти Лілібей в 249 до н. е. провалилася, а в 248 до н. е. римський флот в черговий раз став жертвою шторму.

У 247 до н. е. війська Карфагена на Сицилії очолив енергійний Гамількар Барка. Він розвернув активні дії проти римлян, здійснюючи постійні набіги на береги Італії. Ситуація змінилася тільки тоді, коли римляни великою напругою сил (введення надзвичайного податку) побудували новий флот. У березні 241 до н. е. цей флот розбив ескадру Карфагена поблизу Егатських островів. Розуміючи неминучість падіння Лілібея і Дрепани, Карфаген був вимушений піти на укладення миру, поступившись на користь Риму своїми сицилійськими володіннями і зобов'язавшись сплатити велику контрибуцію. В результаті Першої Пунічної війни Римська республіка перетворилася на сильну державу Західного Середземномор'я.

Друга Пунічна війна (218–201 до н. е.).

TID	Type	Function i...	Temporal ...	Value	Value Fro...	Mod	Anchor Ti...	Anchor Ev...
(t60) 264	DATE		□	BC0264				
(t61) 241 рр. ...	DATE		□	BC0241				
(t62) 218	DATE		□	BC0218				
(t63) 201 рр. ...	DATE		□	BC0201				
(t64) 149	DATE		□	BC0149				
(t65) 146 рр. ...	DATE		□	BC0146				
(t1) 118 років	DURATION		□	P118Y				
(t66) середин...	DATE		□	BC0260		MID		
(t67) 264	DATE		□	BC0264				
(t3) 241 до н....	DATE		□	BC0241				
(t4) 284 до н....	DATE		□	BC0284				
(t68) Після то...	DATE		□	FUTURE_REF				
(t5) 265 до н....	DATE		□	BC0265				
(t69) навесні ...	DATE		□	BC0264-SP				
(t7) 263 до н....	DATE		□	BC0263				
(t8) 262 до н....	DATE		□	BC0262				
(t70) 260 до ...	DATE		□	BC0260				
(t11) 260 до ...	DATE		□	BC0260				

☑ Text Swipe by Word | Font: 12pt. Default | Charset: UTF-8 | Task: Tango TimeML

Fig. 1. Reading and annotation of temporal information with Callisto on example of Punic wars

5.3 Evaluation

After the manual correction of the annotations and creation of the reference data, the results from the automatic annotation are evaluated against these reference data with classical evaluation measures [15]:

– true positives *TP*: number of correctly extracted or normalized temporal expressions;

– precision *P*: percentage of the relevant temporal expressions extracted and normalized divided by the total number of the temporal expressions extracted and normalized;

– recall R: percentage of the relevant temporal expressions extracted divided by the number of the expected temporal expressions;

– F-measure F: the harmonic mean of the precision and recall values $\frac{P \cdot R}{P + R}$.

The evaluation is done with scripts available from previous work [17]. The evaluation measures are computed with strict and relaxed values, according to whether the boundaries of the temporal expressions are detected correctly (strict boundaries) or not (intersection between reference and automatically extracted linguistic units).

6 Results and Discussion

During the manual correction of the results, we observed two main difficulties in the results of the current version of HeidelTime in Ukrainian:

– Ambiguity of *північ* meaning both *midnight* and *north*. Currently, every occurrence of this marker is automatically annotated as temporal information, like in this example:

  У спробах полегшити тиск з <TIMEX3 type="TIME" value="1967-06-12T24:00">півночі</TIMEX3>, <TIMEX3 type="DATE" value="1967-08-09">9 серпня </TIMEX3> мобільна бригада армії Біафри у складі 3000 осіб за підтримки артилерії та бронемашин переправилася на західний берег Нігера. (Trying to ease, the pressure from north, 9th of August mobile group of the Biafra army composed of 3,000 people and supported by artillery and armoured cars crossed the Niger river and reached its western side.)

  The disambiguation of this marker may rely on prepositions it subsumes or on additional analysis of texts, as pre-processing or post-processing step.

– The native tokenization of *HeidelTime* is unable to tokenize text in several situations, for which reason, several dates have been missed by the automatic annotation. For instance, all the dates have been missed in this example: *Пунічні війни – три війни (Перша Пунічна війна 264-241 рр. до н.е., Друга Пунічна війна 218-201 рр. до н.е., Третя Пунічна війна 149-146 рр. до н.е.) між Римом і Карфагеном за панування над Західним Середземномор'ям. (Punic wars – three wars (First Punic war 264-241 BC, Second Punic war 218-201 BC, Third Punic war 149-146 BC) between Pome and Carthage for the conquest of the Western Mediterranean.))*

Further to the manual correction, the reference annotations amount up to 2,719 temporal units. The automatic extraction provides 2,116 temporal units, among which 2,018 are correct (True Positives). For comparison, in the English version of the WikiWars corpus (19 files), the reference data contain 1,858 temporal expressions.

Table 1 indicates global strict and relaxed values of Precision P, Recall R and F-measure F of the extracted results. As expected, relaxed values are better because they accept intersection between reference data and automatically extracted results. Yet, the strict values are high as well, which means that the system is quite successful in the extraction of temporal linguistic expressions.

Table 1. TIMEX3: strict and relaxed values for Precision, Recall and F-measure

	P	*R*	*F*
strict match	*0.85*	*0.66*	*0.75*
relaxed match	*0.95*	*0.74*	*0.83*

Table 2 indicates performance of the system for the detection of types of temporal expressions (which may be of three natures: date, duration, time) and for their normalization. As in other works, we can see that evaluation values obtained for the extraction of linguistic temporal expressions are higher than values for their normalization [19]. In several situations, it may indeed be complicated to compute the normalized values, and typically when past of future events are just mentioned in the texts. Improvement of the computing of the normalization values is one of the challenges for future work.

Table 2. TIMEX3: extraction and normalization values (Precision, Recall, F-measure)

	TP	*P*	*R*	*F*
type	*1.897*	*0.90*	*0.70*	*0.80*
normalization	*1.651*	*0.78*	*0.60*	*0.69*

7 Conclusion and Future Work

The main purpose of this work is the creation of the reference corpus with annotations of temporal expressions and their normalizations. We proposed to build such corpus in Ukrainian as part of the WikiWars corpora. The pre-annotation is done automatically with the Ukrainian version of *HeidelTime*. The annotations are then verified and completed manually. Overall, on 22 Wikipedia articles, we count 2,719 reference temporal units. This reference corpus permits to make the evaluation of the current version of *HeidelTime*: up to 0.80 F-mcasure for the detection of temporal expressions and up to 0.69 F-measurc for their normalization. This corpus is being made available for the research. It will permit to improve the automatic detection of temporal expressions in Ukrainian. One of the challenges is related to the normalization of the temporal expressions and to their link with the events described.

References

1. ACE challenge: The ACE 2004 evaluation plan. evaluation of the recognition of ace entities, ace relations and ace events. Tech. rep., ACE challenge (2004). http: //www.itl.nist.gov/iad/mig/tests/ace/2004
2. Batal, I., Sacchi, L., Bellazzi, R., Hauskrecht, M.: A temporal abstraction framework for classifying clinical temporal data. In: Ann Symp Am Med Inform Assoc (AMIA). pp. 29-33(2009)
3. Bethard, S., Savova, G.,Palmer, M., Pustejovsky, J.: Semeval-2017 task 12: Clinical tempeval. In: Int Workshop on Semantic Evaluation (SemEval-2017). pp. 565-572. Association for Computational Linguistics, Vancouver, Canada (August 2017)
4. Chang, A.X., Manning, C.D.: SUTIME: A library for recognizing and normalizing time expressions. In: LREC. pp. 3735-3740 (2012)

5. Chapman, W.W., Nadkarni, P.M., Hirschman, L., D'Avolio, L.W., Savova, G.K., Uzuner, O.: Overcoming barriers to nlp for clinical text: the role of shared tasks and the need for additional creative solutions. J Am Med Inform Assoc 18(5), 540-543(2011)
6. Cohen, K.B., Xia, J., Roeder, C., Hunter, L.E.: Reproducibility in natural language processing: A case study of two R libraries for mining PubMed/MEDLINE. In: LREC Int Conf Lang Resour Eval. pp. 6-12 (2016)
7. Collins, F., Tabak, L.: Nih plans to enhance reproducibility. Nature 505, 612-613 (2014)
8. Grabar, N., Hamon, T.: Automatic detection of temporal information in ukrainian general-languagetexts. In: COLINS 2018. pp. 1-11 (2018)
9. Grouin, C., Grabar, N., Hamon, T., Rosset, S., Tannier, X., Zweigenbaum, P.: Hybrid approaches to represent the clinical patient's timeline. J Am Med Inform Assoc 20(5), 820-7(2013)
10. Jeong, Y.S., Joo, W.T., Do, H.W., Lim, C.G., Choi, K.S., Choi, H.J.: Korean timeml and korean timebank. In: Chair), N.C.C., Choukri, K., Declerck, T., Goggi, S., Grobelnik, M., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S. (eds.) Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). European Language Resources Association (ELRA), Paris, France (may 2016)
11. Kessler, R., Tannier, X., Hagege, C., Moriceau, V., Bittar, A.: Finding salient dates for building thematic timelines. In: Annual Meeting of the Association for Computational Linguistics. pp. 730-739 (2012)
12. Mazur, P., Dale, R.: WikiWars: A new corpus for research on temporal expressions. In: Int Conf on Empirical Methods in Natural Language Processing. pp. 913-922 (2010)
13. Moskovitch, R., Shahar, Y.: Medical temporal-knowledge discovery via temporal abstraction. In: Ann Symp Am Med Inform Assoc (AMIA). pp. 452-456 (2009)
14. Pustejovsky, J., Lee, K., Bunt, H., Romary, L.: ISO-TimeML: An international standard for semantic annotation. In: Chair), N.C.C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., Tapias, D. (eds.) Int Conf Language Resources and Evaluation (LREC'10). European Language Resources Association (ELRA), Valletta, Malta (may 2010)
15. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys 34(1), 1-47 (2002)
16. Strotgen, J., Gertz, M.: Wikiwarsde: A german corpus of narratives annotatedwith temporal expressions. In: Conf of the German Society for Comp Linguistics and Language Technology (GSCL 2011). pp. 129-134. Hamburg, Germany (September 2011)
17. Strotgen, J., Gertz, M.: Temporal tagging on di erent domains: Challenges, strategies, and gold standards. In: Int Conf on Language Resources and Evaluation (LREC'12). pp. 3746-3753. ELRA (2012)
18. Strotgen, J., Gertz, M.: A baseline temporal tagger for all languages. In: Int Conf on Empirical Methods in Natural Language Processing. pp. 541-547. ACL (2015)
19. Strotgen, J., Armiti, A., Canh, T.V., Zell, J., Gertz, M.: Time for more languages: Temporal tagging of Arabic, Italian, Spanish, and Vietnamese. ACM Transactions on Asian Language Information Processing 13(1), 1-21 (2014)
20. Sun, W., Rumshisky, A., Uzuner, O.: Evaluating temporal relations in clinical text: 2012 i2b2 challenge. JAMIA 20(5), 806-813 (2013)
21. UzZaman, N., Llorens, H., Derczynski, L., Allen, J., Verhagen, M., Pustejovsky, J.: Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In: Int Workshop on Semantic Evaluation (SemEval 2013). pp. 19. Atlanta, Georgia, USA (June 2013), http://www.aclweb.org/anthology/ S13-2001
22. Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Katz, G., Pustejovsky, J.: Semeval-2007 task 15: Tempeval temporal relation identication. In: Int Workshop on Semantic Evaluations (SemEval-2007). pp. 75-80. Prague, Czech Republic (June 2007), http://www.aclweb.org/anthology/S/S07/S07-1014

23. Verhagen, M., Sauri, R., Caselli, T., Pustejovsky, J.: Semeval-2010 task 13: Tempeval-2. In: Int Workshop on Semantic Evaluation. pp. 57-62. Uppsala, Sweden (July 2010), http://www.aclweb.org/anthology/S10-1010

A(n) Assumption in Machine Learning

Dmitry Klyushin[0000-0003-4554-1049], Sergey Lyashko[0000-0003-1016-5231], and Stanislav Zub[0000-0002-5499-0885]

Taras Shevchenko National University of Kyiv, 03680, Kyiv, prospect Glushkova 4D

dokmed5@gmail.com

Abstract. The commonly used statistical tools in machine learning are two-sample tests for verifying hypotheses on homogeneity, for example, for estimation of corpushomogeneity, testing text authorship and so on. Often, they are effective only for sufficiently large sample (n> 100) and have limited application in situations where the size of samples is small (n < 30). To solve the problem for small samples, methods of reproducing samples are often used: jackknife and bootstrap. We propose and investigate a family of homogeneity measures based on A(n) assumption that are effective both for small and large samples.

Keywords: machine learning, sample homogeneity, confidence interval, order statistics, variational series

1 Introduction

Some problems of text analysis, in particular, comparisonof literary styles or testing the homogeneity of the text corpora, can be considered as a comparison of the distributions of random variables. In this case, the natural solution to the problem is the use two-sample tests, which allow one to test the hypothesis that the samples were drawn from the same population.Statistical methods for comparing the literary style using two-sample tests are investigated, in particular, in the papers of Granichin et al. [2] and Zenkov [3] in which the non-parametric Kolmogorov-Smirnov test was used. Statistical methods for assessing the homogeneity of text corpora are investigated, for example, in the papers of Kilgariff [3, 4] and Eder et al. [6, 7], who applied the Wilcoxon-Mann-Whitney test. Despite the widespread use of the Kolmogorov-Smirnov and Wilcoxon-Mann-Whitney tests, they have some drawbacks; in particular, theyare slightly sensitive to changing of distribution parameters [8].

The purpose of this work is to demonstrate a new test based on a family of p-statistics [8] as an alternative to the Wilcoxon-Mann-Whitney test in machine learning applications, and to prove experimentally its independence from the choice of the type of confidence interval for binomial proportion.

2 Two-sample test for verifying hypotheses on homogeneity

Let $x=(x_1,x_2,...,x_n)$ and $y=(y_1,y_2,...,y_m)$ be two samples drawn from the populations X and Y respectively, and z be a sample drawn from one of these two populations. Let's assume that all samples were drawn by simple random sampling. The problem is to identify the population from which the sample z was drawn. This is a common problem in machine learning, for example, in estimation of corpora similarity, testing text authorship and so on.

Solving this problem by classical non-parametric Kolmogorov-Smirnov, Wilcoxon-Mann-Whitney or other non-parametric two-sample tests often leads to uncertainty of an type II error or non-decision depending on the truth or falsity of the null hypothesis. We propose afamily of two-sample non-parametric tests for verifying the hypothesis of samples homogeneity whichsignificance level does not depend on whether the null hypothesis is true or false and has no asymptotic character. Also, we will show the this family does not depend on the choice of the type of the confidence interval for binomial proportion.

Consider the following criterion for the test of hypothesis *H* onhomogeneity of two samples drawn from the populations with distribution functions $F_X(u)$ and $F_Y(u)$, respectively. Let $x=(x_1,...,x_n)\in X$ and $y=(y_1,...,y_m)\in Y$ be the test samples, and $x_{(1)}\le ...\le x_{(n)}$, $y_{(1)}\le ...\le y_{(m)}$ be corresponding variational series. Suppose that $F_X(u)=F_Y(u)$. According to Hill's *A*(*n*) assumption[7],

$$p\left(y_k\in\left(x_{(i)},x_{(j)}\right)\right)=\frac{j-i}{n+1},\ i<j. \qquad (1)$$

Using the sample $y=(y_1,...,y_m)$, we can calculate a frequency h_{ij} of the random event $\left\{y_k\in\left(x_{(i)},x_{(j)}\right)\right\}$ and construct a confidence interval I_{ij} for the probability $p\left(y_k\in\left(x_{(i)},x_{(j)}\right)\right)$ atthegiven significance level β. There are several options for selection ofthe confidence intervals for binomial proportion, therefore there is a family of similarity measures corresponding to different confidence intervals for binomial proportion.

Denote by *L*the number of intervals I_{ij} containing$\frac{j-i}{n+1}$.Then, we can define themeasure of samples homogeneity $h_{xy}=\frac{2L}{n(n-1)}$ as the proportion of I_{ij} containing $\frac{j-i}{n+1}$ among all possible intervals $\left(x_{(i)},x_{(j)}\right), i<j.$ As far as h_{xy} is a frequency of the random event $\left\{\frac{j-i}{n+1}\in I_{ij}\right\}$ having the probability $1-\beta$, we can get

a confidence interval I_{xy} for the probability $p\left(\frac{j-i}{n+1} \in I_{ij}\right)$ which have confidence level β. So, if $1-\beta \in I$ then hypothesis H is accepted, otherwise it is rejected. The statistics h_{xy} is a measure of the homogeneityof the samples x and y. Exchanging x and y, we find the frequency h_{yx} and confidence interval I_{yx} which can give one more test for verifying hypothesis H. Since h_{xy} is not symmetric, we can construct a symmetric homogeneity measure as

$$h = \frac{1}{2}\left(h_{xy} + h_{yx}\right). \quad (2)$$

Theorem 1[8].If 1) $n = m$; 2) $0 < \lim\limits_{n\to\infty} p_q^{(n)} = p_0 < 1$and 3) $0 < \lim\limits_{n\to\infty} \frac{i}{n+1} = p^* < 1$, then the asymptotic level $\beta$of the sequence of confidence intervals $I_{ij}^{(n)}$ for the probabilities $p\left(y_k \in \left(x_{(i)}, x_{(j)}\right)\right)$ does not exceed β.

Let $B_1, B_2, ..., B_n, ...$be a sequence of event that can occur as e result of an experiment E and $\lim\limits_{k\to\infty} p(B_k) = p^*$. Let $h_{n_1}(B_1), h_{n_2}(B_2), ..., h_{n_k}(B_k), ...$ be a sequence of the events $B_1, B_2, ..., B_n, ...$, respectively, and $\frac{k}{n_k} \to 0$ as $k \to \infty$. We shall call the experiment E a *strong random experiment* if $h_{n_k}(B_k) \to p^*$ as $k \to \infty$. Denote that in our case the base trial T is a simple random sampling of one element $x$from the general populations X and$Y$.

Theorem 2 [8]. If in a strong random experiment E samples $x = (x_1, ..., x_n) \in X$ and $y = (y_1, ..., y_m) \in Y$ have the same size, then the asymptotic significance level of the intervalfor the heterogeneity measure (2)does not exceed β.

3 Statistical properties of the family of homogeneity measures

We see that theoretically the heterogeneity measure (2) depends on the choice of the confidence interval I_{ij}. So, it is important to investigate the character of this dependence and find the options that could be considered equivalent. We consider eight methods for calculating confidence intervals binomial proportion: the Clopper-Pearson (I) interval, Bayes's interval (II), the Wilson interval with (III) and without (IV) corrections for continuity, the usual normal approximation with (V) and without (VI) corrections for continuity; methods based on arcsinetransformations, with two types of corrections (VII) and (VIII) [9].

In order to compare the statistics, samples from the following general categories were considered:

1. same variance and different means;

2. same mean and different variances;
3. both different means and variances.

Samples from a parametric family of hypothetical distribution with a sample from the general population with a reference distribution were compared.

4 Results

Comparison of homogeneity measures was performed on samples of 100and 300 elements. For the parameter, the step was set to 0.1. For each pair, ten experiments were conducted, and values were then averaged. Also, for the case of the second case, when we have the same mean and different variances, the value of the parameter was not considered.

To generate pseudorandom numbers with a normal distribution, a non-profit library Infer.NET was used.To generate pseudorandom numbers with a uniform distribution, a pseudo-random number generator, called the Mersenne twister, was implemented.

Analyzing the tables 1-6 we can conclude that changing the confidence intervals I_{ij} does not make significant changes to the final result. Let call two homogeneity measures equivalent if the first measure lies in the confidence interval for the second, and vice versa. To verify and search for equivalent homogeneity measures we constructed an indicator matrix. To construct the indicator matrices we considered only extreme points for a parameter, i.e. $\alpha = 0.0$ (or $\alpha = 0.1$) and$\alpha = 1.0$.

Among the considered 8 methods of constructing confidence intervals, 6 methods are equivalent. We also found two methods that violate the condition of equivalence. These are methods of normal approximation with and without continuity correction. Therefore, we excluded them from further consideration. Finally, we can see that the homogeneity measure (2)is more effective when examining the hypothesis of two samples drawn from populations with different means and the same variance.

We also consideredthemodification of the homogeneity measure (2) using several confidence intervals for averaging results.Supposethatweconstruct M equivalent confidence intervals

$$\forall \alpha, \beta = 1,...M, \ \ I_{ij,\alpha} \cong I_{ij,\beta}, \ \ \alpha \neq \beta,$$

where α, β are the label of the confidence interval (I-VIII) for binomial proportion. Denote by N the total number of all confidence intervals $I_{ij,\alpha}$, i.e. $N = \frac{n(n-1)}{2} \times M$. Let L be the total number of the intervals $I_{ij,\alpha}$, which contain $\frac{j-i}{n+1}$, i.e. $L = \sum_{\alpha} L_{\alpha}$, where L_a is the numberof the intervals constructed using the confidence interval with label α and containing $\frac{j-i}{n+1}$. Then, $h = \frac{L}{N} = \frac{1}{N} \sum_{\alpha} L_{\alpha}$.

Table 2. Homogeneity measure for $N(0,\alpha)$ when N = 100.

α	I	II	III	IV	V	VI	VII	VIII
0,1	0,23467	0,21986	0,23497	0,22582	0,14061	0,16048	0,21081	0,22994
0,2	0,32990	0,31224	0,32913	0,31667	0,23770	0,26022	0,30889	0,32483
0,3	0,42026	0,40162	0,42040	0,40495	0,32218	0,34972	0,40119	0,41624
0,4	0,57040	0,54612	0,56879	0,54814	0,47525	0,50681	0,55323	0,56576
0,5	0,57410	0,55129	0,57321	0,55408	0,49107	0,51885	0,55865	0,57024
0,6	0,70549	0,67873	0,70275	0,68103	0,62194	0,65279	0,69149	0,70206
0,7	0,80465	0,77370	0,79968	0,77562	0,74107	0,77028	0,79640	0,80299
0,8	0,82511	0,79828	0,82117	0,79992	0,75857	0,78766	0,81663	0,82321
0,9	0,84287	0,81655	0,83776	0,81889	0,77952	0,80503	0,83414	0,84117
1,0	0,90121	0,87689	0,89669	0,87879	0,84578	0,87010	0,89535	0,90038

Table 2. Homogeneity measure for $N(0,\alpha)$ when N = 300.

α	I	II	III	IV	V	VI	VII	VIII
0,1	0,10962	0,10591	0,11177	0,10807	0,07395	0,08206	0,10188	0,10963
0,2	0,16266	0,15811	0,16572	0,15972	0,12235	0,13311	0,15437	0,16256
0,3	0,21560	0,20973	0,21935	0,21112	0,17229	0,18485	0,20730	0,21542
0,4	0,32809	0,32015	0,33153	0,32131	0,28375	0,29857	0,32006	0,32789
0,5	0,41340	0,40416	0,41647	0,40514	0,36989	0,38639	0,40659	0,41357
0,6	0,45729	0,44697	0,45960	0,44807	0,41738	0,43420	0,45204	0,45789
0,7	0,68127	0,66868	0,68268	0,66963	0,64476	0,66438	0,67771	0,68231
0,8	0,72489	0,70834	0,72464	0,70940	0,69254	0,71346	0,72343	0,72660
0,9	0,75603	0,74024	0,75597	0,74135	0,72447	0,74386	0,75416	0,75753
1,0	0,82861	0,81437	0,82858	0,81544	0,79845	0,81787	0,82723	0,83018

Table 3. Homogeneity measure for $N(\alpha,1)$ when N = 100.

α	I	II	III	IV	V	VI	VII	VIII
0,1	0,41463	0,38974	0,41093	0,39497	0,34408	0,36693	0,39943	0,41212
0,2	0,42683	0,39929	0,42339	0,40352	0,35299	0,37741	0,41208	0,42360
0,3	0,60737	0,58030	0,60380	0,58327	0,53248	0,55885	0,59368	0,60412
0,4	0,59483	0,56788	0,59166	0,57087	0,51596	0,54558	0,58170	0,59186
0,5	0,56804	0,54030	0,56376	0,54327	0,49164	0,52238	0,55497	0,56515
0,6	0,67869	0,64974	0,67602	0,65212	0,59386	0,62487	0,66352	0,67499
0,7	0,71760	0,68398	0,71259	0,68687	0,64149	0,67380	0,70586	0,71507
0,8	0,80267	0,76927	0,79737	0,77131	0,74026	0,76743	0,79408	0,80048
0,9	0,83911	0,81208	0,83529	0,81366	0,77533	0,80347	0,83085	0,83735
1,0	0,83042	0,80317	0,82596	0,80531	0,77259	0,79691	0,82297	0,82851

Table 4. Homogeneity measure for $N(\alpha,1)$ when N = 300.

α	I	II	III	IV	V	VI	VII	VIII
0,1	0,22493	0,21780	0,22580	0,21965	0,19596	0,20606	0,21985	0,22544
0,2	0,28522	0,27752	0,28714	0,27900	0,25025	0,26278	0,27926	0,28555
0,3	0,33584	0,32470	0,33647	0,32633	0,30455	0,31928	0,33160	0,33666
0,4	0,38989	0,38002	0,39074	0,38144	0,35926	0,37331	0,38574	0,39065
0,5	0,41468	0,40315	0,41582	0,40460	0,38131	0,39783	0,41065	0,41556
0,6	0,47984	0,46601	0,48083	0,46710	0,44600	0,46423	0,47619	0,48105
0,7	0,60955	0,59447	0,60942	0,59558	0,57970	0,59854	0,60788	0,61119
0,8	0,81637	0,80135	0,81628	0,80236	0,78559	0,80369	0,81444	0,81788
0,9	0,81208	0,79776	0,81255	0,79882	0,78048	0,79968	0,81016	0,81347
1,0	0,83792	0,82386	0,83787	0,82496	0,80835	0,82670	0,83655	0,83945

Table 5. Homogeneity measure for $\alpha U(0,1)+(1-\alpha)U(1/2,3/2)$ when N = 100.

α	I	II	III	IV	V	VI	VII	VIII
0,1	0,33384	0,31065	0,33210	0,32059	0,27226	0,28580	0,31945	0,33224
0,2	0,37162	0,35073	0,37018	0,35895	0,31293	0,32758	0,35911	0,37040
0,3	0,43317	0,41095	0,43055	0,41798	0,37913	0,39642	0,42329	0,43226
0,4	0,45727	0,43412	0,45461	0,44020	0,40125	0,41954	0,44754	0,45610
0,5	0,49109	0,46933	0,48832	0,47483	0,44113	0,45727	0,48331	0,49014
0,6	0,60242	0,57725	0,59873	0,58091	0,55374	0,57313	0,59610	0,60170
0,7	0,76356	0,73976	0,76238	0,74244	0,71293	0,73014	0,75638	0,76222
0,8	0,89636	0,88372	0,89640	0,88554	0,85166	0,86414	0,89117	0,89531
0,9	0,99176	0,98970	0,99263	0,99020	0,95562	0,96277	0,98885	0,99115
1,0	1,00000	1,00000	1,00000	1,00000	0,98000	0,98000	1,00000	1,00000

Table 6. Homogeneity measure for $\alpha U(0,1)+(1-\alpha)U(1/2,3/2)$ when N = 300.

α	I	II	III	IV	V	VI	VII	VIII
0,1	0,28141	0,27384	0,28275	0,27719	0,25365	0,26129	0,27608	0,28167
0,2	0,30042	0,29154	0,30056	0,29458	0,27785	0,28619	0,29693	0,30103
0,3	0,30866	0,29983	0,30924	0,30254	0,28365	0,29291	0,30488	0,30908
0,4	0,37128	0,36309	0,37155	0,36542	0,34930	0,35783	0,36830	0,37180
0,5	0,39009	0,38100	0,38919	0,38303	0,37516	0,38404	0,38955	0,39126
0,6	0,46043	0,44959	0,45887	0,45140	0,44637	0,45706	0,46118	0,46197
0,7	0,49939	0,48703	0,49738	0,48846	0,48648	0,49857	0,50122	0,50110
0,8	0,70976	0,69581	0,70833	0,69665	0,69251	0,70718	0,71109	0,71106
0,9	0,96863	0,96684	0,97028	0,96703	0,95169	0,95639	0,96669	0,96856
1,0	1,00000	1,00000	1,00000	1,00000	0,99333	0,99333	1,00000	1,00000

In the computation of the compound heterogeneity measure, we used the same methods, as in the computation of the usual heterogeneity measure (see Section 3). Experiments with the same samples have shown that the using of several confidence intervals at once does not violate the stability of the homogeneity measure, and we can state that such method leads to better results which could be considered as an alternative for the methods of reproducing samples when size of samples is small.

5 Conclusion

We have found that family of homogeneity measures based on A(n) assumption is effective both for small and large samples. The p-statistics and compound heterogeneity measure constructed using a set of different confidence interval for binomial proportion practically independent of the interval selection (except for the methods of normal approximation). These similarity measures may be considered as an effective alternative to the Wilcoxon–Mann–Whitney test in machine learning applications connected with comparisons of distributions.

6 Acknowledgements

The authors would like to express their very great appreciation to Alexei

Kononenko for his valuable and constructive suggestions.

References

1. Granichin, O., Kizhaeva, N., Shalymov, D., Volkovich, Z.: Writing style determination using the KNNtext model. In: Proceedings of the 2015 IEEE International Symposium on Intelligent Control, pp. 900–905. IEEE, Sydney (2015).
2. Zenkov, A., Sazanova, L.: A New Stylometry Method Basing on the Numerals Statistic. International Journal of Data Science and Technology 3(2), 16-23 (2017).
3. Kilgariff,A.: Comparing corpora. International Journal of Corpus Linguistics 6(1):97–133 (2001).
4. Kilgariff, A.: Language is never, ever, ever, random.Corpus Linguistics and Linguistic Theory, 1(2): 263–276(2005).
5. Eder, M., Piasecki, M.,Walkowiak, T.: An open stylometric system based on multilevel text analysis. Cognitive Studies | Études cognitives, 17 (2017).
6. Eder, M., Rybicki, J., Kestemont, M.: Stylometry with R: a package for computational text analysis. R Journal 8(1): 107–121(2016).
7. Hill, B.: Posterior distribution of percentiles: Bayes' theorem for sampling from a population. Journal of American Statistical Association 63(322): 677−691 (1968).
8. Klyushin, D., Petunin, Yu.: A Nonparametric Test for the Equivalence of Populations Based on a Measure of Proximity of Samples. Ukrainian Mathematical Journal, 55 (2): 181-198(2003).
9. Pires A.: Confidence intervals for a binomial proportion: comparison of methods and software evaluation. In: Klinke, S., Ahrend, P., Richter, L. (eds.) Proceedings of the Conference CompStat 2002, Short Communications and Posters(2002).

Intelligence Knowledge-basedSystem Based on Multilingual Dictionaries

Oleksii Puzik[0000-0003-1460-1686]

Kharkiv National University of Radio Electronics
14, Naukyave., 61000, Kharkiv, Ukraine

alphabet308@gmail.com

Abstract. Intelligence knowledge-based systems are important part of natural language processing researches. Appropriate formal models simplify developing of such systems and open new ways to improve their quality. This work is devoted to developing of intelligence knowledge-based system using model based on algebra of finite predicates. The model also isbased on lexicographical computer system which consists of trilingual and explanatory dictionaries. Algebra of finite predicates is used as formalization tool.Problems of distinguishing semantic entities is investigated during research. Method of resolving homonymy ambiguities is used to extract separate entities, thus allowing formalization of semantic relationships. In result formal model of intelligence knowledge-based system was developed.It was shown way to extend the model for different languages.

Keywords: Natural Language Processing, Intelligence Knowledge-based Systems, Algebra of Finite Predicates, AFP, Lexicographical System

1 Introduction

Natural language modeling is important partof the theory of intelligence. The main objects of the language description are words and relations between them.The question is understanding texts written in natural language, create mathematical models for solving linguistic problems and developing programs that function based on these models. The construction of intelligence knowledge-based systems requires the automation of both the syntax of the natural language and its semantics. This part of computational linguistics relates to the section of artificial intelligence, engaged in the development of natural language word processing systems.This problem despite the different approaches to the formalization of semantic problems is still not completely solved because of lack of formalization while describing semantic relations.

This paper proposes formal description of a data model based on algebra of finite predicates (AFP) [1].Thesystem can be used as a part of intelligence system which is based on trilingual terminological dictionary on computer science and radio electronics.

Also, in the study considered formalization of the problem of homonymy ambiguation. There are exist many different approaches [2, 3] regarding multiple-valued structures. Suggested approach allows to resolve homonymy ambiguation by adding explanatory dictionaries to semantic concepts and calculating relevance score based on those explanations.

2 Trilingual dictionary model

Let's assume model of trilingual dictionary as a set of words in different languages for certain knowledge area. In this study we use Ukrainian-Russian-English dictionary of radio electronics and informatics. We need to group all these words by word meanings to build intelligence knowledge-based system. In general case it is hard to achieve this because we have homonymy ambiguity. The classic example is words "*spring"*in English or "*коса"* in Russian and Ukrainianwhich have few meanings. Thus, if we build intelligence system based only on spelling of these words, we can't distinguish semantic. Moreover, if need to get translations we face with issue that invalid translation equivalents pair can be matched. Let's introduce additional semantic metalanguage which has exactly one definition for each semantic conception.In this case we can groupwords from different languages using these abstract words from the semantic metalanguage. Graphical representation is displayed in Fig. 1.

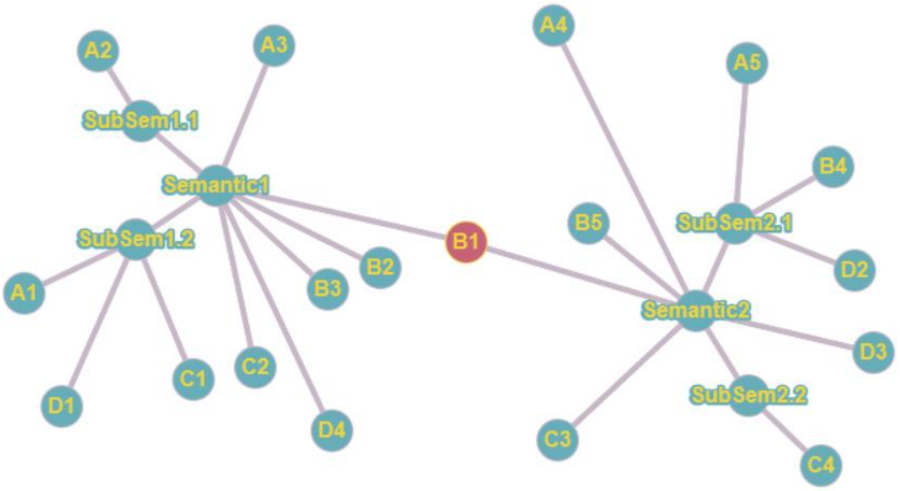


Fig. 3. Graphical representation of words in multilingual dictionary with metalanguage

In this figure Ai, Bi, Ci, Di are some words from different languages where letter identifies some language and index corresponds to i-th word in the language introduced in the dictionary, Semantici – is a semantic concept which connects corresponding words, Semantici, j – is a subsemantic concept which emphasizes semantic features specific to some language.

Thus, we create additional level of indirection. This way used widely in software engineering. For instance, it can be used for machine translations where no direct dictionary between languages but exist two or more dictionaries with common language used in both [4].

Additional feature of such metalanguage is that it is possible to attach explanatory dictionary to the metalanguage entity node and it will have just only one the most appropriate explanation per language per metaword. Also, we can introduce additional features like subsemantics or shadows of meaning attached to main semantic thus we can introduce more specific translations for each language.

Let D – set that represents some dictionary, W – set of all words for all languages, S – set of semantic concepts.

$$D = S_1 \ \vee S_2 \ \vee \ldots \vee S_n$$

Let $w_{S_i} \subseteq$ W – subset of words that represents certain semantic conception S$_i$

$$S_i = w_i^{sem} \vee w_i^{ru} \vee w_{i1}^{ru} \vee w_i^{ua} \vee w_i^{en} \ ,$$

where w$_i$ – words related for certain semantic conception S$_i$for Semantic metalanguage, Russian, Ukrainian and English accordingly. Also, these words may include synonymy words like w_{i1}^{ru} because despite different spelling they represent the same semantic concept.

The following predicate evaluates the equivalence for any two words in the dictionary:

$$P_e(x \ \ ,y \ \ ) = \begin{cases} 1, \ (x \ \ ,y \ \ ) \in S_i \\ 0, \ (x \ \ ,y \ \ ) \notin S_i \end{cases}$$

At the same time $w_i^{sem} \in S_i$ by its definition. So, we can write predicate which introduces translations and synonymy in our dictionary:

$$P_e(x \ \ ,y \ \ ) = P_e(x \ \ ,w_i^{sem}) \wedge P_e(y, w_i^{sem})$$

Equations written above have one issue – they should work on sets of words without intersections, but natural languages mostly have some homonymouswords. Some such word is shown as B1 node in the Fig. 1. Thus, we need to find out method to separate homonymous words. So, in result we need to get following:

$$S_{spring1} = w_{spring1}^{sem} \vee w_{\text{пружина}}^{ru} \vee w_{\text{пружина}}^{ua} \vee w_{spring1}^{en}$$

$$S_{spring2} = w_{spring2}^{sem} \vee w_{\text{весна}}^{ru} \vee w_{\text{весна}}^{ua} \vee w_{spring2}^{en}$$

However, in general case if we just look for translation of separate word, we can get list of possible semantic conceptsof that word. The list can be used to get translations specific for certain language.

3 Homonymy disambiguation

In real world texts we don't have such specific marks like spring1. Moreover, even they exist they will not match our marks in our dictionary. Above we mentioned that semantic nodes may be attached with explanatory nodes. In this case we may consider these explanatory nodes as the same logical entities as word nodes.

$$S_i = w_i^{sem} \vee w_i^{ru} \vee w_{i1}^{ru} \vee w_i^{ua} \vee w_i^{en} \vee x_i^{ru} \vee x_i^{ua} \vee x_i^{en},$$

where w_i – words related for certain semantic conception S_i for Semantic metalanguage, Russian, Ukrainian and English accordingly,x_i – explanations related to certain semantic conception S_i for Russian, Ukrainian and English accordingly.Also, these explanation nodes will help us calculate relevance score for homonymous words.

Homonymy disambiguation is not possible without considering context where homonymous word is used.One of principles which must be used when defining explanations for each sematic concept is that all word used in explanation must be present in the dictionary [5]. We can useapproach described in [6] to select homonym the most relevantto context. In few words, relevance score is calculated for each word used in the context regarding processed word.

Let define context function C(*x, ctx*) which calculatesrelevancescore, where *x* – is some word and *ctx* – is the context where the word *x*is used. Let E(*x*) function which calculates relevance score based on explanations used in the dictionary.Thus we can define function

$$G_i(x_i, y, ctx) = |E(x_i) - C(y, ctx)|,$$

where x_i– word with potential semantic connected to*y*, *y* – word which semantic should be determined, *ctx*–context where word *y* is used.

$$F(y, ctx) = \min_{i=1,n} G_i(x_i, y, ctx)$$

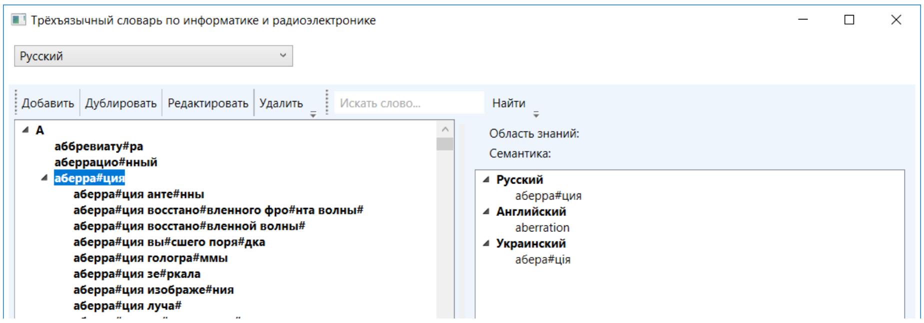
where$F(y, ctx)$is a function which calculatesthe most relevant semantic concept for word *y* based on context where the word is used.

Cognate languages may have similar words but with different semantic. The suggested method may be extended by including explanatory dictionaries for different languages into intelligence knowledge-based system.This allows avoiding such kind of ambiguities.

4 Trilingual electronic dictionary

Based on suggested approach electronic trilingual dictionary of informatics and radioelectronics was developed. The dictionary can be usedby the intelligence knowledge-based systemas a semantic database.

The software system of the trilingual electronic dictionary is based on the paper version of the Russian-Ukrainian terminological dictionary on informatics and radio electronics. An electronic version of the text was obtained by scanning and recognition, which serves as the basis for filling the dictionary with content. The electronic version of the dictionary allows user to add translation equivalents of terms for the English language and proceed to the multilingual dictionary on further development.



А
аббревиату#ра
аберрацио#нный
аберра#ция
аберра#ция анте#нны
аберра#ция восстано#вленного фро#нта волны#
аберра#ция восстано#вленной волны#
аберра#ция вы#сшего поря#дка
аберра#ция гологра#ммы
аберра#ция зе#ркала
аберра#ция изображе#ния
аберра#ция луча#

Русский
аберра#ция
Английский
aberration
Украинский
абера#ція

Fig. 4. Trilingual dictionary UI

The software system of the trilingual dictionary is written in the C# programming language. The architecture is built using the MVVM design pattern (Model-View-ViewModel Model-View-View Model). Thus the software system is clearly divided into separate independent modules. This approach allows reusing and independent changing of individual components of the software system. The user interface is created using WPF technology [7].

The software system of the trilingual electronic dictionary allows user to view, search, edit, add, delete concepts of terms and their translation equivalents for each of languages. The advantage of the proposed approach is the possibility of free switching between the languages of the dictionary and quick access to all translation equivalents of the selected term which corresponds to certain conception semantic.

5 Conclusions and future work

In results was created software system of trilingual terminological Ukrainian-Russian-English dictionary of radio electronics and informatics. The developed model allows quick recognition of synonymous words and translation equivalents. In the study was considered homonymy ambiguities problem and suggested ways to resolve the issue. Suggested model allows extending intelligence system with new languages by not only adding new terms but explanations and specific semantic concepts as well. Intelligence knowledge-based system will use this developed electronic dictionary as a database of low-level terms.

Further development will integrate explanatory dictionary to the intelligence system. Thus, context for each word will be defined and this allows matching context from arbitrary text with semantic concepts stored in dictionary based on suggested approach for homonymy disambiguation.

Long-term future work should consider includingmodules related to processing blocks of texts with extracting complex semantic entitieswhich represent more abstract concepts.

References

1. Bondarenko, M., Shabanov-Kushnarenko, Y.:Teoriyaintellekta: ucheb. Izd-vo SMIT, Kharkov (2006).
2. Chetverikov, G., Vechirska, I.,Tanyanskiy, S.: The methods of algebra of finite predicates in the intellectual system of complex calculations of telecommunication companies. In:2014 24th International Crimean Conference Microwave & Telecommunication Technology, pp. 346-347, Sevastopol (2014).
3. Chetverikov, G., Puzik, O.,Vechirska, I.: Multiple-valued structures of intellectual systems. In: 2016 XIth International Scientific and Technical Conference Computer Sciences and Information Technologies (CSIT), pp. 204-207, Lviv (2016).
4. Johnson, M., Schuster, M., Le, Q.V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F.B., Wattenberg, M., Corrado, G.S., Hughes, M., Dean, J.: Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation., Transactions of the Association for Computational Linguistics, vol. 5, 339-351, https://www.transacl.org/ojs/index.php/tacl/article/view/1081, last accessed 2019/03/20.
5. Shirokov, V., Computer lexicography. Scientific and publishing enterprise "Vidavnitstvo"Naukovadumka" NAN Ukrainy",Kyiv(2011).
6. Chetverikov G., Puzik, O.,Tyshchenko,O.: Analysis of the Problem of Homonyms in the Hyperchains Construction for Lexical Units of Natural Language. In: 2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT), pp. 356-359, Lviv (2018).
7. Chetverikov, G., Vechirska, I., Puzik, O.:Technical Features of the Architecture of an Electronic Trilingual Dictionary. Cognitive studies (16), 143-152 (2016).

STUDENT SECTION

Study of Software Systems Usability Used for Customers Loyalty Identification

Mariia Bilova[0000-0001-7002-4698] and
Oleksandr Trehubenko[0000-0002-6050-322X]

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

missalchem@gmail.com, s219@tmm-sapr.org

Abstract. On the background of software (SW) increase in quantity and complexity and SW versions change, a friendly interface allows enhancing SW competitiveness, reduction in SW development costs, increase in SW users number and users satisfaction, as well as reduction in costs needed for users training and support. The product using which users achieve the goals set and solve various issues in an efficient way, is deemed to be a user-friendly software product.The purpose of the article is to study existing methods for assessing the application usability and analyzing the features of using the main software usability indicators on the example of software for customer loyalty of 'Infotech' consumer society.

Keywords: Software, Usability, Program Interface, CRM-system, Loyalty, Customer Database, Segmentation, RFM Method, Forms, SUS Form, Usability Testing, Main Factors Of SW Usability.

1 Introduction

On the background of software (SW) increase in quantity and complexity and SW versions change, a friendly interface allows enhancing SW competitiveness, reduction in SW development costs, increase in SW users number and users satisfaction, as well as reduction in costs needed for users training and support. The product using which users achieve the goals set and solve various issues in an efficient way, is deemed to be a user-friendly software product.

The study of SW usability acquires particular relevance when designing and applying customer loyalty identification systems, which, due to the features of their intended use usually have a very awkward interface: a large amount of data; entering output data for calculating customer loyalty and generalization of data received via tables, diagrams, charts, etc. The need to collect and analyze respondents' answers, work with a large number of judgments and statements, the probability of error in the calculation of loyalty indices, and a complicated mechanism for comparing the study results which complicates the design and development of the relevant software, are characteristic of various of loyalty assessment methods.

The purpose of the article is to study existing methods for assessing the application usability and analyzing the features of using the main software usability indicators on the example of software for customer loyalty of 'Infotech' consumer society.

2 Recent research and publications analysis

The term of CRM (Customer Relationship Management System) denotes the system for customers relations management. This approach means that when dealing with a customer, the company employee has access to all the necessary information about the relationships with a particular customer and the decision is taken based on this information [1]. Specialists highlight a number of disadvantages of their use or hidden threats in the scientific literature and analysis of CRM-systems practical use. They distinguish three main reasons for the failure of CRM-systems introduction:

- technological reasons (difficulties in ensuring the security of customer's personal data; data export and data exchange with other software failures; long-term development process);
- functional reasons (unfriendly to use for specific tasks; too many functions of redundant nature; unreadiness of staff to encounter technical difficulties in the system introduction);
- organizational reasons (low tolerance on the part of the staff due to the lack of awareness of the benefits; the lack of staff desire to use all available tools; the need of staff and management instruction) [2, 3, 4].

The CRM system helps to optimize and expand customer interaction, while accumulating information about their purchases and interests. This allows further formation more personalized propositions for each particular buyer, which increases the chance of purchasing and tracks customers loyalty features.

The term of *usability* denotes friendliness of use. This is a combination of software features to ensure a predictable users range that reflects its ease and adaptation to changing conditions, operation, operation stability and data preparation, the results clarity, the convenience of introducing changes to software documentation and SW [5, 6].

The software usability can be estimated using a set of characteristics.

1. Efficiency (effectiveness) – positive dynamics in solving the goals set by the user when using a program or a site.

2. Productivity – the ratio of time and resources spent on work with the program, to the efficiency of the *usability* methods used.

3. Satisfaction – users portal assessment in terms of use and solving the tasks set [7]. The users satisfaction is determined by standardized surveys.

Advantages and disadvantages of standardized surveys are given in table 1.

Table 1. Advantages and disadvantages of standardized surveys

No. seriatim	*Method name*	*Advantages*	*Disadvantages*
1.	*ASQ: after scenario polling (3 questions)*	*- speed and low budget;* *- a questionnaire allows detection features and factors of influencing on SW usability and correction of customers retention policy in a relevant way;* *- to assess SW usability many respondents are employed and there are many scenarios of SW testing.*	*A small number of judgments*
2.	*NASA-TLX: NASA load index – an index of mental efforts (6 questions)*	*A tool for work load assessment which allows users meet requirements to the work loads for operators who work with different systems of humane-machine interface.*	*A small number of judgments*
3.	*SEQ: one simple questions (1 question)*	*Speed and low budget*	*- contains only one judgment but this is insufficient for SW usability testing;* *- no sense while using a questionnaire with one judgment only.*
4.	*SUS (usability system scale; 10 questions)*	*- SUS questionnaire advantage is promptness and low budget;* *- SUS questionnaire contains a sufficient number of judgments;* *- SUS questionnaire is designed for SW usability testing.*	
5.	*SUPR-Q (standardized survey; 8 questions)*	*The questionnaire measures basic aspects of web-sites (usability, reliability, loyalty and web-site appearance)*	*- the questionnaire allows detection of specific problems in web-site interface;* *- the questionnaire results are better used together with feedback on a web-site (comments on the web-site).*
6.	*CSUQ (a questionnaire to SW usability; 19 questions)*	*- CSUQ questionnaire is designed for SW usability testing;* *- the questionnaire contains a sufficient number of judgments and covers different elements of SW interface.*	*The questionnaire processing time*
7.	*QUIS (the questionnaire for satisfaction of interaction with a user; 24 questions)*	*- design or systems reorganization management;* *- providing managers with tools for assessment of the system perfection potential regions;* *providing researches with a tested tools for carrying out comparative assessments.*	*- the questionnaire results processing time;* *- a large number of questions.*
8.	*SUMI (a indices of SW usability use; 50 questions)*	*SUMI questionnaire measures users satisfaction: effectiveness; emotional reaction; ability to help; control; comprehensibility.*	*- the questionnaire results processing time;* *- a large number of questions.*

3 Testing usability of customers loyalty assessment system

Testing usability of customers loyalty assessment system. Usability testing implies: test objectives and tasks determining; application environment description; generalization of data obtained; inferencing of conclusions and recommendations.

Diagram of usability test activity is given in figure 1.

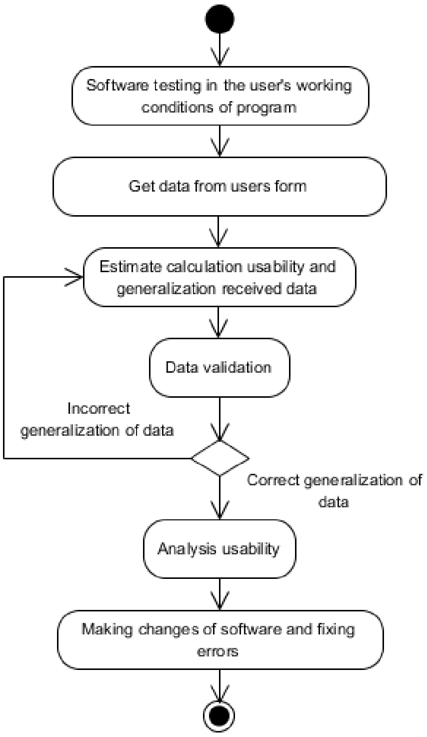

Software testing in the user's working conditions of program
Get data from users form
Estimate calculation usability and generalization received data
Data validation
Incorrect generalization of data
Correct generalization of data
Analysis usability
Making changes of software and fixing errors

Fig. 1. Activity diagram for usability testing

Diagram of application options is given in figure 2.

The system of 'Infotech' consumer society's customer loyalty assessment is chosen as the subject of research.

Purpose and objectives of usability analysis. The software product for 'Infotech' consumer society's customer loyalty assessment shall be simple and usable. The purpose of testing is to check SW uasbility in predictable working conditions. The objectives of testing:

- to study application environment, to define customers needs and all persons concerned;
- to study users, define their needs;
- remove errors associated with a user's interface;
- to estimate the level of customers loyalty assessment SW usability;
- to define possible ways of customers loyalty assessment SW improvement.

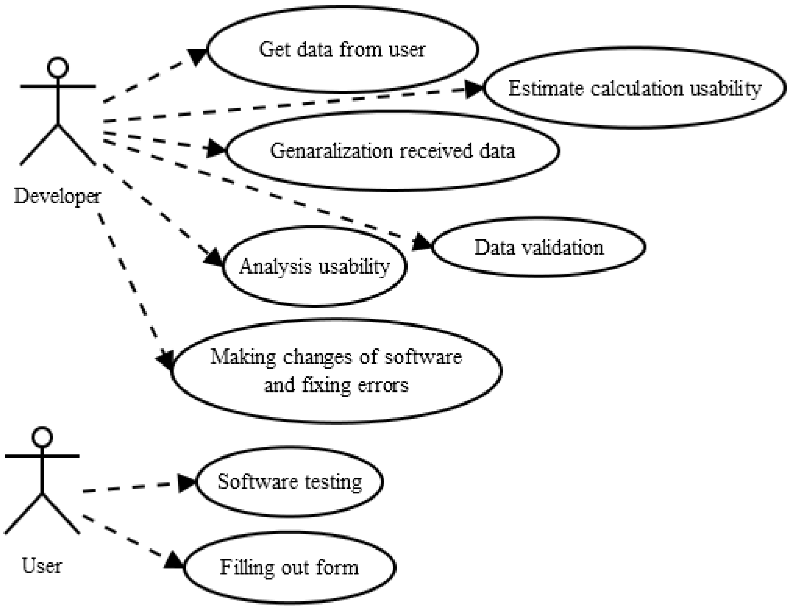

Get data from user
Estimate calculation usability
Genaralization received data
Developer
Data validation
Analysis usability
Making changes of software and fixing errors
Software testing
User
Filling out form

Fig. 2. Precedents diagram

Application environment description. Application environment is the activity of 'Infotech' consumer society enterprise which is associated with customers servicing. The aim of the society is to increase the number of loyal customers number which would result in profitability increase and attraction of new customers.

'Infotech' consumer society is a legal entity acting on the basis of charter [8]. Activity type: 62.03 Computer equipment management. 'Infotech' consumer society renders the following services:

– introduction and support of licensed software;
– system servicing;
– mounting, installation, servicing [9].

Software product based on RFM method use designed for the enterprise operation automation.

The program system is implemented through the Internet store site using which, the customers can order products, and loyalty assessment software application. Information about each customer's orders is stored in the database. The software application ensures automation for data processing solutions with regard to the enterprise customers loyalty.

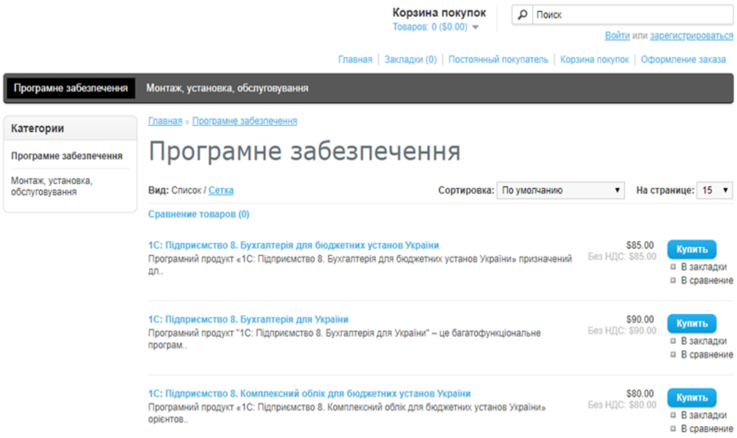
Functional requirements for software are given below.

1. Internet store (the site servicing in the administrator panel; search by categories; review of stock line; authorization; login; choice of product; checkout).

2. Software application (authorization; calculation of customers loyalty; customer's data removal; generalization of data obtained).

3. Non-functional requirements are reliability, usability, safety, expandability.

Site is given in figure 3.


Корзина покупок
Товаров: 0 ($0.00)
Поиск
Войти или зарегистрироваться
Главная | Закладки (0) | Постоянный покупатель | Корзина покупок | Оформление заказа
Програмне забезпечення
Монтаж, установка, обслуговування
Категории
Програмне забезпечення
Монтаж, установка, обслуговування
Главная » Програмне забезпечення

Програмне забезпечення

Вид: Список / Сетка   Сортировка: По умолчанию   На странице: 15

Сравнение товаров (0)

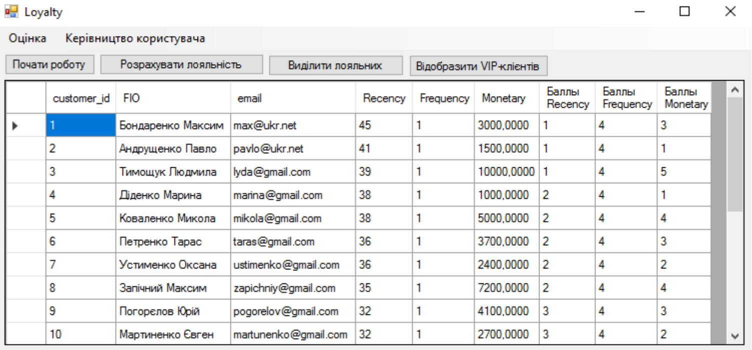
1С: Підприємство 8. Бухгалтерія для бюджетних установ України
Програмний продукт «1С: Підприємство 8. Бухгалтерія для бюджетних установ України» призначений дл..
$85.00
Без НДС: $85.00
Купить
□ В закладки
□ В сравнение

1С: Підприємство 8. Бухгалтерія для України
Програмний продукт "1С: Підприємство 8. Бухгалтерія для України" – це багатофункціональне програм..
$90.00
Без НДС: $90.00
Купить
□ В закладки
□ В сравнение

1С: Підприємство 8. Комплексний облік для бюджетних установ України
Програмний продукт «1С: Підприємство 8. Комплексний облік для бюджетних установ України» орієнтов..
$80.00
Без НДС: $80.00
Купить
□ В закладки
□ В сравнение

Fig. 3. Site

Software is given in figure 4.



customer_id	FIO	email	Recency	Frequency	Monetary	Баллы Recency	Баллы Frequency	Баллы Monetary
1	Бондаренко Максим	max@ukr.net	45	1	3000,0000	1	4	3
2	Андрущенко Павло	pavlo@ukr.net	41	1	1500,0000	1	4	1
3	Тимощук Людмила	lyda@gmail.com	39	1	10000,0000	1	4	5
4	Діденко Марина	marina@gmail.com	38	1	1000,0000	2	4	1
5	Коваленко Микола	mikola@gmail.com	38	1	5000,0000	2	4	4
6	Петренко Тарас	taras@gmail.com	36	1	3700,0000	2	4	3
7	Устименко Оксана	ustimenko@gmail.com	36	1	2400,0000	2	4	2
8	Запічний Максим	zapichniy@gmail.com	35	1	7200,0000	2	4	4
9	Погорелов Юрій	pogorelov@gmail.com	32	1	4100,0000	3	4	3
10	Мартиненко Євген	martunenko@gmail.com	32	1	2700,0000	3	4	2

Fig. 4. Software

Generalization of data obtained. During SW usability testing, 15 employees of 'Infotech' consumer society were proposed to perform certain actions with SW with respect to scenarios.

SUS questionnaire was chosen for testing, since:

- SUS questionnaire advantage is promptness and low budget;
- SUS questionnaire contains a sufficient number of judgments;
- SUS questionnaire is designed for SW usability testing.

SUS questionnaire was compiled after Likert scale principle. The questionnaire has five grade scale from 1 ('strongly disagree') to 5 ('strongly agree') [10].

Algorithm of SUS estimate calculation for each respondent is given below.

1. For questions with odd numbers 1 is deducted from a user's answer.
2. For questions with pair numbers 5 is deducted from a user's answer.
3. All estimates are within an interval from 0 to 4, where 4 – a positive answer.
4. The numbers obtained are summarized and multiplied by 2.5.

SUS estimate is within the interval from 0 to 100 but it should not be confused with per cents. Average is deemed to be an interface which scored 68 points (approximately 50%) and excellent one – 85 and higher. The analysis of testing results of 'Infotech' consumer society customers' loyalty software usability assessment is given in table 2.

Table 2. Analysis of usability assessment as per SUS questionnaire

Respondents	Respondent answers to 10 questions of quest.										SUS assessment
	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	
worker 1	4.	1.	4.	3.	4.	1.	5.	4.	5.	1.	80.
worker 2	5.	1.	4.	3.	4.	3.	4.	5.	5.	2.	70.
worker 3	3.	1.	4.	3.	4.	2.	4.	5.	5.	1.	70.
worker 4	5.	1.	5.	2.	3.	2.	5.	5.	4.	1.	77.5.
worker 5	4.	2.	5.	3.	4.	1.	4.	4.	5.	1.	77.5.
worker 6	5.	3.	4.	1.	4.	1.	5.	4.	4.	2.	77.5.
worker 7	4.	2.	3.	3.	5.	1.	4.	5.	5.	2.	70.
worker 8	5.	2.	5.	4.	4.	2.	5.	3.	5.	3.	75.
worker 9	5.	1.	5.	4.	5.	2.	4.	5.	4.	1.	75.
worker 10	4.	2.	4.	3.	4.	2.	4.	4.	5.	3.	67.5.
worker 11	3.	3.	5.	3.	5.	2.	5.	4.	4.	4.	65.
worker 12	4.	2.	4.	5.	4.	2.	5.	5.	5.	2.	65.
worker 13	5.	2.	3.	5.	3.	1.	4.	4.	4.	3.	60.
worker 14	4.	1.	3.	5.	3.	2.	5.	4.	5.	1.	67.5.
worker 15	3.	2.	2.	4.	3.	3.	4.	5.	4.	1.	52.5.
Average value of SUS assessment											70.

4 Analysis of results obtained

In accordance with analyzed questionnaires of 'Infotech' consumer society employees, average value of SUS assessment equals to 70. The majority of respondents find the SW under study awkward to use. Additional survey detected such faults of the system:

- impossibility of reviewing data on customers orders;
- the software product user cannot create new customer groups by their loyalty level;
- absence of possibility to edit available loyalty groups (work with lists of loyal and VIP customers);
- the software product user cannot edit customers contact information;
- absence of an additional field for adding comments;
- the necessity of saving assessment results to a separate file while specifying generation date.

Some employees indicate that usability is influenced by the absence of possibility of interface customizing (change in font size and theme color). The analysis of usability testing results allows making inference on the need to improve interface of customers loyalty assessment software. Feedback with the program developer is also planned to be made in the software.

5 Conclusions

During the article preparation, the selection, processing and systematization of scientific software development literature from the perspective of its usability, was carried out. The results of the work should be considered as follows.

1. The study of the content and usability characteristic features which indicates usability, was performed.

2. It was noted that the most popular methods of dara collection for users study, are questionnaires and interviews. ASQ, NASA-TLX, SEQ, SUS, SUPR-Q, CSUQ, QUIS, SUMI questionnaires are used to assess SW usability.

3. Features of SW main usability indices were analyzed. It was determined that SW usability can be assessed using a set of such characteristics as effectiveness, productivity, and satisfaction.

4. The conclusion was made that the best way to assess the SW usability is to perform usability testing in order to test the software in the intended operating conditions.

5. The usability of 'Infotech' consumer society customers loyalty assessment system was studied with the use of SUS questionnaire, which resulted in the assessment score of 70. The conclusion was made on the necessity of SW improving due to the possibility of reviewing the data on the customers orders; additional field for adding comments; possibilities of interface customization (changing font sizes and color themes), etc. The analysis of usability testing results allows making inference on the need to improve interface of customers loyalty assessment software.

References

1. Kaverina, I.: Analysis of the existing methods for customer database management to enhance competitiveness of a drug store network. Bulletin of Siberian medicine, vol. 4, 172–176 (2014).
2. Informational web site LIFE-PROG.ru. http://life-prog.ru/1_33645_preimushchestva-i-nedostatki-CRM-sistem.html.
3. Ryazantsev, A.: How to introduce CRM-system for 50 days.Omega-L Publishing House, series 1000 bestsellers (2017).
4. Kinzyabulatov, R.: CRM. In detail and to the point. Publishing solutions Publishing House (2016).
5. Isakov, O., Cherednichenko, O., Mozgin, V., Yangolenko, O.: Study of software products usability testing processes models. The National Technical University Bulletin XIII. Series: Strategic management, portfolio, programs and projects management, vol. 2 (1278), 73–80 (2018).
6. Kirilenko, O., Kuznetsova,Yu.,Sokolova, E., Frolova, G.: Procedure for user interface usability testing.http://ena.lp.edu.ua.
7. Prokhorova, A.: The concept of site usability: design indices and standards.Economics and Law, vol. 9 (67), 87–90 (2016).
8. Official site of Ukraine VerkhovnaRada. http://zakon3.rada.gov.ua/laws/show/436-15.
9. Informational web site of Infotech consumer society.https://www.ua-region.com.ua/38667722.

10. Brooke, J.: SUS: A Retrospective. Journal of usability studies, vol. 8 (2), 29–40 (2013),https://usabilitygeek.com/how-to-use-the-system-usability-scale-sus-to-evaluate-the usability-of-your-website/.

Automated Building and Analysis of Ukrainian Twitter Corpus for Toxic Text Detection

Kateryna Bobrovnyk[0000-0003-3358-1035]

Taras Shevchenko National University of Kyiv, Istitute of Philology
Taras Shevchenko Blvd, 14, Kyiv, Ukraine

katherine.bobrovnik@gmail.com

Abstract. Toxic text detection is an emerging area of study in Inter-net linguistics and corpus linguistics. The relevance of the topic can be explained by the lack of Ukrainian social media text corpora that are publicly available. Research involves building of the Ukrainian Twitter corpus by means of scraping; collective annotation of 'toxic/non-toxic' texts; construction of the obscene words dictionary for future feature engineering; and models training for the task of text classification (com-paring Logistic Regression, Support Vector Machine, and Deep Neural Network).

Keywords: toxic text detection, text corpus, Twitter.

The purpose of this study is to create a Ukrainian text corpus based on posts from Twitter and to perform toxic text detection on it. This area of NLP is relatively new so there are few works concerning this topic [1, 2]. The scope of the work is to create a dictionary of Ukrainian obscene words based on posts from Twitter and to train a toxic text classier using methods of Machine Learning.

Text corpus is a central notion of corpus linguistics. Corpora play an essential role in Natural Language Processing (NLP) research as well as a wide range of linguistic investigations: sentiment analysis, topic modeling, machine translation etc. They provide a material basis and a test bed for building NLP systems. There are thousands of corpora in the world, but most of them are created for specic research projects[3] for a particular language and may not be publicly available.

The first stage of the research is to scrape Twitter posts of hand-picked users with Ukrainian tweets. Scraping was based on Kenneth Reitz's library[4]. The resulting corpus consists of 1.87 million tweets with additional meta-information about time, language, replies, retweets, likes, hashtags, URLs and author nick-names.

The second stage of the research involves corpus and text preprocessing. The data cleanup procedure contains the following steps:

- delete empty texts and duplicates;
- detect the language of each text using fastText model[5] and save texts which were detected as Ukrainian, Belarusian, Bulgarian, Serbian, Macedonian (due to inaccuracies of fastText model);

- perform standard preprocessing procedures such as tokenization, noise re-moval (multiple whitespace/punctuation/new line/quotes, turn all numbers to '0'), substitution (number/html/phone number/email replacers);
- delete texts which contain only URLs, numbers, emails and tags.

The third stage of the research is to annotate texts for further training of the model. Material was distributed amongst 33 people in order to avoid bias. The task was to label tweets as "toxic" (abuses, harassments, threats, obscenity, insults, cyberbullying and identity-based hate texts) or "non-toxic". In total, 55 153 tweets were annotated. To provide features for model training and, conse-quently, improve the accuracy of toxic text detection, a dictionary of obscene words was created. It is based on a list of word roots, word contractions or such combinations as prefix+root or root+suffix. Additionally, Levenshtein edit distance was used. It allows to find the most similar words to those from the dictionary.

The last stage of the research is to train a model that detects toxic texts.

The following steps were made:

- feature engineering (make word embeddings using TF-IDF, bigrams, tri-grams, count number of obscene words in a tweet, number of capitalized words in tweet, number of smiles in a tweet);
- training of models for Text Classification (comparing Logistic Regression, Support Vector Machine and Deep Neural Network);
- evaluation of models' accuracy using cross-validation and F1-score.

The best accuracy is 89% (due to small annotated material and imbalanced classes in training set of 91% of 'non-toxic' and 9% of 'toxic' texts), F1-score is 0.86. There is a room for improvement: to achieve better results, more annotated data is needed. Other possible future directions include generating new features and new classification methods.

References

1. Pradheep, T. and Sheeba, J.I. and Yogeshwaran, T. and Pradeep Devaneyan, S.: Automatic Multi Model Cyber Bullying Detection from Social Networks. In: Proceedings of the International Conference on Intelligent Computing, Salem, Tamilnadu, India. (2017) Available at SSRN: https://ssrn.com/abstract=3123710 or http://dx.doi.org/10.2139/ssrn.3123710
2. Kennedy, G. W., McCollough, A.W., Dixon, E., Bastidas, A.,Ryan, J.,Loo, C., Sahay, S.: Hack Harassment: Technology Solutions to Combat Online Harassment. In: Proceedings of the First Workshop on Abusive Language Online, pp. 73–77, Vancouver, Canada (2017)
3. Rubtsova, Y.: Constructing a corpus for sentiment classification training. SOFT-WARE SYSTEMS 1(109), 72-78 (2015)
4. Twitter Scraper, https://github.com/kennethreitz/twitter-scraper. Last accessed 13 April 2019
5. Language identification, https://fasttext.cc/docs/en/language-identification.html. Last accessed 13 April 2019

Semantic Similarity Identification for Short Text Fragments

Viktoriia Chuiko[0000-0002-4393-3260] and Nina Khairova[0000-0002-9826-0286]

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

viktoriia.chuiko@gmail.com, nina_khajrova@yahoo.com

Abstract. The paper contains review of the existing methods for semantic similarity identification, such as methods based on the distance between concepts and methods based on lexical intersection. We proposed a method for measuring the semantic similarity of short text fragment, i.e. two sentences. Also, we created corpus of mass-media text. It contains articles of Kharkiv news, that were sorted by their source and date. Then we annotated texts. We defined semantic similarity of sentences manually. In this way, we created learning corpus for our future system.

Keywords: semantic similarity, short text fragments, corpus of mass-media text, automatic identification.

The goal of the research is to develop a method for measuring the semantic similarity of short text fragment and to create a program of automatic semantic similarity identification. Existing methods of evaluating similarity have focused mainly on either large documents or individual words [1]. In this work, we focus on computing the similarity between two sentences using corpora of mass-media texts.

The semantic similarity is a quantitative measure that shows how two concepts are close (that is, related or similar) to each other. There are many other connections between words (other than synonymy), in the presence of which one can speak of semantic closeness.

Most of nowadays studies are based on the fact that two sentences that have most of the same words are likely to paraphrase each other. Thereby, we can say that they have semantic similarity [2]. The problem lies in the fact that there are many sentences that convey the same information, but have little resemblance to the surface.

The task of quantitative evaluation of semantic similarity is deeply examined, and now there are many solutions based on different algorithms. Systems, such as Texterra, Semanticus, S-Space, Semantic Vectors and their counterparts, use a semantic distance algorithm. The methods of this group are based on finding the distance between two concepts in a semantic network (for example, WordNet or EuroWordNet). So, between two concepts lies the shortest path and, on its basis, determines the semantic closeness between the words. One of the first such measures was offered by Reznik [3]. The obvious drawback of it was that for some concepts the nesting of the classes to which they belong is greater than for others. To solve this problem, Leacock and Chodorow [4] proposed a method that normalizes the length of the path, considering the depth of the general hierarchy.

Another group is the methods based on lexical intersection. The first algorithm of this type was developed by Lesk [5]. He constructed an algorithm, that basically has the assumption, that related concepts are defined or explained by the same words. Lesk used this approach to solve the problem of finding the correct meaning of a word in some context. The disadvantage of this approach is that articles in common vocabularies are rather short, and may therefore badly reflect the semantic similarity of some words.

There are also systems that offer the use of semantic or parser analyzers to construct the corresponding trees of two comparable sentences, with further analysis and comparison of these trees. An example of such a system can be the MaltParser utility.

All analyzed methods evaluate semantic similarity only for words, but not for larger part of sentences. So, a task of semantic similarity identification for short text fragment (i.e., sentences) is still relevant.

First of all, to make such evaluation we need to have a learning corpus. There are a lot of different corpuses for English language, for example Microsoft Paraphrase Corpus [6]. But only few of them are for Ukrainian and Russian languages.

In this way, the first step of the research was to create corpora of mass-media texts. For this task we choose some sites of Kharkiv news. Then, we automatically extracted news articles content and sorted them according to source.

The next point was to annotate texts. We defined semantic similarity of sentences manually. In this way, we created learning corpus for our future system.

The third stage of the research involves development of method for measuring the semantic similarity of short text fragments using corpora created on previous step. After deep analyze of the existed methods with all their advantages and disadvantages, we propose the following algorithm:

1. From all texts select sentences, which have from 1 to 3 common words.
2. Select two of them and evaluate.
3. Analyze if selected sentences have synonyms.
4. Review the word order of each sentence, using information received on previous stages.
5. Determine a semantic similarity of these two sentences.
6. Repeat steps 2-5 until all texts will be analyzed.

The last stage of the research is to identify the semantic similarity between sentences in the texts of mass-media using developed method.

References

1. Anisimov, A., Glybovets,M., Marchenki,O., Kysenko, V.: The method for calculating of semantic closeness of natural language words meanings (2011)
2. Islam, A., Inkpen, D.: Semantic Similarity of Short Texts. University of information technology & sciences (2009)
3. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In:

International Joint Conference for Artifcial Intelligence (IJCAI-95),pp. 448-453(1995)

4. Leacock, C., Chodorow, M., Miller, G. A.: Using corpus statistics and wordnet relations for sense identifcation. Computational Linguistics, 24(1),pp. 147-165(1998)
5. Lesk, M.: Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In:SIGDOC'86: Proceedings of the 5th annual international conference on Systems documentation, pp. 24-26. New York, NY, USA. ACM(1986)
6. Microsoft Research Paraphrase Corpus Homepage,https://www.microsoft.com/en-us/download/details.aspx?id=52398, last accessed 2019/03/01.

Data-To-Text Generation for Domain-Specific Purposes

Tetiana Drobot[0000−0003−4172−2846]

Taras Shevchenko National University of Kyiv,
Institute of Philology Taras Shevchenko Blvd, 14, Kyiv, Ukraine

tetyanadrobot33@gmail.com

Abstract. The first commercial implementation of Natural Language Generation (NLG) system dates back to the turn of the XXI century. Since then two main methods of NLG – text-to-text generation and data-to-text generation – have grown more complex in order to solve new business challenges. This research project focuses on the full cycle of template-based generation of hotel descriptions from linguistic and non-linguistic input: starting with data scraping and preparation up to rendering the whole text. Also, several improvements to the template- based approach were suggested.

Keywords: Natural Language Generation, data-to-text generation, template-based approach.

Nowadays many industries (e.g., tourism, meteorology, sports journalism, etc.) face a problem of having thousands of data to process and quickly write about. The task is tremendously time-consuming for professional writers. So eventually the choice will fall on data-to-text generation[1] when a computer program converts the incoming data into a text by filling the gaps in a predefined template. The process mentioned above describes a template-based approach to natural language generation (NLG). This method is quite popular due to its simplicity (i.e., no specialized knowledge needed to develop), flexibility (i.e., the ability to be customized to any domain) and good quality of output texts. There are also some drawbacks, such as the possibility to add only handcrafted tem- plates. Another one, little variation in style, can be considered as an advantage if we have synonymized the templates richly. Therefore, a customer gets the feeling that all texts are written by the same qualified author. With the high attention to machine learning (ML) and neural networks (NN), one more question has to be asked: can the NLG system be trainable? Yes, but it takes too much time and resources to train an ML algorithm and, even more, retrain it if need be.

There are a few ways to enhance template-based text generation:

- expand templates to contain information needed to generate more complicated utterances. This task can be done automatically with the help of the WordNet ontology and unannotated corpora of domain-specific texts;
- add linguistic rules to manipulate and maintain templates.

The attempt of their implementation in order to generate hotel descriptions will be shown in the next paragraphs.

According to Reiter and Dale (2000)[2], an NLG system can be decomposed into distinct modules that form a pipeline process. These modules are: document planner with a tree where internal nodes represent structure and leaf nodes represent content as an output; microplanner, also a tree output with internal nodes as structural elements of the document and the leaf nodes as sentences; surface realiser which transforms the sentence representations into text[3].

A template-based approach erases the boundaries between the mentioned modules. But the same tasks to perform are left. The first thing that any NLG system has to take as input is a communication goal. It describes the desired output of the system, e.g., the communication goal of a system that generates hotel information will be expressed in terms of the data it stores, such as “What features does the specific hotel have (nearby attractions, facilities, on-site restaurant, etc.)?”. These features can be scraped from travel websites (Trivago, Hotels.com, TripAdvisor, Booking.com, etc.).

The next step is data preparation. Some changes are applied to the raw data before “filling” the templates' gaps: asserting conjunctions to the lists, synonymization of some phrases, combining two or more expressions together (e.g., “flat-screen TV”, “cable channels” and “satellite channels” to “flat-screen

TV with cable and satellite channels”) etc. The modified data are then stored as dictionary objects.

Using the data sources, the document planning module will decide what information should be included in the produced text (content determination)[2]. As the template-based generator is the object of interest, the order of the sentences is fixed. They are also structured into text sections. A section will be added to the final text only if it includes two or more sentences. For this task, the rules for sentence rendering were written. They work section-by-section, activating the templates of specific sentences in accordance with the given data.

The last step is sections' rendering. It is primarily concerned with the selection of the appropriate synonyms by searching of the word, chosen by the system, in the previous three sentences (or less, depending on the number of sentences that have already been added to the text).

Thus this work offers a solution to domain-specific tasks of text generation for small businesses and start-ups which have restricted hardware resources or a short time for development and implementation.

References

1. Gatt, A., Krahmer, E.: Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. Journal of Artificial Intelligence Research, 61, pp. 65–170 (2018)
1. Reiter, E., Dale, R.: Building natural language generation systems. Cambridge University Press, Cambridge, UK (2000)
2. Learning to tell tales: automatic story generation from Corpora, https://urlzs.com/GUcj. Last accessed 10 Apr 2019

Consideration of the Software Tests Quality Evaluation Problem

Iryna Liutenko[0000-0003-4357-1826] and Oleksiy Kurasov[0000-0003-2518-577X]

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

liv@kpi.kharkov.ua, kurasov.oleksii@gmail.com

Abstract. Software testing problems were considered. Tests quality estimation approaches were determined and justified. There are performance, coverage and implementation factors, which can be used for comprehensive evaluation. Performance approach can be used to estimate testing effectiveness in action by proportions of fixed and not fixed bugs. Coverage approach means volume of fully tested requirements and code structures. Implementation characteristics can be used to evaluate tests as software code. Software tests quality indicators were selected for each of these factors and can be used for assessment. Multicriterion evaluation problems were considered. ПАКС ("Последовательное агрегирование классифицируемых состояний") method was proposed as decision of quality assessment problems.

Keywords: Assessment, Testing, Tests, Software, Indicator, Quality

1 Introduction

Most part of the modern software is a complex, multicomponent system with a significant amount of software code. The list of functional and non-functional requirements is moving forward to software systems, so complex software logic must be implemented.

Testing allows to prevent and fix defects, verify the software compliance with the requirements that have been put forward by the customer and the interested parties.

The testing process is limited by resources. Volumetric and detailed testing of the entire software solution is disadvantageous and too complicated. The testing objects priority and complexity (in this case, a separate component of the subsystem or requirement) must be taken into account, which dictates the corresponding volume of software tests.

A test group that was created without these characteristics cannot be considered as qualitative, as the testing priorities that are misconceived in relation to the PS elements lead to unnecessary time and money costs, which does not guarantee enough level of reliability of the PS that would be released after passing such tests.

The research objective is to find indicators that can be used to determine the value and usefulness of software tests offered for software testing.

2 Evaluation of tests performance

Practice shows that the success of software testing depends on the quality of tests planning and implementation. The effectiveness of testing can be estimated with a relatively small number of indicators.

5. The ratio of not intercepted bugs in the latest version of software to the number of all found (found and corrected / not intercepted) bugs - this indicator can characterize the diligence of testing various options for using FP. The complete coverage of all data, context, and actions is an almost impossible task, so there is a risk that the user may execute an untested sequence of actions which will disrupt the normal operation of the software.

6. The share of bugs repeated in the release - these bugs were corrected in previous versions but became relevant again after the new released version. This indicator differs from the previous one, which may indicate the lack of sufficient regression testing, while the first indicator is more relevant to determine the test quality of the functional that was introduced in the latest version. The disadvantage is the complexity of counting because of the existence of system dependencies of a new code and earlier developed one, which makes possible that a new functionality does not work due to previously unknown defects of the old one [1].

3 Requirements and code coverage assessment

The coverage requirements should show how exhaustively the software compliance with the defined functional requirements, quality attributes, system requirements and constraints are verified.

Comprehensive verification of quality attributes is a complex task due to their probabilistic and subjective nature, therefore it's impossible to say about the adequacy of existing stress tests, information security tests, reliability or usability tests.

Taking the analysis of the code tests coverage, the following metrics can be used:

- coverage of operators - the proportion of validated lines of code;
- coverage of conditions - the proportion of checked branches of execution (calculation of the logical condition);
- covering of roads - the proportion of checked paths through this code snippet;
- coverage of functions - the proportion of proven functions;
- I / O coverage - the proportion of validated calls and outcomes of functions.

There often is a need to verify for the software whether the tests achieve full coverage for one of the indicators, which is crucial in terms of security. It should be noted that the absolute coverage does not have a clear connection with the quality of testing, which, even in this case, does not guarantee impeccable work due to the diversity of performance environment, input data and other factors [2].

4 Assessment by implementation

The volume of test automation can become an important indicator of the testing quality due to its own importance. It reduces the impact of the human factor on testing.

Most software tests are software implemented, which makes it possible to evaluate them as a separate program system with its own interconnected components. When evaluating tests as a program code, the following code properties can be used:

- compliance with conventions - this indicator affects the simplicity of the code perception, which is important when it is accompanied by several developers;
- the purity of the code - the structural simplicity of the code, the absence of superfluous constructions and operators, as well as those constructions that interfere with code tracking and analysis (magic numbers, duplicates).

In case when comparison and assessment are performed taking into account the large number of criteria (more than 20), there is a problem that the formal comparison by criteria values only becomes impossible and attempts to reduce the number of them leads to a decrease in the quality of the final result as a result of its removal from reality. Therefore, it is necessary to find a method that will solve the problem of multi-criteria choice in a space of large size by reducing the number of measurements, based on the specification of the subject domain. The ПАКС method, which is based on the use of verbal analysis methods to reduce the size of the task, can be used to solve the test evaluation problem.

The assessment procedure in this case includes several basic steps:

1. Construction of the hierarchical system of aggregated software tests criteria. The process of constructing is to create integral indicators that aggregate the initial criteria that were chosen by the decision maker as key to assessing the quality of the tests.

2. At the second stage, the successive construction of the scale of each compiled criterion is carried out, which is the most indicative combination of the assessments of the initial criteria.

3. In the third stage, the final selection of the best test list from the resulting complex criteria space is performed using the АРАМИС (“Агрегирование и Ранжирование Альтернатив около Многопризнаковых Идеальных Ситуаций”) method, according to which the lists are ordered in proximity to the reference sample, by distance from the worst sample or by the value of the relative index proximity to the best of the sample.

$$\mathrm{l}(A_q) = \mathrm{d}(A_+, A_q) \,/\, [\mathrm{d}(A_+, A_q) + \mathrm{d}(A_-, A_q)], \tag{4}$$

where $\mathrm{d}(A_+, A_q)$ - distance to the best sample A_+; d (A_-, A_q) – distance to the worst sample A_-.

This methodological approach to reduce the dimension of qualitative traits space has some universality, because it allows to operate both symbolic (the qualitative) and numerical (quantitative) information, representing each composite (aggregate) criterion gradation in the form of combinations of initial indicators evaluation gradations, which allows lowering the subjectivity while evaluating software tests. To build scales of compiled criteria, most of the ranking alternatives methods can be used, which allows to choose the best set and methods of complex criteria construction for practical tasks [3].

5 Conclusions

The effectiveness of testing depends on the tests set formation, which will be enough for the exhaustive software testing and its compliance with the requirements.

Developing an approach to assessing the software tests quality will improve the test results, reduce the time and other resources to find defects in the software system, and will enable you to quickly remedy the shortcomings of the current testing approach in the long run.

References

1. Important Software Test Metrics and Measurements, Homepage, https://www.softwaretestinghelp.com/software-test-metrics-and-measurements,
2. last accessed 2019/04/01.
3. Prause, C., Werner, J., Hornig, K., Bosecker, S., Kuhrmann, M.: Is 100% Test Coverage a Reasonable Requirement? Lessons Learned from a Space Software Project. PROFES 2017, LNCS, vol. 10611, pp 351-367. Springer, Heidelberg (2017)
4. Petrovsky, A.: Decision making theory. “Academy”, Moscow (2009).

Identify of the Substantive, Attribute, and Verb Collocations in Russian Text

Julia Lytvynenko[0000-0001-6087-8155]

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

julialytvynenko12@gmail.com

Abstract. This article describes methods and existing libraries for POS-tagging and collocations extraction, using NLP technologies, processing natural language text in the Python programming language. In addition, it describes one of the possible methods for the selection of collocations for a given pattern.

Keywords: POS-tagging, Python, collocation, corpus linguistics, collocations extraction, morphological marking, computer linguistics, intellectual technologies

There is plenty of different purposes of using certain collocations in text or corpus of text. Collocations play an important role in lexicography (the whole collocation dictionaries are created), they are used in ontology compilation, clusterization, language learning, and some other NLP applications [1].

In our case, this task is part of the development of a program for the identification of closely related text fragments; which may be useful for information retrieval. We have consistently distinguished substantive, attribute, and verb combinations.

Collocation means the co-occurrence of two words in some defined relationship. [2]. We look at several such relationships, including direct adjacency and first word to the left or right having a certain part-of-speech. Currently, the term “collocation” is widely used in corpus linguistics, in which the concept of collocation is rethought or simplified compared with traditional linguistics. This approach is can be called statistical. The frequency of joint occurrence is given a priority, so collocations in corpus linguistics can be identified as statistically stable phrases [1].

To date, scientists have created and considered many different methods for isolating collocations. Among them are statistical methods [4] (association measures, t-score measures etc.), as well as methods based on linguistic models. This idea is laid out and implemented in the well-known system Sketch Engine [3]. It gives typical for a given keyword phrases due to, on the one hand, - syntax, that imposes a restriction on the compatibility of words in a given language, and, on the other hand, probabilistic regularities associated with semantics and linguistic patterns.

In our case, we have created a corpus consisting of approximately 20,000 words, where articles on the subject of information technology are collected. This corpus underwent morphological marking (POS-tagging), on the basis of which we could select the required collocations.

POS-tagging is an automatic morphological markup, which results in each word being tagged. Their values and attributes are determined by the morphological information of each word. For the implementation of morphological marking, the following methods are used: non-verbal morphology, vocabulary morphology based on the base vocabulary, vocabulary morphology on the basis of wordform dictionary, morphemic analysis, Mark chain method, and the N-gram method. Among the existing libraries for POS-tagging in Python there are systems: TreeTager, Pymorphy2, nltk [5].

For morphological processing, we used pymorphy2 and nltk. At the beginning of text processing, we need to normalize the text data and divide the text into tokens. The library nltk is best suited for this. After that, we can start our POS-tagging by using pymorphy2, because it is the best for processing texts of Ukrainian and Russian origin.

When we managed to get the marked text, we need to write the necessary collocations in three files according to three patterns:

1. substantive (NOUN + NOUN(in genitive case)) collocations;

2. attribute (NOUN + ADJECTIVE, which have the same case, number and gender);

3. verb (VERB or INFINITIVE (transitive) with NOUN(in accusative case)

To do this, we create three methods. Each will find the corresponding template. In these methods we again use nltk and pymorphy2. We consider each sentence separately, we look for the first match with the pattern, and if the first is found, we look for a pair for it. If all the conditions are met, we get a list of collocations in the resulting file.

As a result of our text processing, we got 3 lists of collocations, corresponding the template. From the corpus of 20000 words, we have got 668 attributive collocations, 2754 substantive and 452 verb collocations. For example, we got such attributive collocations: краткий обзор, учебно-методический комплекс, учебного заведения etc.” We also got substantive collocations: “форм подготовки, обзор исследований, направления подготовки, система подготовки, использованию технологий etc.” Among verb collocations were: “имеет специфику, исследовать технологию, предполагает организацию etc.”

At the same time, we get a small percentage of collocations that were mistakenly chosen from the text, like substantive collocations: “дисциплине средства, Европе организации, обучения 60-е, годы века etc.” The error arises from the fact that the selected collocation does conform to the pattern, but the words themselves, although met in the same sentence, do not have the indicated type of connection between themselves. This problem can be solved by upgrading the algorithm and adding methods, which can help to determine the syntactic links between the words of the text.

The next step we plan is identifying synonymous collocations or the we can put the results in the collocation dictionaries.

References

1. Khokhlova, M., Zakharov, V.: Study of effectiveness of statistical measures for collocation extraction on russian texts. Computer linguistics and intellectual technologies, 9 (16), pp. 137-143 (2010) In Russian.
2. Yarowsky, D.: One sense per collocation. In: 93 Proceedings of the workshop on Human Language Technology, pp. 266-271 (1993)
3. Kilgarriff, A., Rychly, P., Smrz, P., Tugwell, D.: The Sketch Engine. In: Proceedings of the Eleventh EURALEX International Congress. Lorient, pp.105–116 (2004)
4. Fano, R.: Transmission of Information, Cambridge (MA)(1961)
5. Jurafsky, D.: Speech and Language Processing, Stanford University, University of Colorado at Boulder, 558p. (2018)

Method for ParaphraseExtractionfrom the News Text Corpus

Illia Manuilov, Svitlana Petrasova[0000-0001-6011-135X]

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

banger@ukr.net, svetapetrasova@gmail.com

Abstract. The paper discusses the process of automatic extraction of paraphrases used in rewriting. The researchers propose the method for extracting paraphrases from English news text corpora. The method is based on both the developed syntactic rules to define phrases and synsets to identify synonymous words in the designed text corpus of BBC news. In order to implement the method, Natural Language Toolkit, Universal Dependencies parser and WordNet are used.

Keywords: paraphrase extraction, news text corpus, syntactic rules, synsets, Universal Dependencies,WordNet.

In modern computational linguistics, technologies for identifyingsemantic similarity between linguistic units are widely used. Formally, such a mechanism means the synonymous replacement of elements, extension of the structure (the addition of elements) or shortening of the structure (the omission of elements).

For converting complex text into simpler one andwriting unique texts, the following methods for paraphrasing are used.

1. Transformation of the direct speech into indirect one. This technique allows saving the necessary sense in the text, but at the same time makes information unique for search engines.
2. Reducing the size of the text to simplify it and better understand the content.
3. Text structure processing: moving paragraphs of the text, changing grammatical constructions of sentences which adds not only uniqueness to a new text but a new style of writing without changing its meaning [1].

According toa language level for paraphrasing, the vocabulary, syntactic structure, morphological characteristics of words, their number and order are being changed. In this case, one word can be replaced saving the entire structure or we can change the entire structure retaining lexical units.

There are several ways to paraphrase syntactical units of texts:

– changing the grammatical structure of the sentence, for example, replacing the subject and object;
– replacing words of one part of speechby another, for example, a verb by a noun or adjective;

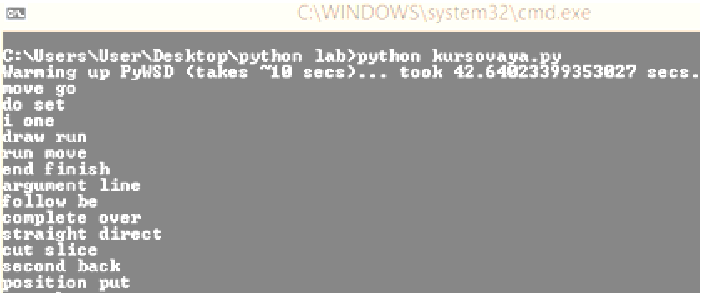
– extending the structure (addition of elements) and vice versa;
– splitting long sentences into several smaller ones and vice versa;
– replacing synonymous words orphrases (collocations) [2].

The paper proposesthe method for paraphrase extraction from the news text corpus based on the developed syntactic rules [3] to define phrases (collocations) and the use of WordNet [4] to identify synonymous words in the text corpus.

The developed corpus consists ofBBCnews articles, the sport section [5].

For preprocessing(POS-tagging), the NLTK's Python language library tools are suggested to use.

Figure 1 shows the synonymous pairs obtained in WordNet.



```
C:\Users\User\Desktop\python lab>python kursovaya.py
Warming up PyWSD (takes ~10 secs)... took 42.64023399353027 secs.
move go
do set
i one
draw run
run move
end finish
argument line
follow be
complete over
straight direct
cut slice
second back
position put
```

Fig. 1. Synonymous Pairs Extracted from WordNet

For extracting paraphrases, we check the correspondence of the grammatical characteristics of collocates (synonymous words of phrases identified at the previous stage) with the syntactic rules.

Thus, phrases whose grammatical characteristics correspond to the rules are considered to be synonymous.As a result, the proposed method for paraphrase extraction from the news text corpus allows identifying a common information space for topical news.

References

1. Koloiev, A.S.: Rewrite as a new phenomenon in modern journalism. In: SPU Bulletin. Philology, vol. 1, 221-226 (2012)
2. Bolshakov, I.A.: Two methods of synonymous paraphrasing in linguistic steganography. In:Proceedings of the International ConferenceDialogue-2004,http://www.dialog-21.ru/media/2496/bolshakov.pdf, last accessed 2019/02/10.
3. Petrasova, S., Khairova, N., Lewoniewski, W.: Building the semantic similarity model for social network data streams. In:Data Stream Mining & Processing, Proceedings of the 2018 IEEE Second International Conference (DSMP), 21-24 (2018)
4. WordNet: https://wordnet.princeton.edu, last accessed 2019/02/10.
5. BBC, https://www.bbc.com/news,last accessed 2019/02/10.

Machine Learning Text Classification Model with NLP Approach

Maria Razno[0000-0003-3356-5027]

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

mari.razno@gmail.com

Abstract. This article describes the relevance of the word processing task that is written in human language by the methods of Machine Learning and NLP approach, that can be used on Python programming language. It also portrays the concept of Machine Learning, its main varieties and the most popular Pythonpackages and libraries for working with text data using Machine Learning methods. The concept of NLP and the most popular python packages are also presented in the article. The machine learning classification model algorithm based on the text processing is introduced in the article. It shows how to use classification machine learning and NLP methods in practice.

Keywords: Machine learning, Python, Pandas, Text classification, NLP, NLTK, Scikit-learn, Artificial Intelligence, Python Library, Deep Learning Texts

Over the last few years machine learning and artificial intelligence have become very hot topics. Nowadays their methods and approaches are a part of a huge amount of products, moreover it is a necessary thing in most applications and appliances. An example of using ML (Machine Learning) can be the automatic determination of important emails and quick responses in Gmail. Nowadays we can confidently say that and artificial intelligence with machine learning can push a person out of many technological processes.

Machine learning is the scientific study of algorithms and statistical methods that computer systems use to effectively perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model of sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to perform the task. There are five types of machine learning algorithms: supervised, semi-supervised, active learning, reinforcement and unsupervised learning [1].

Natural language processing is a subfield of computer science, information engineering, and artificial intelligence concerned with the interactions between computers and human (natural) languages, in particular, how to program computers in order to process and analyze large amounts of natural language data. Tasks in natural

COLINS'2019, Volume II: Workshop. Kharkiv, Ukraine, April 18-19, 2019, ISSN 2523-4013 http://colins.in.ua, online

language processing frequently involve speech recognition, natural language understanding, and natural language generation.

Text classification is one of the most important and typical task in supervised machine learning. Assigning categories of documents, which can be a web page, library book, media articles, gallery etc. has many applications like spam filtering, email routing, sentiment analysis etc. We would like to demonstrate how we can do text classification using the most common python machine learning and natural language processing packages like: Pandas, Scikit-learn, Numpy and little bit of NLTK.

In our study, we are creating the model, that will be able to classify user`s comment and give it a star rate from 1 to 5. Supervised machine learning requires to have prepared labeled data, so we use Yelp_academic_dataset_review in json format. We downloaded the dataset via the link: https://www.kaggle.com/yelp-dataset/yelp-dataset#yelp_academic _dataset_review.json. We got a lot of necessary tools by using Pandas library, that helped us to store data in convenient table, the columns of which are classification parameters and the rows – information for each classified object. This form of data storage is very effective in our study, especially for further accessing a particular column of data during the text processing [2].

The next step is to use natural language processing methods to normalize the text data. During our work we realized, that the package of libraries NLTK(Natural Language Toolkit) is great for our purpose. Thanks to its methods we removed all the stop words, that were not necessary for further analysis, from the text data. Also we needed to use text stemming in order to remove morphological affixes from the text. All of those step helped the model to make an accurate analysis of the text data and get the best clear features for the future classification [3].

The next step of our study was building the machine learning model. We used Scikit-learn due to the fact, that it is a wonderful library with a huge amount of opportunities. It has various types of analysis, moreover, it is the most convenient way of forming a model, because it provides a single interface for all conversion steps and the final result. Instead of using “Bag of words” approach and counting of each word in our text data, we use the tf-idf method for each pair of words in our reviews. Tf-idf normalizes the count by dividing the total sum of the meeting of a certain pair of words into the number of reviews in which these words appear. In such way, we get the model, that will find the most common words for each star rate, in other words it will get the appropriate features and select the best ones. As the result of our study, the model will be able to analyze user`s comment according to found features.

To summarize, in the course of our research, we can say that Python is a wonderful programming language, which provides a lot of great libraries for creating powerful machine learning models and proper natural language processing. For the task of building a machinelearning text classification model with NLP approach, we have reviewed the most popular machine learning libraries like : Pandas, Scikit-learn, Numpy, NLTK and built the text classification model with NLP approach. At the end of our study we get the learning model, that gives user answer with the appropriate star rate to user, according to his comment, and the list of the most common words for each star rate.

References

1. Langley, P.: Human and machine learning.Machine Learning,1, pp. 243–248 (1986)
2. Masch, C.: Text classification with Convolution Neural Net-works on Yelp, IMDB & sentence polarity dataset, https://github.com/cmasch/cnn-text-classification,24/02/2019.
3. Moschitti, A., Basili, R.: Complex Linguistic Features for Text Classification: A Comprehensive Study. In: Lecture Notes in Computer Science vol. 2997, pp. 181-196, Springer Science + Business Media (2004)

Extraction of Semantic Relations from Wikipedia Text Corpus

Olexandr Shanidze and Svitlana Petrasova[0000-0001-6011-135X]

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

s.alexandr21@gmail.com, svetapetrasova@gmail.com

Abstract. This paper proposes the algorithm for automatic extraction of semantic relations using the rule-based approach. The authors suggest identifying certain verbs (predicates) between a subject and an object of expressions to obtain a sequence of semantic relations in the designed text corpus of Wikipedia articles. The synsets from WordNet are applied to extract semantic relations between concepts and their synonyms from the text corpus.

Keywords: semantic relations, rule-based approach, Wikipedia, text corpus, synsets, WordNet.

Due to the growing volume of information, it requires systematization and processing. For example, the increasing number of natural-language texts greatly complicated the process of retrievalof necessary information. Therefore, developing data storage tools and mechanisms for their rapid and efficient processing is an urgent task of NLP. Attempts to cope with this problem led to the development of Information Extraction (IE). According to the extracted information, IE includes the following issues: named entities recognition; attributes/relations extraction; facts/events extraction.

The most challenging task is to get information about semantic relations between objects. A semantic relationis established between lexical units (words, collocations) within the certain semantic field that may be a class, meronymy/holonymy, synonymy, antonymy, and others:

- ISA relation (relation of classification):Object (Member of Class) –>*is a*–> Subject (Class);
- hypernymy: Subject –>*group of* –> Object;
- hyponymy: Object –>*variant of* –> Subject;
- meronymy: Object –>*component of* –> Subject [1].

For extracting information, semantic relations in particular,rule-based methods (using patterns) and machine learning methods (naive Bayes classifier, decision trees, support vector machine (SVM), Hidden Markov Models (HMM), etc.) are applied [2].

The paper proposes the algorithm for automatic semantic relations extraction using the rule-based approach.

Step 1. Preprocessing of the developed text corpus of 200 Wikipedia articles. The Internet encyclopedia presents a system for categorizing pages in the form of a category tree which shows the representativeness of the corpus in a certain category. For our research, we chose articles of Information Technologies category [3].

Step 2. Identifying certain verbs between a subject and an object of expressions in the texts that are assumed as semantic relations, e.g. Subject *->include, consist of, contain* ->Object.

Step 3. Extracting semantic relations (unidentified at the previous stage):

1. search the subjects and objects of predicates (verbs that represent semantic relations identified at the previous stage);

2. obtain synonyms for defined subjects and objects (concepts) from WordNet [4];

3. extract semantic relations between the concepts and their synonyms.

Table 1 shows the semantic relations extracted from the designed text corpus.

Table 3. Semantic Relations Extractedfrom Wikipedia Articles

No	Semantic relation	No	Semantic relation	No	Semantic relation
1	include	11	ability of	21	body of
2	contain	12	aspect of	22	component of
3	consist of	13	member of	23	control of
4	branch of	14	method of	24	mode of
5	class of	15	version of	25	subset of
6	block of	16	part of	26	group of
7	collection of	17	property of	27	quality of
8	description of	18	set of	28	variant of
9	form of	19	type of	29	characteristic of
10	list of	20	use of	30	section of

Consequently, we get the semantic information (the semantic network) of words from the text corpus, i.e. semantic relations between concepts (subjects and objects).

The use of technologies of extractionof semantic relationsfrom texts serves as the basis for developing text analysis tools that operate at a higher level, e.g.text mining. The result of automaticextraction of semantic relations can be used in search engines to extend queries, to construct ontologies, to expand existing and create new thesauri.

References

1. Petrasova, S.V., Khairova, N.F.: Automated semantic network construction based on the glossary. In: Horizons of Applied Linguistics and Linguistic Technologies: International Scientific Conference Megaling–2013,http://megaling.ulif.org.ua/tezi-2013-rik/, last accessed 2019/02/07.
2. Bolshakova, Ye.I., Vorontsov, K.V., Yefremova, N.E.: Automatic natural language texts processing and data analysis. Moscow, Higher School of Economics National Research University, 269 (2017)
3. Wikipedia. Information Technologies Category,https://en.wikipedia.org/ wiki/Category:Information_technology, last accessed 2019/02/07.
4. WordNet,https://wordnet.princeton.edu, last accessed 2019/02/07.

Author index

Reviewers

Babichev, Sergii
Bentayeb, Fadila
Berko, Andrii
Bisikalo, Oleg
Bobicev, Victoria
Bodyanskiy, Yevgeniy
Boytcheva, Svetla
Burov, Eugene
Cekerevac, Zoran
Cherednichenko, Olga
Chetverikov Gregory
Cuprijanow Eugen
Davydov, Maksym
Demchuk, Andrii
Dilay, Marianna
Dosyn, Dmytro
Filatov Valentin
Garasym, Oleg
Godlevskyi, Mykhailo
Gozhyi, Oleksandr
Grabar, Natalia
Hamon, Thierry
Ivasenko, Iryna
Kanishcheva, Olga
Khairova, Nina
Kharchenko, Vyacheslav
Kolbasin, Vladyslav
Kotov, Mykhailo
Krmac, Evelin
Kulchytskyi, Ihor
Kunanets, Nataliia
Kushniretska, Irina
Lande, Dmitry
Levchenko, Olena
Lozynska, Olga
Lytvyn, Vasyl
Lytvynenko, Volodymyr
Martinet, Lucie
Myronovych, Yuriy
Oborska, Oksana

Olakitan, Opeyemi
Orobinska, Olena
Pasichnyk, Volodymyr
Peleshko, Dmytro
Rishnyak, Ihor
Rizun, Nina
Ryabova, Nataliya
Rzheuskyi, Antonii
Savchuk, Valeriia
Shakhovska, Natalya
Sharonova, Natalia
Shepelev, Gennady
Shestakevych, Tetiana
Teslyuk, Vasyl
Veres, Oleh
Veretennikova, Nataliia
Vysotska, Victoria
Werber, Borut
Wodarski, Krzysztof
Yanholenko, Olha
Yurynets, Rostislav
Zakrzewska, Danuta