Zbornik treće nacionalne konferencije sa međunarodnim učešćem pod nazivom

Primena slobodnog softvera i otvorenog hardvera PSSOH 2020

Izdavači / Publishers
Univerzitet u Beogradu - Elektrotehnički fakultet /
University of Belgrade – School of Electrical Engineering
i / and
Akademska Misao / Academic Mind

Štampa / Printed by
Akademska Misao / Academic Mind

ISBN: 978-86-7466-872-6

Tiraž / Number of copies: 50

Mesto i godina izdanja / Place and year of publication
Beograd, 2021. / Belgrade, 2021

Univerzitet u Beogradu – Elektrotehnički fakultet

Zbornik treće nacionalne konferencije sa međunarodnim učešćem pod nazivom

Primena slobodnog softvera i otvorenog hardvera
PSSOH 2020

u Beogradu, februara 2021. godine

University of Belgrade – School of Electrical Engineering

Proceedings of the Third National Conference with International Participation titled

Application of free software and open hardware PSSOH 2020

in Belgrade, February 2021.

Predgovor trećoj PSSOH konferenciji

Sa velikim zadovoljstvom nastavili smo održavanje PSSOH konferencije (Primena slobodnog softvera i otvorenog hardvera) koju smo započeli u 2018. godini sa željom da se u Republici Srbiji organizuju i sa posebnom pažnjom neguju nacionalni skupovi i to uprkos pandemiji COVID-19 virusa. PSSOH je i u 2020. godini dvojezična virtuelna konferencija sa zvaničnim srpskim i engleskim jezikom, pa su radovi u celini štampani u ovom zborniku i na srpskom i na engleskom jeziku.

Pored neizbežnih epidemioloških mera koje smo uveli, a koje su uključile organizaciju virtuelne konferencije, uveli smo i niz drugih izmena počevši od toga da se Zbornik štampa nakon konferencije, kao i to da od 2020. prihvatamo samo radove u celini. Takođe, u 2020. godini odvojili smo organizacioni od uredničkog odbora sa željom da negujemo PSSOH podmladak uz veću efikasnost u organizaciji. Kako nismo želeli da se u potpunosti odreknemo tradicionalnog načina organizacije konferencije, umesto da organizujemo u potpunosti virtuelnu konferenciju, organizovali smo kombinovanu konferenciju. Svi predavači koji su to želeli mogli su da dođu i da na Elektrotehničkom fakultetu Univerziteta u Beogradu uz sve mere predastrožnosti i na licu mesta uz našu tehničku podršku održe predavanje. Ove godine, Zbornik smo štampali nakon konferencije iz dva razloga. Prvi je što nismo stigli ranije da ga pripremimo, jer je kompletan urednički odbor angažovan u nastavi, a kako su letnji semestar školske 2019/2020. i zimski semestar školske 2020/2021. godine bili pretežno onlajn to nam je svima ponestalo vremena. Drugi razlog je što ima više smisla štampati radove nakon javne recenzije zajednice kako na internetu, tako i tokom konferencije.

Ove godine smo po prvi put uveli i plenarnu sesiju na kojoj je predavanje, na naše veliko zadovoljstvo, održao Italo Vinjoli iz *The Document* fondacije povodom godišnjice *Libre Office* projekta. U saradnji sa Descon konferencijom organizovali smo radionicu na temu kako napraviti Klimerka, gde su učesnici u prostorijama Fakulteta i onlajn mogli da prate uputstva i da sastave sopstveni Klimerko otvoreni uređaj. Naši studenti i članovi organizacionog odbora organizovali su i održali PSSOH post-konferencijsku radionicu za instalaciju Linux operativnog sistema. Ove godine, sve prezentacije sa PSSOH konferencije i post-konferencijskih radionica su dostupne i na YouTube-u pod CC licencom, pa naši čitaoci, kritičari, prijatelji i svi znatiželjni, imaju dostupnu PSSOH 2020 onda kada njima to bude odgovaralo.

PSSOH konferencija je bila organizovana u 2020. godini kao jednodnevni događaj koji je uključio niz predavanja po pozivu u četiri sesije. Pored svih izmena, PSSOH konferencija nije menjala svoj društveno odgovorni karakter i teme. I ove godine smo se bavili aktuelnim pitanjima koje se odnose na primenu slobodnog softvera i otvorenog hardvera, ali i zastupljenošću žena i ostalih manjinskih grupa u elektrotehnici i računarstvu. U 2020. godini, naša predavanja po pozivu uključila su teme iz otvorenih formata i otvorenih projekata (*Libre Office*), licence slobodnog softvera i otvorenog hardvera, modele za predikciju potrošnje električne energije zasnovane na primeni slobodnog softvera, simulatore za držanje nastave iz arhitekture i organizacije računara, IoT sisteme, otvorene resurse za obradu srpskog jezika, softver otvorenog koda za vizuelizaciju podataka i otvoreni hardver za realizaciju električnog bicikla. Pored primena u elektrotehnici i računarstvu, PSSOH je uključio i teme iz drugih oblasti, pa su predstavljeni radovi na temu primene slobodnog softvera u meteorologiji, matematici, računarskoj lingvistici, eksperimentalnoj psihologiji i u biliotekama. Takođe, jedan rad je bio posvećen otvorenim psihološkim instrumentima i jedan rad rodnoj raznolikosti na Vikipediji.

Na svim prethodnim, ali i budućim PSSOH konferencijama svi naši gosti i predavači su dobrodošli ako žele da pomognu u skladu sa svojim mogućnostima i sve tretiramo ravnopravno, što je tradicija pokreta slobodnog softvera i otvorenog hardvera. Kako smo najavili u 2018. i realizovali u 2019. godini i ove godine smo ponudili svim autorima koji su na Elektrotehničkom fakultetu Univerziteta u Beogradu objavili otvorene nastavne materijale da ih predstave u našem zborniku. Međutim, do zaključenja zbornika nismo dobili odgovor da li su zainteresovani. Iako je u 2020. godini izostao odgovor autora otvorenih nastavnih materijala, a bilo ih je samo dva u proteklom periodu od godinu dana, to ne znači da su otvorene knjige manje popularne. U Srbiji se upravo rasplamsala diskusija oko Fondacije Aleka Kavčića koja se zalaže za besplatne udžbenike. Mi, urednici PSSOH konferencije, možemo samo da podržimo aktivnosti Fondacije, ali i da predložimo sledeći korak, otvorene referentne udžbenike koji bi bili dostupni pod *Creative Commons Attribution Share*

Alike licencom u pdf formatu na sajtu Ministarstva, pa bi ih mogao štampati i koričiti bilo ko, čime bi otvoreno tržište dobilo pun smisao i donelo dobro krajnjim korisnicima, učenicima i roditeljima koji udžbenike kupuju. Takođe, snimljena referentna predavanja bi trebalo da budu dostupna na sajtu Ministarstva, što bi bila pomoć nastavnicima u pripremi predavanja i učenicima u praćenju nastave. Na današnjem razvoju digitalnih tehnologija, teško je naći izgovor da navedeni nastavni resursi ne budu učinjeni dostupni, ako kao validne argumente isključimo lične ekonomske interese.

Organizacija PSSOH konferencije je podržana od strane velikog broja koleginica i kolega, ustanova, kompanija i udruženja i ovde ih je nemoguće sve pobrojati. Najzahvalniji smo našim donatorima iz Akademske Misli iz Beograda ali i radionici DK za ručnu izradu PSSOH maski. Zahvalni smo i svim predavačima na učešću i na nesebičnom doprinosu u održavanju treće PSSOH konferencije. Takođe, bez Nikole Todorovića i Miloša Bjelića koji predsedavaju organizacionim odborom ne bi smo mogli da zamislimo PSSOH 2020. Veoma smo zahvalni članovima Organizacionog odbora Dejanu Petkoviću, Dragici Nikolić, Živani Garašević, Jovanu Sandiću, Mihajlu Pavloviću, Nenadu Popoviću i Milošu Budimiru. Zahvalnost dugujemo i svim članovima naučnog i spoljnjeg organizacionog odbora i to posebno Dejani Pavlović, ali i Milanu Antiću i Nemanji Jovanoviću. Kako nismo u mogućnosti da sve koji su nam pomogli nabrojimo, unapred se izvinjavamo svima koje smo propustili da spomenemo.

P.S. U duhu PSSOH tema i sa željom da promovišemo slobodan softver, ovaj Zbornik smo pripremili u programskom paketu LibreOffice.

u Beogradu, 1. februara 2021. godine

Uredнички odbor PSSOH konferencije

Foreword to the Third PSSOH Conference

With great pleasure we continued to organize PSSOH conference (Application of Free Software and Open Hardware) despite the COVID-19 virus pandemic in 2020. We established PSSOH in 2018 with the desire to organize and nurture with special attention national gatherings in the Republic of Serbia. PSSOH 2020 is a bilingual virtual conference with official Serbian and English and the papers in this Proceedings are printed in both Serbian and English languages.

In addition to the inevitable epidemiological measures that we had to introduce, including the organization of a virtual conference, we also introduced other changes, starting with the Proceedings being printed after the conference, as well as accepting only full length papers from 2020. Also, in 2020, we separated the Organizational from the Editorial Board with the desire to nurture the PSSOH youth and at the same time by providing more efficient organization. As we did not want to completely abandon the traditional way of organizing the conference, instead of organizing a completely virtual conference, we organized a combined event. All lecturers who desired to come, were able to give a lecture at the School of Electrical Engineering, University of Belgrade, with all pandemic precautions and with our technical support. We printed the PSSOH 2020 Proceedings after the conference for two reasons. The first reason is that we did not have time to prepare it earlier, because Editorial board was enrolled in teaching during 2019/2020 summer semester and during 2020/2021 winter semester that were organized mostly online, so we ran out of time. Another reason is that it makes more sense to print papers after a public review both online and during the conference.

This year, for the first time to our great satisfaction, we introduced a plenary session with guest lecturer Italo Vinjoli from The Document Foundation on the occasion of the anniversary of the Libre Office project. In collaboration with the Descon conference, we organized workshop on how to assemble Klimerko, where participants both at the Faculty and online could follow the instructions and assemble their own Klimerko open source device. Our students and organizing committee members held PSSOH post-conference workshop on how to install the Linux operating system. This year, all presentations from the PSSOH conference and post-conference workshops are available on YouTube under CC license, so our readers, critics, friends, and curious visitors have PSSOH 2020 available whenever they want it.

PSSOH conference was organized in 2020 as a one-day event and it included a series of invited lectures organized in 4 sessions. Despite all the changes, the PSSOH conference did not change its socially responsible character and topics. This year, invited PSSOH talks dealt with current issues related to the application of free software and open hardware as well as to the representation of women and other minority groups in electrical engineering and computing. PSSOH 2020 invited lectures included topics from open formats and open projects (Libre Office), free software and open hardware licenses, models for predicting electricity consumption based on the application of free software, simulators for teaching computer architecture and organization, IoT systems, open resources for Serbian language processing, open source software for data visualization, and open hardware for the realization of an e-bicycle. In addition to the applications in electrical engineering and computing, PSSOH included topics from other fields, such as free software in meteorology, mathematics, computational linguistics, experimental psychology, and libraries. Also, one paper was dedicated to open psychological instruments and one to gender diversity on Wikipedia.

At all previous and future PSSOH conferences, all our guests and lecturers are welcome if they want to help in accordance with their capabilities and we treat everyone equally, which is tradition of the free software and open hardware movement. As we announced in 2018 and realized in 2019, this year we have offered all authors who have published open teaching materials at the School of Electrical Engineering, University of

Belgrade to present them in our Proceedings. However, until the conclusion of the Proceedings, we did not receive answer whether they were interested. Although we will not be presenting open educational resources in this Proceedings and that there are fewer open textbooks last year, open books are not less popular. There is an ongoing discussion about the Alek Kavčić Foundation, which advocates free textbooks in Serbia. We, the editors of the PSSOH conference, can only support the activities of the Foundation, but also propose the next step which would include open reference textbooks that would be available under a Creative Commons Attribution Share Alike license in pdf format on the Ministry website, so anyone could print and use them. We believe that this way, the open market would get its full potential by bringing benefits to end users, students and parents who buy textbooks. Also, recorded reference lectures should be available on the website of the Ministry, which would help teachers in preparing lectures and students in monitoring classes. In today's development of digital technologies, it is difficult to find an excuse not to make the mentioned teaching resources available, if we exclude personal economic interests as valid arguments.

The organization of the PSSOH conference is supported by a large number of colleagues, institutions, companies, and associations, and it is impossible to count them all here. We are most grateful to our donors from Academic Mind from Belgrade, but also to the DK studio for handmade PSSOH masks. We are also grateful to all the lecturers for their participation and generous contribution to the third PSSOH conference. Also, without Nikola Todorović and Miloš Bjelić both chairing the Organizing Committee, we would not have been able to imagine PSSOH 2020. We are very grateful to the members of the Organizing Committee Dejan Petković, Dragica Nikolić, Živana Garašević, Jovan Sandić, Mihajlo Pavlović, Nenad Popović, and Miloš Budimir. We also owe gratitude to all members of our Scientific and External Organizing Committees, especially to Dejana Pavlović, but also to Milan Antić and Nemanja Jovanović. As we are not able to list everyone who helped us, we apologize in advance to everyone we failed to mention.

P.S. According to the PSSOH themes and with aim to promote application of free software, this Proceedings is prepared in LibreOffice.

in Belgrade, February 1, 2021.

Editorial Board of the PSSOH Conference

Sadržaj / Table of Contents

Rezime plenarne sesije / Abstract Plenary Session

LibreOffice 10th Anniversary: Digital Sovereignty & Open Document Standards 6

Slobodan softver i otvoren hardver u elektrotehnici i računarstvu / Free Software and Open Hardware in Electrical Engineering and Computer Science

Licence slobodnog softvera i otvorenog hardvera: Kratko uputstvo za nestrpljive 7

Generalized Additive Model for Electricity Load Prediction in R 15

A Survey of Simulators Suitable for Teaching Courses in Computer Architecture and Organization 19

Towards Deterministic Industrial Internet of Things Networking 24

Otvoreni resursi i tehnologije za obradu srpskog jezika 29

Primena softvera otvorenog koda kod vizuelizacije podataka 34

Otvoreni hardver za realizaciju pogona električnog bicikla 38

Slobodan softver i softver otvorenog koda u društvenim naukama / humanistici, lingvistici, meteorologiji i matematici / Open & Free Software in Social Sciences, Humanities, Linguistics, Meteorology, and Mathematics

Application of Free Software in Meteorology 42

Application of Mathematical Software in Teaching Numerical Mathematics for Students of Electrical Engineering 46

REPOPSI: An Open Repository of Psychological Instruments in Serbian 51

Free Software Tools for Computational Linguistics: An Overview 57

Otvorena biblioteka 71

Slobodni i otvoreni softver u laboratorijskim i mrežnim istraživanjima u eksperimentalnoj psihologiji 79

Žene u elektrotehnici i računarskoj tehnici / Women in Electrical Engineering & Computer Science

Rodna raznolikost na Vikipediji: Primeri iz prakse 84

Spisak autora(ki) / List of Authors 88

LibreOffice 10th Anniversary: Digital Sovereignty & Open Document Standards

Italo Vignoli
One of the founders and team members of The Document Foundation and main spokesperson
italo@documentfoundation.org

Announcement and Outline for Plenary Lecture

We are very proud to announce our Plenary Session for the first time at the PSSOH conference with Italo Vignoli as a spokesperson. This presentation covers history of LibreOffice, its goals and achievements, and the most important: its community. Why Digital Sovereignty? Because "Technology sovereignty is the ability of a state or a federation of states to provide the technologies it deems critical for its welfare, competitiveness, and ability to act, and to be able to develop these or source them from other economic areas without one-sided structural dependency", from the presentation held by Italo Vignoli at PSSOH 2020 and available on Zenodo repository (doi: 10.5281/zenodo.4129851).

Italo Vignoli, also presented how important open document formats are and stressed their significance by some practical examples. We, the Editorial Board of the PSSOH conference, have never aimed to convince anyone, any person, institution, or nation to use free and open source software and LibreOffice. Our aim was limited to give an insight into possibilities and opportunities that come with the free software. If we managed to spread the word on open formats and free and open source software, and to inform at least one person, one institution, or one nation, we can say that our expectations are exceeded.

Keywords: Libre Office; Free and Open Source Software; FOSS; open formats.

Licence slobodnog softvera i otvorenog hardvera
— kratko uputstvo za nestrpljive —

Predrag Pejović[1], Nadica Miljković[1], Miloš Cvetanović[1], Milica Ševkušić[2]
[1]*Elektrotehnički fakultet, Univerzitet u Beogradu, Beograd, Srbija*
[2]*Institut tehničkih nauka SANU, Beograd, Srbija*
peja@etf.rs, nadica.miljkovic@etf.rs, cmilos@etf.rs, biblioteka@itn.sanu.ac.rs

Rezime: U radu je dat prikaz osnovnih ideja slobodnog softvera i način njihove realizacije primenom odgovarajućih licenci, sa posebnim osvrtom na tehniku kopilefta. Razmotrena je generalizacija ideja slobodnog softvera na ostala dela kreativnog rada, a posebno na hardver, sa kratkim osvrtom na odgovarajuće licence primerene tim slučajevima. Naglašena je i obrazložena razlika između slobodnog i besplatnog softvera.

Ključne reči: Licence, otvoreni hardver, slobodni softver, softver otvorenog koda.

I. Uvod

Računari su značajno promenili način života time što su automatizovali algoritamske poslove. Uticaj računara na društvo se često rangira kao industrijska revolucija, samo što ova revolucija nije zamenila ljudsku snagu, već je oslobodila ljudski um dosadnih poslova koji se izvršavaju po unapred utvrđenim algoritmima. Poseban zamah drustvenim promenama je dala sprega računarske tehnike i telekomunikacija, kada je omogućen jednostavan, brz i ekonomičan prenos velike količine informacija na velika rastojanja. Računari su omogućavali prenos, skladištenje i obradu tih informacija. Ove tehnološke inovacije su dovele do značajnih posledica, među kojima su porast ekonomskog značaja nematerijalnih proizvoda, pojava proizvoda sa nultim marginalnim troškovima proizvodnje, značajno povećana složenost proizvoda, problemi saradnje i organizacije u proizvodnji tih složenih proizvoda, kao i pravni, ekonomski i društveni izazovi koje nova tehnologija uzrokuje.

Jedan od osnovnih proizvoda vezanih za računarsku tehniku je softver. Softver je nematerijalni proizvod, karakterisan marginalnim troškom proizvodnje (troškom pravljenja još jednog primerka) koji je praktično jednak nuli. Osim toga, softver se upotrebom ne troši, pa se potražnja ne obnavlja spontano. U odsustvu veštački nametnutih restrikcija, softver lako može da kopira bilo ko. Sve ovo čini softver vrlo specifičnim proizvodom koji zahteva posebne poslovne modele koji bi finansirali njegov razvoj i odgovarajuću pravnu podršku.

Po osobini da je nematerijalni proizvod, softver je sličan nauci, matematici i delu umetnosti. Ove delatnosti su poslovni model nalazile u materijalnom nosiocu, knjizi za dela iz nauke, matematike i književnosti, ili pločama, kasetama i kompakt diskovima u oblasti muzike kada je tehnički postalo moguće snimiti je. Napretkom tehnologije je postajalo sve lakše samostalno praviti kopije ovih dela, bilo na kopir mašini, bilo na kasetofonu ili danas na računaru, a pravna zaštita od ovakve prakse bila je kopirajt: pravo kopiranja su imali samo ovlašćeni proizvođači. Ovakva zaštita poslovnog modela je generalizovana i na softver. Kvantni skok u mogućnostima kopiranja i razmene kopija su izazvale nove računarske i komunikacione tehnologije, što je problem poslovnog modela sa softvera na istovetan način prenelo na sve ostale nematerijalne, a sada digitalizovane, proizvode: marginalni trošak proizvodnje im je pao na nulu, a dostupnost kopiranja je postala opšta. Na primer, kod knjige ono što vredi je sadržaj, mnogo više nego papir i štampanje (mada, ima izuzetaka). Sadržaj je postalo moguće lako kopirati i distribuirati po ceni koja je praktično jednaka nuli, čime je nova tehnologija odvojila i ostale nematerijalne proizvode od materijalnog nosioca. Da bi se poslovni model zaštitio, bile su neophodne restrikcije, uskraćivanje mogućnosti koje tehnologija pruža. Kopirajtom uvedene restrikcije postojale su i ranije, ali se sada pojavila potreba da se kontroliše znatno veći broj potencijalnih proizvođača kopija.

Softver ima niz sličnosti sa ostalim nematerijalnim proizvodima, ali i niz razlika. Jedna od razlika je mogućnost prerade i dorade postojećeg softvera kako bi se prilagodio potrebama korisnika. Ovo je moguće uraditi ako je dostupan izvorni kod programa (*source code*). Osim toga, uvidom u izvorni kod programa može se zaključiti šta i kako program radi sa korisnikovim podacima.

Ekonomski interes, kome je odgovaralo da se za nematerijalne proizvode generalizuju poslovni modeli koji se primenjuju na materijalne proizvode, uticao je na sam softver [1] str. 17-19. Prvo je korisniku izvorni kod programa uskraćen, pa on nije mogao da vidi šta program radi sa njegovim podacima na njegovom računaru. Zatim su u programe ugrađivana i svojstva kontrole korisnika u cilju nametanja poštovanja licence. Dalja evolucija je vodila uvođenju i drugih svojstava u programe koje korisnik ne kontroliše, pre svega u domenu prikupljanja podataka o korisniku i korišćenju. Poslovni model primeren materijalnim objektima primenjen na softver vodio je ka tome da softver kontroliše svog korisnika, a ne da korisnik kontroliše softver koji obrađuje njegove podatke na njegovom računaru. Novim korisnicima računara, koji su počeli da koriste računar u doba kada je koncept vlasničkog softvera već bio formiran, ovo je izgledalo prihvatljivo, nisu imali drugih iskustava. Starim korisnicima računara, iz perioda kada vlasništvo nad softverom nije bilo značajno, koji su navikli da imaju kontrolu nad svojim podacima na svom računaru, ali i da proučavaju i modifikuju tuđe programe u skladu sa

DOI: 10.5281/zenodo.4210351

svojim interesovanjima i potrebama, ovo nije bilo prihvatljivo. Tako je nastao pokret za slobodni softver.

II. Definicija slobodnog softvera

Izvorna ideologija pokreta slobodnog softvera može se naći u [2] u formi zbirke članaka koji su presudno uticali na njegov razvoj. Slobodni softver je definisao Ričard Metju Stolman (*Richard Matthew Stallman*), [2] str. 43, a definicija se sastoji iz četiri slobode koje se tradicionalno numerišu počevši od nule:

Sloboda 0: Sloboda da se program koristi za bilo koju namenu.

Sloboda 1: Sloboda da korisnik prouči kako program radi i da ga prilagodi svojim potrebama. Preduslov za ovu slobodu je mogućnost pristupa izvornom kodu programa.

Sloboda 2: Sloboda da dajete kopije programa kako biste pomogli svojim prijateljima.

Sloboda 3: Sloboda da se program unapredi i da se unapređena verzija publikuje kako bi cela zajednica imala koristi. I za ovu slobodu je preduslov mogućnost pristupa izvornom kodu programa.

Ovde valja naglasiti da se u definiciji slobodnog softvera nigde ne pominju novac, cena i naplata, već samo slobode koje korisnici programa treba da imaju kako bi upravljali svojim podacima i njihovom obradom na svojim računarima.

Navedena definicija slobodnog softvera potiče iz članka koji je napisan 1996. godine ([2] str. 43), ali je slobodni softver koji zadovoljava uslove definicije postojao i ranije. Dva poznata primera su simulator električnih kola SPICE [3], [4] i program za obradu teksta TeX [5], [6]. Takođe, među prvim primerima slobodnog softvera je danas zaboravljeni *Tiny BASIC* [7], [8] koji je nastao kao reakcija na „*Open Letter to Hobbysts*“ [9], [10] koje je 1976. godine napisao Bil Gejts i koje je predstavljalo istorijski značajnu tačku u procesu odvajanja slobodnog i vlasničkog softvera. *Tiny BASIC* je jedan od prvih primera uspeha slobodnog softvera u doba kada sam koncept još nije bio formulisan u danas uobičajenoj formi.

A. Terminološke razlike i zabune: slobodan softver, softver otvorenog koda, besplatan softver

Koncept slobodnog softvera nije bio dobro dočekan u poslovnim krugovima koji su svoje poslovanje zasnivali na vlasničkom softveru. Na ekonomskom nivou pretila im je opasnost od konkurencije sa kojom po ceni nikako nisu mogli da se takmiče. Na ideološkom nivou smetalo im je širenje ideja o pravima korisnika programa. Moćna industrija je pokrenula propagandne mehanizme koji su omalovažavali slobodni softver i napadali njegove protagoniste nazivajući ih dot komunistima (.com). To je dovelo do reakcije [11], ironičnog članka u kome se analiziraju osnove i posledice postojanja slobodnih informacija. Sa druge strane, dostupnost izvornog koda programa i slobode koje slobodni softver donosi nudile su korisnicima niz prednosti, poput veće baze programera koji mogu da pregledaju i menjaju program, konkurencije među njima, a time i potencijalno kvalitetnijih proizvoda i smanjene cene. Želja da se izbegne konfrontacija sa ideologijom vlasništva, pa time i vlasničkog softvera, uz želju da se zadrže prednosti koje slobodni softver nudi, doveli su do odvajanja dela programera od pokreta za slobodni softver i nastanka „Inicijative za otvoreni kod“ [1] str. 14-16, [12], koja je dala alternativnu definiciju slobodnog softvera izbegavajući termin „slobodni softver“ i koristeći termin „softver otvorenog koda“. Ovo je izazvalo žestoko protivljenje Fondacije za slobodni softver [13] koja je insistirala na etičkim pitanjima i terminu „slobodni softver“, [2] str. 57. Sukob je trajao neko vreme, ali se danas može smatrati završenim. I slobodni softver i softver otvorenog koda koriste iste licence, što se naročito odnosi na *GNU General Public License* (GPL) o kojoj će još biti reči. Stoga, za sve praktične primene može se smatrati da su slobodni softver i softver otvorenog koda identični pojmovi jer daju iste slobode korisnicima pošto se distribuiraju pod istom licencom. Stoga se isti pojam označava i terminima FOSS (*Free/Open_Source Software*) i FLOSS (*Free/Libre/Open_Source Software*). Očekivano, u svetu proizvođača vlasničkog softvera softver otvorenog koda dočekan je suštinski isto kao i slobodni softver, samo je neko vreme tolerisan kao manje zlo i željena podela u pokretu za slobodni softver. Jedna slika klime koja je vladala je ideološko propagandni tekst [14] u kome činjenice nisu bile od značaja, već su krajnje bahato i površno pogrešno interpretirane u cilju promovisanja jednog pogleda na svet zasnovanog na pojedinačnim interesima.

Besplatni softver (*freeware*) se bitno razlikuje od slobodnog softvera. Odrednica besplatnog softvera je cena, koja je jednaka nuli, a ne slobode koje su date korisniku. Prilikom korišćenja besplatnog softvera treba biti oprezan i razumeti motive koji uskraćuju neke od sloboda koje slobodni softver ima u proizvodu koji je besplatan. Postoje slučajevi kada se besplatan softver koristi za plasiranje reklama, za prikupljanje informacija o korisniku, za manipulaciju korisnika da se navikne na svojstva besplatnog programa koja se kasnije uskrate, a nastavak korišćenja se uslovljava naplatom. Zabunu oko pojmova slobodnog i besplatnog softvera je delom uzrokovala i dvosmilenost reči „*free*“ na engleskom, sa značenjima „slobodno“ i „besplatno“. Stoga, Ričard Stolman često naglašava da reč „*free*“ u „*free software*“ treba tumačiti kao u „*free speech*“ (sloboda govora), a ne kao u „*free beer*“ (besplatno pivo). Mada, besplatno pivo može da dovede do slobodnijeg govora.

B. GNU

Na ovom mestu treba objasniti i termin koji će se u licencama slobodnog softvera često pominjati, a to je *GNU*. *GNU* je rekurzivni akronim od *Gnu is Not Unix*, kako je Ričard Stolman nazvao svoju viziju operativnog sistema koji će biti nalik na *Unix* [15], [16], ali će za razliku od njega biti slobodan softver. *Unix* je bio pogodna osnova za izgradnju slobodnog operativnog sistema, pre svega zbog svoje prenosivosti na različite hardverske platforme i zbog svoje modularnosti, koja je

omogućavala realizaciju projekta slobodnog operativnog sistema u etapama, modul po modul. *GNU* je najavljen 1984. godine u [2] str. 33 i osnov je svih operativnih sistema koji se danas još kolokvijalno nazivaju *Linux*, iako je *Linux* samo deo tog operativnog sistema, kernel, a korektan naziv je *GNU/Linux*.

III. Klasifikacija licenci

Dominantan način zaštite softvera kao „intelektualne svojine“ jeste kopirajt, mada se u pojedinim pravnim sistemima softver može i patentirati. To su dva bitno različita načina zaštite: kopirajt [17], [18] str. 13-31 štiti autorsko delo u formi u kojoj jeste, trajanje zaštite je različito u zavisnosti od pravnog sistema određene zemlje i vremena nastanka dela, a obično je dugo, čak i preko 100 godina; patent [18] str. 13-31, [19] štiti ideju na osnovu koje je program napravljen, obično traje kratko, oko 20 godina, što takođe zavisi od pravnog sistema. Treba imati u vidu da je pojam „kratko trajanje zaštite“ zavisan i od proizvoda: za softver je 20 godina jako dug period. Po Bernskoj konvenciji [20], [21], svaki rezultat kreativnog rada koji se u pravnom smislu može smatrati autorskim delom podleže kopirajtu automatski, svojim nastankom, nikakva registracija nije potrebna. Ovde se vrlo često greši kada se smatra da je moguće preuzeti bilo koji sadržaj koji podleže kopirajtu, npr. fotografiju koja se nalazi na internetu, za koji ne piše eksplicitno da to nije dopušteno. Upravo suprotno važi: ako ne piše ništa, po Bernskoj konvenciji kopiranje nije dozvoljeno. Prikaz sa objašnjenjima i primerima dat je u [22].

Licenca je dokument kojim se definišu uslovi korišćenja, odnosno kojim se korisniku daju određena prava, [18] str. 51-72, [23]. U slučaju softvera ta prava proističu iz kopirajta i/ili patenta. Ako prava koja su licencom data korisniku omogućavaju četiri slobode navedene u definiciji slobodnog softvera, ta licenca se može smatrati licencom slobodnog softvera.

Licence su pisane jezikom pravnih dokumenata koji nije lako razumljiv laicima. U kontekstu slobodnog softvera koegzistira veliki broj licenci nastalih u različitim periodima i u okviru različitih zajednica, pa ih je moguće klasifikovati na osnovu različitih kriterijuma, [18] str. 69-72. Kako je korisnicima licence bitno da znaju i razumeju koja prava su im licencom data, mi ćemo se u ovom radu držati klasifikacije zasnovane na pravima koja korisnik ima, odnosno podele na restriktivne i liberalne licence.

IV. Restriktivne licence slobodnog softvera: kopileft

Ričard Stolman je imao loše iskustvo da softver koji je razvijan kao slobodan softver bude modifikovan, relicenciran pod vlasničkom licencom i zatvoren, nedostupan svojim originalnim autorima, [1] str. 13-14. Kako bi sprečio takvu praksu, došao je na ideju da kopirajt iskoristi tako da korisnicima omogući slobode koje zahteva definicija slobodnog softvera, ali i da onemogući promenu licence za modifikovane verzije softvera i pretvaranje u vlasnički softver. Ideja za ovakvu primenu kopirajta data je u članku „*The GNU Manifesto*“, [2] str. 39, u kome se prikazuje vizija operativnog sistema *GNU*, ali se raspravlja i metod njegovog licenciranja. Ova tehnika je nazvana kopileft, [2] str. 91 i str. 99, pošto je kopirajt percipiran kao sredstvo kojim se ograničava javna dostupnost, dok kopileft koristi kopirajt kako bi obezbedio ne samo javnu dostupnost izvornog softvera, već i javnu dostupnost njegovih modifikovanih verzija [24], [25]. Sam termin kopileft potiče od izjave o pravima (*copyright notice*) za program *Palo Alto Tiny BASIC* [8], „*Copyleft, All Wrongs Reserved*“, koja je kao zanimljiva igra reči iskorišćena i kasnije često parafrazirana kao „*Copyleft, All Wrongs Reversed*“ (kopileft, sve greške ispravljene) ili kao „*Copyleft, All Rights Reversed*“ (kopileft, sva prava obrnuta).

Proizvođači vlasničkog softvera nisu sa odobravanjem dočekali koncept kopilefta i pežorativno su licence koje ga podržavaju nazivali virusnim, a najpoznatiju kopileft licencu, *GNU General Public License*, nazivali su „*General Public Virus*“ [24]. Marketinšku osnovu za ovakav naziv predstavljaju negativne konotacije pojma računarskog virusa i osobina licence da se prenosi sa programa na program kao virus, da se širi kao zaraza. Očigledno, koncept kopilefta predstavlja ozbiljan izazov za poslovni model vlasničkog softvera.

Kopileft licence se mogu podeliti na jake i slabe. Jake kopileft licence bez izuzetka zahtevaju kopileft, dok slabe zahtevaju kopileft samo za modifikovane verzije programa istog tipa, što je predmet posebne definicije, a to su tipično biblioteke programa [26].

A. GNU General Public License

Kada su jasne pravne tehnike vezane za svojinska prava nad softverom i njihovo licenciranje, kao i ciljevi pokreta za slobodni softver, lako je razumeti smisao *GNU General Public License* (*GPL*) [18] str. 103-140, [27]. *GNU General Public License* je licenca koja se može primeniti na bilo koji softver koji autori žele da distribuiraju kao slobodan softver. Licenca se svrstava u jake kopileft licence jer bez izuzetaka zahteva da modifikovane verzije programa daju korisnicima iste slobode kao i osnovna verzija. Licenca je evoluirala kroz tri glavne verzije, prateći razvoj tehnologije i nove opasnosti koje su nastajale po slobodu korisnika. Detaljan, ali lako razumljiv, prikaz suštine licence sa pravnog stanovišta može se naći u [28].

A. GNU Lesser General Public License

Zahtev koji kopileft nameće programima koji koriste komponente licencirane pod jakom kopileft licencom predstavlja problem pri licenciranju biblioteka programa. U slučaju jake kopileft licence, takve biblioteke se ne mogu koristiti u programima vlasničkog softvera. Kao deo strategije za širenje slobodnog softvera je stoga uvedena liberalnija (popustljivija, dopustljivija) licenca, *GNU Lesser General Public License* (*LGPL*) [18] str. 121-124, [29] koja ne zahteva da program koji koristi biblioteku bude pod licencom koja korisniku daje ista

prava kao i licenca za biblioteku. Međutim, ako modifikacija obuhvata samu biblioteku, tada se zahteva da modifikovana verzija biblioteke bude pod istom licencom.

C. GNU Affero General Public License

GNU General Public License ne pokreće kopileft uslove korišćenjem programa, već samo distribucijom. Korisnik može da modifikuje program i koristi ga, ali dok ga ne distribuira ne mora da izvorni kod modifikovane verzije učini dostupnim. Napretkom interneta, pojavile su se „mrežne aplikacije“ koje se izvršavaju na serveru koji nije u vlasništvu korisnika programa i sam program se ne distribuira korisniku, već se samo koristi na vlasnikovom serveru. Ovaj koncept mrežne aplikacije osnova je prakse koja se danas naziva „*cloud computing*“.

Ideja očuvanja kopilefta u slučaju mrežnih aplikacija dovela je do modifikacije *GPL* dodatnim zahtevom da se u slučaju mrežnih aplikacija korisniku omogući pristup izvornom kodu programa. Ovo je osnovna ideja *GNU Affero General Public License* (*AGPL*) [30], [31]. Licenca spada u jake kopileft licence, čak vrlo jake.

D. GNU Free Documentation License

Slobodan softver treba da ima i slobodnu dokumentaciju. U tom cilju je nastala *GNU Free Documentation License* (*GFDL*) [32] kao licenca primerena tekstu koji sadrži dokumentaciju programa. Licenca se uspešno primenjuje i na druge tekstove, poput udžbenika, bez obzira na temu. Na primer, Vikipedija kosisti *GFDL* u kombinaciji sa *Creative Commons Attribution Share-Alike* licencom. Licenca pripada kopileft tipu i za cilj ima očuvanje moralnih autorskih prava originalnih autora, ali i oslobađanje originalnih autora odgovornosti za sadržaj modifikovanih verzija.

E. Dvojno licenciranje

Koncept kopilefta kao restriktivne licence slobodnog softvera omogućava i pravni fenomen dvojnog licenciranja, [18] str. 262-264: isti program može imati dve i više licenci. Ako neko želi da koristi program izdat pod kopileft licencom u sklopu vlasničkog programa, može dogovoriti sa vlasnicima kopirajta posebnu licencu. Primer takve prakse je program *The Fastest Fourier Transform of the West* (*FFTW*) koji je izdat pod *GPL*, ali i pod vlasničkim licencama [33]. Ovako nešto nije moguće ako se koriste liberalne licence slobodnog softvera.

V. Liberalne licence slobodnog softvera

Liberalne (popustljive, permisivne) licence slobodnog softvera jesu licence koje omogućavaju slobode u skladu sa definicijom slobodnog softvera, ali ne zahtevaju da se za modifikovane verzije programa korisniku daju ista prava kao što ih daje originalna verzija. Ovo ukida smisao dvojnog licenciranja. Liberalne licence obično dokumentuju autorstvo originalnih autora i sadrže odricanje od odgovornosti za korišćenje programa, što su elementi koje imaju i kopileft licence.

U današnje vreme nije sporan stav da rezultati rada finansiranog iz javnih fondova treba da budu javno dostupni. Međutim, sporno je da li treba da budu dostupni pod liberalnom ili pod kopileft licencom. Proizvođači vlasničkog softvera insistiraju na liberalnim licencama i aktivno lobiraju u tom smeru, pošto im to omogućava da slobodni softver modifikuju i modifikovane verzije učine vlasničkim softverom. Time oni u svom poslovnom modelu spasavaju ono što se spasiti može, ali i izvlače

Tabela I: Uporedni pregled licenci slobodnog softvera

licenca	javni domen	BSD	MIT	ISC	Apache	LGPL	GPL	AGPL	
odricanje odgovornosti	✓	✓	✓	✓	✓	✓	✓	✓	liberalne
autorstvo		✓	✓	✓	✓	✓	✓	✓	
ista licenca za nemodifikovane delove					✓	✓	✓	✓	
kopileft za modifikovane verzije						✓	✓	✓	restriktivne
kopileft							✓	✓	
javna upotreba pokreće kopileft								✓	

profit iz rezultata rada zajednice koja razvija slobodan softver.

A. BSD licence

BSD licenca je nastala za potrebe licenciranja programskog paketa *Berkeley Software Distribution* koji je verzija *Unix* operativnog sistema razvijanog na Univerzitetu Kalifornije u Berkliju, [34], [35] str. 35-49. Licenca je imala nekoliko verzija, od kojih je verzija sa četiri klauzule imala reklamnu klauzulu koja je zahtevala pominjanje organizacije koja je razvila originalnu verziju programa u svim reklamnim materijalima modifikovanih verzija. Kasnije je ova klauzula napuštena. Licenca i u svojoj najkraćoj verziji ("*Zero Clause BSD*") zadržava pominjanje institucije u kojoj je originalna verzija programa razvijena, rečenicu kojom se sva prava koja zahteva definicija slobodnog softvera prenose korisniku, kao i odricanje od odgovornosti, odredbu koju sadrže sve *BSD* licence.

B. MIT licenca

MIT licenca [35] str. 49-62, [36] je po suštini veoma slična *BSD* licenci u verziji koja nema reklamne klauzule. Obe su veoma kratke, zadržavaju podatak o nosiocu kopirajta za originalnu verziju programa, sadrže odredbu koja sva prava zahtevana definicijom slobodnog softvera daje korisniku i sadrže odricanje od odgovornosti.

C. ISC licenca

Slična *BSD* i *MIT* licencama je *ISC* licenca [37] koju je uveo *Internet Systems Consortium*. Suština ove licence je ista kao kod pomenute dve srodne licence, ali je jezik dodatno redukovan izbacivanjem teksta koji se smatra nepotrebnim posle ratifikacije Bernske konvencije.

Da rezimiramo tri zajednička svojstva navedenih liberalnih licenci slobodnog softvera: sadrže podatak o originalnom nosiocu kopirajta, odredbu o davanju prava koje zahteva definicija slobodnog softvera, kao i odricanje od odgovornosti.

D. Softver u javnom domenu

Za softver u javnom domenu [38] nisu zadržana nikakva prava poput kopirajta ili patenta. Kako je po Bernskoj konvenciji svako delo koje podleže kopirajtu automatski zaštićeno, a time i softver, za postavljanje u javni domen potreban je dokument kojim se autor odriče autorstva i pripadajućih prava. Obično ovakav dokument sadrži i odricanje od odgovornosti. U pravnim sistemima nekih zemalja, prevashodno evropskih, nije moguće odreći se svih prava, a pre svega moralnih autorskih prava, što onemogućava predaju softvera u javni domen u punom smislu tog pojma. Pre pojave kopilefta i problema koji su njegov nastanak uzrokovali, značajni programski paketi koji spadaju u slobodan softver bili su u javnom domenu [38], zato što zakonska ragulativa u Sjedinjenim Američkim Državama to dopušta.

Komparativni pregled svih razmatranih licenci i ograničenja koja nameću prikazan je u Tabeli I.

VI. Kompatibilnost licenci

Praksa modifikovanja i nadgradnje programa koja postoji u proizvodnji softvera dovela je do otvaranja pitanja kompatibilnosti licenci, [18] str. 241-254, [39]: pod kojim uslovima program izdat pod jednom licencom može sadržati delove koda koji su izdati pod drugom licencom? Pravo se ovde sreće i sa tehničkim pitanjem dovoljnog odvajanja komponenti softverskih sistema da bi se mogle smatrati zasebnim celinama, kada relicenciranje komponenti nije potrebno.

Kompatibilnost dve licence nije uzajamna. Na primer, softver iz javnog domena se može uključiti kao komponenta u program izdat pod *GPL*, ali se program izdat pod *GPL* ne može uključiti u program koji se prenosi u javni domen. Primer složenosti problema je slučaj da stara verzija *BSD* licence ne može da se uključi u *GPL* program, zbog klauzule o reklamiranju, a nova, koja tu klauzulu nema, može. Štaviše, verzije 2 i 3 *GPL* nisu direktno međusobno kompatibilne pošto verzija 3 nameće dodatna ograničenja.

Zbog problema kompatibilnosti, povoljno je koristiti postojeće licence za licenciranje novog softvera i ne uvoditi nove licence bez valjanog razloga, [40], [41]. Za postojeće licence koje se široko koriste unapred su poznata prava korisnika i kompatibilnost licenci, pa nije potrebno proučavati svaki put formalnim jezikom pisan tekst licence i sagledavati njegove implikacije. Mrežni servis za objavljivanje koda i upravljanje razvojem softvera GitHub ima uputstvo za korisnike sa listom licenci koje mogu izabrati [42], autor koji postavlja svoj kod treba samo da izabere licencu. Informacije koje pomažu u izboru licence se mogu naći na [43].

VI. Razlika između slobodnog softvera i besplatnog softvera

U ovom odeljku ništa novo neće biti rečeno, a postoji samo da dodatno naglasi da slobodan softver i besplatan softver nisu isto i da među njima postoji fundamentalna razlika. Slobodan softver daje korisniku slobode navedene u drugom odeljku ovog rada. U njima se nigde ne pominje cena. Besplatan softver (*freeware*) je karakterisan samo činjenicom da mu je cena jednaka nuli i ne ispunjava neke od uslova definicije slobodnog softvera. Ovo dovodi do razlike u samom softveru: kod slobodnog softvera je jako malo verovatno da ćete naći agresivno reklamiranje ili da će imati neku zlonamernu funkciju. Jednostavno, licenca slobodnog softvera omogućava da se nepoželjne funkcije izbace iz programa i modifikovana verzija učini dostupnom kako korisnici ne bi morali da trpe neželjena svojstva programa motivisani time što je program besplatan. Stoga, ne postoji motiv za uvođenje zlonamernih i/ili eksploatišućih svojstava u slobodni softver.

VII. Generalizacija: otvorena kultura i Creative Commons licence

Softver je oduvek bio u digitalnom domenu, tu je i nastao. Napretkom tehnologije i ostala dela kreativnog

rada koja podležu kopirajtu su dobijala digitalnu formu i postalo je moguće nadgrađivati ih [44]. Kopirajt takvu praksu ograničava. Uspeh slobodnog softvera je motivisao generalizaciju ideja i prakse koju su uvele licence slobodnog softvera na ostale vidove stvaralaštva koji podležu kopirajtu. Takva generalizacija je uspešno izvršena kroz *Creative Commons* (*CC*) licence [45], [46], [47]. Kod ovih licenci se kopileft tretira kao *„share-alike“* opcija licence koja zahteva da se modifikovane verzije dela distribuiraju pod licencom koja korisnicima daje ista prava. Osim opcije kopilefta, licence otvaraju još dve dimenzije: pravo modifikacije dela koje može biti dato ili uskraćeno, kao i pravo na komercijalnu eksploataciju dela, koje isto tako može biti dato ili uskraćeno. Kako opcija za zabranu modifikacija ukida smisao kopileft opcije, navedene tri dimenzije licence daju šest smislenih kombinacija [46] str. 49-53, od kojih autor bira licencu koja odgovara njegovim namerama. Kao i licence koje se primenjuju na softver, *Creative Commons* licence stalno se dopunjavaju i ažuriraju kroz nove verzije, [46] str. 56. Trenutno aktuelna verzija 4.0, između ostalog, uvodi i novi modul, *CC0*, koji autorima omogućava da se u granicama douštenim zakonom odreknu prava i omoguće neograničeno korišćenje svoga dela. *Creative Commons* licence imaju svoju tabelu kompatibilnosti prilikom stvaranja modifikovanih verzija [39].

Creative Commons licence su našle široku primenu zato što pokrivaju širok spektar sadržaja i mogućnosti, a njihovoj popularnosti doprinosi i činjenica da se koriste u projektima Vikimedija fondacije, te da se u okviru platformi otvorenog pristupa primenjuju kao standard. Danas je dovoljno samo pogledati logotip *CC* licence i biće jasno koja su prava data korisniku. Od pomoći je i troslojna struktura ovih licenci: osnovni sloj je sadržaj ugovora, tekst razumljiv pravnicima; drugi sloj je opis licence, tekst razumljiv pravnim laicima koji objašnjava nameru koju licenca treba da ostvari; treći sloj je digitalni kod, deo licence namenjen mašinskom čitanju metapodataka u cilju lakšeg pretraživanja. Upravo je drugi sloj licence nivo kojim se bavimo u ovom radu i koji pokušavamo da generalizujemo na ostale licence i prenesemo čitaocima.

VIII. Otvoreni hardver

Generalizacija ideja slobodnog softvera na hardver nije direktna i kasnila je oko petnaest godina u odnosu na vreme kada su se ideje slobodnog softvera iskristalisale, [48], [49]. Za ovo postoje dva razloga [50]:

1. Hardver ima značajan materijalni nosilac, čine ga fizički objekti. Kopirati hardver je znatno teže nego kopirati softver, primerenije je reći da se hardver prema nekom planu pravi, ne da se kopira. Plan se može kopirati. Kod hardvera nema etičkog prigovora zašto se ne sme kopirati kada je to tako lako moguće.
2. Dominantan mehanizam zaštite hardvera su patenti, a ne kopirajt, pod uslovom da je određeni hardverski dizajn dovoljno inovativan da se može patentirati, što je predmet posebnih rasprava i osporavanja [51]. Električno kolo ne može biti pod kopileft licencom, pošto ne može biti zaštićeno kopirajtom.

Kopiranjem softvera se dobija kompletan proizvod. Kod otvorenog hardvera korisnik dobija samo ekvivalent izvornog koda programa, projektnu dokumentaciju po kojoj može da napravi proizvod, ne dobija ekvivalent izvršne verzije programa, sam proizvod. U tom smislu, svaki kuvar, u značenju zbirke recepata za pripremanje hrane, jeste biblioteka projektne dokumentacije otvorenog hardvera i u ovom tekstu će biti korišćen za ilustrovanje razlika u odnosu na slobodni softver. Kopileft bi u slučaju kuvara bio ekvivalentan zahtevu da svaki recept izveden iz u kuvaru datog takođe bude objavljen, otvoren, kao i da njegova primena bude slobodna. Međutim, materijalni objekti se dominantno štite patentom. Uprkos stalnom proširivanju tipova objekata koji se mogu patentirati, jela još uvek nisu uključena, a dominantni tip njihove zaštite je poslovna tajna. Kako onda sprovesti ideju kopilefta? Dodatno, kopirajt nad receptom je moguće zaobići jednostavnim prepričavanjem recepta, pisanjem recepta koji na drugi način opisuje isto jelo. Napisati novi operativni sistem po ugledu na neki postojeći, mnogo je teže. Stoga je otvoreni hardver drugačiji od slobodnog softvera, slobode koje definišu slobodan softver nemaju pun smisao kada se direktno generalizuju na hardver. Postupak generalizacije ideja slobodnog softvera na hardver liči na generalizaciju poslovnih modela proizvodnje materijalnih objekata na nematerijalne, samo izveden u suprotnom smeru. Ipak, motivi nastanka slobodnog softvera, pravda uz minimum restrikcija slobode korisnika, privlačni su da budu generalizovani na hardver.

Reč hardver ima dvostruko značenje: može se odnositi na bilo koji materijalni objekt, a može se odnositi i na znatno uži pojam računarskog hardvera, materijalnog dela računarskog sistema. Ovde je razvoj tehnologije doveo do zamagljivanja granice između hardvera i softvera. Rad programabilne logičke komponente, hardverskog elementa, definiše program. Kod ugrađenih računarskih sistema za specifične namene (*embedded systems*) inovativni deo sistema je softver, a hardver se sastoji iz malog broja integrisanih komponenti visoke složenosti čiju funkciju definiše softver. Time se mehanizmi zaštite softvera direktno prenose na hardver, pošto je specifičnost navedenih sistema u softveru, ne u hardveru. Takođe, dokumentacija koja opisuje hardver direkno je podložna kopirajtu i može se objaviti pod *GFDL* ili *CC* licencama.

Uspešan primer na ovim principima zasnovanog projekta otvorenog hardvera je *Arduino* [52], gde se za softver koriste *GPL* i *LGPL*, a za dokumentaciju *CC* licence. Tip projekta je takav da ga u potpunosti opisuju softver i projektna dokumentacija za hardver, pa je za zaštitu moguće direktno primeniti postojeće licence zasnovane na kopirajtu. Upravo je otvorenost softvera i dokumentacije za izradu hardvera dovela do izuzetnog uspeha *Arduino* projekta kroz stvaranje zajednice i njegovog društvenog, tehnološkog, a pre svega obrazovnog uticaja. Sličan primer je i niz projekata računara na jednoj ploči (*single board computers*).

Povećavanje složenosti računarskih sistema opravdava generalizaciju druge slobode iz definicije slobodnog softvera u delu koji se odnosi na proučavanje na hardver. Primer [53] ukazuje na bezbednosne slabosti hardvera koje su otkrivene uvidom grupe istraživača i koje se, nakon spoljnjeg uvida u njihovo postojanje, mogu softverski otkloniti. U izgledu je da će za bezbednost podataka od sve većeg interesa biti poznavanje detalja realizacije hardvera koji su danas uglavnom zatvoreni i štiti ih poslovna tajna. Veliki korak napred je učinjen kroz nastanak otvorene arhitekture skupa instrukcija, tačke spoja hardvera i softvera, [54]. Ovde se, uz otvorene alate za projektovanje pomoću računara, očekuje veliko proširenje mogućnosti i liberalizacija projektovanja digitalnog hardvera. Već sada su dostupne hardverske implementacije *RISC-V* koje su otvorene [55], pa korisnici znaju šta i kako hardver radi sa njihovim podacima, što u nizu primena daje prednost.

Open Source Hardware Association, (*OSHWA*) od 2016. godine [56] vrši sertifikaciju otvorenog hardvera u skladu sa publikovanim kriterijumima. Do 24.08.2020. sertifikovano je ukupno 977 uređaja. Sertifikovani uređaji stiču pravo na repoznatljiv logo koji ih identifikuje kao otvoreni hardver.

Jedna osobina koja prati hardver koji nije otvoren je ograničeno servisiranje. Za svaku popravku jedino je moguće koristiti usluge ovlašćenog servisa, koji je vrsta monopola, što proizvođačima omogućava značajan profit. Reakcija je pokret *Right To Repair* [57] koji se zalaže za prava vlasnika da popravljaju i modifikuju uređaje koje poseduju, čemu otvorenost hardvera značajno pomaže.

Napredak tehnologije je doveo do toga da i kuća od drveta može biti otvoreni hardver. Danas se drvo obrađuje *Computer Numerical Control* (*CNC*) mašinama kojima upravlja softver prema ulaznim podacima koje dobije. Projektna dokumentacija za kuće podložna je kopirajtu. Ovo je dovelo do *WikiHouse* projekta otvorenih kuća [58]. Koliki je korak do postojanja otvorene biblioteke projekata kuća? Kakve licence su primerene takvim projektima? Statika zgrade i odricanje od odgovornosti, može li to zajedno? Kakva je saglasnost sa propisima u građevinarstvu? Koliki je korak do generalizacije na druge materijale i sisteme gradnje? Hoće li odobravanje građevinske dozvole biti svedeno na automatsku proveru da li „*design files*“, dati u standardizovanom otvorenom formatu, zadovoljavaju usvojene urbanističke uslove? Da li je za otvorene projekte zgrada potrebna drugačija licenca od postojećih? Ovakva pitanja je otvorilo svođenje opisa materijalnih objekata na digitalnu formu.

Do sada izneti primeri su koristili licence slobodnog softvera i *Creative Commons* licence za generalizaciju ideja slobodnog softvera na hardver. U [59] su date licence specifično namenjene otvorenom hardveru kao podrška za repozitorijum otvorenog hardvera. Ove licence još nisu dostigle popularnost *Creative Commons* licenci, ali se razvijaju i predstavljaju pokušaj generalizacije ideja i prilagođavanja licenci slobodnog softvera na hardver.

Može se zaključiti da proces generalizacije ideja slobodnog softvera na hardver nije direktan, a najlakše je primenljiv kod digitalnog hardvera gde programabilne komponente čine osnovu projekta. Ideja otvorenosti i mogućnosti uvida se može generalizovati i sve više dobija na značaju. Zbog drugačijeg tipa zaštite materijalnih objekata, ideja kopilefta nije uvek direktno primenljiva i teže se realizuje. Takođe, moguće je da su za različite tipove materijalnih objekata primerenije njihovim specifičnostima prilagođene licence.

IX. Zaključak

Mnogo godina posle Platona vratili smo se svetu ideja i podeli sveta na materijalne i nematerijalne objekte, samo su nematerijalni objekti sada kodirani u digitalnoj formi. Digitalna tehnologija je omogućila lako kopiranje i široku distribuciju informacija. Softver je nematerijalni proizvod, a pokušaji da se poslovni model materijalnih objekata prilagodi softveru suočili su se sa nizom apsurda. To je dovelo do pokreta za slobodni softver i licenci slobodnog softvera sa posebnim naglaskom na kopileft tehniku koja od korisnika zahteva da drugima pruži ista prava koja je i sam dobio. Vremenom se koncept slobodnog softvera pokazao kao uspešan, pa je prilagođen na druge oblasti. U ovom radu su na nivou razumljivom pravnim laicima prikazane osnovne licence koje omogućuju slobodu razvoja i razmene ideja.

X. Zahvalnica

Autori se zahvaljuju docentu dr Vladimiru Milovanoviću sa Fakulteta inženjerskih nauka Univerziteta u Kragujevcu što im je skrenuo pažnju na *ISC* licencu.

Predrag Pejović i Nadica Miljković su delimično podržani sredstvima projekta Ministarstva prosvete, nauke i tehnološkog razvoja broj TR33020.

Nadica Miljković i Milica Ševkušić su delimično podržane sredstvima sa projekta broj 0079 "Boosting EOSC readiness: Creating a scalable model for capacity building in RDM" koji je finansiran od strane Evropske unije u okviru H2020-EU.1.4.1.1. EOSC Secretariat projekta br. 831644."

Literatura

[1] S. Chopra, S. D. Dexter, *Decoding Liberation: The Promise of Free and Open Source Software*. Routledge, 2008.

[2] R. M. Stallman, *Free software, free society: Selected essays of Richard M. Stallman*. GNU Press, 2002, pristupljeno 01.11.2020.

[3] Wikipedia contributors, “SPICE”, *Wikipedia, The Free Encyclopedia*, pristupljeno 17.08.2020.

[4] T. Tuma, Á. Buermen, *Circuit Simulation with SPICE OPUS: Theory and Practice*. Basel: Birkhäuser, 2009. str. xv-xvi.

[5] Wikipedia contributors, “TeX”, *Wikipedia, The Free Encyclopedia*, pristupljeno 06.08.2020.

[6] S. Levy, R. Seroul, *A Beginner's Book of TEX*. Springer Science & Business Media, 2012.

[7] Wikipedia contributors, “Tiny BASIC”, *Wikipedia, The Free Encyclopedia*, pristupljeno 19.08.2020.

[8] R. Pasqualini, B. Edmundson, J. Handy, "On-chip Tiny Basic dumps development systems", *Electronic Design*, vol. 28, no. 24, pp. 235-240, 1980.

[9] Wikipedia contributors, "Open Letter to Hobbyists", *Wikipedia, The Free Encyclopedia*, pristupljeno 07.08.2020.

[10] B. Gates, "An Open Letter To Hobbyists". *Homebrew Computer Club Newsletter*. Mountain View, California: Homebrew Computer Club. vol. 2, no. 1, p. 2, Jan. 1976.

[11] E. Moglen, "The dotcommunist manifesto", Online, January 2003, pristupljeno 01.11.2020.

[12] Open Source Initiative, home page, pristupljeno 01.11.2020.

[13] Free Software Foundation, home page, pristupljeno 01.11.2020.

[14] Wikipedia contributors, "Samizdat: And Other Issues Regarding the 'Source' of Open Source Code - Wikipedia", *Wikipedia, The Free Encyclopedia*, pristupljeno 01.11.2020.

[15] Wikipedia contributors, "Unix", *Wikipedia, The Free Encyclopedia*, pristupljeno 16.08.2020.

[16] B. W. Kernighan, *UNIX: A History and a Memoir*. Kindle Direct Publishing, 2020.

[17] Wikipedia contributors, "Copyright", *Wikipedia, The Free Encyclopedia*, pristupljeno 15.08.2020.

[18] L. E. Rosen, *Open source licensing: software freedom and intellectual property law*. Upper Saddle River, NJ: Prentice Hall PTR, 2005.

[19] Wikipedia contributors, "Patent", *Wikipedia, The Free Encyclopedia*, pristupljeno 14.08.2020.

[20] Wikipedia contributors, "Berne Convention", *Wikipedia, The Free Encyclopedia*, pristupljeno 29.07.2020.

[21] *Berne Convention for the Protection of Literary and Artistic Works*. WIPO, 1984.

[22] M. Ševkušić, "Autorska prava i slobodne licence (za istraživače)", Online, pristupljeno 01.11.2020.

[23] Wikipedia contributors, "License", *Wikipedia, The Free Encyclopedia*, pristupljeno 18.08.2020.

[24] Wikipedia contributors, "Copyleft", *Wikipedia, The Free Encyclopedia*, pristupljeno 18.08.2020.

[25] S. Dusollier, "Open Source of Copyleft: Authorship Reconsidered", *Colum. J. L. & Arts*, vol. 26, no. 3+4, pp. 281-296, 2002.

[26] A. J. Hall, "Open-Source Licensing and Business Models: Making Money by Giving It Away", Santa Clara Computer & High Tech. L. J., vol. 33, no. 3, pp. 427–437, 2016.

[27] Wikipedia contributors, "GNU General Public License" *Wikipedia, The Free Encyclopedia*, pristupljeno 04.08.2020.

[28] B. Kuhn, A. K. Sebro Jr, D. Gingerich, "Copyleft and the GNU General Public License: A Comprehensive Tutorial and Guide", *Copyleft.Org*, 2015.

[29] Wikipedia contributors, "GNU Lesser General Public License" *Wikipedia, The Free Encyclopedia*, 04.07.2020.

[30] Wikipedia contributors, "GNU Affero General Public License", *Wikipedia, The Free Encyclopedia*, pristupljeno 28.05.2020.

[31] M. MacDonald, "Open Source Licensing in the Networked Era", Masaryk U. J.L. & Tech., vol. 7, no. 2, pp. 229-240, 2013.

[32] Wikipedia contributors, "GNU Free Documentation License", *Wikipedia, The Free Encyclopedia*, pristupljeno 19.08.2020.

[33] Wikipedia contributors, "FFTW", *Wikipedia, The Free Encyclopedia*, pristupljeno 28.02.2020.

[34] Wikipedia contributors, "BSD licenses", *Wikipedia, The Free Encyclopedia*, pristupljeno 11.08.2020.

[35] A. M. St. Laurent, *Understanding open source and free software licensing*. Beijing ; Sebastopol, CA: O'Reilly, 2004.

[36] Wikipedia contributors, "MIT License" *Wikipedia, The Free Encyclopedia*, pristupljeno 31.10.2020.

[37] Wikipedia contributors, "ISC License" *Wikipedia, The Free Encyclopedia*, pristupljeno 18.08.2020.

[38] Wikipedia contributors, "Public-domain software", *Wikipedia, The Free Encyclopedia*, pristupljeno 26.07.2020.

[39] Wikipedia contributors, "License compatibility", *Wikipedia, The Free Encyclopedia*, pristupljeno 16.08.2020.

[40] Wikipedia contributors, "License proliferation", *Wikipedia, The Free Encyclopedia*, pristupljeno 21.06.2020.

[41] R. W. Gomulkiewicz, "Open Source License Proliferation: Helpful Diversity or Hopeless Confusion Open Source and Proprietary Models of Innovation: Beyond Ideology - Part III: Open Source and Proprietary Software Development", Wash. U. J. L. & Pol'y, vol. 30, no. 1, pp. 261-292, 2009.

[42] Licensing a repository, Online, pristupljeno 01.11.2020.

[43] Choose an open source license, Online, pristupljeno 01.11.2020.

[44] Lawrence Lessig, *Remix: Making art and commerce thrive in the hybrid economy*. Penguin, 2008.

[45] Creative Commons Srbije, home page, pristupljeno 01.11.2020.

[46] Creative Commons (Organization), Ed., *Creative commons for educators and librarians*. Chicago: ALA Editions, 2020.

[47] M. Välimäki, *The rise of open source licensing: a challenge to the use of intellectual property in the software industry*, Helsinki, Finland: Turre, 2005. str. 154-161.

[48] Wikipedia contributors, "Open-source hardware", *Wikipedia, The Free Encyclopedia*, pristupljeno 27.06.2020.

[49] J. R. Ackerman, "Toward Open Source Hardware", U. Dayton L. Rev., vol. 34, no. 2, str. 183-222, 2008.

[50] Richard Stallman, "On 'Free Hardware'", Linux Today, June 22, 1999.

[51] Patent Absurdity: how software patents broke the system, home page, pristupljeno 01.11.2020.

[52] Arduino, home page, pristupljeno 01.11.2020.

[53] Meltdown and Spectre, home page, pristupljeno 01.11.2020.

[54] Krste Asanović, David A. Patterson, "Instruction Sets Should Be Free: The Case For RISC-V", EECS Department, University of California, Berkeley, *Tech. Rep. UCB/EECS-2014-146*, 2014.

[55] Krste Asanović, "Keynote: State of the Union", September 3, 2020, Online video, pristupljeno 01.11.2020.

[56] Open Source Hardware Association, home page.

[57] IFIXIT, home page, pristupljeno 01.11.2020.

[58] WikiHouse, home page, pristupljeno 01.11.2020.

[59] CERN Open Hardware License, home page, pristupljeno 01.11.2020.

Generalized Additive Model for Electricity Load Prediction in R

Sovjetka Krstonijević
Mihajlo Pupin Institute, Belgrade, Serbia
sovjetka.krstonijevic@pupin.rs

Abstract: The electricity load forecasting is an important activity in power grid which becomes a critical one in condition of growing demands, the presence of renewable energy sources and electricity market. This entails a question of an adequate electricity load model for forecasts generation. In this work the attention is on the generalized additive model, GAM, the configurable modelling framework, available in R environment, which is suitable for developing various tools for analysis, simulation and prediction. The recent results confirmed its potential and applicability for actual load prediction problems.

Keywords: generalized additive model (GAM); prediction model; electricity load prediction

I. Introduction

The power grid is the most complex man-made system which serves more than five billion people around the globe [1]. All in order to timely deliver the electricity to consumers. In the absence of the electricity storage on a large scale, the energy has to be generated and delivered as it is consumed. Thus, the most of the effort in managing power grid is about balancing the supply and demand as discrepancy between those two could cause the grid instability and unreliability. Consequently, the prediction of the electricity load is one of the most important activities in the power system. The prediction of electricity load is about recognizing and identifying the factors that drive the electricity, deduced from the past load behavior. Then, identified patterns are used to forecast the load expectations. This is the task for load prediction model.

The history of load prediction is over a century long. When lighting was the only electricity consumer, it was enough to estimate the rise of consumption in the evening, by counting the number of the installed bulbs. With the involvement of various electric appliances, the overall consumption exhibited the characteristic weekly and daily patterns, reflecting the consumers' habits of appliance usage. The prediction was based on the identified load patterns. With the further involvement of the heating/cooling systems into the grid, the electricity consumption became sensitive to meteorological factors. So the weather variables, such as temperature, wind, humidity, etc. were included into the model. In today's complex power system, present on electricity market, some economic and socio-demographic indices are, also incorporated into the model.

In the pre-PC era, all load prediction was made manually, using charts and tables. The first computationally supported load models are based on traditional statistical approaches like classical multiple linear regression or various time-series models [2]. Due to their simplicity, they are, still, widely used in practice, for load forecasting functionality, typically on a system or regional level. The new approaches are based on Machine Learning (ML) and Artificial Intelligence (AI) techniques, [3] [4] [5]. Their general advantage over the models of the previous generation is in increased ability to handle the nonlinearities in load, providing better accuracy.

In Smart Grid Era, the efficient grid management is based on ability to predict the load on more local level of aggregation, e.g. cities, neighborhoods, homes, etc. The renewable energy sources and changes in a way how the energy is consumed set new challenges in keeping the system reliable. Such trend changes the standard specification of load model. Not only the accuracy is critical, but, also, its fastness, robustness, adaptability. Most of the previously established models are developed under stable load conditions and customized to a particular part of the system. Although simple for implementation or successful in capturing the nonlinearities, they have some limitations in adapting to data, either because they need human expertise as in ARIMA models, or are computationally expensive as e.g. deep neural networks [6].

As a compromise, the generalized additive model, GAM, offers flexible and configurable framework for developing the load prediction model.It is useful extension of linear model, able to capture complex non-linear relationships, still retaining the simple estimation procedure.Because of its favorable properties it is suitable, both, for simulation and forecasting.

II. GAM in R environment

The generalized additive model, GAM, has been introduced by Hastie and Tibshirani in [7] [8], followed by several important publications concerned with the supporting mathematical foundations and its computational improvements, e.g. as in [9] [10]. However, for practitioners the recommended reading is a book by Simon Wood, *Generalized Additive Models: An Introduction with R* that offers some useful hints for GAM implementation. The GAM is a statistical framework that, generally, covers a wide range of model structures with some common properties:

1) Additivity, as the response variable that can be from the broad range of distribution is obtained as a summation of individual effects, represented with one or more terms.

Table 1: Overview of the most important R packages for GAM model development

Package	Description	Reference
`mgcv`	Package for GAM model fitting, applicable for family of response distribution. The fitting is based on quadratically penalized likelihood type approach, (PIRLS).Several procedures for regulation of the model terms smoothness are available (GCV/UBRE/AIC/REML).The methods for variable selection, embedded into the fitting procedure are available	[20]
`gam`	Package for GAM model fitting, based on *back-fitting algorithm* that combines different smoothing and fitting methods. (Original GAM fitting version)	[7]
`GAMboost`	`Package for` fitting the GAM model by likelihood based boosting. It is recommended for models with a large number of terms. The procedure implements the mechanism for variable selection.	[11]
`Gamlss`	Package for GAM model development with the extension for Location, Scale and Shape (GAMLSS).It enables modeling all parameters of the response distribution via additive functions of the covariates.	[12]
`SpAM`	*Sparse Additive Model* for sparse estimator that simultaneously enforces smoothness of each term and sparsity across components.	[13]
`COSSO`	*Component Selection Shrinkage Operator* is a package for GAM variable selection. It imposes the lasso-type penalty term for constraining the sum of terms' norms and excludes them from the model through the optimization procedure.	[14]
`CASA`	*Component Automatic Selection in Additive models* variable selection method	[15] [16]
`gamsel`	*Generalized Additive Model Selection* uses penalized likelihood approach for fitting sparse generalized additive models in high dimension. The variable selection is embedded into the estimation procedure.	[17]
`gratia`	Improved *ggp*lot-based graphics and functions for visualization, adapted to the GAMs fitted using the `mgcv` package.	[18]
`mgcViz`	Advanced GAM visualization tool that enables interactive use, GAM additive structure exploitation, scale to large data sets that can be used in conjunction with a wide range of response distributions.	[19]

2) The model terms are smooth functions of a single or more variables and can be, either, linear or nonlinear, mostly represented with the splines. (The splines are typical GAM model building blocks. However, the GAM inherited, also, all types of terms from linear model.)

3) All model terms could be estimated simultaneously and in prediction, their individual contributions are simply summed up. In this regard, some recent developments of the GAM fitting methods, represented in [9] [10] enables its use for exploring very large data sets.

All tools for splines and variable selection, model fitting and validation as well as visualization needed for developing GAM model are available in R. Taxonomy of mostly used R packages for GAM, with the short description and references is given in Table 1. The list is by no means complete, as there are many other GAM-specific and common tools that support GAM development in R.

III. GAM Electricity Load Prediction

The GAM applicability to electrical load forecasting problem at the different network aggregation levels is confirmed in several publications. The good forecasting accuracy (about 1% MAPE error) for French load at the national level is reported in [21]. In addition, the French energy company *Electricite de France*, EDF, has implemented a GAM-based day-ahead load forecasting functionality, on a system level [9]. The improvement for online forecasting based on GAM is documented in [22].On the regional level, GAM performances were evaluated on data from National Electricity Market of Australia [23] and US utility company [24]. In [25] its application to the load forecasting at the substation level in France is shown. The flexibility of GAM model applicability to the load prediction problem is illustrated in the paper [26]. A platform for massive-scale load simulation in Smart Grids, starting from the individual households, over low- to medium-voltage network, up to the national level, is presented in this work. Three different scenarios covering the cases of network dynamic reconfigurations, abrupt and gradual changes in consumers' behaviour, and increasing capacity of distributed renewable energy sources is performed using GAM. Regarding the model, it has been shown that GAM is capable to simulate all aggregated loads for over 70 individual households.

A. GAM Load Prediction Model: A Toy Example

A simple GAM structure that implements basic load effects is used to demonstrate the GAM handily implementation in R. The load could be decomposed into the components on different temporal scales, related to the annual, weekly and daily cycles. Then one could suppose that in each instance in day, the load is a summation of the weekly and yearly effects, described with the spline `s(week)` and `s(year)`, respectively. The variable `week` is defined for each day in week, while `year` is for day in the year. Also, the weekly and

annual cycles interacts, as weekly pattern changes during the year, that could be described with the multivariate spline `ti(week, year)`, representing only variation in weekly pattern due to year changes. The temperature impact on the load level is incorporated into the model and this nonlinearity is modelled with the spline `s(temperature)`. Based on these assumptions, the load GAM model structure is defined with the sum of identified load components.

For testing the above model, a data set with the hourly loads and daily temperatures for two years, collected from the site [27] was used. The separate model for each hour in day was fitted by using `gam` function from the `mgcv` package for one year sample of data. We used the penalized regression splines, with the control knots, 7 for week (daily knots), 12 for year (monthly knots) and 10 for temperature. For fitting and smoothing methods, the PIRLS/REML was chosen, (see `mgcv`).

Each package for GAM fitting has the standard statistical tools for model validation, residual checking, partial residual checking, etc. (In `mgcv`, typicaly, function `summary`, `residuals`, `gam.check`, `qq.gam` are used for preliminary model checking). The specificity of GAM is that it, also, provides the results for each individual effect that could be observed separately. The `draw` function from the `gratia` package was used to visualize the results of fitting. The weekly, yearly and temperature-based smooths, for one non-working and one working hour, learned from the data, are shown in Figures 1 and 2, respectively.

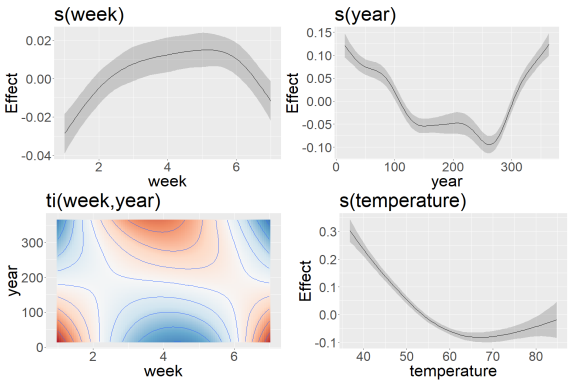


Figure 1: The load effects for the 1.h of the day

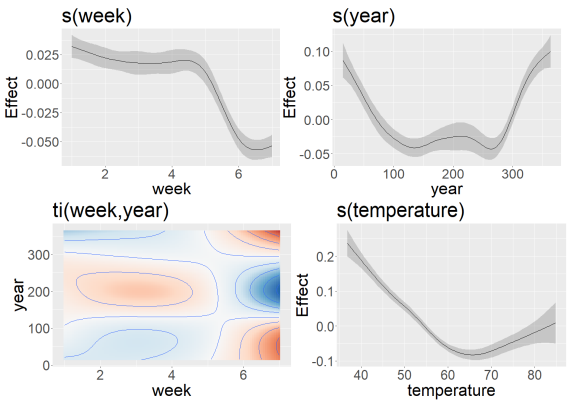


Figure 2: The load effects for the 14.h of the day

Some useful conclusions related to the differences in the weekly and yearly cycles, as well as response to a temperature, for different hours in the day could be drawn!

Based on the previously trained model and new values for temperature, the `mgcv` function `predict` produces the load predictions. The result for each hourly GAM model is combined for obtaining the whole day load prediction. For randomly chosen day, not included in model training, the results of prediction are represented in the Figure 3. The blue line is for predicted values and red for real one. The MAPE error for this day is 2.2%.

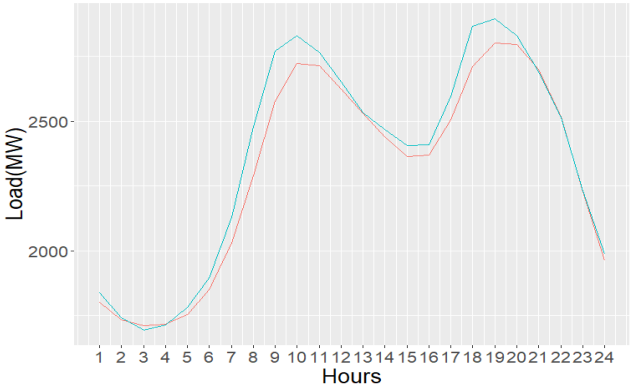


Figure 3: The results of prediction for one day

A stepping forward is in adding more assumption on the load model, which would improve the basic one. The terms for the load from the previous days, weeks, the temperature from the previous days, the other meteorological variables (humidity, wind, cloud cover), as well as economic parameters, could be incorporated into the model. To prevent the model overfitting, some of the variable selection procedure, listed in a Table 1, is available.

IV. Conclusion

The generalized additive model, GAM, and supporting resources for its development in R environment have been briefly overviewed. We are primary motivated by its potential to resolve the important actual problem of electricity load prediction. However, being flexible and easy to implement and interpret, providing high accuracy in prediction and free-software-based it is recommendable for wider application.

References

[1] T. Hong, Load Forecasting: White Paper and Case Study-NARUC, pubs.naruc.org, 2015

[2] J. R. Cancelo, A. Espasa, R. Grafe, "Forecasting the electricity load from one day to one week ahead for the Spanish system operator," *International Journal of Forecasting* 24(4), pp. 588–602,2008

[3] R. Wang, J. Wang, Y. Xu, "A Novel Combined Model Based on Hybrid Optimization Algorithm for Electrical Load Forecasting," *Applied Soft Computing Journal*, vol. 82, September 2019.

[4] J. G. Jetcheva, M. Majidpour, W. P. Chen, "Neural Network Model Ensembles for Building Level Electricity Load Forecasts," *Energy and Buildings*, vol. 84, pp. 214–223, 2014

[5] E. Ceperic, V. Ceperic, A. Baric, "A Strategy for Short-Term Load Forecasting by Support Vector Regression Machines," *IEEE Transactions on Power System*, Vol.28, Issue: 4 , Nov. 2013

[6] X. Liu, Z. Zhang, "A comparative study of the data-driven day-ahead hourly provincial load forecasting methods: From classical data mining to deep learning, "*Renewable and Sustainable Energy Reviews*, Vol. 119 , March 2020

[7] T. J. Hastie and R. J. Tibshirani, *Generalized additive models*, London: Chapman & Hall, 1990

[8] T. J. Hastie, R. J. Tibshirani, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, Springer series in statistics, Springer, New York, January 2017

[9] S. Wood, Y. Goude, S. Shaw, "Generalized additive models for large data sets,*" Journal of the Royal Statistical Society: Applied Statistics*, Vol.64 (1), pp. 139–155, 2015

[10] S. N. Wood, Z. Li, G. Shaddick, N. H. Augustin, "Generalized Additive Models for Gigadata: Modeling the U.K. Black Smoke Network Daily Data," *Journal of the American Statistical Association*, Vol.0, No.0,1-12,2017

[11] G. Tutz, H. Binder, "Generalized additive modeling with implicit variable selection by likelihood-based boosting," *Biometrics*, Vol. 62, pp. 961-971, December 2006

[12] R. A. Rigby, D. M. Stasinopoulos, "Generalized Additive Models for Location, Scale and Shape," *Journal of the Royal Statistical Society: Applied Statistics*, Vol. 54(3), pp. 507–554, 2005

[13] P. Ravikumar, J. Lafferty, H. Liu, L. Wasserman, "Sparse Additive Model," *Journal of the Royal Statistical Society: Statistical Methodology*, Vol. 71(5), November 2009

[14] Y. Lin, H. H. Zhang, "Component Selection and Smoothing in Multivariate Nonparametric Regression," *The Annals of Statistics*, Vol. 34(5), pp. 2272-2297, October 2006

[15] A. Antoniadis, X. Brossat, Y. Goude, J.M. Poggi, V. Thouvenot, *R package CASA: Component Automatic Selection in Additive models*, UseR! `a Aalborg, 2015

[16] V. Thouvenot, *Estimation et sélection pour les modèles additifs et application à la prévision de la con sommation électrique*, Ph.D. dissertation, Université Paris-Saclay, Français, 2015

[17] A. Chouldechova, T. Hastie, "Generalized Additive Model Selection*", Annals of Applied Statistics*, Stanford University, 2015

[18] G. Simpson, *Introducing gratia*, Oct. 23, 2014. [Online]. Available: https://fromthebottomoftheheap.net/2018/10/23/introducing-gratia/

[19] M. Fasiolo, R. Nedellec, Y. Goude, and S. Wood, "Scalable visualisation methods for modern Generalized Additive Models", *Journal of Computational and Graphical Statistics*, Vol.29(1), July 2019

[20] Wood, *Generalized Additive Models: An Introduction with R*, Second Edition, Chapman & Hall/CRC Texts in Statistical Science, 2006

[21] A. Pierrot, Y. Goude, "Short-term electricity load forecasting with generalized additive models", in *Proc. of ISAP power*, pp.593-600, 2011

[22] A. Ba, M. Sinn, Y. Goude, P. Pompey, "Adaptive learning of smoothing functions: Application to electricity load forecasting," *Advances in Neural Information Processing Systems*, 25, pp. 2519–2527, 2012

[23] S. Fan. R. J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model", *IEEE Transactions on Power Systems*, 27(1): 134–141, 2012

[24] R. Nedellec, J. Cugliari, Y. Goude, "GEFCom2012: Electricity load forecasting and back casting with semi-parametric models", *International Journal of Forecasting*, 2013

[25] Y. Goude, R. Nedellec, N. Kong. "Local short and middle term electricity load forecasting with semi-parametric additive models," *IEEE Transactions on Smart Grid*, 2013

[26] P. Pompey, A. Bondu, Y. Goude, M. Sinn "Massive-Scale Simulation of Electrical Load in Smart Grids Using Generalized Additive Models," in *Modeling and Stochastic Learning for Forecasting in High Dimensions*, Vol 217. pp. 193-212, Springer, 2015

[27] Available: http://fkeynia.googlepages.com/loaddata [Accessed 08.09.2020]

Pregled simulatora pogodnih za držanje nastave arhitekture i organizacije računara

Zaharije Radivojević[1], Saša Stojanović[1], Miloš Cvetanović[1]
[1]*Univerzitet u Beogradu – Elektrotehnički fakultet, Beograd, Srbija*
{zaki, stojsasa, cmilos}@etf.bg.ac.rs

Apstrakt: Kursevi iz oblasti arhitekture i organizacije računara obično obrađuju veći broj tema za čiju se praktičnu demonstraciju najčešće koriste softverski simulatori. Ovaj rad objedinjuje preglede simulatora koji su dostupni u otvorenoj literaturi. U radu su predstavljeni kriterijumi za evaluaciju na osnovu kojih su dati simulatori evaluirani, a koji mogu biti od koristi prilikom izbora simulatora koji bi se koristili u nastavi na nekom kursu. U radu je takođe prikazan izbor simulatora koji se koriste na Elektrotehničkom fakultetu Univerziteta u Beogradu.

Ključne reči: Arhitektura i organizacija računara; softverski simulatori; logičko projektovanje; evaluacija simulatora.

I. Uvod

Arhitektura i organizacija računara se izučava na mnogim univerzitetima širom sveta. Kursevi iz ove oblasti obično daju pregled računarske arhitekture i koncepata organizacije kao i uvid u rad tipičnog računarskog sistema. Pored toga, kursevi teže da što više upoznaju studente sa raznim alatima koji su im potrebni da bi mogli da vrše istraživanje i razvoj u ovoj oblasti nakon završetka studija.

Praktično korišćenje alata u nastavi obično obuhvata faze analize i logičkog dizajna računarskih sistema. U zavisnosti od toga da li se primenjuje obrazovni model od vrha ka dnu ili od dna ka vrhu alati treba da omoguće rad sa komponentama na odgovarajućem nivou apstrakcije. Ovakav rad je moguće najefikasnije ostvariti pomoću odgovarajućih simulatora računarskih sistema. Koristeći simulatore studenti imaju mogućnost da posmatraju ponašanje računarskih sistema na nivou funkcionalnih blokova ili komponenata, kao i na nivou interakcija između njih. Takođe studenti imaju mogućnost da na neki način modifikuju posmatrane sisteme. Simulatori se obično razlikuju po pitanju teme koju pokrivaju, nivou detalja na koji to čine, nivou do kojeg mogu biti modifikovani, načinu simulacije i načinu prikaza rezultata.

Ovaj rad objedinjuje preglede dela simulatora dostupnih u otvorenoj literaturi koje su autori razvijali ili evaluirali u dužem nizu godina. U radu su predstavljeni kriterijumi za evaluaciju na osnovu kojih su dati simulatori evaluirani, a koji mogu biti od koristi prilikom izbora simulatora koji bi se koristili u nastavi na nekom kursu. U radu je takođe prikazan izbor simulatora koji se koriste na Elektrotehničkom fakultetu Univerziteta u Beogradu.

Ostatak rada organizovan je na sledeći način. Glava II daje pregled simulatora pogodnih za nastavu arhitekture i organizacije računara. Glava III uvodi kriterijume za njihovu evaluaciju. Glava IV daje pregled korišćenih simulatora koji se koriste na Elektrotehničkom fakultetu Univerziteta u Beogradu koji su opisani u skladu sa kriterijumima evaluacije. Glava V zaključuje rad.

II. Pregled odabranih simulatora

U otvorenoj literaturi dostupan je veliki broj simulatora koji su pogodni za držanje nastave i učenje u oblasti arhitekture i organizacije računara sa posebnim fokusom na logičko projektovanje. U Tabeli 1. prikazana je dostupnost svakog od razmatranih simulatora i programski jezik u kome je isti razvijen. Simulatori su odabrani tako da pored arhitekture računara prikazuju i delove koji se odnose na organizaciju računara, dok simulatori koji to ne podržavaju nisu uključeni u analizu. Ovo znači da asembleri, emulatori i simulatori koji prikazuju samo izvršavanje programa bez ulaženje u detalje implementacije sistema koji simuliraju nisu uključeni u ovu analizu. Detaljan opis svih ovih simulatora nije dat u samom radu zbog obima koji bi takav rad imao već se detalji mogu naći u prethodnim istraživanja autora. Simulatori nastali u periodu do 2008. godine su opisani u istraživanjima koja su dostupna u radu [1] i na adresi [2], dok su istraživanja o simulatorima koji su nastali u periodu od 2009. od 2018. ili se u tom periodu inteziviralo njihovo korišćenje su dostupni u radu [3].

Razmatrani simulatori se grubo mogu razvrstati u tri grupe prema načinu na koji je moguće specificirati organizaciju računara koja se simulira. Ove tri grupe bi bile: simulatori koji omogućavaju kompletan dizajn organizacije računara; simulatori koji omogućava samo pregled postojeće organizacije računara i kojima je fokus na arhitekturi računara; i simulatori koji se nalaze između ove dve krajnosti koji imaju mogućnost za konfigurisanje jednog dela organizacije računara.

Simulator iz prve grupe omogućavaju korisniku da najpre napravi proizvoljnu konfiguraciju računarskog sistema, a da je potom simuliraju. Iako su neki simulatori napravljeni kao alati za dizajn digitalnih logičkih kola opšte namene dok si drugi eksplicitno napravljeni za proučavanje arhitekture i organizacije računara mogu se tretirati kao jedinstvena grupa. Pravljenje novih konfiguracija računarskih sistema se obično sastoji od pravljenja komponenata i njihovog povezivanja u složenije celine, a kasnije i složeno hijerarhijsko povezivanje datih komponenata. Ono po čemu se pojedini simulator razlikuju je dostupnost unapred pripremljenih komponenata, kao i nivoa apstrakcije samih komponenata koje se simuliraju, npr. nivo logičkih kola, ili nivo čitave periferije kao što su ekrani ili tastature.

DOI: 10.5281/zenodo.4273750

Tabela 1. Karakteristike odabranih simulatora

		Dostupnost		
		Komercijalna	**Besplatan**	**Otvoreni kod**
Programski jezik	**Java**	-	COCONUT, CPU Sim, DLD-VISU, EASE, Easy CPU, EDCOMP, EDUCache, EduMIPS64, HASE, Java based Virtual Lab, JCachesim, MARIE, MIPS X-Ray, SDLDS, Simple CPU Architecture, VSMIS	JHDL, Logisim
	C, C++	-	FastCache, SIMCA, SimFlex, SimOS, SimpleScalar	DigLC2, DLXview, RSIM
	Pajton/C++	Simics	-	M5
	Ostalo	CCSTUDIO, CodeWarrior, ISE Design Suite, Quartus II, Virtual Vulcan	DEEDS, ESCAPE, HASE-Dinero, RM, SMOK, ViLLE plug-in, Visual CPU simulator, VSDS	-

Simulatori iz druge grupe omogućavaju korisniku da simulira unapred definisane sisteme. Ovi simulatori obično imaju mogućnost postavljanja određenih parametara pre početka same simulacije. Parametrima je obično moguće definisati broj korišćenih periferija, širinu memorijske reči, početne vrednosti registara i memorijskih lokacija i tome sličnim detaljima. To znači da se ovim simulatorima može jednostavno napraviti veći broj konfiguracija koje bi poslužile za testiranje određenih programa. Izveštaji o izvršavanju simuliranih programa su obično dati u formi tabela sa sadržajima registara i memorijskih lokacija, vremenskim dijagramima, ili tekstualnih datoteka sa različitim tragovima izvršavanja.

Simulatori iz treće grupe korisniku pružaju mogućnosti koje su između prethodne dva opisane grupe simulatora. Ovi simulatori imaju delove koji su fiksne strukture i ne mogu se menjati već im se samo mogu postavljati početni parametri. Sa druge strane ovi simulatori omogućavaju izvestan stepen fleksibilnosti u opisu organizacije računara time što se određeni delove mogu u potpunosti redefinisati i opisati koristeći raspoložive komponente. Ovi simulatori imaju za cilj da naprave kompromis između kontrole prilikom projektovanja i performansi prilikom izvršavanja simulacija.

Sve tri grupe simulatora su zastupljene u nastavi na kursevima koji pokrivaju oblast arhitekture i organizacije računara, ali izbor simulatora zavisi od tema koje se pokrivaju na konkretnom kursu i načina na koji se teme obrađuju. Prva grupa simulatora je pogodna kada je sve delove organizacije potrebno prikazati na nivou transfera između registara i gde se simulacije obavljaju na relativno malom broju taktova. Na osnovnim kursevima ova grupa simulatora je pogodna za logičko projektovanje računara i njegovih sastavnih delova. Druga grupa simulatora je namenjena temama kod kojih je važno posmatrati izvršavanje čitavih programa, što bi u slučaju da se koriste simulatori prve grupe bilo vremenski zahtevno. Posmatranje čitavih programa daje uvid u performanse izvršavanja programa na određenoj arhitekturi i na određenoj organizaciji računara. Potreba za simulatorima iz ove grupe obično postoji prilikom obrade tema arhitekture računara, keš memorije, protočne obrade i sličnih koje se obično pojavljuju na naprednim kursevima iz ove oblasti. Treća grupa simulatora namenjena je kursevima na kojima ima dovoljno vremena da se nekoj temi posveti više vremena i da se omogući da studenti samostalno predlažu rešenja u datoj oblasti. Tom prilikom studenti ne moraju da projektuju čitav sistem od početka već samo delove od značaja i to sa dovoljnim nivoom detalja. Na ovaj način se pravi kompromis između toga koliko vremena posvetiti pojedinoj oblasti i vremenu koje je potrebno za izvršavanje simulacije.

III. Evaluacija odabranih simulatora

Razmatrani simulatori se pored načina moguće specifikacije organizacije računara koja se simulira mogu razmatrati i sa drugih aspekata od važnosti za obrazovni proces. Obrazovni proces može pored oblasti koje se obrađuju diktirati i način prikaza rezultata simulacije kao i postupak dolaska do tih rezultata. U Tabeli 2. dat je prikaz razmatranih simulatora u kontekstu svih ovih kriterijuma.

Neki od ovih simulatora su razvijeni za uvodne kurseve iz oblasti arhitekture i organizacije računara dok su ostali namenjeni naprednijim kursevima. U zavisnosti od toga da li se primenjuje obrazovni model od vrha ka dnu ili od dna ka vrhu simulatori treba da omoguće prikaz osnovnih komponenti pogodnih za izgradnju jednostavnih računarskih sistema, ili složenih komponenti pogodnih za opis sistema na konceptualnom nivou, kao i komponenti koje opisuju prelaze između ovih nivoa. Nivoi za podelu simulatora su: G1 – komponente omogućavaju kompletan dizajn; G2 – komponente omogućavaju pregled već realizovanih složenih sistema i njihovu parametrizaciju; G3 – parametrizacija i konfigurabilnost komponenata u određenim delovima simuliranog sistema dok je ostatak simuliranog sistema nepromenljiv.

Kursevi na kojima se razmatraju teme iz oblasti arhitekture i organizacije računara se dosta razlikuju na pojedinim univerzitetima, ali je ta razlika uglavnom rezultat broja kurseva i fonda časova na datom univerzitetu koji su posvećeni toj oblasti. Na kursevima se mogu uočiti četiri grupe povezanih tema za koje razmatrani simulatori imaju dostupnu podršku, pri čemu određeni simulatori mogu imati podršku i za veći broj grupa tema. U slučaju nekih simulatora iz grupe G1 ovu podršku može i sam korisnik razviti. Razmatrane grupe tema su: AR – Arhitektura računara; MEM – Memorijski podsisitem; UI – Ulazno-izlazni podsisitem; OR – Organizacija računara.

Mnogi simulatori imaju veoma dobru grafičku prezentaciju pogodnu za predstavljanje detalja simuliranog sistema, dok ima i onih simulatora koji svoje rezultate predstavljaju u tekstualnoj formi. Što se tiče rezultata simulacije neki simulatori omogućavaju samo uvid u konačni rezultat simulacije dok drugi omogućavaju uvid i u međurezultate simulacije. Grupe za

podelu simulatora su: ZK - Zbirni prikaz konačnih rezultata; PK - Pojedinačni prikaz konačnih rezultata; ZM - Zbirni prikaz međurezultata; PM - Pojedinačni prikaz međurezultata.

Kada se razmatraju pojedine oblasti korisnika mogu zanimati rezultati nastali u različitim vremenskim trenucima. Na primer kada se razmatraju teme iz arhitekture računara korisnika obično zanima ponašanje sistema nakon obavljenih pojedinih instrukcija dok u slučaju razmatranja detalja organizacije računara korisnika zanimaju rezultati nakon svakog takta izvršavanja simulacije. U nekim slučajevima, kao što je rad keš memorije, korisnika zanimaju samo podaci dobijeni nakon kompletnog izvršavanja programa. Grupe za podelu simulatora su: Takt – Praćenje simulacije na nivou takta; Instrukcija – Praćenje simulacije na nivo pojedinačnih instrukcija; Program – Praćenje simulacije nakon izvršenja čitavog programa.

IV. Pregled korišćenih simulatora

Odabirom kriterijuma od važnosti mogu se odrediti simulatori koji bi bili kandidati za uključivanje u nastavu na pojedinim kursevima. Odabir se sprovodi u zavisnosti od kriterijuma uslovljenih pristupom čitavoj oblasti (primenjenog obrazovnog modela i oblasti koje kurs obuhvata), kao i od kriterijuma uslovljenih nivoom očekivanih rezultata simulacije (načina za prikaz i praćenje rezultata simulacije). Na primer, na Elektrotehničkom fakultetu u Beogradu oblast arhitekture i organizacije računara se obrađuje na više povezanih kurseva pokrivajući sve navedene teme koristeći model od dna ka vrhu. Na uvodnim kursevima postoji potreba da se posmatra izvršavanje na nivou pojedinih taktova, na narednim kursevima na nivou instrukcija, dok je na naprednim kursevima cilj praćenje rezultata na nivou programa.

Osnovi računarske tehnike 1 (ORT1) je kurs na prvoj godini u drugom semestru. Kurs se bavi logičkim projektovanjem i program predmeta obuhvata analizu i sintezu kombinacionih i sekvencijalnih mreža, standardne module, kao i osnove operacione i upravljačke jedinice računara. Za demonstraciju svih ovih koncepata se u prošlosti koristio simulator VSDS [4], dok se sada koristi simulator Logisim [5], zbog jednostavnijeg prikaza povratne sprege. Na kursu je važno praćenje izvršavanja simulacije na nivou takta i projektovanje jednostavnih komponenti i njihovo povezivanje u hijerarhijske mreže.

Praktikum iz osnova računarske tehnike (PORT) je kurs koji se istovremeno održava sa kursom ORT1. Predmet predstavlja dopunu kursa ORT1 primerima praktične primene obrađenih koncepata i oblasti. Kako bi se obrađivani primeri implementirali na dostupnom hardveru (Altera Cyclone III na DE0 razvojnoj ploči i Altera Cyclone V na DE0-CV razvojnoj ploči) koristi se razvojno okruženje Quartus II [6]. Simulator dostupan unutar razvojnog okruženja predstavlja profesionalni alat i nije prilagođen uvodnim kursevima na kojima se razmatra logičko projektovanje i arhitektura i organizacija računara te se sama simulacija obavlja koristeći pomenuti hardver.

Osnovi računarske tehnike 2 (ORT2) je kurs na drugoj godini u trećem semestru. Predmet se bavi nastavkom logičkog projektovanja uređaja (jednostavni interfejsni uređaji), arhitekturom računara (izvršavanjem instrukcija, skupom instrukcija i načinima adresiranja) i organizacijom računara (logičkim projektovanjem operacione i upravljačke jedinice računara). Na početnom delu kursa za logičko projektovanje uređaja se koristi isti hardver i razvojno okruženje kao na kursu PORT, dok se za potrebe arhitekture i organizacije računara koristi simulator COCONUT [3] kojim se projektuju deo procesora (deo za dohvatanje i dekodovanje instrukcija) i deo upravljačke jedinice (mikroprogramske realizacije), a simulacije se prate na nivou takta.

Arhitektura računara (AR) je kurs na drugoj godini u četvrtom semestru. Kurs se bavi arhitekturom računara (složene instrukcije), mehanizmom prekida, osnovama memorijskog podsistema, i osnovama ulazno-izlaznog podsistema. Na kursu se koristi simulator EDCOMP [7] pomoću koga studenti prate izvršavanje pojedinih instrukcija, prete prekide i pristupe periferijama. U situacijama kada se od studenata traži da prate izvršavanje zadatih programa onda se simulacije obavljaju na nivou takta, ali u situacijama kada studenti samostalno pišu programe onda se simulacije obavljaju na nivou instrukcija ili čitavog programa.

Tabela 2. Evaluacija odabranih simulatora

Naziv	**Autor**	**Konfigurabilnost**	**Teme**	**Prikaz rezultata**	**Vremenski kvant**
CCSTUDIO	Texas Instruments Inc.	G2	AR,UI,OR	PM	Instrukcija
COCONUT	University of Belgrade, Serbia	G3	AR,MEM,UI,OR	PM	Takt
CodeWarrior	Freescale Semiconductor, Inc.	G2	AR,UI,OR	PM	Instrukcija
CPU Sim	Colby College, USA	G2	AR,OR	PM	Takt

Naziv	Autor	Konfigurabilnost	Teme	Prikaz rezultata	Vremenski kvant
DEEDS	University of Genoa, Italy	G1	OR	PM	Takt
DigLC2	University Paris-Sud, France	G2	AR,OR	PM	Instrukcija
DLD-VISU	Technical University of Darmstadt, Germany	G1	OR	PM	Takt
DLXview	Purdue University, USA	G2	AR,OR	PM	Takt
EASE	University of Northern British Columbia, Canada	G2	AR	PM	Instrukcija
Easy CPU	Holon Institute of Technology, Israel	G2	AR,OR	PM	Takt
EDCOMP	University of Belgrade, Serbia	G2	AR,MEM,UI,OR	PM	Takt
EDUCache	Ss. Cyril and Methodius University, FYRM	G2	MEM	PM	Instrukcija
EduMIPS64	University of Catania, Italy	G2	AR,OR	PM	Takt
ESCAPE	Ghent University, Belgium	G2	AR,OR	PM	Takt
FastCache	University of Wisconsin Madison, USA	G2	MEM	ZK	Program
HASE	University of Edinburgh, UK	G1	MEM,UI,OR	PM	Takt
HASE-Dinero	University of Edinburgh, UK	G2	AR,MEM,OR	PM	Takt
ISE Design Suite	Xilinx Inc.	G1	MEM,UI,OR	PM	Takt
Java based Virtual Lab	SRM University, India	G2	-	PM	Instrukcija
JCachesim	University of Siena, Italy	G2	AR,MEM	PK	Instrukcija
JHDL	Brigham Young University, USA	G1	MEM,UI,OR	PM	Takt
Logisim	Hendrix College, USA	G1	OR	PM	Takt
M5	The University of Michigan, USA	G1	AR,MEM,UI,OR	ZK	Takt
MARIE	Mackenzie Presbyterian University, Brazil	G2	AR	PM	Instrukcija
MIPS X-Ray	Federal Center for Technological Education of Minas Gerais, Brazil	G2	AR,OR	PM	Takt
Quartus II	Altera Corporation	G1	MEM,UI,OR	PM	Takt
RM	University of Catalonia (UPC), Spain	G2	AR,OR	PM	Takt
RSIM	Rice University Houston Texas, USA	G2	AR,MEM,OR	ZK	Takt
SDLDS	University of Belgrade, Serbia	G1	OR	PM	Takt
SIMCA	University of Minnesota, USA	G2	AR,MEM,OR	ZK	Program
SimFlex	Carnegie Mellon University, USA	G2	AR,MEM,UI,OR	ZK	Instrukcija
Simics	Virtutech AB Stockholm, Sweden	G1	AR,MEM,UI,OR	ZK	Instrukcija
SimOS	Stanford University, USA	G2	AR,UI,OR	ZK	Program
Simple CPU Architecture	Redeemer University College, Canada	G2	AR,OR	PM	Takt
SimpleScalar	University of Wisconsin-Madison, USA	G2	AR,OR	ZK	Takt
SMOK	University of Washington, USA	G1	MEM,OR	PM	Takt
ViLLE plug-in	University of Turku, Finland	G1	OR	PM	Takt
Virtual Vulcan	YOERIC Corporation	G1	-	PM	Takt
Visual CPU simulator	Kagawa University, Japan	G2	AR,OR	PM	Takt
VSDS	University of Belgrade, Serbia	G1	AR,MEM,UI,OR	PM	Takt
VSMIS	University of Belgrade, Serbia	G2	MEM	PM	Takt

Arhitektura i organizacija računara 1 (AOR1) je kurs na trećoj godini u petom semestru. Kurs se bavi memorijskom hijerarhijom (keš memorija, virtuelna memorija, preklapanje pristupa memorijskih modula) i protočnom obradom. Na kursu se pored simulatora koji obuhvataju teme keš memorije, virtuelne memorije i protočne obrade koristi i simulator koji opisuje rad računarskog sistema sa preklapanjem pristupa memorijskim modulima VSMIS [8]. Svi ovi simulatori se prvo koriste na nivou takta kako bi se studenti upoznali sa radom sistema, a kasnije se simulacije obavljaju na nivou kompletnih programa.

V. Zaključak

Većina kurseva iz oblasti arhitekture i organizacije računara na univerzitetima širom sveta uključuje ne samo teorijsku nastavu već i praktičan rad koristeći simulatore računarskih sistema. Ovaj rad je objedinio preglede dela simulatora dostupnih u otvorenoj literaturi koje su autori razvijali ili evaluirali u dužem nizu godina. Na početku je dat pregled simulatora, nakon čega su dati kriterijumi evaluacije po kojima su dati simulatori evaluirani i na kraju je dat pregled simulatora koji se koriste na kursevima arhitekture i organizacije računara na Elektrotehničkom fakultetu u Beogradu.

Evaluacija je pokazala da nema jednog simulatora koji se može koristiti bez obzira na pristup izučavanju odabranih oblasti (primenjenog obrazovnog modela i oblasti koje kurs obuhvata) i od očekivanog nivoa detalja unutar simulatora (načina za prikaz i praćenje rezultata simulacije). Koristeći kriterijume opisane u radu moguće je odrediti simulatore koji bi bili kandidati za uključivanje u nastavu na pojedinim kursevima prilikom izučavanja pojedinih tema. Jedan takav izbor simulatora je predstavljen na primeru Elektrotehničkog fakulteta Univerziteta u Beogradu gde se oblast arhitekture i organizacije računara izučava u okviru 5 kurseva i gde se koristi 10 različitih simulatora od kojih su 6 pomenuti u ovom radu.

Zahvalnica

Rad na ovom projektu je delimično bio finansiran od strane Ministarstva prosvete, nauke i tehnološkog razvoja Republike Srbije (III44009 i TR32047), kao i Fonda za nauku Republike Srbije (AVANTES).

Reference

[1] B. Nikolic, Z. Radivojevic, J. Djordjevic and V. Milutinovic, "A Survey and Evaluation of Simulators Suitable for Teaching Courses in Computer Architecture and Organization," IEEE Transactions on Education, vol. 52, no. 4, pp. 449-458, Nov. 2009, doi: 10.1109/TE.2008.930097.

[2] B. Nikolic, Z. Radivojevic, J. Djordjevic, and V. Milutinovic, "Survey of simulators," Available from: http://rti.etf.bg.ac.rs/rti/ri3aor/Simulators/index.html (pristupano oktobar 2020).

[3] Z. Radivojevic, Z. Stanisavljevic, M. Punt, "Configurable simulator for computer architecture and organization," Computer Applications in Engeneering Education, vol. 26, no. 5, pp. 1711– 1724, Sep. 2018, doi: 10.1002/cae.22034.

[4] N. Grbanovic, B. Nikolic, J. Djordjevic, "The VSDS environment based laboratory in computer architecture and organisation," Computer Applications in Engeneering Education, vol. 19, no. 4, pp. 685–696, Nov. 2011, doi: 10.1002/cae.20353.

[5] C. Burch, "Logisim: A graphical system for logic circuit design and simulation," Journal on Educational Resources in Computing, vol. 2, no. 1, pp. 5–16, Feb. 2002, doi: 10.1145/545197.545199.

[6]Altera Corporation, "Quartus II Software," Available from: http://www.altera.com/products/software/products/quartus2/qts-index.html (pristupano maj 2008).

[7] J. Djordjevic, B. Nikolic, A. Milenkovic, "Computer Architecture Laboratory," Available from: http://rti.etf.bg.ac.rs/rti/ef2ar/labvezbe/CISCA.zip (pristupano oktobar 2020).

[8] K. Milenković, Ž. Stanisavljević, J. Đorđević, "Visual software system for memory interleaving simulation," Serbian Journal of Electrical Engineering, vol. 14, no. 1, pp. 51–66, Feb. 2017, doi: 10.2298/SJEE1701051M

Towards Deterministic Industrial Internet of Things Networking

Gordana Gardašević, Milica Lekić
University of Banja Luka
Faculty of Electrical Engineering
Banja Luka, Bosnia and Herzegovina
{gordana.gardasevic, milica.lekic}@etf.unibl.org

***Abstract*—The challenge raised by the explosion of Internet of Things (IoT) scenarios and applications is permanently shaping the networking and communications landscape, with a significant social impact. The ongoing research activities in the field of industrial IoT (IIoT) are directed towards designing deterministic wireless networks and reliable transmission protocols, but there are still many issues requiring a global consensus before the final deployment. The paper discusses the requirements of next-generation IIoT applications based on Wireless Sensor Networks (WSNs) standards and technologies. A particular attention is given to prerequisites for deterministic networking in IIoT environments, as well as to benefits of using open-hardware and open-software IoT platforms.**

***Index Terms*—Industrial IoT, IEEE 802.15.4e, IEEE 802.15.4g, TSCH, 6TiSCH, scheduling, OpenMote-B**

I. Introduction

The development of industrial Internet of Things (IIoT) standards is a main prerequisite for the transformation of traditional systems into a new generation known as Industry 4.0. This transition can be explained by the following statement: "Industry 4.0 is a concept that relies on communication technologies, automation and production processes, as well as on efficient data transmission in industrial environments. Within modularly structured smart factories, Cyber-Physical Systems (CPSs) will enable the performance monitoring, the creation of virtual resources and decentralized decision-making, as well as human-machine interfaces (HMI) in real time." [1], [2].

Intensified research activities have been focused on the design of energy-efficient and reliable industrial Wireless Sensor Networks (WSNs). These networks provide the infrastructure for a large number of applications ranging from automation and control of industrial processes, to large smart grid systems in electric power plants and intelligent transport networks. WSNs provide numerous advantages, primarily in terms of low implementation cost, flexibility and scalability, as well as network operational efficiency. The development of sensor technologies is one of the key elements for mass production of low cost devices that can be used efficiently in different environments. Current predictions indicate that the number of smart devices worldwide will reach 70 billion by 2025. This dramatic increase in the number of IoT devices requires new forms of energy generated from the immediate environment. The new generation of industrial applications will be driven by advanced technologies, such as cloud/edge/fog computing, big data analytics, artificial intelligence and machine learning, Software Defined Networking (SDN), etc. It is expected that their use will significantly reduce operating costs and increase work efficiency.

In the past ten years, the Internet Engineering Task Force (IETF) has been standardized key IPv6-based protocols adapted to constrained devices in WSNs, such as IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN), Constrained Application Protocol (CoAP), Routing Protocol for Low-Power and Lossy Networks (RPL), IPv6 over Time Slotted Channel Hopping - TSCH mode of IEEE 802.15.4e (6TiSCH) and others [3]. The networking of constrained devices is also the subject of standardization activities of the IETF CoRE (Constrained RESTful Environments) working group. The integration of sensor devices into the IPv6 Internet environment is based on the concept known as the Semantic Web of Things that allows access to sensor data through standardized Web interfaces.

The support of real-time communications over license-free bands is a challenging task and requires a strict timing control within the IIoT network. There is an increased effort in terms of research and standardization activities towards deterministic Medium Access Control (MAC) protocols. As a promising solution, the Time-Slotted Channel Hopping (TSCH) as a synchronous MAC protocol was introduced in IEEE 802.15.4e standard [4]. It has attracted significant attention from the research community as it promises more reliable and predictable wireless networking, particularly for the challenging IIoT environments. Following this concept and the urgent need for the standardized architectures, the 6TiSCH working group (WG) was formed with the aim of defining a single protocol-stack adapted to industrial applications. 6TiSCH is based on open standards, with support for different scheduling strategies, as well as for deterministic packet switching over the TSCH MAC sublayer [5]. Moreover, the 6TiSCH mechanisms are of particular importance for the further adoption of

DOI: 10.5281/zenodo.4064277

IPv6 in industrial standards [6].

The organization of the paper is as follows. Section II presents an overview of emerging IIoT requirements and applications. Section III introduces the concept of deterministic networking and addresses challenges of wireless communications in IIoT environments. Promising technologies and solutions, such as synchronous MAC scheduling in IEEE 802.15.4e - TSCH, and multimodal and re-defined physical layer in IEEE 802.15.4g have been discussed. Finally, a conclusion highlights key findings of this survey by indicating open issues and future trends in the emerging field of IIoT.

II. Requirements of IIoT Networking

Characteristics of radio-communication links in the industrial automation field are such that performance and reliability guarantees must be provided. Design objectives are directly related to the specifics of the applications and industrial systems, where the following parameters are of key importance:

- efficiency of available resources (battery power, processor capabilities, bandwidth, etc.),
- effective mechanisms to prevent network congestion,
- reliable time synchronization,
- real-time transmission,
- transmission security and safety,
- modular design and scalability,
- interoperability with existing infrastructure, and
- Quality of Service (QoS) support.

Depending on the application requirements (reliability, time constraints, MAC mechanisms), the following types of traffic can be distinguished in industrial networks:

- Traffic in emergency situations. It represents the category of the highest priority. This type of traffic commonly has an asynchronous character, and is generated by special circumstances, e.g. explosions, plant fires and similar major anomalies in industrial environments. Modeling of this type of traffic implies fulfillment of strict time requirements and high reliability of work, so it is necessary to define priorities in the scheduling and access to the communication channel.
- Regular traffic in closed control loops. In these systems, the sampling rate of sensor data is significantly higher compared to other types of IIoT networks, and the traffic is generated periodically. At the same time, the interruption of communication leads to the instability of the complete control process. Performance optimization is one of the basic requirements in the operation of this type of network.
- Traffic in open control loops. Compared to the previous two categories, the restrictions are not so strict, because the occurrence of an error in the system will not have consequences due to slow changes in the control process. An example of this type of control are frequency components in the generator units of industrial plants.
- Surveillance traffic. This type of traffic is mostly one-way, because it realizes process monitoring by collecting appropriate information in process control and automation. Based on this information, improvements and upgrades of the system are planned. Traffic can be classified in the category of low priority, because occasional packet losses can be tolerated.

The process of selection and design of particular MAC and routing protocol must take into account the characteristics of the wireless links and the capabilities of the sensor device. Since devices usually are battery-powered, protocols that require a large number of transactions will very quickly reduce the available resources. Variations in link quality have a direct impact on protocol design, especially if transmission time characteristics are taken into account. Moreover, the dynamic nature of links and node metrics inevitably reflects on protocol design. Interference, multipath fading, changes in node power supply, current CPU overload, etc. have the significant impact on transmission quality.

III. Concept of Deterministic Networking

Deterministic networking (DetNet) represents an important element of IIoT system design. The basic idea is to limit packet transmission delays, while ensuring very high transmission reliability, which is of particular importance for M2M operations (for example, process automation in industry, audio and video streaming, vehicle control, etc.). The IETF DetNet WG defines mechanisms to implement deterministic data paths for real-time applications that require very low packet loss, low latency variation (jitter), and guaranteed latency limits [7]. In these networks, QoS requirements can be expressed based on the following parameters:

- Minimum and maximum end-to-end delay that allows packet delivery within the defined delay limits and jitter reduction/elimination.
- Probability of packet loss, taking into account the state of nodes and links in the network.
- Guarantees regarding deterministic delay can be achieved by eliminating collisions between nodes and appropriate access to time slots. Using multiple channels provides a support for multiple transmissions in the same time period, based on different channel offsets, thus increasing network capacity. This avoids the effects of interference and multipath fading, thus increasing the reliability of communications and reducing the energy required to retransmit packets.

Energy efficiency and communication reliability are often conflicting requirements. Novel solutions such as Wake-up Radio (WuR) allow nodes to work at a power

consumption level that is 1000 times lower than that of the traditional radio [8].

The concept of time-sensitive networking is commonly associated with applications from the domains of industrial automation and the automotive industry [9]. These requirements should overcome some of the limitations of traditional Internet technologies for the realization of transmission at the MAC layer. In the case of classic industrial automation networks, the transmission is usually performed at distances of up to several kilometers, and the number of hops can vary from e.g. 5 in indoor environments, up to 70 hops in large plants. In such environments, it is necessary to provide real-time traffic control, as well as the transmission of video content and large files. One of the key requirements in these networks are precise time synchronization and the deterministic delay.

A. TSCH Scheduling

The TSCH mode on the MAC sublayer is organized to support application requirements such as industrial automation and process control [10], [11]. Common areas of application are robotics, oil and gas industry, healthcare, transport systems, smart grid systems, etc. TSCH combines time synchronization with channel hopping to support deterministic delay guarantees, communication reliability (i.e., resistance to interference and multipath fading), and higher network throughput in comparison to other MAC protocols. The time-slotted operation of TSCH reduces collisions, enables ultra low-power communications, and provides deterministic properties on wireless medium. TSCH enables an ultra-low duty cycle less than 0.1%, thus extending battery life by up to 10 years. The channel hopping is a well known and efficient technique to combat multipath fading and co-channel interference. Current 6TiSCH implementations use the 2.4 GHz band, with 16 frequencies available.

The communication in a 6TiSCH network is orchestrated by a schedule. A slotframe consists of a matrix of cells of equal length (typically 10 or 15 ms), each cell being defined by a pair of timeslot and channel offsets. Slotframes repeat over time to enable nodes to have periodic access to the medium. TSCH defines two types of cells: dedicated and shared. A dedicated cell is contention-free providing that only one transmitter can send a packet. If cells are shared between multiple nodes, than the random access mechanism is applied.

The 6TiSCH protocol stack is a result of joint IETF and ETSI effort to provide all relevant mechanisms for routing, transport, security and application-level interface. The architecture and mechanisms have been developed in order to provide an open and standardized communication stack and to speed up the adoption of IPv6 in industrial standards. The scheduling in 6TiSCH networks has attracted considerable research interest [10], [12]–[14].

B. Using Open-hardware and Open-software in IEEE 802.15.4.g-based Experimentation

The proliferation of low-power and low-cost devices with IP connectivity based on open protocols and architectures results in explosion of diverse IoT applications. In recent years, research and education in the emerging field of telecommunication technologies have been benefited from using open-software and open-hardware platforms. Therefore, for the experimentation purposes, we use the OpenMote-B hardware devices and RIOT OS.

The OpenMote-B is an open-hardware platform for IIoT applications based on IPv6 protocol stack, Fig. 1. This board consists of a Texas Instruments CC2538 System on Chip (SoC) and an Atmel AT86RF215 dual-band radio transceiver. The CC2538 includes an ARM Cortex-M3 micro-controller (32 MHz, 32 kB RAM, 512 kB Flash) and a IEEE 802.15.4-compatible radio transceiver. The AT86RF215 completely supports the IEEE 802.15.4g standard and provides data transmission in sub-GHz and 2.4 GHz. More information on this board can be found in [15].

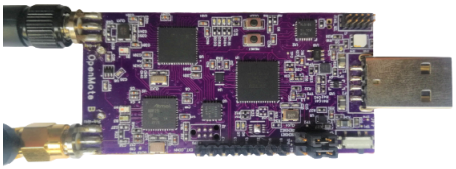


Fig. 1. The OpenMote-B hardware platform.

RIOT is an open-source operating system (OS) for memory constrained systems with focus on the wireless low-power IoT devices [17]. Memory size is around 10 KB and it is based on micro-kernel and modular architecture (8, 16, 32-bit). RIOT OS provides support to protocol stacks such as IPv6, 6LoWPAN, as well as standard protocols: RPL, UDP, TCP and CoAP. RIOT runs on several platforms/architectures including embedded devices as well as common PCs. RIOT OS source code is available on GitHub repository.

The IEEE 802.15.4g standard has developed a new set of physical layers (PHYs) for outdoor low data rate wireless Smart Utility Networks (SUN) applications [18]. It defines three PHYs: Multi-rate Multi-regional Offset Quadrature Phase Shift Keying (MR-OQPSK), Multi-rate Multi-regional Frequency Shift Keying (MR-FSK) and Multi-rate Multi-regional Orthogonal Frequency Division Multiplexing (MR-OFDM).

Some of the key IEEE 802.15.4.g specifications are:

- Operation in free 700-1000 MHz and 2.4 GHz bands,
- Data rates from 40 kb/s to 800 kb/s,
- Provision of a communication in multiple frequency bands and use of multiple data rates,
- Payload maximum length is 2047 bytes (B) where a complete IPv6 packet can be transmitted without fragmentation, and

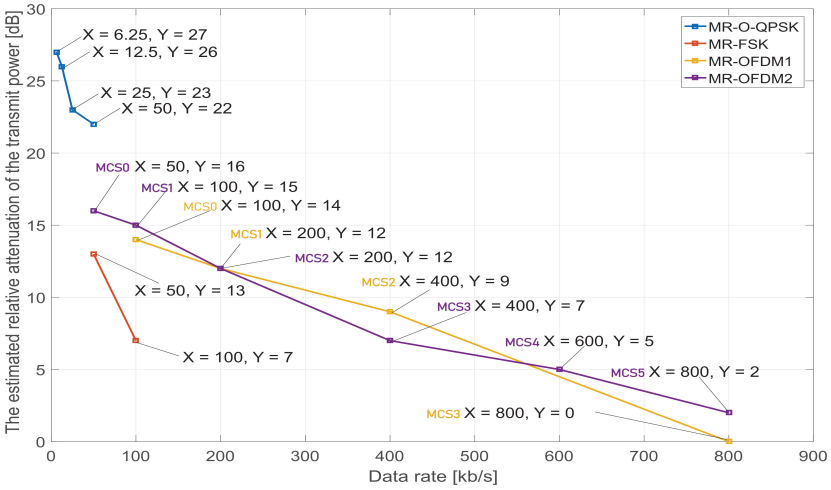


Fig. 2. The estimated relative attenuation of TX power when the connection is broken [16].

- Coexistence with other systems operating in the same band (IEEE 802.11, 802.15 and 802.16).

Given the raising interest in the IIoT networks, characterized by low-rate and low-power, exploration of the range and reliability of the different modulation types helps in choosing a proper PHY depending on the application requirements. In [16], the experimental performance evaluation of the IEEE 802.15.4g applications has been conducted in three test scenarios using the OpenMote-B hardware and RIOT software platforms. The overall results present a dataset obtained from the deployment of 2 nodes using the IEEE 802.15.4g SUN modulations. The following metrics were used for data traffic analysis: packet loss [%], average Received Signal Strength Indicator (RSSI) [dBm], min/avg/max Round Trip Time (RTT) [ms] and PHY configuration.

1) First test scenario: Measurements have been conducted in a Faraday cage which has ensured idealized conditions. The communication performance has been tested both in sub-GHz (863-870 MHz) and 2.4 GHz frequency bands. The purpose of this scenario has been to verify accuracy of used hardware platform. This experiment implies that MR-FSK is the most robust PHY in both bands according to the RSSI value analysis. An interesting observation is that the RSSI value difference between MR-O-QPSK and MR-OFDM2 is less than 2 dB. Taking into account that the signal bandwidth of these PHYs is 5 MHz for MR-O-QPSK and 0.8 MHz for MR-OFDM2, these results prompt the MR-OFDM usage in low-power and low-rate IIoT.

2) Second test scenario: Here, measurements have been taken in sub-GHz band and in an in-door environment with 10 m device spacing. The aim of this setup is to evaluate PHY configuration performances in real conditions. Here we noticed an enormous human impact on communication link with variations of packet losses between 20% and 90%. In this scenario, it has been measured relative attenuation at the transmitter at the moment of communication interruption (packet losses $>$ 90%). The overall conclusion is that MR-O-QPSK provides the most robust propagation waves.

The measured relative attenuation at the transmitter for all tested PHY configurations is depicted on Fig. 2. Results indicate that MR-FSK, MR-OFDM1/MCS3, and MR-OFDM2/MCS5 PHYs require the highest level of transmit power. MR-O-QPSK is the most robust PHY, while MR-OFDM and MR-FSK PHYs have similar results. A significant decrease of attenuation appears in the case of MR-OFDM1 MCS2 and MCS3. One reason for this is data reduction by half (MCS3 - 800 kb/s, MCS2 - 400 kb/s). The other reason is the use of frequency repetition technique in MCS2 configuration. Fig. 2 provides one more crucial conclusion based on the attenuation difference between MCS1 and MCS2 configurations for both MR-OFDM options, which are the same and equal 3 dB. This dataset confirms that BPSK modulation provides extra 3 dB over QPSK modulation. The comparison of MR-OFDM options indicates that as the signal's bandwidth is smaller, the less robust is its PHY.

3) Third test scenario: The last scenario has been tested in sub-GHz frequency band to evaluate the PHY resistance to the influence of noise. The experiments have been conducted in controlled environment (RF coaxial cables and a coupler). The signal and noise powers have been measured at the moment of the connection interruption. The gathered dataset shows that the increase of data rate reduces the resistance of PHY to the noise influence and that MR-O-QPSK PHY is the the most sensitive to the noise impact.

The collected dataset provides an overview of estimated PHY configuration performances giving the reason for further research of the IEEE 802.15.4g standard in IIoT. The overall results show that MR-O-QPSK provides the longest range in the real conditions, but the crucial conclusion is that MR-OFDM PHY is the most resistant to the influence of noise which prompts its usage in low-power and low-rate

wireless networks.

IV. Conclusion

As a result of rapid development of IIoT connectivity, many heterogeneous devices will be operational at close range and within a limited spectrum, which poses a challenge in terms of coexistence in the ISM band. The new generation of IIoT devices will have the ability to detect, classify and avoid areas affected by external interference. These devices will have multimode radio chips, flexibility in the selection of software and protocols, as well as communication support for the advanced applications. Next generation industrial standards based on low-power WSN technologies are expecting to provide high immunity against interference and multipath fading, and to support the QoS differentiation of traffic flows. Preliminary results obtained by using IIoT open-hardware and open-software platforms confirm that schedule-based deterministic MAC protocols, such as TSCH, as well as deterministic networking mechanisms can provide such guarantees.

References

[1] M. Hermann, T. Pentek, and B. Otto, "Design Principles for Industrie 4.0 Scenarios," in *2016 49th Hawaii International Conference on System Sciences (HICSS)*, 2016, pp. 3928–3937.

[2] C. Lin, D. Deng, Z. Chen, and K. Chen, "Key design of driving industry 4.0: joint energy-efficient deployment and scheduling in group-based industrial wireless sensor networks," *IEEE Communications Magazine*, vol. 54, no. 10, pp. 46–52, 2016.

[3] I. Ishaq, D. Carels, G. Teklemariam, J. Hoebeke, F. Abeele, E. Poorter, I. Moerman, and P. Demeester, "IETF Standardization in the Field of the Internet of Things (IoT): A Survey," *Journal of Sensor and Actuator Networks*, vol. 2, no. 2, p. 235–287, Apr 2013. [Online]. Available: http://dx.doi.org/10.3390/jsan2020235

[4] "IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer," *IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011)*, pp. 1–225, April 2012.

[5] "Minimal IPv6 over the TSCH Mode of IEEE 802.15.4e (6TiSCH) Configuration," RFC 8180, May 2017.

[6] P. Thubert, "An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4," Internet-Draft, Dec. 2018, work in Progress. [Online]. Available: https://tools.ietf.org/id/draft-ietf-6tisch-architecture-19.html.

[7] N. Finn and P. Thubert, "Deterministic Networking Problem Statement," Internet Engineering Task Force, Internet-Draft, September 2017. [Online]. Available: https://tools.ietf.org/id/draft-ietf-detnet-problem-statement-02.html.

[8] M. Magno, V. Jelicic, B. Srbinovski, V. Bilas, E. Popovici, and L. Benini, "Design, Implementation, and Performance Evaluation of a Flexible Low-Latency Nanowatt Wake-Up Radio Receiver," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 2, pp. 633–644, 2016.

[9] "IEEE Standard for Local and Metropolitan Area Networks–Timing and Synchronization for Time-Sensitive Applications," *IEEE Std 802.1AS-2020 (Revision of IEEE Std 802.1AS-2011)*, pp. 1–421, 2020.

[10] R. Teles Hermeto, A. Gallais, and F. Theoleyre, "Scheduling for IEEE802.15.4-TSCH and Slow Channel Hopping MAC in Low Power Industrial Wireless Networks: A Survey," *Comput. Commun.*, vol. 114, no. C, pp. 84–105, Dec. 2017. [Online]. Available: https://doi.org/10.1016/j.comcom.2017.10.004

[11] M. R. Palattella, N. Accettura, L. A. Grieco, G. Boggia, M. Dohler, and T. Engel, "On optimal scheduling in duty-cycled industrial IoT applications using IEEE802.15.4e TSCH," *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3655–3666, 2013.

[12] M. R. Palattella, T. Watteyne, Q. Wang, K. Muraoka, N. Accettura, D. Dujovne, L. A. Grieco, and T. Engel, "On-the-Fly Bandwidth Reservation for 6TiSCH Wireless Industrial Networks," *IEEE Sensors Journal*, vol. 16, no. 2, pp. 550–560, Jan 2016.

[13] E. Livolant, P. Minet, and T. Watteyne, "The Cost of Installing a 6TiSCH Schedule," in *Ad-hoc, Mobile, and Wireless Networks*, N. Mitton, V. Loscri, and A. Mouradian, Eds. Cham: Springer International Publishing, 2016, pp. 17–31.

[14] T. Devaja, D. Bajovic, D. Vukobratovic, and G. Gardaševič, "Scheduling in 6TiSCH Networks via Max-Product Message-Passing," in *IEEE EUROCON 2019 -18th International Conference on Smart Technologies*, 2019, pp. 1–6.

[15] P. Tuset-Peiró, R. D. Gomes, P. Thubert, E. Cuerva, E. Egusquiza, and X. Vilajosana, "A Dataset to Evaluate IEEE 802.15.4g SUN for Dependable Low-Power Wireless Communications in Industrial Scenarios," *Data*, vol. 5, no. 3, p. 64, Jul 2020. [Online]. Available: http://dx.doi.org/10.3390/data5030064

[16] M. Lekić and G. Gardašević, "Experimental Performance Evaluation of IEEE 802.15.4g Applications Using OpenMote-B Devices," in *Proceedings of the Twenty-nine International Electrotechnical and Computer Science Conference ERK 2020*, September 2020, to be published.

[17] E. Baccelli, C. Gündoğan, O. Hahm, P. Kietzmann, M. Lenders, H. Petersen, K. Schleiser, T. Schmidt, and M. Wahlisch, "Riot: an open source operating system for low-end embedded devices in the iot," *IEEE Internet of Things Journal*, vol. PP, pp. 1–1, 03 2018.

[18] "IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 3: Physical Layer (PHY) Specifications for Low-Data-Rate, Wireless, Smart Metering Utility Networks," *IEEE Std 802.15.4g-2012 (Amendment to IEEE Std 802.15.4-2011)*, pp. 1–252, 2012.

Otvoreni resursi i tehnologije za obradu srpskog jezika

Vuk Batanović[1], Nikola Ljubešić[2], Tanja Samardžić[3], Maja Miličević Petrović[4]

[1]*Inovacioni centar Elektrotehničkog fakulteta, Beograd, Srbija*
[2]*Institut Jožef Stefan, Ljubljana, Slovenija*
[3]*Univerzitet u Cirihu – URPP "Language and Space", Cirih, Švajcarska*
[4]*Univerzitet u Beogradu – Filološki fakultet, Beograd, Srbija*
vuk.batanovic@ic.etf.bg.ac.rs, nikola.ljubesic@ijs.si, tanja.samardzic@uzh.ch, m.milicevic@fil.bg.ac.rs

Rezime: Otvorenost jezičkih resursa i alata je od velike važnosti za povećanje kvaliteta i brzine razvoja tehnologija za računarsku obradu prirodnih jezika. U ovom radu predstavljeni su otvoreni resursi za obradu srpskog jezika. Opisani su ručno anotirani korpusi, kao i širi spektar alata i računarskih modela, uključujući i veb servis koji omogućava njihovo jednostavno korišćenje.

Ključne reči: računarska lingvistika; korpusi tekstova; jezički alati; NLP; otvorena kultura.

I. Uvod

U poslednjih nekoliko godina došlo je do razvoja otvorenih, slobodno dostupnih resursa i tehnologija za računarsku obradu tekstova na srpskom jeziku, uključujući anotirane jezičke korpuse, alate za automatsku analizu i označavanje tekstova, kao i različite vrste modela za obradu prirodnih jezika (engl. *natural language processing* – NLP). Otvorenost i javna dostupnost omogućavaju veći stepen ponovljivosti rezultata NLP istraživanja, pospešuju saradnju istraživača i stimulišu zajedničko unapređivanje postojećih korpusa, alata i modela, umesto stalnog zasebnog pravljenja novih resursa istog tipa iz početka. Ova otvorenost je naročito važna u manjim jezicima poput srpskog, kod kojih se NLP istraživanjima bavi dosta ograničen krug naučnika i koji nisu od naročitog interesa za širu svetsku istraživačku zajednicu. Poseban doprinos promovisanju aktuelnosti i važnosti otvorenih jezičkih resursa za srpski i srodne jezike dao je projekat *Regional Linguistic Data Initiative* (ReLDI) [1], koji je doveo do nastanka većeg broja alata primenjivih na više južnoslovenskih jezika, kao i korpusa anotiranih korišćenjem standardizovane metodologije označavanja.

U ovom radu su ukratko prikazani aktuelni otvoreni, slobodno dostupni jezički resursi koji se mogu koristiti za analizu i obradu tekstova na srpskom jeziku. To uključuje kako resurse koji su proistekli iz ReLDI projekta, tako i one koje su autori rada razvili naknadno ili nezavisno od pomenutog projekta. Pri tome, pod otvorenim resursima ili alatima se podrazumevaju oni koji su javno dostupni na određenom repozitorijumu ili sajtu i jasno objavljeni pod nekom od odgovarajućih slobodnih licenci kao što su Creative Commons, GPL, i sl.

Ostatak rada je strukturiran na sledeći način: najpre su prikazani anotirani korpusi tekstova, a nakon toga alati i modeli za računarsku obradu tekstova na srpskom jeziku. Na kraju su izloženi planovi vezani za pravce daljeg razvoja.

II. Korpusi

U ovom odeljku su opisani ručno i automatski anotirani korpusi tekstova na srpskom jeziku koji su javno dostupni. Najpre su predstavljeni ručno anotirani korpusi standardnog i nestandardnog jezika, zatim korpusi posebno anotirani za određenu problematiku, dok je na kraju opisan veb korpus srpskog jezika. Izrada mnogih anotiranih korpusa je koordinisana sa sličnim poduhvatima na drugim jezicima, pre svega na hrvatskom i slovenačkom.

A. Ručno anotirani korpusi

SETimes.SR korpus [2], [3], izgrađen na osnovu *SETimes* paralelnog korpusa novinskih članaka [4], predstavlja ručno anotirani korpus tekstova pisanih standardnim srpskim jezikom namenjen obučavanju i evaluaciji računarskih modela na većem broju problema iz obrade prirodnih jezika. On sadrži 163 dokumenta podeljena na 3891 rečenicu, odnosno 86 726 tokena. *SETimes.SR* korpus je anotiran u pogledu segmentacije (na tokene, rečenice i dokumente), morfosintaktičkih oznaka, lema, sintaksnih dependencija kao i imenovanih entiteta. Korišćene morfosintaktičke oznake su u skladu sa MULTEXT-East v6 standardom za srpsko-hrvatski makrojezik[1]. Sintaksne dependencije su označene prema specifikaciji Universal Dependency v2 (UDv2)[2]. Oznake imenovanih entiteta su date u IOB2 formatu, uz razlikovanje pet tipova entiteta — osobe (PER), prisvojni pridevi izvedeni od imena (DERIV-PER), lokacije (LOC), organizacije (ORG) i razno (MISC). *SETimes.SR* je anotiran po uzoru na i uz pomoć modela obučenih na *SETimes.HR* [5] korpusu na hrvatskom jeziku, koji sada predstavlja deo većeg *hr500k* korpusa [6], [7].

ReLDI-NormTagNER-sr korpus [8], [9] je ručno označen skup tvitova napisanih na srpskom jeziku. Njegova primarna namena jeste prilagođavanje računarskih modela fenomenima koji su česti u nestandardnom jeziku koji se koristi na internetu. Korpus se sastoji od 3748 tvitova koji sadrže ukupno 91 781 token. Slojevi anotacije prisutni u korpusu su sledeći: tokenizacija i podela na rečenice, normalizacija na nivou reči, morfosintaktička anotacija (na osnovu standarda MULTEXT-East i Universal Dependencies), lematizacija i imenovani entiteti (sa istih 5 tipova entiteta kao i u *SETimes.SR* korpusu). Ovaj korpus je izgrađen uporedo

[1] http://nl.ijs.si/ME/V6/msd/html/msd-hbs.html
[2] http://universaldependencies.org

DOI: 10.5281/zenodo.4113229

sa sličnim resursom za hrvatski jezik – *ReLDI-NormTagNER-hr* [10] – a oba su oblikovana po uzoru na slovenački korpus *Janes-Tag* [11], [12].

B. Specijalizovani ručno anotirani korpusi

Na srpskom jeziku su javno dostupni ručno anotirani specijalizovani korpusi koji se odnose na semantičke probleme određivanja semantičke sličnosti, detekcije parafraza i analize sentimenta.

Za problematiku određivanja semantičke sličnosti kratkih tekstova izrađen je korpus *STS.news.sr* [13]. On sadrži 1192 para rečenica iz novinskog domena anotirana u pogledu stepena semantičke sličnosti između rečenica u paru. Ocene sličnosti su granulirane, na skali od 0 do 5, i dobijene su usrednjavanjem individualnih ocena petoro anotatora. *STS.news.sr* je kreiran korišćenjem sadržaja ranijeg korpusa parafraza *paraphrase.sr* [14], [15], koji sadrži samo binarne ocene sličnosti, ručno zadate od strane jednog anotatora, koje govore da li se rečenice u okviru para mogu smatrati parafrazama ili ne.

Za probleme analize sentimenta tekstova na srpskom predstavljena su dva anotirana korpusa tekstova iz domena filmova – jedan na nivou dužih dokumenata, i drugi na nivou kratkih komentara. *SerbMR* [16] je izbalansiran korpus filmskih recenzija koji je dostupan u varijantama sa dve klase polarnosti sentimenta (pozitivna i negativna) i sa tri klase polarnosti (pozitivna, neutralna i negativna). Dvoklasna varijanta *SerbMR-2C* sadrži ukupno 1682 dokumenta, a troklasna *SerbMR-3C* ukupno 2523, odnosno 841 dokument po klasi. Oznake sentimenta u ovom skupu podataka su dobijene automatski, konverzijom numeričkih ocena pridruženih svakoj recenziji od strane njenog autora.

SentiComments.SR korpus kratkih tekstova na srpskom jeziku [17] je ručno anotiran oznakama sentimenta koje omogućavaju više nivoa interpretacije. Zbog toga se ovaj korpus može upotrebiti u obučavanju i evaluaciji klasifikatora na većem broju užih problema u analizi sentimenta, uključujući određivanje polarnosti, određivanje subjektivnosti, detekciju sarkazma, itd. *SentiComments.SR* sadrži 3490 kratkih komentara i zajednički je anotiran od strane dvoje anotatora.

C. Veb korpus

Veb korpus *srWaC* [18], [19] predstavlja najveći javno dostupni korpus tekstova opšteg tipa na srpskom jeziku. U aktuelnoj verziji 1.1, on sadrži 555 miliona tokena i preko 25 miliona rečenica raspoređenih u oko 1,3 miliona dokumenata. Ovaj korpus je izgrađen prikupljanjem celokupnog sadržaja sa *.rs* domena, nakon čega je sprovedeno uklanjanje duplikata na nivou pasusa, vraćanje dijakritičkih oznaka u tekstovima, kao i automatsko morfosintaktičko označavanje i lematizacija.

III. Alati i računarski modeli

U ovom odeljku su opisani javno dostupni alati i modeli koji se mogu koristiti za rešavanje konkretnih NLP zadataka u obradi tekstova na srpskom jeziku. Oni su izloženi po rastućem nivou obrade, počevši od osnovnih alata za podelu i korekciju tekstova, preko alata za morfosintaktičku i sintaktičku obradu, do alata i modela vezanih za semantičke probleme. Na kraju je opisan *ReLDIanno* veb servis [20] u okviru koga je integrisan veliki broj navedenih alata.

A. Segmentacija i tokenizacija

Uobičajen prvi korak u obradi tekstualnog sadržaja jeste njegova segmentacija i tokenizacija, tj. podela dokumenata na rečenice, i rečenica na elementarne jedinice – tokene – koji predstavljaju reči, brojeve, znakove interpunkcije, itd. Za tokenizaciju srpskog, hrvatskog, slovenačkog, makedonskog i bugarskog jezika može se koristiti ReLDI tokenizator, koji sadrži odvojene modele za standardne i nestandardne tekstove na svim navedenim jezicima. ReLDI tokenizator pruža izlaz u vertikalizovanom, CoNLL-U formatu, koji predstavlja ulazni format podataka za alate za vraćanje dijakritičkih oznaka, morfosintaktičko označavanje i lematizaciju, sintaktičko parsiranje i označavanje imenovanih entiteta.

B. Vraćanje dijakritičkih oznaka

U obradi latiničnih tekstova na srpskom jeziku prikupljenih sa društvenih mreža, veb foruma, komentara posetilaca na različitim sajtovima, itd. često se susreće odsustvo dijakritičkih oznaka, tj. pisanje tekstova engleskom/ASCII latinicom, bez slova č, ć, š, đ, ž i dž. Nedostatak dijakritika može znatno otežati pravilnu automatsku obradu tekstova, jer dovodi do ortografske istovetnosti različitih, često morfosintaktički i/ili semantički veoma udaljenih reči (npr. glagol *sesti* i redni broj *šesti* se svode na isti oblik). Stoga je preporučljivo pre dalje obrade sprovesti vraćanje nedostajućih dijakritičkih oznaka, za šta se može iskoristiti statistički alat za redijakritizaciju [21], koji, pored srpskog, podržava i hrvatski i slovenački jezik.

C. Morfosintaktičko označavanje

Morfosintaktičko označavanje, tj. automatsko dodeljivanje morfosintaktičkog opisa svakom tokenu u rečenici, što uključuje informacije o vrstama reči, predstavlja važan korak u NLP obradi jer su ovi opisi vredan ulazni podatak za mnoge složenije zadatke. Trenutno najbolje javno dostupno rešenje za morfosintaktičko označavanje srpskog jezika je skup alata centra znanja za južnoslovenske jezike CLASSLA [22], koji predstavlja izmenjenu i proširenu verziju *Stanza* biblioteke [23]. Unutar CLASSLA paketa, za srpski jezik su dostupni model za standardni jezik [24] i model prilagođen nestandardnom jeziku [25]. Posebna snaga ovog i svih drugih modela za nestandardne tekstove u okviru CLASSLA paketa je što su izrazito otporni na izostavljanje dijakritičkih znakova u tekstu. Alat generiše morfosintaktičke opise po MULTEXT-East standardu, ali i po sve popularnijem Universal Dependencies standardu, po kojem se opis sastoji od univerzalne vrste reči i univerzalnih morfosintaktičkih odlika.

D. Morfološka normalizacija

Morfološka normalizacija tekstova omogućava svođenje različitih oblika reči na istu zajedničku osnovu i moguće ju je sprovesti pomoću stemera ili lematizatora. U okviru paketa *SCStemmers* [16], napisanog u programskom jeziku Java, reimplementirana su četiri algoritma stemovanja koja su primenjiva na srpski jezik:

- Optimalni i pohlepni stemer Kešelja i Šipke [26]
- Unapređenje njihovog pohlepnog stemera koje je predstavio Milošević [27]
- Stemer za hrvatski jezik Ljubešića i Pandžića, koji predstavlja unapređenje pristupa iz [28]

Iako do sada nije sprovedena uporedna intrinzička evaluacija ovih algoritama, ekstrinzičke evaluacije na semantičkim problemima određivanja semantičke sličnosti kratkih tekstova i analize sentimenta [13], [17], [29], [30] konzistentno pokazuju da je stemer Ljubešića i Pandžića obično najbolji izbor.

Trenutno najtačnije otvoreno rešenje za lematizaciju srpskog jezika je lematizator koji je deo CLASSLA lanca alata [22]. On koristi flektivni leksikon *srLex* [31], [32] za sve oblike koji su pokriveni leksikonom. Za ostale oblike alat koristi seq2seq model koji na temelju oblika i njegovog morfosintaktičkog opisa oblikuje lemu. Kao i za morfosintaktičko označavanje, i za lematizaciju su dostupni modeli za standardne [33] i za nestandardne tekstove [34].

E. Sintaktičko parsiranje

Sintaktičko parsiranje predstavlja automatsku izradu sintaksnog stabla rečenice kroz obeležavanje sintaktičkih veza između tokena. Iako se parsiranje ređe koristi kao pretprocesiranje za zadatke višeg nivoa, ono je veoma bitno za potrebe lingvističke analize teksta, a može se upotrebiti i za analizu i interpretaciju rada računarskih modela. Od 2017. godine, srpski jezik je kroz anotaciju korpusa *SETimes.SR* [35] prisutan u međunarodnom projektu standardizacije označavanja sintaktičkih dependencija Universal Dependencies [36], pri čemu se anotirani resursi u UD repozitorijumima regularno ažuriraju. Već pomenuti CLASSLA paket takođe pruža mogućnost i za sintaktičko parsiranje tekstova na srpskom. Za razliku od morfosintaktičkog označavanja i lematizacije, u parsiranju se koristi isti model za obradu i standardnih i nestandardnih tekstova [37].

F. Označavanje imenovanih entiteta

Kao i u slučaju morfosintaktičkog i sintaktičkog označavanja i lematizacije, CLASSLA lanac alata [22] postiže trenutno najbolje rezultate i na problemu označavanja imenovanih entiteta. Pri tome, alat razlikuje istih pet tipova entiteta označenih u *SETimes.SR* i *ReLDI-NormTagNER-sr* korpusima, koji su i korišćeni pri obučavanju modela, i koristi isti IOB2 format oznaka. Isto kao i kod rešenja za morfosintaktičko označavanje i lematizaciju, alat omogućuje obradu standardnih [38] i nestandardnih tekstova [39].

G. Određivanje semantičke sličnosti kratkih tekstova

Za dobijanje granuliranih ocena semantičke sličnosti kratkih tekstova dostupan je paket *STSFineGrain* [13] u okviru koga je implementirano više modela za rešavanje ovog problema, koji se mogu obučiti i evaluirati korišćenjem *STS.news.sr* ili nekog sličnog korpusa na drugom jeziku. To uključuje kako osnovne modele zasnovane na leksičkom preklapanju tekstova i/ili sličnosti vektora značenja reči u njima, tako i nešto naprednije modele koji se oslanjaju i na informacije o frekventnosti reči [15], vrsti reči [40] ili na oba tipa informacija [13].

H. ReLDIanno veb servis

ReLDIanno veb servis [20] omogućava jednostavno korišćenje jezičkih alata za srpski, hrvatski i slovenački jezik, uključujući alate za tokenizaciju, morfosintaktičku analizu, lematizaciju, sintaktičko parsiranje i označavanje imenovanih entiteta. Servisima se može pristupiti putem veb aplikacije ili programski, kroz biblioteku za programski jezik Python. Unutar veb aplikacije omogućena je obrada različitih formata fajlova, kao što su TXT, DOCX, PDF, i ZIP arhive. Alati koji se trenutno nalaze u pozadini servisa predstavljaju, uz izuzetak tokenizatora, stariju generaciju otvorenih alata za obradu srpskog jezika koji su razvijeni unutar projekta ReLDI [1].

IV. Zaključak

U ovom radu su ukratko izloženi aktuelni otvoreni, slobodno dostupni anotirani korpusi, kao i alati i modeli za računarsku obradu tekstova na srpskom jeziku. U planu je dalje proširivanje skupa otvorenih resursa novim vrstama anotiranih podataka, poput anotacija koreferentnih odnosa. Pored toga, u skoroj budućnosti se planira i zamena modela korišćenih u okviru veb servisa *ReLDIanno* novijim verzijama iz CLASSLA paketa [22]. Na kraju, u toku su i prvi eksperimenti sa velikim jezičkim modelima poput BERT-a [41], koji će verovatno predstavljati sledeći korak u razvoju računarske obrade srpskog jezika.

Zahvalnica

Autori su zahvalni velikom broju saradnika i anotatora bez čije pomoći izrada navedenih otvorenih resursa ne bi bila moguća.

Rad na ovoj publikaciji delimično je podržan od strane Ministarstva prosvete, nauke i tehnološkog razvoja, i sproveden uz podršku Fonda za nauku Republike Srbije, projekat 6526093, VI-AVANTES.

Dostupnost podataka: Otvoreni resursi su dostupni preko sajta ReLDI centra za jezičke podatke (https://reldi.spur.uzh.ch/hr-sr/resursi-i-alati) i CLARIN.SI repozitorijuma (https://www.clarin.si). Veb servis *ReLDIanno* (http://www.clarin.si/services/web) je dostupan preko sajta CLASSLA centra (https://www.clarin.si/info/k-centre), dok je novi CLASSLA paket alata dostupan preko *GitHub*-a (https://

github.com/clarinsi/classla-stanfordnlp) i *pypi* usluge (https://pypi.org/project/classla).

Literatura

[1] T. Samardžić, N. Ljubešić, and M. Miličević, "Regional Linguistic Data Initiative (ReLDI)," in P*roceedings of the Fifth Workshop on Balto-Slavic Natural Language Processing (BSNLP 2015)*, 2015, pp. 40-42

[2] V. Batanović, N. Ljubešić, T. Samardžić, and T. Erjavec, "Training corpus SETimes.SR 1.0." Slovenian language resource repository CLARIN.SI, 2018, [Online]. Available: http://hdl.handle.net/11356/1200

[3] V. Batanović, N. Ljubešić, and T. Samardžić, "SETimes.SR – A Reference Training Corpus of Serbian," in *Proceedings of the Conference on Language Technologies & Digital Humanities 2018 (JT-DH 2018)*, 2018, pp. 11–17

[4] F. M. Tyers and M. S. Alperen, "South-East European Times : A parallel corpus of Balkan languages," in *Proceedings of the Workshop on Exploitation of Multilingual Resources and Tools for Central and (South) Eastern European Languages, LREC 2010*, 2010, pp. 49–53

[5] Ž. Agić and N. Ljubešić, "The SETIMES.HR Linguistically Annotated Corpus of Croatian," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, 2014, pp. 1724–1727

[6] N. Ljubešić, Ž. Agić, F. Klubička, V. Batanović, and T. Erjavec, "hr500k – A Reference Training Corpus of Croatian," in *Proceedings of the Conference on Language Technologies & Digital Humanities 2018 (JT-DH 2018)*, 2018, pp. 154–161

[7] N. Ljubešić, Ž. Agić, F. Klubička, V. Batanović, and T. Erjavec, "Training corpus hr500k 1.0." Slovenian language resource repository CLARIN.SI, 2018, [Online]. Available: http://hdl.handle.net/11356/1183

[8] N. Ljubešić and M. Miličević, "Tviterasi, tviteraši or twitteraši? Producing and analysing a normalised dataset of Croatian and Serbian tweets," *Slov. 2.0 Empirical, Appl. Interdiscip. Res.*, vol. 4, no. 2, pp. 156–188, 2016, doi: 10.4312/slo2.0.2016.2.156-188

[9] N. Ljubešić, T. Erjavec, V. Batanović, M. Miličević, and T. Samardžić, "Serbian Twitter training corpus ReLDI-NormTagNER-sr 2.1." Slovenian language resource repository CLARIN.SI, 2019, [Online]. Available: http://hdl.handle.net/11356/1240

[10] N. Ljubešić, T. Erjavec, V. Batanović, M. Miličević, and T. Samardžić, "Croatian Twitter training corpus ReLDI-NormTagNER-hr 2.1." Slovenian language resource repository CLARIN.SI, 2019, [Online]. Available: http://hdl.handle.net/11356/1241

[11] D. Fišer, N. Ljubešić, and T. Erjavec, "The Janes project: language resources and tools for Slovene user generated content," *Lang. Resour. Eval.*, vol. 54, no. 1, pp. 223–246, 2020, doi: 10.1007/s10579-018-9425-z

[12] T. Erjavec *et al.*, "CMC training corpus Janes-Tag 2.1." Slovenian language resource repository CLARIN.SI, 2019, [Online]. Available: http://hdl.handle.net/11356/1238

[13] V. Batanović, M. Cvetanović, and B. Nikolić, "Fine-grained Semantic Textual Similarity for Serbian," in *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, 2018, pp. 1370–1378

[14] V. Batanović, B. Furlan, and B. Nikolić, "Softverski sistem za određivanje semantičke sličnosti kratkih tekstova na srpskom jeziku," *Zbornik radova sa 19. telekomunikacionog foruma (TELFOR 2011)*, Nov. 2011, pp. 1249–1252, doi: 10.1109/TELFOR.2011.6143778

[15] B. Furlan, V. Batanović, and B. Nikolić, "Semantic similarity of short texts in languages with a deficient natural language processing support," *Decis. Support Syst.*, vol. 55, no. 3, pp. 710–719, Feb. 2013, doi: 10.1016/j.dss.2013.02.002

[16] V. Batanović, B. Nikolić, and M. Milosavljević, "Reliable Baselines for Sentiment Analysis in Resource-Limited Languages: The Serbian Movie Review Dataset," in *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, 2016, pp. 2688–2696

[17] V. Batanović, "Metodologija rešavanja semantičkih problema u obradi kratkih tekstova napisanih na prirodnim jezicima sa ograničenim resursima," Univerzitet u Beogradu, 2020

[18] N. Ljubešić and F. Klubička, "{bs,hr,sr}WaC –Web corpora of Bosnian, Croatian and Serbian," in *Proceedings of the Ninth Web as Corpus Workshop (WaC-9)*, 2014, pp. 29–35

[19] N. Ljubešić and F. Klubička, "Serbian web corpus srWaC 1.1." Slovenian language resource repository CLARIN.SI, 2016, [Online]. Available: http://hdl.handle.net/11356/1063

[20] N. Ljubešić *et al.*, "Easily Accessible Language Technologies for Slovene , Croatian and Serbian," in *Proceedings of the Conference on Language Technologies & Digital Humanities*, 2016, pp. 120–124

[21] N. Ljubešić, T. Erjavec, and D. Fišer, "Corpus-Based Diacritic Restoration for South Slavic Languages," in *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, 2016, pp. 3612–3616

[22] N. Ljubešić and K. Dobrovoljc, "What does Neural Bring? Analysing Improvements in Morphosyntactic Annotation and Lemmatisation of Slovenian, Croatian and Serbian," in *Proceedings of the Seventh Workshop on Balto-Slavic Natural Language Processing (BSNLP 2019)*, 2019, pp. 29–34

[23] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, and C. D. Manning, "Stanza: A Python Natural Language Processing Toolkit for Many Human Languages," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations (ACL 2020)*, 2020, pp. 101–108, doi: 10.18653/v1/2020.acl-demos.14

[24] N. Ljubešić, "The CLASSLA-StanfordNLP model for morphosyntactic annotation of standard Serbian 1.1." Slovenian language resource repository CLARIN.SI, 2020, [Online]. Available: http://hdl.handle.net/11356/1349

[25] N. Ljubešić and V. Štefanec, "The CLASSLA-StanfordNLP model for morphosyntactic annotation of non-standard Serbian 1.0." Slovenian language resource repository CLARIN.SI, 2020, [Online]. Available: http://hdl.handle.net/11356/1332

[26] V. Kešelj and D. Šipka, "Pristup izgradnji stemera i lematizora za jezike s bogatom fleksijom i oskudnim resursima zasnovan na obuhvatanju sufiksa," *INFOTEKA - časopis za Bibl. i Inform.*, vol. 9, no. 1–2, pp. 21–31, 2008

[27] N. Milošević, "Stemmer for Serbian language." arXiv 1209.4471, 2012

[28] N. Ljubešić, D. Boras, and O. Kubelka, "Retrieving Information in Croatian: Building a Simple and Efficient Rule-Based Stemmer," in *INFuture2007: Digital Information and Heritage*, Zagreb, Croatia: Department for Information Sciences, Faculty of Humanities and Social Sciences, 2007, pp. 313–320

[29] V. Batanović and B. Nikolić, "Sentiment Classification of Documents in Serbian: The Effects of Morphological Normalization and Word Embeddings," *Telfor J.*, vol. 9, no. 2, pp. 104–109, 2017, doi: 10.5937/telfor1702104B

[30] V. Batanović and B. Nikolić, "Sentiment Classification of Documents in Serbian: The Effects of Morphological Normalization," in *Proceedings of the 24th Telecommunications Forum (TELFOR 2016)*, 2016, pp. 889–892, doi: 10.1109/TELFOR.2016.7818923

[31] N. Ljubešić, F. Klubička, Ž. Agić, and I.-P. Jazbec, "New Inflectional Lexicons and Training Corpora for Improved Morphosyntactic Annotation of Croatian and Serbian," in *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, 2016, pp. 4264–4270

[32] N. Ljubešić, "Inflectional lexicon srLex 1.3." Slovenian language resource repository CLARIN.SI, 2019, [Online]. Available: http://hdl.handle.net/11356/1233

[33] N. Ljubešić, "The CLASSLA-StanfordNLP model for lemmatisation of standard Serbian 1.2." Slovenian language resource repository CLARIN.SI, 2020, [Online]. Available: http://hdl.handle.net/11356/1355

[34] N. Ljubešić, "The CLASSLA-StanfordNLP model for lemmatisation of non-standard Serbian 1.1." Slovenian language resource repository CLARIN.SI, 2020, [Online]. Available: http://hdl.handle.net/11356/1351

[35] T. Samardžić, M. Starović, Ž. Agić, and N. Ljubešić, "Universal Dependencies for Serbian in Comparison with Croatian and Other Slavic Languages," in *Proceedings of the Sixth Workshop on*

Balto-Slavic Natural Language Processing (BSNLP 2017), 2017, pp. 39–44

[36] D. Zeman *et al.*, “CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies,” in *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, 2018, pp. 1–21, doi: 10.18653/v1/K18-2001

[37] N. Ljubešić, “The CLASSLA-StanfordNLP model for UD dependency parsing of standard Serbian.” Slovenian language resource repository CLARIN.SI, 2019, [Online]. Available: http://hdl.handle.net/11356/1260

[38] N. Ljubešić, “The CLASSLA-StanfordNLP model for named entity recognition of standard Serbian 1.0.” Slovenian language resource repository CLARIN.SI, 2020, [Online]. Available: http://hdl.handle.net/11356/1323

[39] N. Ljubešić, “The CLASSLA-StanfordNLP model for named entity recognition of non-standard Serbian 1.0.” Slovenian language resource repository CLARIN.SI, 2020, [Online]. Available: http://hdl.handle.net/11356/1341

[40] V. Batanović and D. Bojić, “Using Part-of-Speech Tags as Deep-Syntax Indicators in Determining Short-Text Semantic Similarity,” *Comput. Sci. Inf. Syst.*, vol. 12, no. 1, pp. 1–31, 2015, doi: 10.2298/CSIS131127082B

[41] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019)*, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423

Primena softvera otvorenog koda kod vizualizacije podataka

Vladimir Otašević, Biljana Kosanović
Računarski centar Univerziteta u Beogradu, Beograd, Srbija
vladimir.otasevic@rcub.bg.ac.rs, biljana@rcub.bg.ac.rs

Rezime: Rad je napisan sa ciljem da se pokažu mogućnosti različitih alata otvorenog koda koji se mogu koristi za potrebe vizualizacije i grafičke interpretacije podataka. Takođe, navedeni su i objašnjeni osnovi aspekti koji utiču na grafičku interpretaciju podataka. Ovaj rad sadrži kratak pregled funkcionalnosti dve *javascript* biblioteke za vizualizaciju i to: *d3.js* i *chart.js*.

Ključne reči: javascript, otvoren kod, vizualizacija podataka, grafikon, interakcija

I. Uvod

Oblast vizualizacije podataka se vekovima razvijala 1. Nije oduvek bila posebna oblast, već se dug period posmatrala kako sastavni element drugih oblasti kao što su fizika, matematika, kartografija i dr. Sa nastankom računara dolazi i do ubrzanog razvoja tehnika vizualizacije podataka. Danas, kada je većina poslova nezamisliva bez računara, vizualizacije predstavlja važan segment prilikom prenosa informacija.

Vizualizacija podataka bi se mogla definisati kao tehnika kojom se informacije i podaci grafički predstavljaju. Grafički objekti mogu biti različiti grafikoni, skale, stabla, histogrami i dr. Postoji još jedna važna odlika vizualizacije podataka, a to je da ona ne treba samo da grafički predstavlja informacije i podatke već i da svojim grafičkim objektima i interaktivnošću privuče korisnikovu pažnju kako bi jasnije razumeo predstavljenu informaciju 2.

Postoji veći broj alata za vizualizaciju podataka baziranih na *javascript* jeziku 3. U ovom radu su opisane *d3.js* (dostupna na https://d3js.org/) i *chart.js* (dostupna na https://www.chartjs.org/) biblioteke zbog iskustva u radu sa njima. Navedene biblioteke iako se koriste za vizualizaciju podataka, one se međusobno razlikuju. Biblioteka *d3.js* je pogodna zbog svoje fleksibilnosti prilikom implementacije ali to uvodi veću kompleksnost u kodu. Sa druge strane *chart.js* je jednostavnija za primenu ali to dovodi do manje fleksibilnosti prilikom implementacije. Zbog načina implementacije i velikog izbora grafikona koji se mogu proizvesti pomoću ovih biblioteka, one predstavljaju jedno od mogućih rešenja za vizualizaciju.

II. Grafička interpretacija podataka

A *Istorija*

Oblast razvoja tehnika grafičke interpretacije podataka je već dugo u fokusu pažnje istraživača 1. Danas vizualizacija podataka predstavlja posebnu granu, a u prilog tome ide i to što postoje eksperti i stručnjaci koji se isključivo bave razvojem rešenja grafičke interpretacije podataka, a da pri tome ne moraju razumeti oblast kojoj pripadaju podaci koje obrađuju. Ovakav pristup ovoj oblasti nije postojao oduvek. Ranije je oblast vizualizacije podataka bila samo segment neke veće oblasti za koju postoji potreba da se određeni skup podataka prikaže na adekvatan način.

Počeci razvoja oblasti vizualizacije vezuju se za početak 16. veka. Period od 16. veka pa sve do danas moguće je podeliti po epohama. Svaka epoha obeležena je različitim otkrićima koja su doprinela razvoju vizualizacije 1.

Prva epoha obuhvata period u kojem je od značaja bilo da se podaci zapisuju na struktuiran način. To je podrazumevalo da se podaci npr. pišu u tabelama. Takođe, istu epohu su obeležile i nove kartografske tehnike. Za drugu epohu se vezuju dešavanja koja se odnose na merenja u fizici. Epoha je obeležena otkrićem Dekartovog koordinatnog sistema. Tada se javlja potreba za ozbiljnim analitičkim pristupom prema prikupljenim podacima koji su sakupljeni putem različitih popisa. Javljanje ideje i prihvatanje važnosti grafičke interpretacije podataka se dešava u 18. veku. Ta nova epoha obeležena je događajima koji se vezuju za otkrića iz oblasti statistike usled čega se javila potreba za razvojem i ispitivanjem novih oblika i formi koje mogu poslužiti za predstavljanje podataka. Od ovog trenutka pa sve do početka 20. intenzivno se razvijaju nove forme grafičke interpretacije podataka koje su posledica ubrzanog razvoja nauke, privrede i društva kao i dolazak do velikog broja novih otkrića. Usled dešavanja u Evropi i svetu, prestaje razvoj oblasti vizualizacije podataka sve do 1950 kada ponovo počinje intenzivan razvoj. Pojava računara omogućila je razvoj novih tehničkih rešenja koja treba da pomognu da se podaci koje koriste računari interpretiraju na adekvatan način.

Danas kada je svakodnevni život nezamisliv bez upotrebe uređaja poput računara ili mobilnog telefona vizualizacija podataka dobija sasvim novi značaj. Vizualizovani podaci nisu više samo podaci koji se tiču naučnih krugova, već se ozbiljno pristupa razmišljanu kako je najbolje preneti informacije ljudima. Važno je razmotriti kako se informacije koje treba preneti grafički interpretiraju kada čulo vida predstavlja značajno čulo kojim čovek prikuplja i brzo obrađuje informacije oko sebe. Posmatrano sa aspekta računarstva, vizualizacija predstavlja značajan segment u oblasti mašinskog učenja i virtualne stvarnosti.

B *Elementi koji utiču na prenos informacija*

Grafička interpretacija podataka i informacija zavisi od nekoliko elemenata. Prilikom implementacije i odabira grafikona neophodno je razmotriti svaki element. To su elementi kao što su boja, dimenzije, oblici, dodatne funkcionalnosti i dr. Navedeni elementi su samo neki od atributa koji utiču na to kako će grafikon biti protumačen. Jedan pogrešno definisan element može dovesti do toga

DOI: 10.5281/zenodo.4064269

da se pogrešno protumače predstavljeni podaci i informacije.

Boja predstavlja značajan element prilikom implementacije grafikona 4. Često je ovaj element taj koji se koristi kada je potrebno pronaći način kako jasno kategorizovati podatke. Pogrešno odabrane boje mogu dovesti do toga da se podaci izmešaju i na taj način dovesti do toga da korisnik dođe do pogrešnih zaključaka. Postoje dve značajne osobine koje se često koriste prilikom odabira boja, a to su kontrast i nijanse. Kontrast je tehnika kod koje se boje biraju na takav način da su jasno uočljive i da se međusobno odabrane boje jasno razlikuju. Za potrebe prikazivanja kategoričkih podataka ova tehnika može biti od značaja. Sa druge strane nijansa je tehnika kod koje se podaci predstavljaju istom bojom različitog intenziteta. Kada je potrebno predstaviti podatke koji imaju hijerarhijsku strukturu onda se podaci koji pripadaju različitim grupama strukture mogu obojiti različitim nijansama iste boje i to bi ukazivalo da ti podaci pripadaju istoj nadgrupi. Treba voditi računa i da odabrane boje budu u skladu sa ostalim elementima aplikacije.

Dimenzije predstavljaju element koji se može posmatrati i tumačiti na više načina. Važno je da dimenzije grafikona budu usklađene sa ostatkom aplikacije na takav način da ne postoji prazan prostor niti da postoje grafički elementi koji se usled svojih dimenzija preklapaju. Sa druge stane dimenzije se mogu odnositi i na sastavne elemente grafikona. Ako govorimo npr. o koordinatnom sistemu, dimenzija tačaka u njemu može predstavljati dodatnu informaciju koja govori o broju elemenata sa istim koordinatama.

Dimenzije se često kombinuju sa oblicima sastavnih elemenata grafikona. Kao i boja, oblici mogu nositi informacije o grupnoj/klasnoj pripadnosti podataka. Oblici bi trebali da budu dovoljno jasni kako bi se međusobno razlikovali i da budu što jednostavniji kako ne bi nepotrebno skretali korisnikovu pažnju.

Element koji se razlikuje u potpunosti od svih navedenih elemenata jeste implementacija dodatnih funkcionalnosti. Savremene tehnologije i moderne tekovine su načinile da korisnik očekuje interaktivne mogućnosti od strane grafičko-korisničkog interfejsa. Vizualizacija podataka se može posmatrati kao deo grafičko-korisničkog interfejsa koji je pogodan za implementaciju dodatnih funkcionalnosti. Postoji veći broj različitih alata i biblioteka koje pružaju proširenje grafikona na način da im omogućavaju dinamičko ponašanje. Treba biti umeren kao i kod ostalih elemenata. Ne treba implementirati sve mogućnosti svim korišćenim grafikonima, jer bi to dovelo samo do zabune i nerazumevanja informacija. Biblioteke pružaju jedan deo mogućnosti za implementaciju dinamičkih karakteristika grafikona. Prilikom implementacije grafikona neophodno je da se u zavisnosti od razvojnog okruženja programiraju i delovi koji služe da se grafikon poveže sa ostatkom sistema.

III. Softverski alati otvorenog koda za vizualizaciju podataka

Postoji čitav spektar alata koji vizualizuju podatke. Alati se mogu svrstati po različitim kriterijumima. Neki od tih kriterijuma su: licenca softvera, tip podataka koji se vizualizuje, vrsta platforme za koju su grafičke predstave namenjene i dr. U zavisnosti od licenciranja alati mogu biti redovno održavani, transparenti i bezbedni za korišćenje. Što se tiče vrste podataka, ona utiče na ograničenje skupa dostupnih grafikona koji se mogu primeniti. Ako govorimo o vrsti platforme onda treba praviti razliku da li će grafikoni biti deo eksternog alata (veb rešenja) ili deo internog alata (desktop rešenja).

A Biblioteka d3.js

Biblioteka kao što je *d3.js* (pun naziv na eng. *Data-Driven Documents*) je rešenje koje se može primeniti kada je potrebno kreirati složeniji oblik grafičke interpretacije podataka. Biblioteka je namenjena vizualizaciji podataka i obuhvata veliki spektar mogućnosti koje nisu direktno povezane sa samom vizualizacijom kao što je npr. čitanje i obrada formata fajlova kao što su *JSON*, *csv* ili *psv* formata. Biblioteka objedinjuje i koristi tehnologije kao što su *HTML*, *CSS*, *JavaScript* i *SVG*. Biblioteka *d3.js* je opisana u ovom radom pod pretpostavkom da se biblioteka upotrebljava sa klijentske strane, iako se ista može primeniti i sa serverske strane. Alat kao što je *d3.js* karakteriše 3 osobine i to: kompatibilnost, debagovanje i performanse. Kompatibilnost podrazumeva da se alat može kombinovati sa drugim *JavaScript* bibliotekama ili uopšte ostalim bibliotekama sa ciljem da se ostvari što bolji osećaj prilikom korišćenja veb aplikacija 5. Debagovanje kao važan mehanizam otklanjanja grešaka i nedostataka. Kao poslednje, performanse sa ciljem da se rezultati primene *d3.js* biblioteke iskoriste kako bi se implementacija samog veb servisa poboljšala na način da korisnik nikada ne oseti kako nešto radi sporo ili ne radi uopšte.

Alat poseduje veliki broj funkcionalnosti. Neke od funkcija koje su od značaja za vizualizaciju podataka kod veb aplikacija su: kreiranje grafičkog objekta, dohvatanje objekata, operacije nad podacima, interaktivni dodaci 6. Zbog hijerarhiske strukture, alat pruža mogućnost da se prilikom implementacije pristupi pojedinačno svakom od elemenata strukture i na taj način omogućava da se rade fina podešavanja ili obrade izuzeci. Dohvatanje objekata predstavlja važan segment alata. Nije uvek jednostavno sa složene veb stranice izvući tačno određen objekat pogotovu ako su stranice veb servisa obogaćene različitim bibliotekama ili su objekti definisani složenim identifikatorima. *D3* biblioteka ima implementirane kvalitetne funkcije za dohvatanje objekata koje mogu u zavisnosti od prosleđenih parametara objekat identifikovati po različitim kriterijumima. Takođe, dohvatanje objekata ne mora biti samo slučaj kada se traži eksplicitno jedan objekat koji ispunjava uslov, već se mogu dohvatati nizovi objekata koji ispunjavaju kriterijume prosleđene kao argument funkcija za dohvatanje objekata. Podaci koji se vizualizuju se mogu obrađivati uz pomoć *D3* biblioteke. To omogućava da se isti podaci mogu obrađivati na različite načine što dovodi do toga se na kraju od istih početnih podataka dobiju različite interpretacije podataka i to ne samo po grafičkoj strukturi već i po nameni. Kako bi se korisnik zainteresovao za elemente grafikona koji su od značaja ili

da bi se veliki skup podataka predstavio višeslojnom strukturom, alat *d3.js* pruža mogućnost implementacije interaktivnih elemenata. Interaktivni elementi služe da prilikom korisnikovog rada sa grafikonom, koji može biti prelazak „miša“ preko grafikona, pritisak tastera i dr., dođe do transformacije grafikona i na taj način se prikažu novi elementi. Treba biti umeren sa interaktivnim funkcionalnostima iz razloga što preterana primena može dovesti do toga da se korisniku u potpunosti skrene pažnja sa suštine i time dođe do ne razumevanja informacija.

U okviru *NI4OS* projekta, realizovan je veći broj različitih grafikona primenom *d3.js* biblioteke. Razlog za primenu *d3.js* biblioteke jeste taj što je bilo neophodno razviti složenu logiku i kompleksnu grafičku interpretaciju. Realizovana rešenja dostupna su na https://ni4os.eu/survey-results/

Alat sadrži veliki skup primera koji se mogu primeniti. Kao početan izvor grafikona veoma je korisna lista gotovih primera dostupnih na https://observablehq.com/@d3/gallery. Navedeni servis nudi mogućnost da pre nego što se donese odluka o primeni određenog grafikona izvrše modifikacije i testiranja sa konkretnim podacima i na taj način uštedi na vremenu ako se ispostavi da željeni grafikon nije najadekvatniji za primenu. Odabrani grafikon se može naknadno modifikovati i prilagoditi. Dodatno, kod većine primera postoji i hijerahiska struktura koja dozvoljava izmene i promene kako bi se detalji prilagodili određenoj vrsti podataka.

Jedan primer izvorne verzije grafikona dostupan je na https://gist.github.com/kerryrodden/766f8f6d31f645c39f488a0befa1e3c8. Isti navedeni primer je korišćen prilikom realizacije grafičkog rešenja za potrebe NI4OS projekta. Usklađivanje i modifikacije se ogledaju pre svega u potrebi da se grafikon podesi na način da adekvatno predstavlja podatke sa složenom strukturom i da tako predstavljeni podaci budu jasni korisniku. Na slici 1 je prikazana prilagođena verzija grafikona. Takođe, bilo je neophodno pronaći adekvatan način obeležavanja delova grafikona kako bi oni bili usklađeni sa ostalim elementima veb stranice.

Slika 1: Prilagođena verzija realizovana za potrebe NI4OS projekta

B Biblioteka chart.js

Još jedna biblioteka koja se može koristiti za vizualizaciju podataka jeste *chart.js* biblioteka. Ta biblioteka je takođe pogodna da se kombinuje sa programima na *JavaScript* jeziku. Poseduje vrlo jednostavan interfejs za korišćenje. Pogodna je za implementaciju iz razloga što se veoma brzo i lako mogu napraviti grafikoni koji ispunjavaju većinu korisničkih zahteva. Poseduje veliki broj gotovih grafikona dostupnih na https://www.chartjs.org/samples/latest/. Osobina koju treba istaći jeste mogućnost transformisanja grafikona. Grafikoni su jednostavnije strukture i vrlo lako se može jedan grafikon transformisati u neki drugi oblik grafikona. Kada je reč o podešavanju postojećih grafikona, postoji veći broj funkcija koje imaju jednostavan oblik pozivanja i koje omogućavaju implementaciju dodatnih funkcionalnosti. Nedostatak kod *chart.js* je taj što ako korisnik želi određenu modifikaciju koja nije predviđena od strane samog alata za vizualizaciju mora se dodatno razviti deo programa koji će implementirani željenu modifikaciju.

Biblioteka *chart.js* sadrži veliki broj gotovih primera koji se uz promenu skupa podataka koji se vizualizuju i uz često male promene u programu kreira željeni grafikon. Postoji mnoštvo grafikona koji se po tipu mogu klasifikovati kao ista vrsta grafikona ali zbog sitnih detalja međusobno se razlikuju. Sitni detalji po kojima se grafikoni razlikuju predstavlja jednu vrstu modifikacije grafikona. Primer grafikona sa ovom osobinom može se videti na https://www.chartjs.org/samples/latest/charts/area/line-boundaries.html.

Značajna prednost *chart.js* biblioteke je jednostavna integracija sa radnim okruženjima kao što su *Angular* i *React*. Ako govorimo o *Angular* radnom okruženju može se u projekat dodati paket pod nazivom *ng2-charts* 7. Navedeni paket omogućava da se u *Angular* razvojno okruženje dodaju grafikoni kreirani *chart.js* bibliotekom. Biblioteka *chart.js* zbog svoje osobine da se veličina grafikona dinamički prilagođava veličini ekrana uređaja, predstavlja čest izbor i dobru kombinaciju kada se veb servisi razvijaju tehnologijom kao što je *Angular*. Ako se koristi *React* radno okruženje, *chart.js* se sa lakoćom može integrisati pre svega zbog svoje jednostavne i jasno definisane strukture objekata. Primer paketa koji bi se mogao primeniti kako bi se koristila *chart.js* biblioteke jeste *react-charts-2* 89. Navedeni paket predstavlja omotač za *chart.js*.

IV. Zaključak

Postoji veći broj elemenata koji utiču na izgled grafikona i na način na koji će grafikoni preneti informacije. Navedene biblioteke su samo jedna od mogućih rešenja. Prilikom implementacije bilo kakvih grafičkih elemenata treba biti obazriv. Prenos informacija vizualizacijom podataka je svakako jedan način kako se ljudi mogu informisati.

Navedene biblioteke imaju svoje prednosti i mane. Biblioteka *d3.js* može biti veoma zahtevna po pitanju neophodnog znanja za njenu primenu i mnoge funkcionalnosti su nepotrebne za konkretnu implementaciju ali biblioteka pruža veliki stepen prilagođavanja i pogodna je za vizualizaciju velikih količina podataka. Sa druge strane *chart.js* biblioteka poseduje manji skup grafikona i teže je prilagoditi grafikon konkretnoj implementaciji dok sa druge strane je

biblioteka jednostavna za primenu i lako se integrira sa postojećim *JavaScript* programom 10.

Zahvalnica

Autori su zahvalni Milici Ševkušić, bibliotekarki u Tehničkom institutu SANU i prof. Nadici Miljković na datoj prilici i pruženoj podršci.

Rad je napisan u okviru projekta NI4OS-Europe - National Initiatives for Open Science in Europe (H2020 no. 857645).

Literatura

1 Friendly, Michael & Chen, Chun-houh & Härdle, Wolfgang Karl & Unwin, Antony. (2008). A Brief History of Data Visualization. 10.1007/978-3-540-33037-0_2

2 Marvin A. Ruder. The Visual Display of Quantitative Information. https://hci.iwr.uni-heidelberg.de/system/files/private/downloads/385875839/marvin-ruder_report.pdf

3 Jakub Majorek. 14 JavaScript Data Visualization Libraries in 2020. https://www.monterail.com/blog/javascript-libraries-data-visualization

4 Maureen Stone. Choosing Colors for Data Visualization. January 17, 2006. (dostupno na https://www.perceptualedge.com/articles/b-eye/choosing_colors.pdf)

5 Deqing Li, Honghui Mei, Yi Shen, Shuang Su, Wenli Zhang, Junting Wang, Ming Zu, Wei Chen. ECharts: A declarative framework for rapid construction of web-based visualization. Visual Informatics. Volume 2, Issue 2. 2018. Pages 136-146. ISSN 2468-502X. https://doi.org/10.1016/j.visinf.2018.04.011 (http://www.sciencedirect.com/science/article/pii/S2468502X18300068)

6 Guldamlasioglu, Selin. (2015). Web-based information visualization using JavaScript. (Dostupno na https://trepo.tuni.fi/handle/10024/97846)

7 Sebastian Eschweiler. Angular & Chart.js (with ng2-charts). (dostupno na https://medium.com/codingthesmartway-com-blog/angular-chart-js-with-ng2-charts-e21c8262777f)

8 Peter Cook. Creating a dashboard with React and Chart.js. Jan 28, 2019. (dostupno na https://www.createwithdata.com/react-chartjs-dashboard/)

9 Dmitry Rogozhny. Quick Introduction to Displaying Charts in React with Chart.js and react-chartjs-2. March 10th, 2020. (dostupno na https://www.newline.co/@dmitryrogozhny/quick-introduction-to-displaying-charts-in-react-with-chartjs-and-react-chartjs-2--a85b4e2e)

10 Omar Almootassem and Syed Hamza Husain and Denesh Parthipan and Qusay H. Mahmoud. A Cloud-based Service for Real-Time Performance Evaluation of NoSQL Databases. (2017). 1705.08317. arXiv. (dostupno na https://arxiv.org/ftp/arxiv/papers/1705/1705.08317.pdf)

Otvoreni hardver za realizaciju pogona električnog bicikla

Aleksandar R. Milić
Univerzitet u Beogradu – Elektrotehnički fakultet, Beograd, Srbija
milic.aleksandar@etf.bg.ac.rs

Rezime: U ovom radu je predstavljen oblik sistemskog korišćenja otvorenog hardvera i slobodnog softvera u otvorenom obrazovanju u formi studentskog tima. Rad sadrži prikaz materijala dostupnih na *GitHub*-u i drugim lokacijama koji se koriste za rad studentskog tima i unapređenje nastave na završnim godinama studija Elektrotehničkog fakulteta - Univerziteta u Beogradu. Kao reprezentativni primer otvorenog hardvera u formiranom sistemu dat je aplikativni prikaz kroz realizaciju projekta pogona električnog bicikla kao i procedura za repliciranje istog. Na kraju su dati predlozi za aktivnije korišćenje otvorenih materijala u cilju poboljšanja performansi predstavljenog sistema. Pogon električnog bicikla prezentovan u radu projektovao je *H-Bridges* tim studenata Elektrotehničkog fakulteta iz Beograda tokom učešća na svetskom takmičenju u inovativnim rešenjima *IEEE International Future Energy Challenge 2019.*

Ključne reči: pogon; električni bicikl; slobodno obrazovanje; otvoreni hardver

I. Uvod

Primena otvorenog hardvera i slobodnog softvera u svrhu otvorenog obrazovanja je sve šire zastupljena na prestižnim univerzitetima širom sveta [1]. Elektrotehnički fakultet Univerziteta u Beogradu (ETF) u tom duhu pruža u okviru mnogobrojnih kurseva na različitim nivoima studija sveobuhvatnu otvorenu literaturu [2]. Da je trend otvorenog hardvera i softvera sve prisutniji u univerzitetskom obrazovanju pokazuje i činjenica da se sve više profesora, ali i studenata oslanja na deljenje stručne literature, ali i na druge prakse otvorenog obrazovanja. Na taj način studentima se pruža mogućnost da prate rad na projektima svojih kolega sa drugih univerziteta, kao na primer Univerziteta u Koloradu ili Virdžinija Teka korišćenjem jednostavih platformi kao što je *GitHub* [3], [4].

U duhu otvorene kulture, 2018. godine na ETF-u je osnovan tim studenata pod nazivom *H-Bridges.* Tim je osnovan sa ciljem da se studentima pruži drugačiji pristup obrazovanju iz različitih oblasti elektrotehnike. *H-Bridges* svake godine učestvuje na svetskom takmičenju u oblasti inovativnih i energetski efikasnih rešenja pod pokroviteljstvom IEEE organizacije (eng. *Institute of Electrical and Electronics Engineers*) pod nazivom *IEEE Interantional Future Energy Challenge* – IEEE IFEC [5]. Glavni cilj takmičenja jeste deljenje hardverskih i softverskih rešenja između studenata sličnih profila na svetskom nivou. Ovo je rezultovalo formiranjem sistema otvorenog obrazovanja u okviru *H-Bridges*-a u cilju postizanja vrhunskih rezultata.

Osnovna aktivnost *H-Bridges* tima jeste rešavanje jednogodišnjeg multidisciplinarnog tehničkog zadatka projektovanjem inovativnog hardvera i softvera. Pored tehničkog dela, studenti se tokom jednogodišnjeg rada u laboratoriji upoznaju se radom u timu i stiču niz organizacionih i *soft skills* veština kroz kontinualnu komunikaciju sa kompanijama (eng. *fundraising*) i medijima (eng. *public relations*). Od 2018. godine na ETF-u je na taj način započet trend otvorene kulture i formiranje drugačijeg pristupa tradicionalnom obrazovanju u različitim oblastima elektrotehnike kroz sistemsko korišćenje otvorenih materijala i prenos znanja sa generacije na generaciju. Tehnička i ogranizaciona rešenja svih generacija studentskog tima, kao i u radu predstavljeno rešenje električnog bicikla, integrisani su u sistem funkcionisanja u *H-Bridges*-u, a dodatno su dostupni svima na analizu i korišćenje.

Pogon za električni bicikl snage 700 W razvijen je u Laboratoriji za električna vozila na ETF-u. Tehničko rešenje, koje je timu donelo i prvu nagradu, predstavljeno je jula 2019. godine u poslednjoj fazi takmičenja u Viskonsinu, SAD. Rešenje je predstavljeno kroz aplikativnu realizaciju električnog bicikla, modifikacijom tradicionalnog *Off-Road* bicikla kompanije *Capriolo.* Realizovani prototip prikazan je na Slici 1.

Pored projekta električnog bicikla koji će u poglavlju III biti detaljnije predstavljen, projekti koji su takođe delom ili potpuno otvoreni nalaze se u Tabeli 1. To su projekat brzog punjača za baterije sa galvanskom izolacijom, sistem napajanja nanosatelita i drugi. Hardverska i softverska tehnička rešenja koja su nastala kao rezultat projekata navedenih u Tabeli 1 sistemski su integrisana u različite aktivnosti rada studentskog tima i kontinualno se koriste u nastavi.

Slika 1. Realizovani električni bicikl u okviru *H-Bridges* tima na Elektrotehničkom fakultetu u Beogradu sa osnovnim elementima: 1) električna mašina, 2) invertorska jedinica, 3) ručica za gas i 4) baterija

DOI: 10.5281/zenodo.4064695

Tabela 1: Pregled delimično ili potpuno otvorenih projekata koji se mogu naći na [6] i sajtu ddc.etf.rs

Projekat	Nominalna snaga [W]
Sistem za brzo punjenje baterija sa galvanskom izolacijom	1100
Monofazni ispravljač za priključenje obnovljivih izvora na mrežu	1300
Pogon za električni bicikl	700
Sistem za napajanje nanosatelita	80
Integrisani monofazni pogon sa asinhronom mašinom sa kaveznim rotorom	900

II. *H-Bridges* studentski tim i otvoreno obrazovanje

Za svaki od projekata u Tabeli 1 postoje skupovi materijala koji su dostupni za studente ETF-a i druge korisnike. Jedan skup sadrži izvorni (eng. *source*) kod sa kontrolnim algoritmom, paket simulacija, spisak hardverskih elemenata, BOM (eng. *Bill Of Materials*) i gerber fajlove za projektovanje štampane ploče – PCB (eng. *Printed Board Circuit*). Otvoreni materijali koji su rezultat naučno - istraživačkog rada *H-Bridges* tima i proisteklih master i diplomskih radova mogu se pronaći na [6] i sajtovima ddc.etf.rs i emp.etf.rs/rddc.htm (pristupljeno 26. avgusta 2020).

Materijali predstavljaju hardverska i softverska tehnička rešenja za aplikacije navedene u Tabeli 1. Grupisani su uz prateće tekstualne zapise prema subjektivno definisanom težinskom faktoru i intezivno se koriste u okviru tri kursa na osnovnim i master studijama ETF-a. Na trećoj godini studenti korišćenjem skupova sa nižim težinskim faktorom obnavljaju i proširuju elementarna znanja. Skupove sa najvišim težinskim faktorom studenti koriste na master studijama za rešavanje kompleksnijih tehničkih problema u okviru zasebnih oblasti za istraživanje. Tehničke probleme najpre rešavaju korišćenjem otvorenih softverskih rešenja, a zatim na eksperimentalnim postavkama koje su rezultat prethodnih projekata. U realnom okruženju na dostupnom hardveru imaju priliku da se upoznaju sa nemodelovanim problemima koji se u teorijskim delovima kurseva uglavnom zanemaruju.

Pored upotrebe otvorenih hardverskih i softverskih rešenja u tradicionalnim nastavnim aktivnostima, ista su integrisana u sistemsku strukturu *H-Bridges* tima. Njihova upotreba može se grubo predstaviti kao:

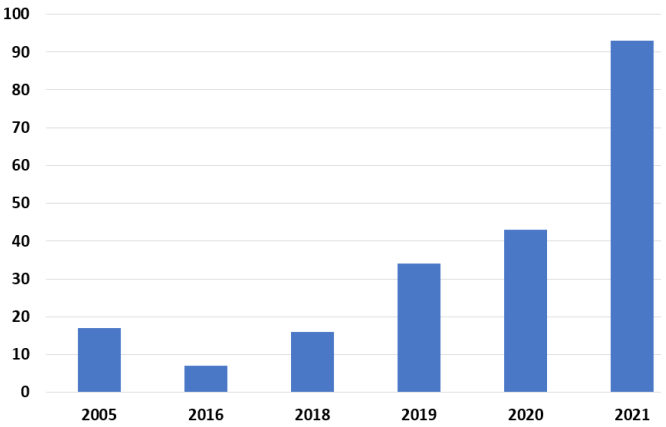
1) Upotreba materijala pri formiranju novog tima
2) Upotreba materijala tokom izrade novog rešenja

Tokom prvog meseca formiranja nove generacije tima, otvorena tehnička i organizaciona rešenja prethodnih timova se intezivno koriste kroz stručne obuke i treninge koje stariji članovi drže za mlađe. Prve četiri radne nedelje po formiranju novog tima nova generacija studenata ima zadatak da replicira kosture projekata prethodnih timova uz svakodnevne konsultacije sa starijim članovima, mentorom i korišćenjem otvorenih hardverskih i softverskih rešenja po ugledu na dobre prakse primene otvorene nauke u obrazovanju [7]. Na kraju četvrte nedelje za nove članove se organizuje KTW (eng. *Knowledge Transfer Weekend*). KTW predstavlja finale uvodnih obuka, na kojem studenti u roku od tri dana posećuju 10 treninga i radionica. Prvih 8 treninga je posvećeno prethodnim projektima i rešavanju problema sa kojima su se susretale prethodne generacije uz korišćenje otvorenih tehničkih rešenja. Poslednja dva treninga služe kao uvod za naredni projekat i njih po pravilu drže predstavnici iz industrije.

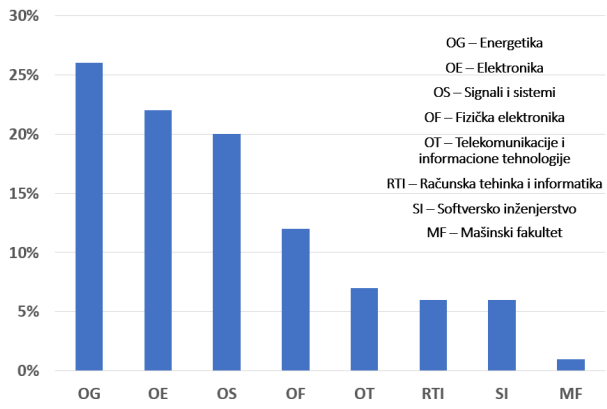
Nakon uvodnog meseca, tokom rešavanja aktuelnog zadatka za tekuću generaciju, studenti kontinualno koriste i pozivaju se na dostupna tehnička rešenja. Kroz ovakav sistemski pristup nove generacije studenata se upoznaju sa problemima sa kojima su se susretale njihove kolege, uče da rade zajedno na rešavanju istih i ubrzano stiču praktično iskustvo kroz ponavljanje ranijih projekata. Ovako se postiže velika efikasnost pri učenju i brzo savladavanje niza problema koji su u startu zajednički za sve generacije. Dodatno, prethodna otvorena rešenja postaju temelj za rešavanje aktuelnog zadatka.

Tokom aktivnog rada na projektu i korišćenja otvorenih rešenja, ista se redovno unapređuju, održavaju i spremaju za narednu generaciju. Formiranim sistemom znanje se prenosi kroz generacije preko studenata, odstranjujući potencijalni jaz na relaciji student – predavač. Rezultat predstavljenog sistema otvorenog obrazovanja koji odstupa od tradicionalnog rezultovao je brojnim svetskim nagradama i priznanjima za *H-Bridges* tokom prethodnih godina u konkureciji najvećih svetskih univerziteta [5].

Drugi rezultat je broj studenata u timu koji raste iz godine u godinu, Slika 2. Poslednjoj generaciji tima pristupilo je rekordna 93 člana (39% studentkinja, 61% studenata) sa svih odseka ETF-a, što je u nivou najbrojnih odseka na Fakultetu.



Slika 2. Broj prijavljenih studenata za *H-Bridges* tim po generaciji



Slika 3. Raspodela prijavljenih studenata po odsecima 2020/21

Pregled novih članova tima po odsecima ETF-a u 2020. godini dat je grafički na Slici 3. Tokom izrade zadatka timu takođe u toku godine pristupi određeni broj studenata Mašinskog, Ekonomskog i Fakulteta organizacionih nauka Univerziteta u Beogradu. Broj studenata na izbornim predmetima na kojima se otvorena tehnička rešenja koriste takođe je u konstantnom porastu, ali statistički podaci neće biti izneti u ovom radu zbog mnogobrojnih dodatnih parametara koji na iste utiču.

III. Otvoreni hardver za realizaciju pogona električnog bicikla

U nastavku je predstavljen jedan aplikativni prikaz otvorenog hardvera kroz tehničko rešenje električnog bicikla prikazanog na Slici 1. Dodatno, predstaviće se metoda za repliciranje tehničkog rešenja električnog bicikla bez izmena u hardverskom i softverskom dizajnu.

U Tabeli 2 predstavljeni su svi elementi električnog bicikla i njihova cena. Osim invertorske jedinice (Slika 1 element broj 2) svi elementi se mogu kupiti i montirati prema uputstvu proizvođača. Za projektovanje invertorske jedinice, postoji otvoreno hardversko rešenje sa svim potrebnim materijalima: gerber fajlovi, BOM, izvorni kod i 3D model za projektovanje kućišta [6]. U nastavku svaki element električnog bicikla biće ukratko predstavljen kao i procedura za repliciranje invertorske jedinice. Korisnik navedene elemente može montirati na tradicionalni bicikl koji već ima u vlasništvu ili na drugi proizvoljni model. Sistem je organizovan tako da je adaptivan za montiranje na bicikl proizvoljnih dimenzija.

Bicikl na Slici 1 na sebi ima *Brushless DC* električnu mašinu snage 3000W. Prednja strana električne mašine integriše Holov senzor koji prosleđuje invertorskoj jedinici informacije o realizovanoj vučnoj sili. Zadnja strana integriše 1:6 mehanički reduktor koji ima za cilj da visoke brzine na kojima radi električna mašina spusti na brzine potrebne prosečnom korisniku, od 0 do 100 obrtaja u minutu. Mašina je dostupna u slobodnoj prodaji sa integrisanim elementima i sistemom sa nošenje [8].

Na guvernali bicikla postavljena je ručica za gas kojom korisnik kontroliše vučnu silu na obodu točka. Ručica za gas sa uputstvom za monitranje dostupna je za kupovinu u slobodnoj prodaji [9].

Na realizovan prototipu korišćene su litijum polimer baterije kapaciteta 250 Wh, 55 V, 10 A. Pri kupovini elemenata, baterije, električne mašine i gas ručice, preporučuju se elementi navedeni u otvorenom BOM-u. Predložene baterije pružaju korisniku dovoljnu autonomiju u gradskim uslovima i potrebne karakteristike iz ugla elektro parametara za odgovarajući rad invertorske jedinice. Varijacije u izboru baterije mogu dovesti do neželjenih pojava i kvara pri trajnom radu. Uz kupovinu predloženih baterija korisnik dobija odgovarajući punjač za njihovo brzo punjenje koje može iskoristiti na običnoj monofaznoj utičnici. Uz kupljeni baterijski paket dolazi i torba za njihovo postavljanje na bicikl.

Razvijeni prototip invertorske jedinice prikazan je na Slici 4. Razvijena invertorska jedinica sa prilagodnim kućištem predstavlja glavni doprinos u naučno – istraživačkom radu u okviru predstavljenog projekta. Za razvoj invertorske jedinice u prvom koraku, bez otvorenih materijala, bilo bi neohodno odraditi inicijalni dizajn hardvera i softvera, simulacije karakterističnih režima i problema, odabir svih hardverskih elemenata, projektovanje štampane ploče i kontrolnog algoritma, itd. Sa otvorenim, dostupnim i organizovanim materijalima, procedura projektovanja ovakvog rešenja je dosta pojednostavljena i ubrzana. Kako se materijali konstantno unapređuju u cilju korišćenja istih od strane korisnika sa elementarnim znanjima iz elektrotehnike, projektovanje ovakvog tehničkog rešenja od strane posrednih korisnika koji ranije nisu imali prilike da se susretnu sa tematikom postaje izvodljivo. U nastavku su navedeni koraci koje je neophodno sprovesti do repliciranja predstavljenog rešenja za električni bicikl.

U prvom koraku neophodno je izraditi štampanu ploču, što se po niskoj ceni može izvesti od strane podizvođača iz NR Kine [10]. Dokumentacija potrebna za izradu iste je dostupna u vidu gerber fajlova [6]. Paralelno je potrebno poručiti sve elemente iz BOM-a koji sadrže pored elementa za invertorsku jedinicu i električnu mašinu sa sistemom za nošenje, baterijski paket i ručicu za gas. Nakon prikupljanja svih elemenata potrebno je zalemiti iste na štampanoj ploči. Na ploči svaki element ima svoju jedinstvenu oznaku koja odgovara jednom od elemenata u BOM-u. Nakon lemljenja potrebno je odgovarajući izvorni kod spustiti na procesor koji se nalazi u okviru invertorske jedinice. Za ovaj korak, postoji intuitivna procedura u vidu tekst dokumenta, ali je poželjno učešće stručnog tehničkog lica. Po postavljanju invertorske jedinice u kućište, pobrojani elementi se mogu montirati na proizvoljni tradicionalni bicikl. Na kraju je potrebno povezati izlaze svih elemenata za jednoznačno obeležene ulaze na invertorskoj jedinici.

Nakon ispravnog povezivanja sistema u okviru invertorske jedinice se izvršava kontrolni algoritam, bez posredstva korisnika, koji na bazi prikupljenih informacija (sa baterije, ručice za gas i električne mašine) na obodu točka korisniku pruža željenu vučnu silu.

Slika 4. Razvijeni otvoreni prototip invertorske jedinice u okviru *H-Bridges* tima za električni bicikl

Invertorska jedinica na sebi poseduje *Bluetooth* modul uz koji je pridružena jednostavna Android aplikacija otvorenog koda za telefon. Aplikacija pruža korisniku informacije o translatornoj brzini, trenutnom kapacitetu baterije, srednjoj brzini i prosečnoj potrošnji baterije.

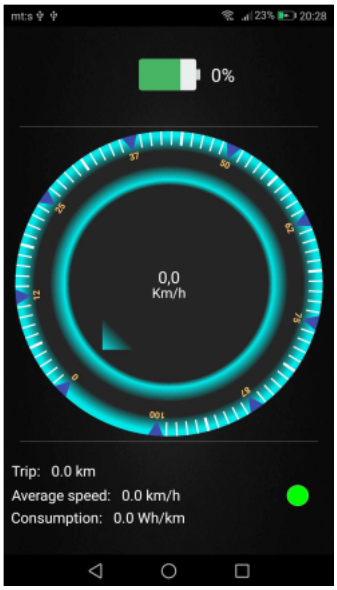
Tabela 2: Cena elemenata sistema za pravljenje replike električnog bicikla izražena u dolarima

Elementi sistema	Cena [$]
Električna mašina sa sistemom za montrinje, dostavom i carinom	350
Izrada štampanih ploča sa dostavom i carinom	60
BOM i usluga lemljenja	200
Izrada kućišta i monitranje sistema	30
Baterija 250 Wh sa adekvatnim punjačem	200
Android aplikacija	0
Proizvoljni tradicionalni bicikl	200

Korisnički interfejs razvijene aplikacije prikazan je na Slici 5. Pri prvom korišćenju aplikacije korisnik unosi svoj jedinstveni nalog, naziv bicikla i lozinku, nakon čega samo vlasnik bicikla može da upravlja istim. Pored osnovnih opservabilnih funkcija aplikacije, korisnik ima mogućnost da definiše ograničenja po snazi, brzini i vučnoj sili pogona. Na ovaj način pruža se prilika korisniku da definiše ograničenja pri vožnji u slučaju da bicikl koristi osoba za koju je potrebna posebna pažnja, kao što su deca niskog uzrasta. Fajl za instaliranje aplikacije je takođe otvoren [6].

Razvijeni algoritam za kontrolu u okviru invertorske jedinice omogućava korisniku da biciklom upravlja bez korišćenja pedala, odnosno da brzinu kretanja diktira isključivo pomoću ručice za gas. Takođe, zbog implementiranog „*clutch*“ sistema u okviru reduktora na električnoj mašini, ovakav adaptivni sistem korisnik može da koristi i kao tradicionalni bicikl bez učešća elektromotornog pogona u vožnji. Potrebno je jedino isključiti pogon pomoću opcije u aplikaciji nakon čega se inercija pogona neće osećati pri vožnju.

Iako je navedeni postupak na prvi pogled jednosmeran, prosečan korisnik pri pokušaju realizacije navedenog rešenja može naići na mnogobrojne probleme.



Slika 5. Korisnički interfejs razvijene Android aplikacije u okviru projekta električnog bicikla *H-Bridges* tima

Deo problema pri realizaciji predstavljenog tehničkog rešenja ogleda se pri lemljenju invertorske jedinice gde može doći do niza problema usled nestručnog rukovanja. Takođe, pri motiranju sistema za nošenja električne mašine takođe može doći do neželjenih mehaničkih problema pri režimima polaska, kada korisnik pokreće svoj bicikl iz režima mirovanja. Ovi problemi se mogu otkloniti dodatnom automatizacijom procedura za projektovanje i montiranje sistema formiranjem materijala za potrebe korisnika sa skromnijim tehničkim znanjem.

IV. Zaključak

U radu je dat aplikativni prikaz otvorenog hadrvera i slobodnog softvera kroz realizaciju pogona električnog bicikla. Ukratko je predstavljena procedura za izradu prototipa. Predstavljen je sistemski pristup za korišćenje otvorenih hardverskih i softverskih rešenja na proizvoljnom broju studenata u formi studentskog tima. Predstavljeni su statistički rezultati u formi odziva kod studenata za učešće u predstavljenom sistemu. Nedovoljna vidljivost predstavljenih rešenja može se promeniti većim korišćenjem materijala u okviru različitih kurseva na ETF-u. Na ovaj način bi i studenti koji nisu aktivno uključeni u *H-Bridges* tim mogli da raspolažu sa različitim hardverskim i softverskim rešenjima. Na kraju, veći podsticaj za studente da nastave sa naučno – istraživačkim radom na predstavljenim projektima bi sigurno doveo do bogatijeg sadržaja istih i bolje organizacije dostupnih materijala.

Zahvalnica

Autor je zahvalan prof. dr Nadici Miljković za veliku podršku i vredne komentare tokom pisanja rada. Posebnu zahvalnost autor duguje svim studentima koji su prošli kroz *H-Bridges* tim od 2018. godine i što su velikom željom, požrtvovanjem i disciplinom gradili sistem koji danas postoji. Zahvalnost u ime svih članova tima upućena je i svim kompanijama koje su prepoznale predstavljenu ideju iza *H-Bridges* tima i nesebično je podržale.

Dostupnost podataka: Podaci su dostupni na GitHub stranici (https://github.com/Mostovi).

Literatura

[1] https://ocw.mit.edu/ans7870/global/05_Prog_Eval_Report_Final.pdf? [pristupljeno sep. 3, 2020]
[2] N. Miljković, "Otvoreni nastavni materijali: Interna iskustva," *Primena slobodnog softvera i otvorenog hardvera PSSOH 2018*: 16.
[3] https://github.com/jackdski/ifec [pristupljeno avg. 25, 2020]
[4] https://github.com/jhensal33/IFEC [pristupljeno avg. 25, 2020]
[5] http://energychallenge.weebly.com/ [pristupljeno avg. 25, 2020]
[6] https://github.com/Mostovi [pristupljeno sep. 1, 2020]
[7] J. R. Wagge, M. J. Brandt, L. B. Lazarevic, N. Legate, C. Christopherson, B. Wiggins, J. E. Grahe, "The collaborative replications and education project," *Frontiers in psychology*, vol 10, pp. 247, 2019
[8] https://lunacycle.com/cyclone-mid-drive-3000w-planetary-kit/ [pristupljeno avg. 25, 2020]
[9] https://www.amazon.com/slp/electric-throttle/f4ppa4356yddvbt [pristupnljeno avg. 25, 2020]
[10] https://jlcpcb.com/ [pristupljeno avg. 25, 2020]

Free software in Meteorology

Ivana Tošić
University of Belgrade-Faculty of Physics, Institute of Meteorology, Belgrade, Serbia
itosic@ff.bg.ac.rs

Abstract: In meteorology, the importance of open science and open data is recognized. Huge amount of data is needed for weather and climate forecast. Traditionally, GFortran is used for development of Numerical Weather Prediction models and number crunching. One of the most applied free software packages is the Grid Analysis and Display System (GrADS), that has been developed for manipulation and visualization of meteorological data. The University of Belgrade - Faculty of Physics is the only educational and research institution in Serbia where the meteorology is studied. Students learn about processes in the atmosphere by applying free software in several courses. The application of free open source software in education and research in meteorology is presented.

Keywords: Meteorology; GFortran; Xmgr; GrADS; LibreOffice.

I. Introduction

Meteorology applies physical and mathematical methods to understand and forecast the processes in the atmosphere that determine a weather and climate. Meteorology makes extensive use of the high-performance computing and tools of mathematics and physics, adapted to analyze atmospheric phenomena using large datasets.

The Faculty of Physics in Belgrade is the oldest national and leading educational and research institution in the field of physical and meteorological sciences in Serbia. The Faculty of Physics is organizing four undergraduate and master groups of studying: General Physics, Theoretical and experimental Physics, Computer and applied Physics, and Meteorology. About 25 students enroll meteorology each year, which encourage individual work on computers.

II. Free Software for Meteorology Calculation, Visualization, and Modelling

There is a computer laboratory equipped with 16 personal computers in Institute of Meteorology at the Faculty of Physics. Linux 4.9 and LibreOffice 5 are installed at all computers. Free software is applied at the following courses: Statistics in Meteorology, Fortran in Meteorology, Micrometeorology, Modelling of the Atmosphere I, Modelling of the Atmosphere II, Climatology, Applied Meteorology and Weather Forecast. Students use LibreOffice Calc, LibreOffice Writer, Xmgr, GrADS and GFortran to complete their works.

LibreOffice is a free and open-source office suite, a project of the Document Foundation (https://www.libreoffice.org/). It was forked in 2010 from OpenOffice.org, which was an open-sourced version of the earlier StarOffice. The LibreOffice suite consists of programs for word processing, creating and editing of spreadsheets, slideshows, diagrams and drawings, working with databases, and composing mathematical formulae.

The GNU General Public License (GNU GPL or simply GPL) is a series of widely-used free software licenses that guarantee end users the freedom to run, study, share, and modify the software. Historically, the GPL license family has been one of the most popular software licenses in the free and open-source software domain. Prominent free software programs licensed under the GPL include the Linux kernel and the GNU Compiler Collection (GCC).

The purpose of the **GNU Fortran (GFortran)** project is to develop the Fortran compiler front end and run-time libraries for GCC, the GNU Compiler Collection (https://gcc.gnu.org/fortran/). GFortran development is part of the GNU Project. It is suitable for free number crunching to a broad spectrum of platforms and users. The GFortran compiler is fully compliant with the Fortran 95 Standard and includes legacy F77 support. GFortran development follows the open development process.

Xmgr (an early name for Grace) is one of the two most prominent graphing packages for Linux (https://en.wikipedia.org/wiki/Grace_(plotting_tool). The name stands for "**GRaphing, Advanced Computation and Exploration of data**". The Grace project was started as a fork, released under the GPL. **Xmgr** is a 2D plotting tool for workstations or X-terminals. A few of its features are:

- User defined scaling, tick marks, labels, symbols, line styles, colors;
- Batch mode for unattended plotting;
- Read and write parameters used during a session;
- Polynomial regression, splines, running averages, DFT/FFT, cross/auto-correlation;
- Hardcopy support for PostScript, HP-GL, jpeg, pdf, and png format.

While **Xmgr** has a convenient point-and-click interface, most parameter settings and operations are available through a command line interface. Weakness is that is no undo feature.

The Grid Analysis and Display System (GrADS) is an interactive desktop tool that is used for easy access, manipulation, and visualization of earth science data (http://cola.gmu.edu/grads/). GrADS, developed and supported at COLA/George Mason University, is freely available. GrADS supports many data file formats, including binary (stream or sequential), GRIdded Binary or General Regularly-distributed Information in Binary form (GRIB), Network Common Data Form (NetCDF), and the Binary Universal Form for the Representation of meteorological station data (BUFR). GrADS has been implemented worldwide on a variety of commonly used operating systems and is freely distributed over the Internet. GrADS uses a 5-Dimensional data environment: the four conventional dimensions (longitude, latitude,

DOI: 10.5281/zenodo.4064715

vertical level, and time) plus an optional fifth dimension for grids that is generally implemented but designed to be used for ensembles. Data from different data sets may be graphically overlaid, with correct spatial and time registration. Operations are executed interactively by entering FORTRAN-like expressions at the command line. A rich set of built-in functions are provided, but users may also add their own functions as external routines written in any programming language. GrADS has a programmable interface (scripting language) that allows for sophisticated analysis and display applications.

For example, to create a plot of wind field in GrADS, one can use commands similar to these:

```
ga-> sdfopen uwnd.mean.nc
Found displayable variable uwnd with 0 levels in SDF file
Data file uwnd.mean.nc is open as file 1
LON set to 0 360
LAT set to -89 89
LEV set to 0 0
Time values set: 1980:1:1:0 1980:1:1:0
ga-> d uwnd
```

III. Free Software in Meteorology Education

A. Statistics in Meteorology

Students use LibreOffice Calc and LibreOffice Writer for completing their homework. Precipitation and temperature data are downloaded from the European Climate Assessment & Dataset site (https://eca.knmi.nl/dailydata/predefinedseries.php).

Example of precipitation distribution for Belgrade during the period 1951-2000 is shown in Fig. 1. Absolute frequency obtained from observations is presented with solid line, while theoretical distribution (Normal pdf) with dashed line. It can be seen that the Normal pdf can be used to describe distribution of the annual precipitation sums in Belgrade.

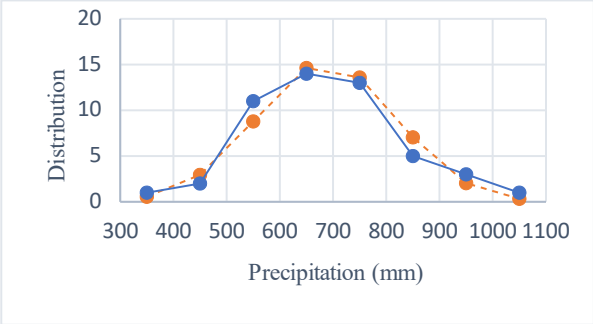


Figure 1: Frequency distribution (solid line) and Normal probability density function (dashed line) of annual precipitation sums in Belgrade.

Precipitation regime, i.e., monthly distribution of precipitation in Belgrade, using the LibreOffice spreadsheets, is presented in Fig. 2. The continental regime is characterized with a maximum of precipitation in June.

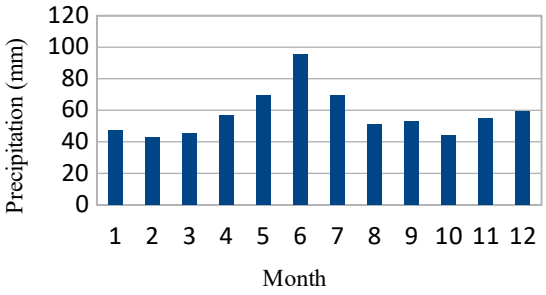


Figure 2: Precipitation regime in Belgrade during the period 1961-2018.

B. Climatology

Climatology describes the complex interrelations between the atmosphere, land, oceans, ice and the biosphere. It is defined as weather conditions averaged over a period of time. Temperature and precipitation are the main factors determining regional climate.

Students run codes in GFortran, and use Xmgr for visualization of obtained results. Example of the annual mean temperature along with a trend for Belgrade during the period 1961-2018 is shown in Fig. 3. Applying the Mann-Kendall test, the significant increase of the annual mean air temperature in Belgrade is found [1].

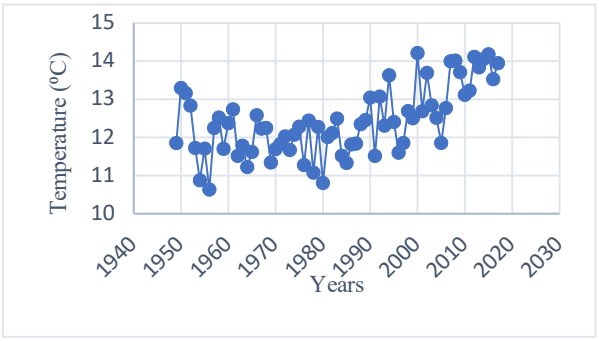


Figure 3: Annual mean temperature in Belgrade with linear regression trend during the period 1961-2018.

C. Modelling of the Atmosphere

While many different methods for numerical solution of partial differential equations are available, the approach that is followed in the course “Modelling of the Atmosphere” is a traditional, Eulerian, grid-point, finite-differencing method. This course is concerned almost exclusively with dynamics, i.e., the approximations of those processes that govern motion of the air [2]. One of examples that students solve at the end of the course is to derive an approximation for the one-dimensional linear advection equation using the Adams-Bashforth scheme for time differencing, and centered and backward finite difference quotient of the first-order of accuracy for space differencing. Then, the advection of the initial perturbation should be calculated and displayed for different time step.

Consider the one-dimensional linear advection equation

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0 \quad ,$$

where $u = u(x, t)$ is a function of two independent variables space and time, and c = const.

The Adams-Bashforth scheme is a time scheme,

$$\frac{U^{n+1} - U^n}{\Delta t} = \frac{3}{2} f^n - \frac{1}{2} f^{n-1} \quad , \quad \frac{dU}{dt} = f(U, t) \quad , \quad U = U(t) \quad .$$

Centered finite difference quotient over two grid intervals represents an approximation of the spatial derivative in the form

$$\left(\frac{\partial u}{\partial x}\right)_{j,t_0} \approx \frac{u(x_{j+1}, t_0) - u(x_{j-1}, t_0)}{2\Delta x}$$

Approximation of the spatial derivative by the quotient backward over one grid interval is

$$\left(\frac{\partial u}{\partial x}\right)_{j,t_0} \approx \frac{u(x_j, t_0) - u(x_{j-1}, t_0)}{\Delta x}$$

Initial perturbation in one point (top) and in three points (bottom) is advected without change in shape (Fig. 4). Centered finite difference quotient (CKKR) better preserved amplitude of initial perturbation than the backward quotient (UKKR). However, CKKR produced negative values.

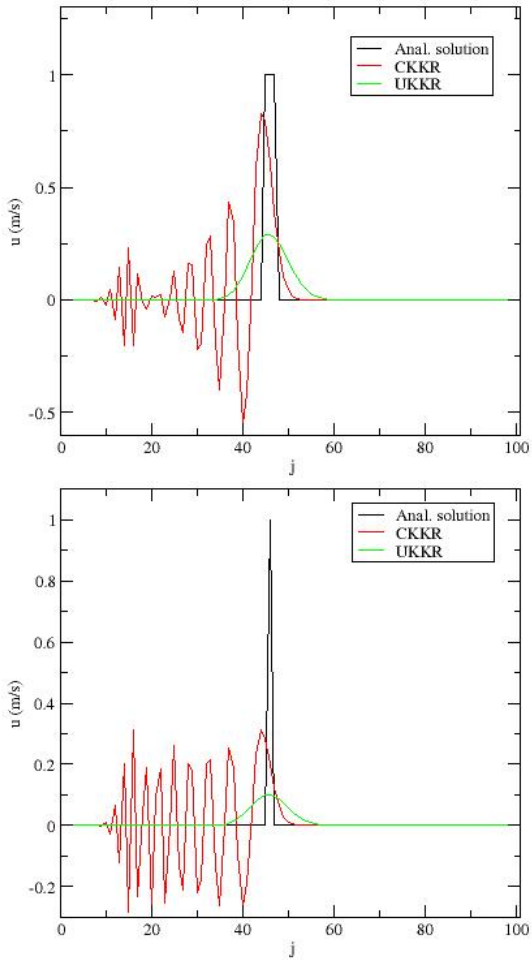


Figure 4: Analytical solution (black), centered finite difference quotient (CKKR, red) and backward quotient (UKKR, green) for initial perturbation in: one point (top) and three points (bottom).

IV. Free Software in Meteorological Research

The Institute of Meteorology pursues a wide range of research topics ranging from data analysis and numerical forecasting to regional climate research.

GrADS is extensively used free software in meteorological research for displaying meteorological fields. Daily values of air temperature, geopotential height, and vector wind are obtained from the gridded dataset of the National Center for Environmental Prediction/National Center for Atmospheric Research (NCEP/NCAR) Reanalysis Project (https://psl.noaa.gov/data/composites/day/). All gridded values are saved as the Network Common Data Form (NetCDF) and imported in GrADS for a mapping. An extraordinary heat wave occurred in Serbia from July 14 to July 24 in 2007 [3]. Record values of the maximum temperatures were observed over almost the whole territory of Serbia and in Smederevska Palanka, a temperature of 44.9 ºC was registered, which was the absolute maximum value ever recorded. Temperature anomaly observed on 24 July 2007 based on the period 1981-2010 is shown in Fig. 5. The atmospheric circulation at 500 hPa (not shown) resulted in the horizontal advection of warm air masses from northern Africa across central and eastern Mediterranean towards the Balkans.

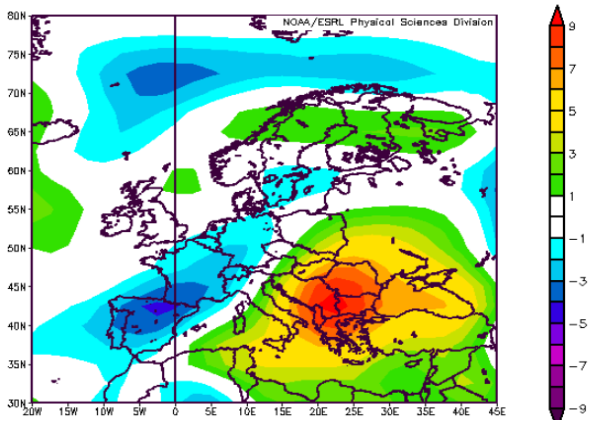


Figure 5: Temperature anomaly of 24 July 2007 at 1000 hPa over Europe, based on the 1981–2010 reference period.

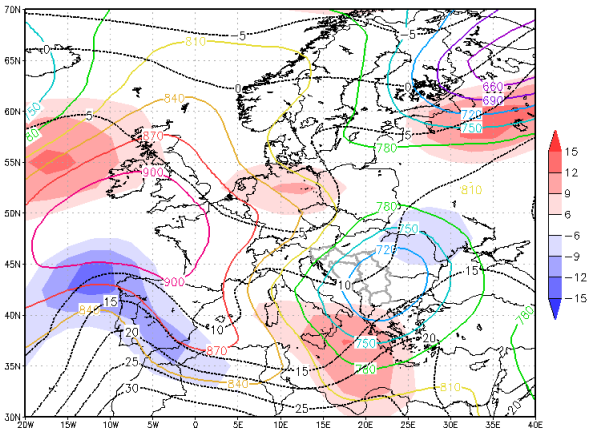


Figure 6: Synoptic situation on 14 May 2014: a geopotential height (gpm, continuous line), temperature (ºC, dashed line) and horizontal wind speed (shaded) at 925 hPa.

Extreme precipitation was recorded in May of 2014 in western Serbia [4]. On 15 May 2014, the daily rainfall broke previous historical records in Belgrade (109.8 mm), Valjevo (108.2 mm) and Loznica (110 mm) as a result of the cyclonic activity. Precipitation exceeded 200 mm in 72 h, producing the most catastrophic floods in the recent history of Serbia. Synoptic situations were analyzed in [4] and presented in Fig. 6.

Interdisciplinary research such as the study of long-range transport is a good example for application of GrADS [5]. Although the phenomenon of long-range dust transport is generally present, North Africa, more precisely the Sahara, is the largest source of mineral dust, with about 0.8 x 10^9 tons per year. In the Sahara region, air currents lead to dust build-ups, which are then transported to the Mediterranean [6] [7] and further north to Arctic regions or west to the Atlantic. A composite geopotential height and wind flow map for synoptic situations that favored a long-range dust transport from Sahara towards the Balkans is presented in Fig. 7. As a consequence, colored rain can be observed in Belgrade and Serbia.

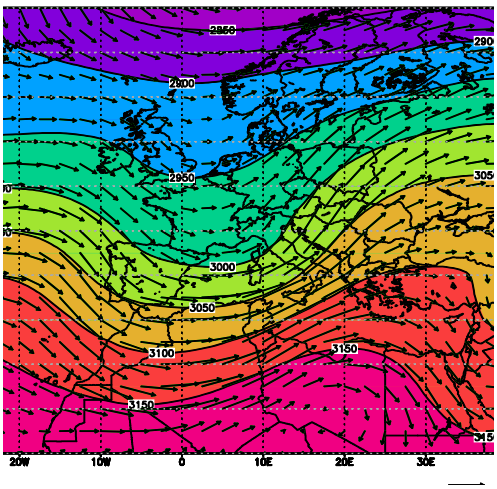


Figure 7: A composite geopotential height (m) and wind flow (m/s) map for days of 18 events analysed from March 2012 to December 2013.

V. Conclusion

Meteorology is a geophysical science and a good example of a discipline that relays on complex visualization of scientific data. Consequently, free open source software might be good candidate for both students and researchers in the field of data calculation and visualization. There are many free software programs that are applied for calculation and visualization of meteorological data. Xmgr is a 2D plotting tool for X-terminals used in education. GrADS is extensively applying in visualization of meteorological fields in education and research. Students in Meteorology use free software at several courses at the Faculty of Physics in Belgrade. However, there is a space for incorporation of other free software, for example Python, which will be done in the near future.

References

[1] M. Unkašević, D. Vujović, I. Tošić, *Zbirka rešenih zadataka u Klimatologiji i Primenjenoj meteorologiji*. Univerzitet u Beogradu-Fizički fakultet, Savezni hidrometeorološki zavod, 238 pp, 2002

[2] B. M. Gavrilov, I. Tošić, M. Rančić, *Numerical Methods in Meteorology: Solved problems*. Lap Lambert, 172 pp, 2014

[3] M. Unkašević, I. Tošić, "The maximum temperatures and heat waves in Serbia during the summer of 2007," *Climatic Change*, vol. 108, pp. 207-223, 2011

[4] I. Tošić, M. Unkašević, S. Putniković, "Extreme daily precipitation: the case of Serbia in 2014," *Theoretical and Applied Climatology*, vol. 128, pp. 785–794, 2017

[5] D. S. Đorđević, I. Tošić, S. Sakan, S. Petrović, J. Đuričić-Milanković, C. D. Finger, P. Dagsson-Waldhauserova, "Can volcanic dust suspended from surface soil and deserts of Iceland be transferred to Central Balkan similarly to African dust (Sahara)?" *Frontiers in Earth Sciences*, vol. 7:142, 2019

[6] Z. Vukmirović, M. Unkašević, L. Lazić, I. Tošić, S. Rajšić, M. Tasić, "An alysis of the Saharan dust regional transport," *Meteorology and Atmospheric Physics*, vol. 85, pp. 265–273, 2004

[7] S. Rodríguez, X. Querol, A. Alastuey, M.-M. Viana, E. Mantilla, "Events affecting levels and seasonal evolution of airborne particulate matter concentrations in the western Mediterranean," *Environment and Science Technology,* vol. 37, pp. 216– 222, 2003

Application of Mathematical Software in Teaching Numerical Mathematics for Students of Electrical Engineering

Nataša Ćirović
University of Belgrade, School of Electrical Engineering, Belgrade, Serbia
natasa@etf.bg.ac.rs

Abstract: The successful education of numerical mathematics for engineering students requires usage of software tools for various applications. In this paper we present our experiences in using both open and proprietary software tools in teaching numerical mathematics for bachelor students of electrical engineering at the School of Electrical Engineering, University of Belgrade in different applications. We compare the mathematical software used for implementation of numerical solutions in class and present an illustrative example showing the application of mathematical software in teaching process.

Keywords: numerical mathematics; higher education; open software; proprietary software.

I. Introduction

With the increase of computer power in the past few decades, the nature of introductory courses in numerical mathematics for students of engineering is changing from purely theoretical to application driven implementation with ready-to-use tools. Software intended for numerical computation is continuously being developed and improved. The usage of such tools in the classroom allows the shift in focus from tedious repeated computation by-hand to getting insight in the limitations of various numerical methods.

During bachelor studies at the School of Electrical Engineering, University of Belgrade, numerical methods are thought within courses Numerical analysis and discrete mathematics for Computer Engineering and Information Theory and Software Engineering students, and Numerical mathematics for Electrical Engineering second year students. Part of the students' grade is realized by the implementation of programming assignments in mathematical software or programming language of choice. Various tools are used during the teaching process by the lecturers and implementation of programming assignments by the students. In this paper we present and compare the software tools, both propriety and open, used for the needs of numerical mathematics courses for various applications.

II. Software applications

Software tools are used for various applications in teaching numerical mathematics courses for students of electrical engineering. These include online educational platforms as administrative support to the educational process, text processing tools and mathematical software tools. Also, tools for spreadsheet manipulation have shown useful in the educational process.

A. Educational platforms

As support in administrating the educational process online educational platforms are used, which facilitate access to additional learning material such as examples of solved problems in various mathematical software used in the classroom. Also, selection, uploading and grading of programming assignments implemented by students is conducted through these platforms.

The Computer Center of the University of Belgrade hosts and administrates the eLearning platform based on **Moodle** [1]. Moodle is a learning management system and it is a free and open source software package designed to provide educators, administrators, and learners with a single robust, secure and integrated system to create personalized learning environments [2]. The capabilities of Moodle greatly depend on the administrative settings done by the host.

During the spring semester 2020 the COVID-19 pandemic forced the learning process from the classroom to the online environment. School of Electrical Engineering, University of Belgrade provided the **Microsoft Teams** (MST) platform for online teaching. Even though MST was not primarily developed as learning platform, it has all the necessary features such as assignment posting and grading. Live streaming is very easily implemented through MST, together with live recording, which was essential in the situation when no on-site classes were possible. The feature of live streaming is not directly available in Moodle, but it could possibly be realized through additional plug-ins.

B. Text processing tools

Text processing software tools are utilized both by students and teachers in various applications.

LaTeX [3] is a free and open source document preparation system. It uses markup-level directions to define the formatting. Typing and formatting mathematical equations is easily implemented through LaTeX editors. For offline installation we have good experiences with MiKTeX [4] with TeXstudio editor [5] on Microsoft Windows operating systems. Overleaf [6], an online collaborative LaTeX editor, allows fast editing without installation, which can be very useful feature for students. All the above mentioned software tools related to LaTeX are open software.

LibreOffice Writer is part of LibreOffice, one of the leading open source office software suite. LibreOffice was forked from OpenOffice.org in 2010, that contains

DOI: 10.5281/zenodo.4081705

programs for word processing, creating and editing of spreadsheets, presentations, diagrams and drawings, databases, and composing mathematical formulas.

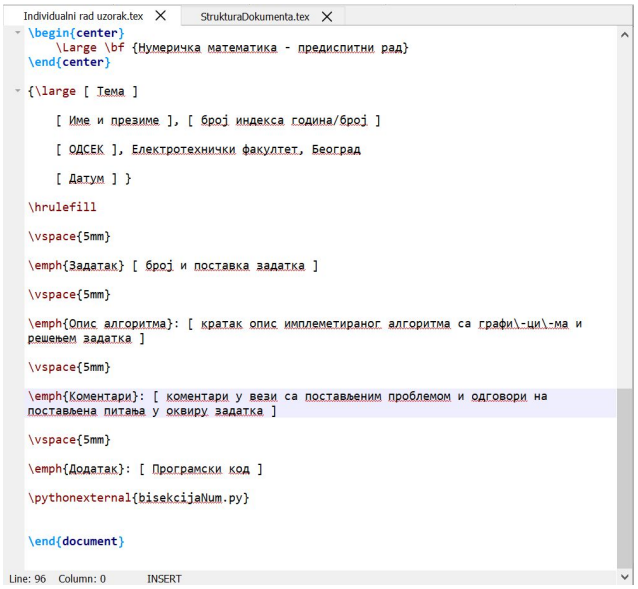


Figure 1: Part of LaTeX template code for programming assignment report with automatic importing of the program code

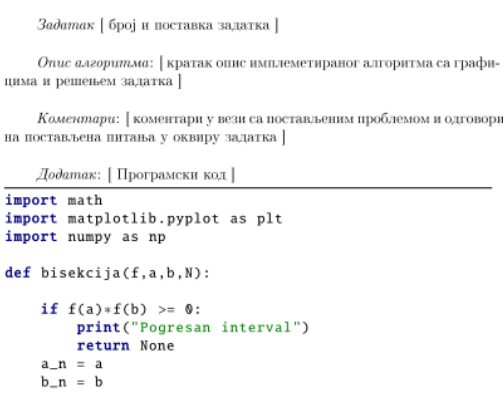


Figure 2: Part of LaTeX template pdf output for programming assignment report with automatic importing of the program code

As its native file format LibreOffice uses the Open Document Format for Office Applications (ODF), and it also supports the file formats of most other major office suites, including Microsoft Office, through a variety of import and export filters [7].

Microsoft Word is part of Microsoft Office, a propriety office package with wide range of applications including word processing, spreadsheet, presentation, but also e-mail client and MS Teams app. Since Office 2013, Microsoft has promoted Office365 which allows the use of the software and other services on a subscription business model. Since 2017 School of Electrical Engineering University of Belgrade implemented Office365 cloud platform for students with a large number of online services [8].

The official literature for the numerical mathematics courses is the electronic textbook by the author of this paper and it is freely available through the School website [9]. The book is prepared in LaTeX with TeXstudio editor and the graphics are prepared in GeoGebra [10], an interactive mathematics software suit for learning and teaching mathematics.

Text processing applications are utilized by students for creation of programming assignment reports. Two templates are available for reporting, in LibreOffice Writer/Microsoft Word and in LaTeX. Figure 1 shows part of the LaTeX code for reporting programming assignment template using the package *listings* for source code printing, and Figure 2 shows the output of this code.

C. Mathematical software tools

Programming assignments are part of the grade and students need to solve two problems that require programming solution in mathematical software tools or programming language of their own choice. During the teaching process the mathematical software is also used to provide deeper insight in various numerical methods.

Spreadsheet software is used for automatic calculations. Even though spreadsheet software allows writing programs (macros), in classroom we implement numerical methods by using the worksheet formulas. This approach has shown especially useful in the teaching process of numerical mathematics, allowing students to go through the steps of the numerical method in detail while avoiding tiresome calculation by hand. It is suitable for solving problems of lower dimensions which are typically considered within the undergraduate numerical mathematics courses [11]. For this purpose LibreOffice Calc or Microsoft Excel can be used with equal success.

Besides the spreadsheet applications, mathematical software tools developed particularly for numerical mathematical computations is also used in courses. A number of software packages have been developed to produce numeric and symbolic mathematical computations. The predominant propriety packages in the academic environment are **Maple** [12], **Mathematica** [13] and **MATLAB** [14], [15]. In parallel, a number of free and open source packages were developed during the past decades, with more or less success.

Maple is a proprietary software created and developed by Maplesoft company, primarily for numeric and symbolic mathematical computations. Within the project supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia in 2019 we were able to obtain the license for Maple, which enabled us to introduce Maple in the teaching process of numerical mathematics. Maple includes the Student package, a collection of routines designed to assist the teaching and learning of standard undergraduate mathematics. It includes a sub-package specifically designed for numerical analysis topics and it contains computation commands, visualization commands and interactive routines, allowing the students to gain insight on how a specific numerical method operates [12].

GNU Octave is one of the most successful alternatives to MATLAB. It is compatible with many MATLAB scripts and has available online version, which

is an advantage for students with modest technical capacities [16].

SageMath is an open source mathematical software built on top of many existing open source packages with a unified Python interface. SageMath can be used on a local computer, in a local network and online. The online version is called CoCalc and it can operate as a collaborative and educational platform as well [17], [18].

Python is a high-level general purpose programming language [19]. It is highly extensible, fast-growing environment, allowing creation of large number of libraries that support numerical and symbolic computation. Some of the most valuable libraries that support numerical methods include:

- *NumPy*, defines a multi-dimensional array object and associated fast math functions that operate on it [20];
- *SymPy*, a Python library for symbolic mathematics, aiming to become a full-featured computer algebra system [21];
- *SciPy*, a Python library build on NumPy, used for solving scientific and mathematical problems. It allows manipulation and visualization of data with a wide range of high-level commands. SciPy includes modules for linear algebra, optimization, integration, special functions, ODE solvers [22];
- *matplotlib*, a comprehensive library for creating static, animated, and interactive visualizations in Python [23];

SciPy with matplotlib library allows creation of interactive routines and animations of various numerical methods. There is a large number of routines available online that are created with the aim to improve the learning process and to provide visualization of various numerical methods. Also, using the above mentioned Python libraries, it is not hard to create such routines, even for inexperienced programmers.

Other general purpose programming languages can also be successful in programming numerical methods, such as **C**, **C++**, **C#**, **Java**.

In our past experience with programming assignments we found that students tend to use the software that they are already familiar with. Thus, students of Computer Engineering and Software Engineering mostly implement their assignments in Python or C++ while students of Electrical Engineering mostly use MATLAB/GNU Octave. Since both MATLAB and GNU Octave use file with m extension and having in mind that GNU Octave is compatible with many MATLAB scripts, it is often hard to tell in which package the code was originally written.

III. Classroom example

We present an example with the aim to illustrate the application of mathematical software tools in teaching numerical mathematics methods for students of electrical engineering. Here we show how we can implement the solution of nonlinear equation with the Newton-Raphson iterative method using Maple Student package, Python and LibreOffice Calc. As an illustrative example we solve the equation $(x-1)e^{3x}=0$. Knowing the correct solution, $x=1$, allows us to compare the exact error in each iteration.

A. Maple Student package

For the implementation of the Newton-Raphson method we use the command *Newton* within the Student [NumericalAnalysis] subpackage. The *output* option displays the sequence of iteration values, *plot* option provides geometric interpretation of the method and *animation* option enables animated display of the iterative process. Figure 3 shows the Maple code and graphical interpretation for our example function.

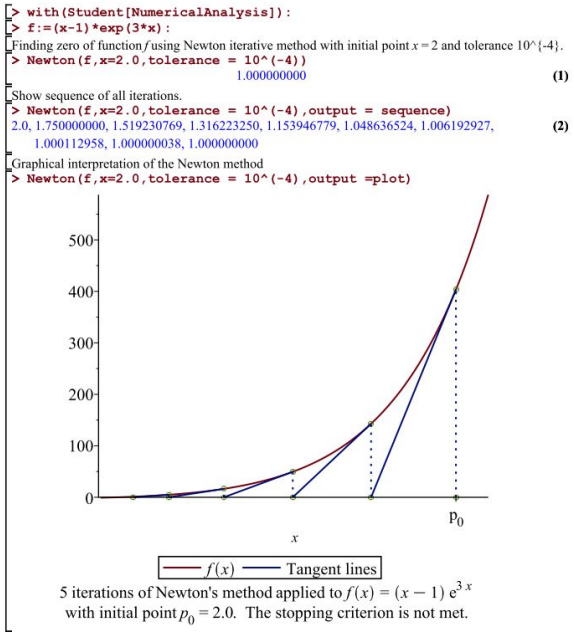


Figure 3: Maple code and output for implementation of Newton-Raphson method with Student package.

B. Python with libraries

SciPy library contains a large number of routines for implementation of numerical methods. For the solution of our example problem $(x-1)e^{3x}=0$, we use the scipy.optimize.newton routine. Additionally, we created the routine that shows the graphical interpretation of Newton-Raphson method using the SciPy and matplotib libraries. Figure 4 shows part of the python code with iteration values and graphical interpretation of Newton-Raphson method implemented on our example problem.

C. LibreOffice Calc

When introducing a numerical method in class we often use spreadsheet software and worksheet formulas. Here we show the solution of our example problem $(x-1)e^{3x}=0$ in LibreOffice Calc. Figure 5 shows the table obtained during implementation of the Newton-Raphson method for solving our example problem. We compare two adjacent iteration values and the actual error.

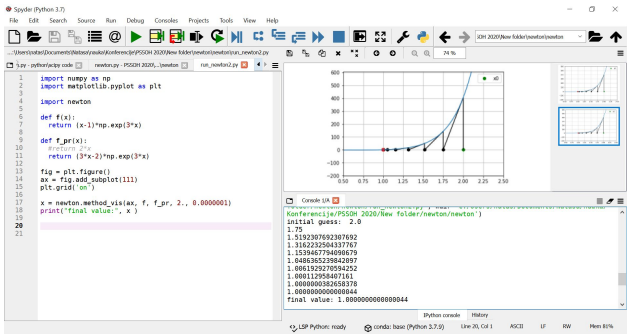
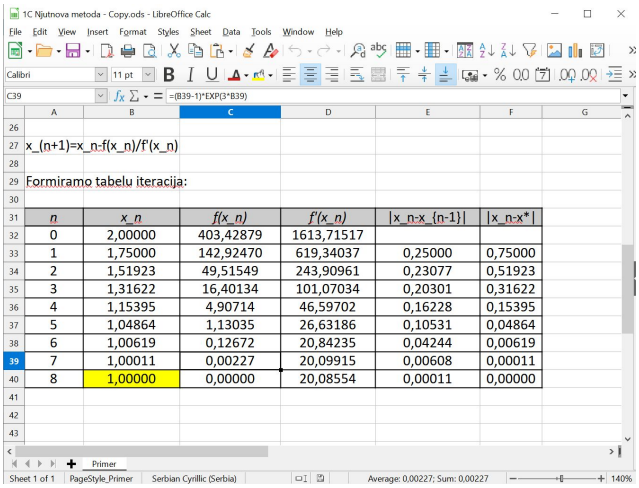
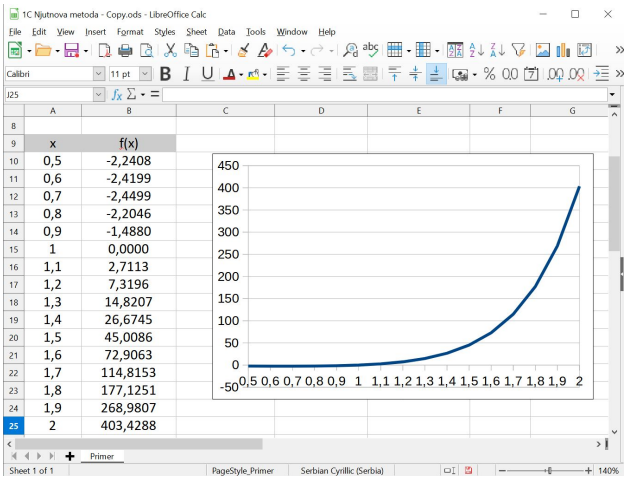


Figure 4: Part of the python code with iteration values and graphical interpretation of Newton-Raphson method for solving $(x-1)e^{3x}=0$.



n	x_n	f(x_n)	f'(x_n)	|x_n-x_{n-1}|	|x_n-x*|
0	2,00000	403,42879	1613,71517		
1	1,75000	142,92470	619,34037	0,25000	0,75000
2	1,51923	49,51549	243,90961	0,23077	0,51923
3	1,31622	16,40134	101,07034	0,20301	0,31622
4	1,15395	4,90714	46,59702	0,16228	0,15395
5	1,04864	1,13035	26,63186	0,10531	0,04864
6	1,00619	0,12672	20,84235	0,04244	0,00619
7	1,00011	0,00227	20,09915	0,00608	0,00011
8	1,00000	0,00000	20,08554	0,00011	0,00000

Figure 5: Table obtained during implementation of the Newton-Raphson method in LibreOffice Calc



x	f(x)
0,5	-2,2408
0,6	-2,4199
0,7	-2,4499
0,8	-2,2046
0,9	-1,4880
1	0,0000
1,1	2,7113
1,2	7,3196
1,3	14,8207
1,4	26,6745
1,5	45,0086
1,6	72,9063
1,7	114,8153
1,8	177,1251
1,9	268,9807
2	403,4288

Figure 6: Graphic of the function $f(x)=(x-1)e^{3x}$ *created in LibreOffice Calc using linear chart.*

Graphing of functions in LibreOffice Calc can be realized by using the chart of linear type on function values, which basically creates linear interpolation of data. This approach does not handle function discontinuities, so one should be especially careful when dealing with function graphing in this way. Figure 6 shows the result of graphing our example function $f(x)=(x-1)e^{3x}$ with linear chart in LibreOffice Calc.

IV. Conclusion

Software tools are used in bachelor numerical mathematics courses for 3 main applications: educational platforms, text processing and mathematical software tools. While educational platforms and text processing tools are used irrelevant of topic, mathematical software is crucial in numerical mathematics courses for understanding and implementing numerical methods. Spreadsheet software tools have shown very useful when a numerical method is introduces in class. Both open source LibreOffice Calc and proprietary Microsoft Excel can be successfully used. Numerical software allows more detailed analysis of numerical methods. For this purpose the Maple Student package, a proprietary software, and Python with SciPy and mathplotlib are used. In comparison to Maple, Python may require creation of new routines, but many are already freely available. Also, such routines can be easily created for any numerical method in Python, while Maple Student package covers only the most common numerical methods. Other relevant software can be used with the same purpose and we plan to include SageMath in teaching process. For implementation of programming assignments students tend to use tools that they are already familiar with and that are available.

Acknowledgements

This work was partially supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia through project NUMMMO [RVO-41/2019].

References

[1] http://elearning.rcub.bg.ac.rs [Accessed Aug. 25, 2020]

[2] E. Costello, "Opening up to open source: looking at how Moodle wasadopted in higher education", *Open Learning: The Journal of Open, Distance and e-Learning*, vol. 28, no. 3, pp. 187-200, 2013, doi: 10.1080/02680513.2013.856289

[3] https://www.latex-project.org [Accessed Aug. 25, 2020]

[4] https://miktex.org [Accessed Aug. 25, 2020]

[5] https://texstudio.org [Accessed Aug. 25, 2020]

[6] https://www.overleaf.com [Accessed Aug. 25, 2020]

[7] https://www.libreoffice.org [Accessed Aug. 25, 2020]

[8] https://www.microsoft.com/en-us/microsoft-365 [Accessed Aug. 25, 2020]

[9] N. Ćirović, "Numerička matematika", School of Electrical Engineering, University of Belgrade, Belgrade, 2018, https://www.etf.bg.ac.rs/uploads/files/udzbenici/Natasa%20Cirovic%20Numericka%20matematika%20CIP.pdf

[10] https://www.geogebra.org [Accessed Aug. 25, 2020]

[11] H. Djamila, "Excel spreadsheet in teaching numerical methods", *IOP Conf. Series: Journal of Physics: Conf. Series*, vol. 890, 012093, 2017, doi:10.1088/1742-6596/890/1/012093

[12] https://www.maplesoft.com/ [Accessed Sep. 1, 2020]

[13] https://www.wolfram.com/mathematica/ [Accessed Sep. 1, 2020]

[14] https://www.mathworks.com/ [Accessed Sep. 1, 2020]

[15] R.L. Burden and J.D. Faires, "Numerical Analysis", 9th Edition, Brooks/Cole, Cengage Learning, 2011.

[16] https://www.gnu.org/software/octave/ [Accessed Sep. 1, 2020]

[17] https://www.sagemath.org/ [Accessed Sep. 1, 2020]

[18] P. Szabó and J. Galanda, "Sage math for education and research," 15th Int. Conf. on Emerging eLearning Technologies and Applications (ICETA), Stary Smokovec, 2017, pp. 1-4, doi: 10.1109/ICETA.2017.8102535

[19] https://www.python.org/ [Accessed Sep. 1, 2020]

[20] https://numpy.org/ [Accessed Sep. 1, 2020]

[21] https://www.sympy.org [Accessed Sep. 1, 2020]

[22] https://scipy.org/ [Accessed Sep. 1, 2020]

[23] https://matplotlib.org/ [Accessed Sep. 1, 2020]

REPOPSI: The open repository of psychological instruments in Serbian

Aleksandra Lazić[1,3], Ljiljana B. Lazarević[2,3], Danka Purić[1,3], Iris Žeželj[1,3]
[1]*Department of psychology, University of Belgrade, Belgrade, Serbia*
[2]*Institute of psychology, University of Belgrade, Belgrade, Serbia*
[3]*Laboratory for research of individual differences, University of Belgrade, Belgrade, Serbia*
aleksandra.lazic@f.bg.ac.rs, ljiljana.lazarevic@f.bg.ac.rs, dpuric@f.bg.ac.rs, izezelj@f.bg.ac.rs

Abstract: The Repository of Psychological Instruments in Serbian (REPOPSI; https://osf.io/5zb8p/), run by the Laboratory for Research of Individual Differences at the University of Belgrade and hosted on the Open Science Framework, is an open-access repository of psychological instruments. REPOPSI is a collection of over 130 instruments (e.g., scales, tests) commonly used in social and behavioral science research. Documented are Serbian, English and multilingual instruments, which can be used free of charge for non-commercial purposes (e.g., academic research or education). We argue that REPOPSI enables scientists to increase the efficiency of their research and the visibility of their output. We analyze REPOPSI's commitment to ensure that its (meta)data is findable, accessible, interoperable, and reusable (the FAIR Data Principles) and its trustworthiness with respect to transparency, responsibility, user focus, sustainability, and technology (the TRUST principles). Finally, we describe how the FAIR and the TRUST principles will support the process of continuous improvement of REPOPSI.

Keywords: digital repository; FAIR principles; TRUST principles; research data; psychological measures.

I. Introduction

To assess individual differences, social and behavioral science researchers either adapt existing instruments or develop new ones. The majority of the standardized instruments have been originally created in English and need to be translated and adapted to other cultural contexts. Although there are notable exceptions where researchers digitally keep a record of all of the translations/adaptations of a specific instrument (e.g., HEXACO-PI-R [1] or IPIP Items [2]), until recently, there was no centralized repository of such translations/adaptations in Serbian. It was not uncommon for several research groups to unknowingly work on the same translations/adaptations either simultaneously or at different points in time. This decreases the efficiency of the research process by causing redundant expenses and overlapping efforts. At the same time, local researchers have developed original instruments for the assessment of novel constructs (e.g., new personality traits, [3]) or for the assessment of constructs underrepresented in the research toolkit (e.g., superstition, [4]). However, the opportunities to ensure the visibility and reusability of such instruments were limited.

The Repository of Psychological Instruments in Serbian (REPOPSI; https://osf.io/5zb8p/) is an open-access digital repository for psychological instruments (e.g., items, scales, questionnaires, tests). Most of the instruments measure characteristics of individuals (e.g., personality, attitudes, beliefs, emotions, values, intentions) and are developed to be used in social and behavioral science research. REPOPSI is designed to document primarily instruments in Serbian as well as instruments translated into Serbian and/or adapted for the Serbian population. Also documented are versions of these instruments in English as well as in other languages. Because of this, not only Serbian but also international, and especially regional, researchers can benefit from REPOPSI, with the Serbian translations often being a useful starting point for further adaptations for the Bosnian-Croatian-Montenegrin-Serbian (BCMS) language. The instruments are used free of charge for non-commercial purposes (e.g., academic research or education). More precisely, all Repository content is under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International license [5].

REPOPSI was established and is maintained by researchers and students at the Laboratory for Research of Individual Differences (LIRA; https://lira.f.bg.ac.rs/en/) at the University of Belgrade Faculty of Philosophy (UBFZF; http://www.f.bg.ac.rs/en2). REPOPSI is hosted on the Open Science Framework (OSF; https://osf.io/) platform. OSF – the flagship product of the nonprofit Center for Open Science (COS; https://www.cos.io/) – is "a free, open source web application that connects and supports the research workflow" [6]. Researchers at LIRA typically use OSF for managing their projects and collaborations and rely on OSF's secure cloud environment for data storage (e.g., [7], [8], [9], [10]).

II. REPOPSI structure, contribution, and maintenance

REPOPSI is a "*project*" on OSF, which is the largest unit of categorization supported. The project is titled "Repository of psychological instruments in Serbian [Repozitorijum psiholoških instrumenata na srpskom jeziku] (REPOPSI)". It has been assigned a unique, persistent URL (https://osf.io/5zb8p/) and a Digital Object Identifier (https://doi.org/10.17605/OSF.IO/5ZB8P). The customized tags in English are automatically indexed by the search for public content on OSF and the project may be displayed in search engine results. The Repository has its own built-in, publicly available analytics (https://osf.io/5zb8p/analytics/) that tracks unique visits

and popular pages over time. The project's Wiki page – available in both English (https://osf.io/5zb8p/wiki/home/) and Serbian (https://osf.io/5zb8p/wiki/srpski/) – provides the following information to users: REPOPSI's scope and purpose, instructions for use and for contributing, terms of use, and a disclaimer.

REPOPSI's OSF project was made publicly available on March 6, 2020. It currently contains 140 instrument records (last updated: August 29, 2020): 101 instruments are open-access and available in the Repository; 3 instruments are open-access but are available elsewhere online; and 36 instruments are available upon request from the contact author. The latter could not be made open because of, for example, ethical constraints or the instrument administration protocol. All records are listed in the Inventory (https://osf.io/mxrc2/), which is available in two file formats (CSV and XLSX). The Inventory is directly searchable by users within the OSF application via a search box. Additional materials for the Repository (e.g., promotional posters) are made available in a public GitHub repository (https://github.com/liralab-bgd/repopsi). Individuals who wish to contribute to REPOPSI are invited to fill in the *contribution form*, which is available upon request at the LIRA email address. Once the filled-in form is sent back to LIRA, it is reviewed as quickly as possible and the instrument is deposited in REPOPSI's OSF project.

The contribution form is available in various file formats, as needed (e.g., CSV, ODS, XLSX, GSHEET). It contains detailed instructions in English for the contributors, along with input formats and example inputs. The form contains both mandatory and optional fields divided into two parts.

The first part of the contribution form provides a brief overview of the instrument as well as the authorship and citation details. The mandatory fields are the title in English and Serbian, title abbreviation, terms of use, keywords in English, languages in which the instrument is to be deposited, followed by the citation, authors, and date of creation for the original and, if applicable, the translated/adapted instrument. The optional fields include specialized tags in English which differentiate between different versions of the instrument (e.g., short-form, self-report, version for service providers), publisher, funding sponsors and contractual arrangements, a maximum 300-word abstract in English, contact person, an external open-access link to the instrument, and additional notes in English (e.g., about the availability of additional translations). Information provided in this part of the contribution form is entered as metadata in the Inventory.

The second part of the contribution form provides information on the instrument itself, that is, on the original instrument and, if applicable, on one or more translations/adaptations of the same version of the instrument. Here, the only mandatory field contains the items of the instrument and, if applicable, their presentation order, while all other fields are optional. They include the introductory sentences and instructions that respondents receive as well as the response scale on which respondents answer the items (e.g., the number of response categories together with their numerical or verbal labels); this is followed by instructions for instrument administrators (which may be provided either in English or in the language of the instrument), including scoring instructions (e.g., whether an item or scale recoding is necessary, how items are combined to form the total score), and additional notes (e.g., about the translation procedure). Together with the form, contributors may also submit the scoring code as a file format of their choosing (e.g., R code file, SPSS syntax file).

REPOPSI's *maintenance team* handles the contributions. The team currently consists of eight members, who are all researchers, PhD students or graduate students at LIRA: the project administrator (admin), four associates, and three supervisors. The admin communicates with the contributors, organizes and coordinates team activities, and trains the associates. The associates have at their disposal a manual for reviewing, cleaning, and depositing the contributions. The manual is continually updated and revised; it details the procedure and provides screenshots as examples. The team collaborates using a private folder on Google Drive (https://www.google.com/intl/en/drive/), which doubles as back-up storage for the Repository contents.

Each instrument is deposited into its own public "*component*", that is, a sub-project in REPOPSI's top-level OSF project. The component holds all of the versions, translations, and adaptations of the same instrument. Each contribution is documented as a separate file, available in two open formats – the Open Document Format for Office Applications (ODT) and the Portable Document Format (PDF). The scoring code file is also added to the component. All files follow file naming conventions: "abbreviation_version_language(s)" and "abbreviation_version_syntax". A built-in version control maintains a record of all previous copies of a file. Each component and each file has a unique, persistent URL so that it can be cited or linked to individually. For example, the URL of the "Disintegration Scale - DELTA9" component is https://osf.io/s9ezy/, while the URL of the "delta9_20_srp_eng_ger.odt" document in this component is https://osf.io/p5shr/. The customized component name and contributor-generated tags in English are automatically indexed by search for public content on OSF and the component may be displayed in search engine results. The download count of files is publicly available in the "Files" page of the project (https://osf.io/5zb8p/files/).

III. The FAIR and TRUST guiding principles

Many in the Open Access community are familiar with the *FAIR Data Principles* [11], [12], which state that all research objects should be Findable, Accessible, Interoperable, and Reusable (i.e., FAIR) both for people and for machines. However, to make data FAIR and to keep it FAIR over time requires trustworthy digital repositories [13]. Recently, representatives from across the digital repository community have published the collaboratively-developed *TRUST Principles* that offer guidance for maintaining the trustworthiness of digital repositories, especially those responsible for the stewardship of research data [13], [14]. Transparency,

Responsibility, User focus, Sustainability, and Technology are the five TRUST Principles. Together, the TRUST Principles are complementary to the FAIR Data Principles and aim to facilitate discussion around implementing best practices in the area of digital preservation.

To allow for more efficient and effective use of data, REPOPSI has adopted the FAIR Data Principles and has implemented the TRUST Principles. Currently, both sets of principles are adopted/implemented in part rather than completely. In the future, the FAIR and TRUST principles will lead REPOPSI along the continuum towards its optimal state; in other words, we will strive to make the data as FAIR and the Repository as TRUSTworthy as possible.

Here, we will provide a brief self-assessment of the extent to which REPOPSI, in its current state (as described above), has adopted the FAIR Data Principles as well as of the extent to which it has implemented the TRUST Principles. In addition to being measures of performance, these principles serve as REPOPSI's internal objectives, which will also be presented later in the paper.

A. The FAIR Data Principles as a tool for self-assessment

The FAIR Data Principles, along with their identifiers, are available in the Scientific Data comment published in 2016 [13]. To evaluate REPOPSI's (meta)data, we used the 15 FAIRsFAIR (https://www.fairsfair.eu/) data assessment metrics [15]. The brackets contain the FAIR principle identifier.

The data is assigned a globally unique (F1) and a persistent identifier (F1), that is, a URL. Metadata includes some of the descriptive core elements (i.e., author, title, data identifier, publication date, and keywords) for every instrument, as well as the publisher (where applicable) and a summary (for some of the instruments) (F2) to support data findability. Furthermore, the main metadata document (i.e., the Inventory) specifies the identifier of the data (i.e., component URL) such that users can access the data through the metadata (F3). Metadata is registered or indexed in a searchable resource (e.g., the Inventory; OSF tags), that is, it is available in a machine-readable format (F4). Metadata specifies levels of instrument availability as well as its public-domain-equivalent license (A1). Although, loosely speaking, REPOPSI's metadata uses a formal, accessible, shared, and broadly applicable language, it is not represented using a formal knowledge representation language (such as RDF or OWL) (I1), nor does it incorporate additional terms from semantic resources (e.g., from ontologies, thesauri or taxonomies) (I1). Metadata links the instrument to its related resources (when the contributor provides such resources), including publication, an external repository, platform or site; although persistent identifiers are used when possible (e.g., DOI for publication), links between an instrument and its related entities are not expressed through relation types (such as the ones listed in, for example, the DataCite Metadata Schema, https://schema.datacite.org/) (I3). Metadata richly describes the contents of the instruments, using a number of accurate and relevant properties, such as language and instrument version; however, the mandatory and optional fields of the contribution form may influence metadata completeness (R1). Metadata includes license information under which data can be reused. However, a clear and accessible data usage license with the conditions under which data can be reused is provided only in the OSF project's Overview and Wiki pages; other metadata contains only the Creative Commons license code (i.e., CC BY-NC-SA 4.0) (R1.1). Metadata contains some useful information on the instrument provenance, such as instrument source, instrument creation date, maintenance team members involved in depositing data, data publication and versioning information, references to related works (e.g., scientific articles or associated instruments) (R1.2). Metadata follows a standard recommended by the target research community (e.g., the machine-readable Inventory) (R1.3). The instruments are available in file formats widely used by the target research community; for example, the PsychArchives repository [16] also deposits psychological tests as PDF files However, even though some of the files are very human-readable, they are not machine-readable (R1.3).

B. The TRUST Principles as a tool for self-assessment

A short description of each of the TRUST principles is reproduced in Table 1. To evaluate REPOPSI's organizational, managerial, stewardship, and technological capabilities, we used the guidelines provided in the 2020 Scientific Data comment by Lin and colleagues [13] (see also [17]). The brackets contain the TRUST principle identifier (Tr for Transparency, R for Responsibility, U for User focus, S for Sustainability, and Tec for Technology).

Table 1: The TRUST Principles for digital repositories (reproduced from [13], http://creativecommons.org/licenses/by/4.0/)

Principle	Guidance for repositories
*T*ransparency	To be transparent about specific repository services and data holdings that are verifiable by publicly accessible evidence.
*R*esponsibility	To be responsible for ensuring the authenticity and integrity of data holdings and for the reliability and persistence of its service.
*U*ser Focus	To ensure that the data management norms and expectations of target user communities are met.
*S*ustainability	To sustain services and preserve data holdings for the long-term.
*T*echnology	To provide infrastructure and capabilities to support secure, persistent, and reliable services.

REPOPSI's "data preservation", "data persistence", and "data integrity" capabilities are largely dependent on OSF (Tr; R; S). First, OSF stores files with long-term and robust preservation in mind (e.g., multiple locations, multiple services, data integrity checks, and data recovery) [18]. Second, others can access all public content via OSF's open API (documentation is available at https://developer.osf.io/). Third, COS established a

preservation fund to host a static version of OSF in the event that COS ceases to function. This fund is sufficient for 50+ years of read access hosting at present costs [19]. Fourth, OSF provides a high-level of security to ensure data integrity (e.g., two-factor account authentication, multi-regional storage buckets, three types of hashes for files, advanced database backup features) [18].

The mission statement and scope of REPOPSI are clearly stated in the OSF project's Wiki pages (Tr). Terms of use, both for the Repository and for the instruments it holds, are transparently declared (Tr). Evidence of the data documentation and curation services that REPOPSI offers is not entirely publicly accessible (Tr). Specifically, the work that the maintenance team puts into improving users' contributions is not visible until the data is ready to be deposited into the OSF component. REPOPSI adheres to the designated community's metadata and curation standards (R). The extent to which REPOPSI has adopted the FAIR Data Principles so far can assure the potential users that the data is likely to be useful, discoverable, and interoperable. REPOPSI provides various data services (e.g., it provides a searchable inventory with rich metadata); furthermore, users can benefit from different OSF services (such as version history or viewing, downloading, and sharing files) (R). REPOPSI demonstrates some responsibility for the stewardship of the instruments (R). While it provides long-term persistence as well as technical validation and documentation, both quality control and authenticity protection are rather low as they include only informal vetting of the contributors and the instrument translations/adaptations. REPOPSI relies on users to decide under what conditions it is appropriate to open their work and to seek approval from original authors if needed (R). Only the contact information provided in the contribution form is deposited in REPOPSI (R). When flagged by users, REPOPSI will remove instruments from public access, if needed (R). REPOPSI strives to enforce standards and norms for data practices of its target user community (i.e., social and behavioral science researchers and students). By using OSF and adopting FAIR Data Principles, REPOPSI enables its community to find, explore, and understand its contents with regard to potential (re)use (U). REPOPSI encourages its contributors to fully describe the instruments and its users to provide feedback on any issues with the data (U). Relevant data metrics (e.g., the download count) are available to users (U). REPOPSI's plans and practices to ensure long-term data preservation and use are still in the conceptualization phase (S). While no funding is needed at this point, REPOPSI will have to sufficiently plan for risk mitigation, disaster recovery, and succession. REPOPSI relies on OSF to provide the infrastructure necessary to support its secure, persistent, and reliable services (Tec). The mechanisms to mitigate cyber or physical security threats have not yet been considered (Tec).

IV. Applications of REPOPSI

A. Applications in collaborative research projects

The quality of large-scale collaborations or cross-cultural research mainly depends on the quality of the materials used in the studies, as it determines the veracity, reliability and validity of the scientific investigation [20]. Translating and preparing instruments for cross-cultural research is essential and a complex process [21]. The main goal is to achieve conceptual equivalence between translations of the instrument as it heavily influences the quality of the research [22]. When translating and/or adapting instruments to Serbian, researchers typically follow high-quality standards – that is, either the standard forward-backward translation process [23] or recommendations for translation developed by the Psychological Science Accelerator [24] – and this procedure is a requirement if research using the instrument is to be published in a peer-reviewed scientific journal. Therefore, one of the main advantages of REPOPSI is that researchers and students have access to a large number of already prepared questionnaires, scales, and inventories. This facilitates participation in large-scale collaborations (e.g., [24], [25]), especially when a fast response from researchers is required (e.g., in the times of health crises like the COVID-19 pandemic, [8], [9], [26]).

B. Applications in teaching and learning

Starting from the academic year 2019/2020, second-year psychology students at UBFZF are actively encouraged to make use of REPOPSI. Namely, within the Psychology of Individual Differences, Psychometrics 1, and Psychometrics 2 courses, groups of four to five students work together to create, and later validate, a psychological instrument assessing individual differences in personality, ability, attitudes, and so on. To create a novel instrument, students can benefit from an insight into existing instruments measuring the same or similar constructs. Moreover, the process of construct validation typically entails administering – apart from the instrument being validated – an already established instrument assessing the desired construct as well. For both of these purposes, students can search REPOPSI and download the needed instruments, whenever they are available. Apart from being able to look for specific instruments by their name, students can also search the Repository by keywords (tags), and thus potentially discover additional instruments they were previously unaware of. To our knowledge, all student groups searched the Repository at some point during the academic year, and many of them used the instruments downloaded from the Repository in their validation studies. Finally, students can also be contributors to REPOPSI. When an instrument needed for the validation study has not yet been translated into Serbian, students often translate the instrument themselves. When the quality of the translation is assessed as satisfactory by the course teachers, students are invited to fill in the contribution form and help the Repository grow in this way.

V. REPOPSI's plans for improvement and growth

As noted earlier, we will use the FAIR Data Principles and the TRUST principles as guidelines to improve REPOPSI. Here, we present four concrete ways in which we plan to do so.

First, we plan to make the instruments deposited into REPOPSI more machine-readable, which should help increase their *Reusability*. This can be achieved by copying the items, the introductory sentences, the instructions, and the response scale of the instrument into a CSV or an XML file[1].

Second, to increase the *Transparency* of REPOPSI's specific services, we plan to move the work of the maintenance team to a public repository on GitHub and to make the maintenance manual open-access. That way, the enhancements to the initial contributions (such as edits and additions to the metadata) would be more visible.

Third, we plan to make the contribution form open-access and to offer it in an online survey form that is easier for the users to fill in. That way, REPOPSI would demonstrate greater adherence to the principles of *Transparency*, *User Focus*, and *Technology*.

Fourth, we plan to increase *User Focus* by promoting REPOPSI to a wider audience. So far, undergraduate and graduate psychology students at the UBFZF as well as LIRA members have been invited to use REPOPSI and to contribute to it. REPOPSI is also visible on LIRA's webpage dedicated to open science [27] and is promoted on LIRA's Facebook (https://www.facebook.com/LIRAlab.bgd/) and Twitter (https://twitter.com/LIRAlab_bgd) accounts. In the future, we plan to promote REPOPSI on the UBFZF website, to create a Wikipedia (https://www.wikipedia.org/) entry for it, and to promote it on academic social networking sites such as ResearchGate (https://www.researchgate.net/) or LinkedIn (https://www.linkedin.com/). We also plan to spread the word about REPOPSI to international, and especially regional, associations. REPOPSI could be added to a global registry of research data repositories such as re3data (https://www.re3data.org/) as well. Such efforts would likely not only facilitate the discovery of the instruments deposited in the Repository but would also increase the number of contributions to it. We expect that REPOPSI would thus continue to grow over the following years.

VI. Conclusion

The open-access repositories of psychological instruments adapted for non-English speaking countries are few. Notable exceptions include the PsychArchives [16] and the ZIS [28] repositories, which store English, but also German, as well as multilingual measurement instruments. REPOPSI (https://osf.io/5zb8p/), run by LIRA and hosted on OSF, is the first open-access repository of psychological instruments in Serbian.

A 2017 report by the Organization for Economic Co-operation and Development (OECD) concluded that "Research data repositories are an essential part of the infrastructure for open science. They bring considerable economic, scientific, and social benefits" ([29], p. 58). The benefits of a repository of open original psychological instruments and their translations are manifold – and this applies to REPOPSI as well. It streamlines the research process by making it easier to collect, find, reuse, and preserve the translations. Researchers and students benefit from having open access not only to the instruments but also to their scoring and administration principles. Moreover, the Repository saves them the resources that would otherwise go into the often laborious process of translating and adapting measures. REPOPSI allows local researchers to contribute their own original instruments or translations – predominantly in BCMS but also in other languages. This offers them the opportunity to reach a wider, international audience.

Based on the presented self-assessment of the extent to which REPOPSI has adopted the FAIR Data Principles, we can safely conclude that the Repository's (meta)data fulfills at least the minimal criteria for FAIRness [30]: a) data is assigned a globally unique and a persistent identifier; b) there is sufficient amount of metadata to make the instruments findable but also understandable and reusable by users; and c) (meta)data has clear license information. Furthermore, based on the self-assessment of the extent to which REPOPSI has implemented the TRUST Principles, we believe that the Repository, in its current state, demonstrates satisfactory TRUSTworthiness, that is, adherence to the principles of Transparency, Responsibility, User focus, Sustainability, and Technology. Finally, we have concrete plans to improve the FAIRness of REPOPSI's data as well as its overall TRUSTworthiness. REPOPSI will monitor and identify its target community's changing needs so that it can continue to provide a reliable, responsive, and consistent service.

Acknowledgements

We are grateful to all the researchers and students who contributed their original instruments and translations to REPOPSI. We wish to thank the four maintenance team associates who have been helping to run REPOPSI: Marija Petrović, Ana Ivanov, Mia Medojević, and Jovan Ivanović.

References

[1] "HEXACO-PI-R Materials for Researchers," http://hexaco.org/hexaco-inventory [Accessed Sept. 20, 2020].

[2] "Translating IPIP Items into Other Languages," https://ipip.ori.org/newItemTranslations.htm [Accessed Sept. 20, 2020].

[3] G. Knezevic, D. Savic, V. Kutlesic, and G. Opacic, "Disintegration: A reconceptualization of psychosis proneness as a

1 Some online survey software tools most commonly used for social science research that allow the user to import entire question packages (such as Qualtrics, https://www.qualtrics.com/core-xm/survey-software/ or SoSci Survey, https://www.soscisurvey.de/) rely on different file formats to achieve that (e.g., TXT or XML). On the other hand, most of the survey software tools allow the user to paste several items at once (even from a non-machine-readable format) into a question editor. This lack of uniformity will make it challenging to choose the machine-readable file format which is the most appropriate for REPOPSI's target community.

personality trait separate from the Big Five," *Journal of Research in Personality*, vol. 70, pp. 187–201, Oct. 2017, https://doi.org/10.1016/j.jrp.2017.06.001.

[4] I. Zezelj, M. Pavlovic, M. Vladisavljevic, and B. Radivojevic, "Construction and behavioral validation of superstition scale," *Psihologija*, vol. 42, no. 2, pp. 141–158, 2009, https://doi.org/10.2298/PSI0902141Z.

[5] "Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)," Creative Commons, https://creativecommons.org/licenses/by-nc-sa/4.0/ [Accessed Sept. 20, 2020].

[6] S. Bowman *et al.*, "OSF," OSF, https://osf.io/4znzp/ [Accessed Sept. 1, 2020].

[7] G. Knežević, L. B. Lazarević, C. Montag, and K. Davis, "Relations between lexical and biological perspectives on personality: New evidence based on HEXACO and Affective Neuroscience Theory," *Journal of Personality Assessment*, vol. 102, no. 3, pp. 325–336, Apr. 2019, https://doi.org/10.1080/00223891.2018.1553782.

[8] P. Teovanovic, P. Lukic, Z. Zupan, A. Lazić, M. Ninković, and I. Zezelj, "Irrational beliefs differentially predict adherence to guidelines and pseudoscientific practices during the COVID-19 pandemic," PsyArXiv, May 20, 2020, https://doi.org/10.31234/osf.io/gefhn.

[9] L. B. Lazarevic, D. Purić, P. Teovanovic, G. Knezevic, P. Lukic, and Z. Zupan, "What drives us to be (ir)responsible for our health during the COVID-19 pandemic? The role of personality, thinking styles and conspiracy mentality," PsyArXiv, Jun. 22, 2020, https://doi.org/10.31234/osf.io/cgeuv.

[10] L. B. Lazarevic *et al.*, "Many Labs 5: Registered replication of LoBue and DeLoache (2008)," *Advances in Methods and Practices in Psychological Science*, 2020, https://doi.org/10.1177/2515245920953350.

[11] "Guiding principles for findable, accessible, interoperable and re-usable data publishing version b1.0," FORCE11, https://www.force11.org/fairprinciples.

[12] M. D. Wilkinson *et al.*, "The FAIR Guiding Principles for scientific data management and stewardship," *Scientific Data*, vol. 3, no. 1, Mar. 2016, https://doi.org/10.1038/sdata.2016.18.

[13] D. Lin *et al.*, "The TRUST Principles for digital repositories," *Scientific Data*, vol. 7, no. 1, May 2020, https://doi.org/10.1038/s41597-020-0486-7.

[14] "The TRUST principles: An RDA community effort," RDA, https://www.rd-alliance.org/rda-community-effort-trust-principles-digital-repositories [Accessed Sept. 1, 2020].

[15] A. Devaraju *et al.*, "FAIRSFAIR data object assessment metrics," Zenodo, Jul. 10, 2020, https://doi.org/10.5281/zenodo.3934401.

[16] PsychArchives, https://www.psycharchives.org/ [Accessed Sept. 20, 2020].

[17] "The TRUST principles for trustworthy data repositories – An update," RDA, https://www.rd-alliance.org/trust-principles-trustworthy-data-repositories-%E2%80%93-update [Accessed Sept. 1, 2020].

[18] B. A. Nosek *et al.*, "NSF 19-501 AccelNet proposal: Community of Open Scholarship Grassroots Networks (COSGN)", MetaArXiv, Jan. 30, 2020, https://doi.org/10.31222/osf.io/d7mwk.

[19] S. Bowman, "FAQs," OSF Guides, https://help.osf.io/hc/en-us/articles/360019737894-FAQs [Accessed Sept. 1, 2020].

[20] H. Moshontz *et al.*, "The Psychological Science Accelerator: Advancing psychology through a distributed collaborative network," *Advances in Methods and Practices in Psychological Science*, vol. 1, no. 4, pp. 501–515, Oct. 2018, https://doi.org/10.1177/2515245918797607.

[21] P. S. Jones, J. W. Lee, L. R. Phillips, X. E. Zhang, and K. B. Jaceldo, "An adaptation of Brislin's translation model for cross-cultural research," *Nursing Research*, vol. 50, no. 5, pp. 300–304, Sep. 2001, https://doi.org/10.1097/00006199-200109000-00008.

[22] J. Choi, K. E. Kushner, J. Mill, and D. W. L. Lai, "Understanding the language, the culture, and the experience: Translation in cross-cultural research," *International Journal of Qualitative Methods*, vol. 11, no. 5, pp. 652–665, Dec. 2012, https://doi.org/10.1177/160940691201100508.

[23] R. W. Brislin, "Back-translation for cross-cultural research," *Journal of Cross-Cultural Psychology*, vol. 1, no. 3, pp. 185–216, Sep. 1970, https://doi.org/10.1177/135910457000100301.

[24] C.-P. Hu *et al.*, "Data from the Human Penguin Project, a cross-national dataset testing social thermoregulation principles," *Scientific Data*, vol. 6, no. 1, Apr. 2019, https://doi.org/10.1038/s41597-019-0029-2.

[25] J. H. Zickfeld *et al.*, "Kama muta: Conceptualizing and measuring the experience often labelled being moved across 19 nations and 15 languages.," *Emotion*, vol. 19, no. 3, pp. 402–424, Apr. 2019, https://doi.org/10.1037/emo0000450.

[26] I. M. Veer *et al.*, "Psycho-social factors associated with mental resilience in the Corona lockdown," PsyArXiv, Apr. 22, 2020, https://doi.org/10.31234/osf.io/4z62t.

[27] "Open science and cross-cultural studies," Laboratory for research of individual differences, https://lira.f.bg.ac.rs/en/open-science-and-cross-cultural-studies/ [Accessed Sept. 20, 2020].

[28] "Open Access Repository for Measurement Instruments," GESIS – Leibniz Institute for the Social Sciences, https://zis.gesis.org/en [Accessed Sept. 20, 2020].

[29] OECD, "Business models for sustainable research data repositories," OECD Science, Technology and Industry Policy Papers, Dec. 2017, https://doi.org/10.1787/302b12bb-en.

[30] CESSDA Training Team, "CESSDA Data Management Expert Guide," CESSDA ERIC, Bergen, Norway, Jan. 2020, https://doi.org/10.5281/zenodo.3820473.

Free Software Tools for Computational Linguistics: An Overview

Miloš D. Đurić
Faculty of Electrical Engineering, University of Belgrade, Belgrade, Serbia
djuric@etf.bg.ac.rs

Abstract: In the past six decades or so, we witnessed a rapid growth in the study of what is now known as Computational Linguistics. During the last decade, free software tools received increasing attention by the computational linguistic research community, and attracted the interest of computational linguists. This paper looks into the implications of utilizing free software tools in the domain of acoustic phonetics, discourse analysis and computational text analysis. In other words, the present paper is a descriptive exploration of free software tools, which I utilized for my research purposes. Therefore, this paper sets out to explore some useful aspects of these tools in order to get a better understanding of the roles they may have in computational linguistics. The aim of the paper is to provide a unitary descriptive account of Praat, KH Code and NLTK and shed light on their benefits from the point of view of their user. By way of demonstrating some of their features, certain concrete proposals are given. It is hoped that this investigation may spark interest for further research on the subject.

Keywords: Computational Linguistics; Free Software Tools; Praat; KH Coder; NLTK.

I. Introductory Remarks

Generally speaking, according to the pertinent literature, since their first appearance, in the late 1940s, computers have become increasingly familiar to the general public [1]. However, at the time, computational linguistics was seen mainly through mechanical translation, which was considered as the best known and most glamorous aspect of computational linguistics. With the advent of the Internet, Computational Linguistics has witnessed a revived interest precisely because it has become part and parcel of the phenomena Computer Science and Artificial Intelligence have set about to explain. The terminological inconsistency is also spotted in the pertinent literature, particularly since some researchers equate the terms 'computer speech and language processing' with 'human language technology', 'natural language processing', and ultimately, with 'computational linguistics' [2].

Broadly speaking, computational linguistics is said to be interdisciplinary, since in its methods of analysis it takes into account a variety of diverse perspectives. More specifically, in linguistics this area has the implications and applications in the domain of Second Language Acquisition and Computer-Assisted Language Learning [3], and, it can be also applied in the domain viewed from information-processing perspective in works which treat humans as limited-capacity processors. As regards the theoretical framework, which was fused in an eclectic way in order to analyse the corpus-based data, my analysis has been informed by the following studies: [4]–[26].

The paper aims to describe three free software tools in terms of their usability for a linguist, and a computational linguist, for that matter, and is, therefore, primarily descriptive in its orientation. The practical part of the study is devoted to bringing together linguistically and computationally motivated analyses as a rationale behind the inspection of free software tools for computational linguistics.

II. Free Software Tools

Before one proceeds, one is tempted to provide at least one broad and tentative working definition of free software. Broadly speaking, free and open-source software (or, simply FOSS) might be defined as a piece of software that can be classified as both free software and open-source software. According to the pertinent literature, the terms "free software" and "open source software" might be said to refer to software products distributed under terms, which allow users to use the respective software, modify the software according to one's needs, and ultimately, redistribute the software [27].

Since my intention is not to clarify this delimitation in depth, I shall adopt solely this working definition and apply it to the tools that were utilised in the research, and this survey, for that matter.

Equally, the term "tools" will be used very loosely, since one encompasses an acoustic tool (Praat), a text chunking tool (KH Coder), and a powerful library based on the Python programming language (NLTK), respectively.

A. Praat

Broadly speaking, Praat is a free software package with open source code aimed at linguists intending to analyse speech, i.e. spoken discourse. In addition to this, according to the pertinent literature, Praat is generally defined as computer software for phonetic analysis [28], and more specifically, as a standard tool for transcription of speech, and classification of speech events [29]. In addition to these definitions, one comes across the definition of Praat being described as a versatile, open-source platform, which provides a whole lot of features. Furthermore, the quoted reference [28] asserts that Praat might be utilised in the context of the pronunciation teaching process by allowing the learners to individually analyse the generated visual patterns of their own speech thereby making them aware of nuances and diverse

DOI: 10.5281/zenodo.4073970

distinctions within the target language pronunciation. In addition to this, the research has been undertaken in order to explore the ways in which the learners of foreign languages could improve their pronunciation by using Praat.

Chronologically speaking, Praat is also defined as an application developed for speech researchers. The creators of Praat are Paul Boersma and David Weenik (both from the University of Amsterdam). Even though the main purpose of Praat was to apply it in the realm of speech analysis and speech synthesis, its application has been developed further in the direction of facilitating manipulation and labelling processes whilst, simultaneously, offering a powerful apparatus for phoneme identification. Additionally, Praat researchers enhanced format plotting, amongst other things, thereby providing a sound foundation for teaching vowel and diphthong production processes.

It goes without saying that the Praat program can be downloaded free of charge [30]. The Internet source also provides the description of the features pertaining to the Praat software tool as well as useful guides.

However, Praat is not only used in the context of Second Language Acquisition (i.e. SLA), but also in the context of prosody conversion [31]. Praat seems to allow for all sorts of articulatory and acoustic analyses. These analyses comprise segmental and prosodic characteristics of spoken discourse.

According to some researchers, Praat might tackle dialect research and may even be used for forensic purposes, since it already enables a detailed acoustic analysis and annotation of speech data, both in phonetic and phonological domains [32]. Voice analysis using Praat tool has also been fruitful so far, particularly in the domain of assessing a user's emotional state [33].

The Praat research is anchored in different theories, one of which is Optimality Theory (OT), particularly as a way of understanding the Optimality-theoretic driven stochastic grammars [34]. A particularly striking example of the application of OT theory in the Praat analysis environment lies in Boersma's Gradual Learning Algorithm enabled by the Praat program to help you rank Optimality-Theoretic constraints in ordinal and stochastic grammars.

In the domain of language teaching, Praat is considered to have been designed to be used by serious speech researchers, whilst complex computer readouts related to formant plots demand a sophisticated level of understanding [35]. In teaching English pronunciation practice, the focus is primarily on segmental and suprasegmental pronunciation [36]. Nevertheless, Praat is also utilised in looking into its effectiveness in helping students to acquire prosodic features of the English language [37].

Before I embark on the concrete application, let us see the plausible application, guaranteed by the creators of Praat. Firstly, Praat can be used in speech analysis, which is comprised of spectral analysis, pitch analysis, formant analysis, intensity analysis, analysis of jitter, shimmer and voice breaks, generating cochleagram and excitation pattern. Secondly, it is used in the domain of speech synthesis, which brings into the focus pitch, formant, intensity, articulatory synthesis, as well as Klatt acoustic synthesis. Praat also marks the borderline in the domain of listening experiments and labelling and segmentation. The former comprises identification and discrimination tests, whilst the latter includes label intervals and time points on multiple tiers, the use of phonetic alphabet, and the use of sound files up to two gigabytes 2 GB, or in terms of corpus length three hours of spoken data.

One ought to mention other functionalities of Praat for the purpose of a more comprehensive picture. These functionalities would include: 1. speech manipulation (encompassing: change pitch, duration contours and filtering), 2. learning algorithms (bringing about a biologically-inspired feed forward neural networks, followed by discrete and stochastic Optimality Theory), 3. statistics couched in multidimensional scaling, principal component analysis and discriminant analysis, 4. graphics (high quality for scientific papers and theses, production of encapsulated PostScript files, integrated mathematical and phonetic symbols), 5. programmability (easy programmable scripting language and well-established communication with other programs), 6. Portability (including well-organised machine-independent binary files, and possibility of reading and writing diverse sound and other file types, and finally, 7. configurability [38]. It should be mentioned that Praat abounds in plug-ins, which are resorted to in prosody analysis [39], amongst other things.

Now let us see the screen capture of the Praat working environment.

It can be spotted in Figure 1, that the Praat working environment comprises two principal elements: the Praat Objects, and the Praat Picture. According to the pertinent literature, the Praat Objects window is the location for the majority of workflows, and this menu is used to open, create and save files, with further possibility of opening various editors and queries one needs in order to work with sound files [40]. One should select a sound and then the option "View and Edit". Afterwards, whilst examining a sound file, the editor window shows the sound's waveform on the top and a spectrogram on the bottom. Within this working environment, the cursor

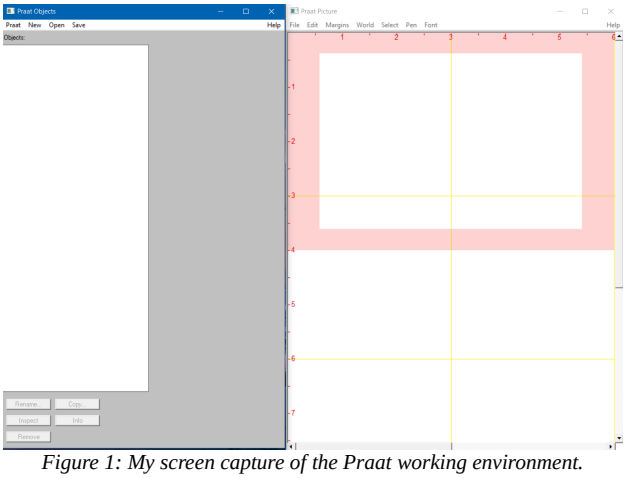


Figure 1: My screen capture of the Praat working environment.

allows a researcher to carry out selections and perform measurement. Generally speaking, Praat is particularly useful in corpus-based analysis. A spoken corpus typically consists of a set of sound files, each of which is paired with an annotation file, and metadata information. In the part that follows, I shall describe the application of Praat in my research.

The observations I make in this investigation are based on the data that have been collected from the oral medium in the form of academic lectures. Namely, academic discourse manifests a wealth in the number and variety of compounds. Generally speaking, delimiting binary/two-constituent/non-canonical compounds has not always been fairly easy and not without problems. Since stress is considered to be one of the reliable criteria (for example, see [41]–[45]), it was interesting to look into the role of stress in compounds by means of acoustic analysis provided by Praat. However, generally speaking, non-canonical i.e. multi-constituent compounds have been out of the focus due to certain delimitation problems, among other things. If they have been discussed at all, this mainly occurred in connection with standard language and written medium, as well as fairly informal styles. It seems that multi-constituent constructs in academic discourse have been left aside. As a consequence of such tendencies, multi-constituent constructs have been delimited as a separate, though not syntactically clearly delimited category of lexical items. Strictly speaking, this Praat-motivated investigation turns attention to the issue of a more adequate delimitation of multi-constituent constructs, particularly to the set of linguistic units that display variation in stress, this being illustrated by the corpus-based data.

The motivation lying behind the decision to select compounds as an object of study could be found in the claims from the pertinent literature, according to which, the analysis of nominal compound constructions has proven to be an unmanageable and recalcitrant problem, which poses serious challenges for natural language processing systems [46].

More precisely, in this Praat-motivated study, I focus on stress of multi-constituent constructs. Since stress is often used as the delimitation marker between phrases and compounds in the English language (see, for example: [47]–[49]), and yet, many examples taken from the language contradict this general rule, I have decided to analyse the authentic oral corpus, and to analyse how multi-constituent constructs behave in this discourse type with respect to this prosodic feature.

Our speech data come from a specific register of the oral/spoken medium in the form of academic lectures. I have chosen this type of spoken discourse because I have assumed that there could be either consistency or variability in the prosodic pattern of certain multi-constituent constructs, which are used relatively frequently in academic discourse. Furthermore, this discourse type provides a relatively narrow domain of knowledge in which such constructs are used. The examples that I considered relevant for my hypothesis showed that relevant factors for the occurrence of compound stress consistency might be the processes of domain-specific lexicalization of certain constructs.

In order to avoid the mentioned problems, I extended the empirical scope.

The question that might be posed is: Why Computer Science academic discourse? The first reason, according to the pertinent literature would be that from the perspective of the traditional lexicon designer working within computational linguistics, complex nominals, i.e. compounds are formed generatively and therefore do not merit explicit listing except when clearly non-compositional [50]. In this context, according to the quoted reference, in this spectrum of compounds, technical terminology holds the attention of a significant location, being highly productive and encapsulating the essential concepts of a particular technical domain.

The second reason for selecting five academic lectures was that I wanted to avoid using fairly small data sets. The third reason pertains to my wish to avoid using my own intuition, the practice not uncommon in the linguistic research of researchers operating within the tradition of transformational-generative paradigm and transformational grammarians who have long used their own native speaker intuitions [51].

I have extended the empirical scope and studied the prominence found in the actual speech (i.e. speech data from more specialised genres and language registers), and tried to analyse these linguistic items by means of acoustic analysis. I have established five classes of constructs: 1. Dictionary-attested constructs (DAC), 2. Frequent and repeated constructs (FRC), 3. Discourse community constructs (DCC), 4. Domain-specific constructs (DSC) and 5. Multi-constituent constructs (MCC).

Multi-constituent constructs (MCCs) were selected for the analysis. My initial assumption is that there could be either consistency or variability in the prosodic pattern of MCCs in academic discourse. Corpus comprises high-quality recordings of lectures (the duration of which totals six hours and fourteen minutes in the MP3 format which was subsequently converted into .wav format so as to be able to undergo the Praat analysis. The Transcript of Lectures (ToL) consists of 75 pages comprising 45187 words.

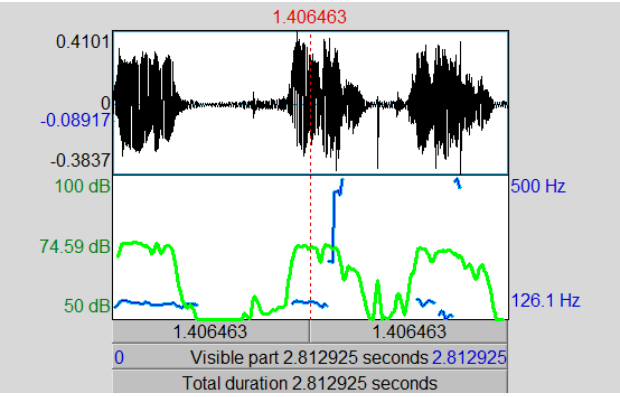


Figure 2: The Praat-generated token 1 of the MCC "random number generator" from my corpus.

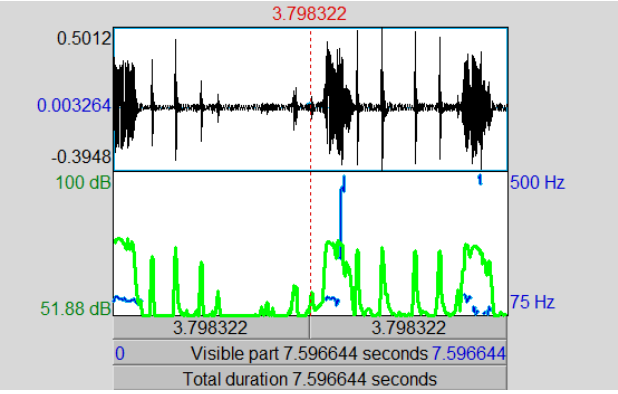


Figure 3: The Praat-generated token 2 of the MCC "random number generator" from my corpus.

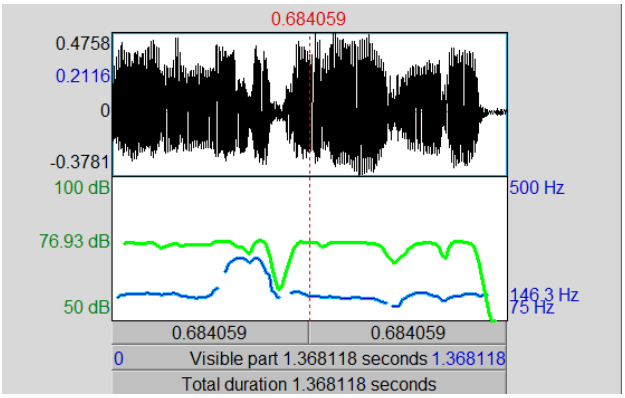


Figure 4: The Praat-generated token 3 of the MCC "random number generator" from my corpus.

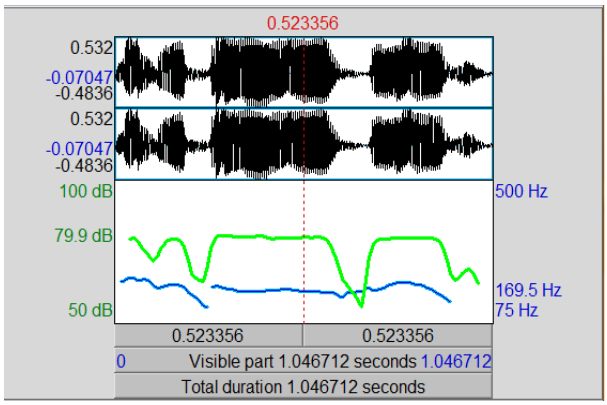


Figure 5: The Praat-generated token 1 of the MCC "hundred dollar bills" from my corpus.

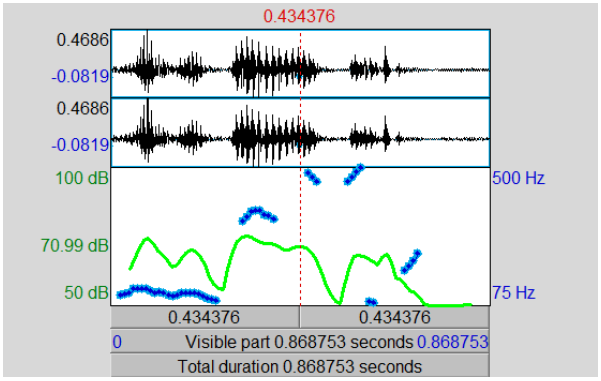


Figure 6: The Praat-generated token 2 of the MCC "hundred dollar bills" from my corpus.

Set apart from mostly clear-cut cases of binary compounds is a group of multi-constituent compounds which may exhibit somewhat different acoustic behaviour. Let us see the following example from our corpus. Specifically, I there are three tokens of the MCC "random number generator" in my corpus. All tokens have been analysed by means of Praat, and the results of the analysis are displayed in the following figures.

The maximum pitch for the first token equals 489.71 Hz (Figure 2), whilst for the second token this value equals 488.74 Hz (Figure 3), and 258.47 Hz in the case of the third token (Figure 4). The duration, displayed in seconds, varies, so, on the one hand the duration of the first token totals 2,81 s, whilst the second token lasts for 7,59 s and the third one 1,36 s. Linguists are feeling their way on a slippery terrain in the cases, like this one, when the intra-speaker variation has been spotted and then acoustically-confirmed. So, in a nutshell, the Praat-provided visualisation facilitates better understanding of the subtle differences in intra-speaker variation, otherwise perceived by introspection, but not confirmed acoustically by proper measurement. Let us now see the case of the MCC that has two tokens in total. The example in question is the unit "hundred dollar bills".

When juxtaposed, as in the case of this Praat-generated visualisation (Figure 5 and Figure 6), one cannot but notice that these two tokens of the MCC "hundred dollar bills" exhibit the variation in the maximum pitch. More specifically, the maximum pitch of the first token equals 208.16 Hz, whilst the latter one equals 493.80 Hz. Additionally, the variation in duration has been spotted. Namely, the first token lasts for 1.04 s, and the second token lasts for 0.86 s.

In my previous research, all the analysed examples have shown that compoundhood of a MCC is well-established in discourse unless for some discoursal reason the significance of the construct is to be underlined, for example, at the end of the sentence, or at the end of the discoursal subtopic in generalised conclusive utterances. This conclusion is enabled through the analysis by means of Praat. In the next section, we shall see some plausible advantages and disadvantages of using Praat.

In this part, I have tried to show how some central acoustic parameters provided by Praat can be applied in Computational Linguistics by focusing on a small group of compounds (i.e. MCCs) that might mark the borderline between binary i.e. canonical and multi-constituent i.e. non-canonical compounds. I have argued that the Praat visualisation and Praat-generated parameters could change the fairly static picture provided by non-acoustic approaches. Moreover, the non-acoustic analysis seems to be inadequate to grapple with items that cannot be easily captured in compoundhood-driven terms. In addition to this, intuition-based analyses of MCCs have equally brought about a host of problems, which can be resolved by means of pretty straightforward visualisations, such as those generated by means of Praat.

Perhaps the paramount feature of Praat might be considered to be its all-embracing help function, which is brought up-to-date regularly. It should be stressed that

this represents a circumstance which seems convenient for both expert and non-expert users. Over and above, another eye-catching and attractive Praat-feature refers to its offering of its own scripting language, which is another reason to utilise this tool in computational linguistics.

However, this software tool is not without its problems. More precisely, the felicitous handling and usage depends on the user. Namely, computational linguists might find this free software tool very useful and handy, whilst perhaps some non-acoustically oriented users with the lack of knowledge in the domain of acoustic phonetics and computational linguistics, for that matter, might find the utilisation of Praat as something pretty complex and demanding thereby opting for the more intuitive approach in spoken language data analysis.

To conclude, intuitive analyses based on sort of introspection have tended to obfuscate rather than clarify speech sound phenomena and suprasegmental properties of sounds, MCCs and their stress not being an exception to this problem set. The described free software tool Praat, which is intended for acoustic analysis, seems to offer both precise measurement and accurate description of the given speech phenomena under investigation.

The successful utilisation of this free software tool depends on the nature of the user. Namely, those users who are not familiar with concepts within computational linguistics will perhaps use Praat less successfully than those who are computational linguists. However, they are not without problems, particularly if we take into account computer scientists, software developers and engineers who can master this free software tool to overcome any acoustic problem. Therefore, it can be said that Praat depends on the nature of its user.

B. KH Coder

If one tries to define this software tool, one comes across the definition of KH Coder provided by its author. I have slightly modified the given definition by adding the item “tool” in the description. Namely, KH Coder is usually defined as a free software tool for quantitative content analysis or text mining, and it is also utilised for computational linguistics [52]. Furthermore, it is also characterised as a software tool intended for computer-assisted qualitative data analysis. KH Coder was developed by Koichi Higuchi.

The survey of the literature shows that KH Coder is successfully implemented in diverse text analyses, such as the analysis of occupational accidents and their prevention in Spanish digital press [53]. It is also used in analysing students’ course evaluation through text mining, which is predominantly based on co-occurrence network analysis provided by KH Coder [54]. KH Coder is also used in the context of SLA and EFL and ESL in preparing specific teaching materials for advanced reading comprehension based on specific text mining [55]. Furthermore, this free software tool is highly suitable for specific tasks, such as the analysis of specific keywords with the help of co-word mapping comparison between two types of newspapers [56].

Certain authors explore the big data realm as a completely novel field for both scholars and practitioners dealing with big data conceptualisation based on diverse case studies [57]. The relevant features of KH Coder, such as multi-dimensional scaling, cluster analysis and co-occurrence network, are employed by researchers whose aspirations are to be found in the domain of specific language register. In this sense, the researchers implement KH Coder in order to carry out multi-dimensional scaling and co-occurrence network analysis on the academic journal dataset [58].

The emerging field of quantitative text analysis also represents a fruitful field of research particularly for the authors utilising the given free software tool, which has proven to be a satisfactory testing ground both for written and oral data [59]. Similarly, KH Coder is used by researchers exploring news articles databases and comparing their local and international media reports [60]. Finally, there are authors who employ text analytics visualisation provided by the free software tool in question in order to explore and visualise student comment data in the discourse of science and technology [61].

In the part that follows I shall briefly describe one previous research of mine, in which I utilised KH Coder for computational discourse analysis [62]. The research was part of a wider interdisciplinary field of discourse studies, more specifically, digital art museum discourse, which explored various aspects of language expression that is manifested in this discourse type. CAT was applied to the text contained within the web pages of six digital museums of digital art. I must emphasise that I have utilised some parameters, or, more specifically, textual dimensions elaborated in the pertinent literature (for instance, see [63]). At this point, one should also add the remark from the literature that text collections and corpora in digital form (like my corpus) represent important resources for empirical research [64].

Since KH Coder belongs to free software tools for quantitative content analysis and text mining, it is, consequently utilised for computational linguistics, and as such offers a plethora of features that might analyse the language material and facilitate CTA. By way of illustration, we shall see the actual implementation of this software tool.

The given visualisation (Figure 7), provided by KH Coder, lends support to the assumption that individual language items might be followed easily, even though a lot of combinations would appear within these clusters. Additionally, some overlapping clusters might have gone further on the analysis path leaving the most distant ones stranded. One can notice that the lexical unit “programming” collocates with the units “software”, “code” and “package” thereby generating the following clusters: “programming software”, “programming code” and “programming package”, to mention but a few.

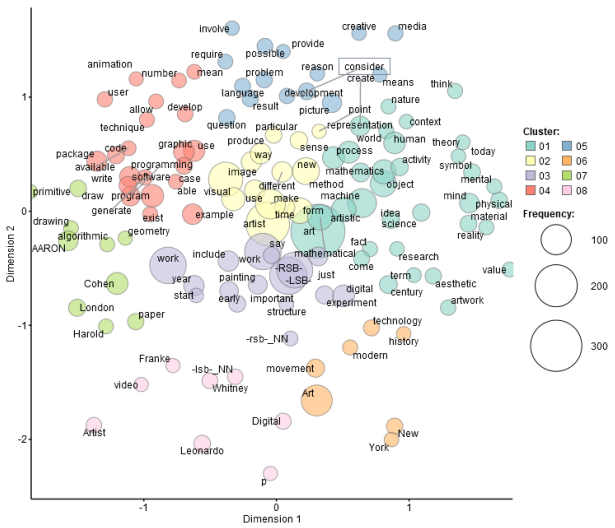


Figure 7: The two-dimensional solution for non-metric multidimensional scaling (2D Cruscal) for the text excerpt from my DAM corpus.

Now, let us see the three-dimensional solution.

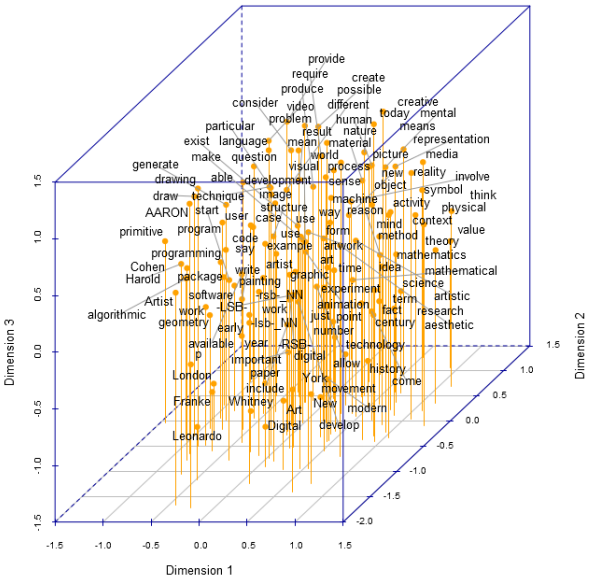


Figure 8: The three-dimensional solution for non-metric multidimensional scaling (3D Cruscal) for the text excerpt from my DAM corpus.

The first impression is that 3D cruscal (Figure 8) seems not to be neatly organised as is the case with the 2D cruscal. Perhaps, this might be the case due to the corpus size. However, 3D visualisation seems to offer less satisfactory data when it comes to cluster analysis. Nonetheless, a host of collocations can be traced and spotted without looking into separate tables, for that matter.

Now let us see the visualisation of the previously sifted corpus data.

The lexical unit “VR” collocates with the items “people”, “platform”, “time”, etc. And the adjective “virtual” generates the clusters “virtual museum”, “virtual exhibition” and “virtual experience” (Figure 9).

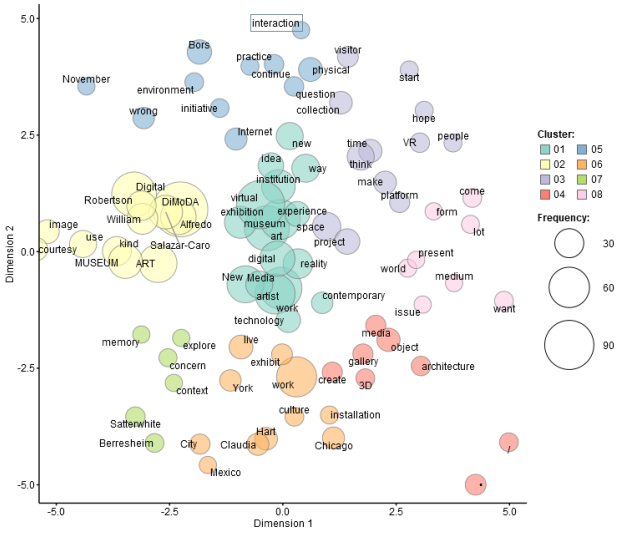


Figure 9: 2D Cruscal for the text excerpt from my DiMoDA corpus.

And now let us see another case of the given corpus-based analysis.

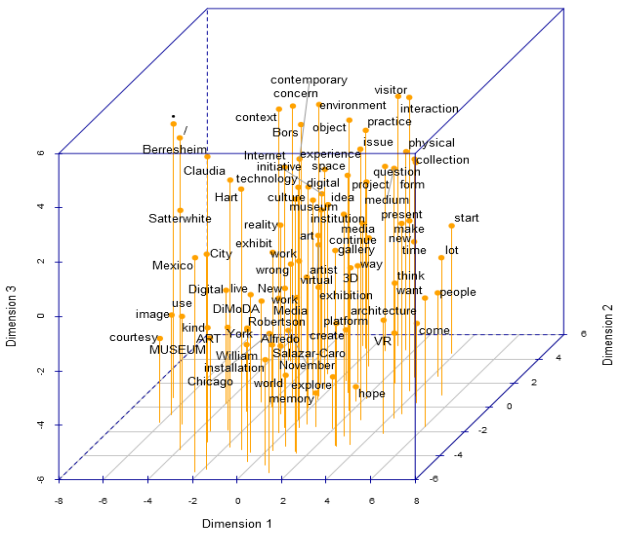


Figure 10: 3D Cruscal for the text excerpt from my DiMoDA corpus.

This time (Figure 10), 3D cruscal fits neatly in the representational-computational approach to the analysed lexical items. The centrality is taken by the lexical item “museum” which is located near items with which it enters into the most frequently occurring collocation patterns. Of course, this visualisation is not sufficient enough on its own, but ought to be accompanied by statistical tables and other numerical parameters that are obtainable in KH Coder.

Sometimes automatically-driven part-of-speech tagging might be problematic, as can be seen in the previous illustrative example (Table 1). Namely, the semantic unit “New” is treated as an instance of a proper noun, even though we cannot see the context in which it appears immediately and the sole indicator for this

Table 1: The KH Coder-generated illustrative table for the text excerpt from my La TurboAvedon corpus.

Lexical items	Part of Speech	Frequency
AVEDON	ProperNoun	21
space	Noun	17
LATURBO	ProperNoun	16
work	Noun	16
virtual	Adj	10
New	ProperNoun	8
artist	Noun	8
live	Verb	8
avatar	Noun	7
consider	Verb	7
experience	Noun	7
media	Noun	7
paraspace	Noun	7
production	Noun	7
sculpture	Noun	7
surface	Noun	6
Sculpt	ProperNoun	5
authorship	Noun	5
identity	Noun	5
object	Noun	5
parasubject	Noun	5
polygon	Noun	5
social	Adj	5
term	Noun	5

decision seems to be the initial capital letter “N”, which must have evoked the item “New” in the proper noun“New York”. This is the reason why one should use simultaneously statistical tables and visualisations provided by KH Coder. Therefore, classical tables should be interpreted jointly with graphs and 2D and 3D visualisations. It is in this way that the computational discourse analysis might be extended.

KH Coder enables analysing lexical clusters and collocations that are not quite susceptible to classical text analysis, or discourse analysis, for that matter. However, this computational discourse analysis is in stark opposition to non-computational accounts, precisely because it provides better language data manipulation through precisely adjusted measurement methods in terms of statistical analysis.

As can be seen from the neat examples from my corpus cited so far, sometimes certain clusters signal to the analyser to drop certain definitional characteristics, as was the case with letter capitalisation influencing the specific part-of-speech tagging (see: Table 1). The analysis restriction that seems problematic refers to the situation when certain lexical units may remain unspecified in terms of part-of-speech. This is why one should not combine Chomskyan (i.e. computational) manners of analysis with non-Chomskyan (i.e. impressionistic) ways of analysis. It should be added that, according to the literature, in the period before Chomsky, linguistics tended to be a taxonomic enterprise, which was dubbed verbal botany [65]. However, this is not to say that KH Coder lacks the essential features of a free software tool intended for computational discourse analysis. On the contrary, this tool provides an upgrade of a sort. Following the standard picture, the benefits of KH Coder refer to the analyses it provides: 1. word frequency list, 2. the context in which the lexical item is used, 3. co-occurrence network of words, 4. correspondence analysis of words, to name just a few. However, some challenges remain unresolved, such as those referred to in the literature concerning the pitfalls on the path to formulating a unique query which could extract information from aligned texts [66], among other things.

Furthermore, statistical analyses of automatically extracted words are suitable for gaining a whole picture of the data since traditional problems of how to represent lexical items and their respective lexical clusters and collocations in standard glossaries and general dictionaries alike have become more apparent than real. Unfortunately, various coding rules to count concepts, a topic no less interesting in the computational discourse analysis could not be taken up in my investigation, and this paper, for that matter. In the part that follows, I present another free software tool.

C. NLTK

The Natural Language Toolkit (NLTK) represents a collection of libraries and programs for symbolic and statistical NLP written in the Python programming language. More precisely, according to the pertinent literature, the NLTK is a suite of open source program modules, tutorials and problem sets, providing ready-to-use computational linguistics courseware [67]. Furthermore, NLTK is said to be a platform for building Python programs to work with human language data. Additionally, it is asserted in the literature that Python Natural Language Processing Toolkit plays an important role as a platform for building Python programs to work with human language data [68].

NLTK arrives with a large collection of corpora, followed by large-scale and extended documentation, making NLTK unique in providing a comprehensive framework for students to develop a computational understanding of language [69]. The quoted reference asserts that NLTK’s code base of 100,000 lines of Python code includes support for corpus access, tokenising, stemming, tagging, chunking, parsing, clustering, classification, language modeling, semantic interpretation, unification, among other things.

Furthermore, NLTK has many third-party extensions. There are plenty of approaches to each NLP task in the NLTK environment. Related to this are also fast sentence tokenisation, and other relevant features for computational linguistics. And now let us consider the choice for selecting the Python programming language.

It should be mentioned that the creators of NLTK are Steven Bird and Edward Loper (both from the Department of Computer and Information Science at the University of Pennsylvania). NLTK has been used successfully as a teaching tool and, so far, many researchers have chosen Python as their implementation language for NLTK, mainly because Python’s syntax and semantics are transparent with good string-handling functionality. On the one hand, Python is an interpreted language which makes it suitable for facilitating interactive exploration. On the other hand, it is an object-oriented language, which entails that Python allows data

and methods to be encapsulated and readily and easily re-used.

Additionally, according to the literature, Python is heavily used in the industrial context and scientific research alike. Nevertheless, it also offers programming possibilities in educational contexts around the world. The same source claims that Python is said to be often praised for the way it facilitates productivity, quality, and maintainability of software [70].

Some of the features that might be useful to computational linguists can be carried out by means of the NLTK. For example, tokenising text into sentences, tokenising sentences into words, tokenising sentences using regular expressions, filtering stop words in a tokenised sentence, stemming words, lemmatizing words, creating custom corpora, part-of-speech tagging, extracting chunks, text classification and parsing specific data, to list just a small portion of features, functionalities and possibilities from the representative literature [71]. In the following section, I shall briefly describe the actual use of NLTK in CTA.

The first obvious application of NLTK in CTA refers to the main features pertaining to computing with language. More specifically, NLTK enables the following: categorising and tagging words, processing raw text, accessing text corpora and lexical resources, writing structured programs, learning how to classify a text, extracting specific text information from text. Moreover, one can also analyse the sentence structure by NLTK. This tool can also analyse the meaning of the language data in general, and the meaning of sentences, in particular.

The already existing corpora may be sufficient for a scholar interested in the basic computational linguistics. Still, they seem to be representative enough in terms of corpus representativeness. Let us see an example of NLKT corpora.

These nine texts (Figure 11), or more precisely, corpora are sufficiently equipped so as to serve as input data for a computational linguistic analysis. A linguist is provided with neatly modified ways of exploring the given corpora. For example, let us search for concordances of the lexical item “lucky” in the first corpus (Figure 12).

Counting vocabulary is another convenience provided in the NLTK environment. Let us see an example in Figure 13.

As seen from my illustrative example (Figure 13), the described free software tool seems to be very user-friendly and convenient for a computational linguist who wishes to find out the length of a corpus. Strictly speaking, the number refers to the words and punctuation symbols which occur. The term len is utilised to obtain the length of something, in my case, a text, which has been applied to the corpora at hand.

I was particularly interested in generating tokens and tokenisation process within the NLTK context. Not surprisingly, tokens have become one of the highly explored language phenomena within the current linguistic research both of cognitive and computational provenance (see, for instance, [72] and [73]). In the vast literature existing today, a token has been referred to as an instance of a unit, as distinct from the unit that is instanced [74].

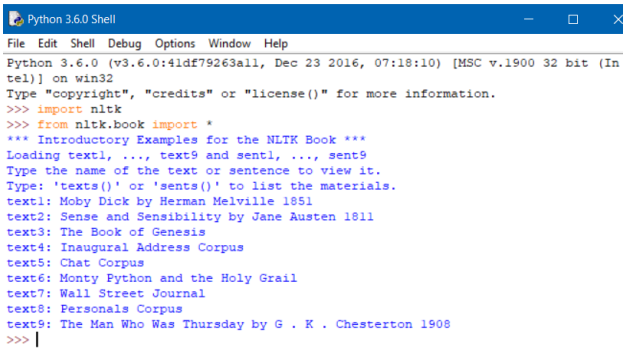


Figure 11: My screen capture of an illustrative example of the NLTK corpus structure.

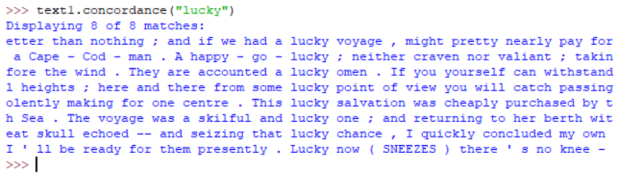


Figure 12: My screen capture of the NLTK-generated concordance of the lexical item “lucky” from the first NLTK corpus.

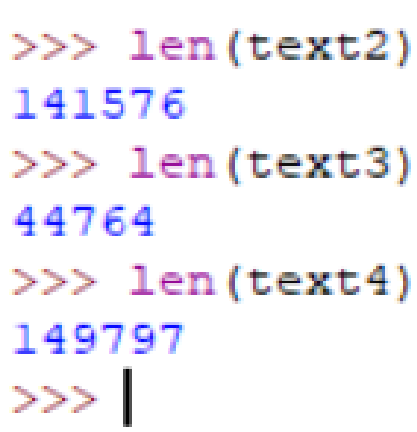


Figure 13: My screen capture of an illustrative example of vocabulary counting of NLTK corpora.

More specifically, in linguistics, the term “token” is simply defined as a particular example of a general type [75]. According to some computationally-motivated sources, however, the vocabulary of a text is just the *set* of tokens that it uses, since in a set, all duplicates are collapsed together [76]. Now, let us see a typical example of tokenisation by means of NLTK.

The immediate problem that is noticed (Figure 14) is the presence of the occurrence of orthographic symbols,

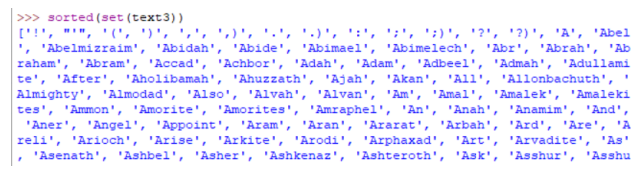


Figure 14: My screen capture of an illustrative specimen of tokens in the NLTK corpus number 3.

```
>>> text3.dispersion_plot(["light", "dark", "love", "Lord", "sin"])
```

Figure 15: My screen capture of an illustrative example written in Python in order to obtain the lexical dispersion plot for NLTK corpus 3 (i.e. The Book of Genesis).

which are redundant for a linguistic analysis, at least the one that is lexeme-focused. According to the literature on the subject, by wrapping `sorted()` around the Python expression `set(text3)`, a sorted list of vocabulary items is obtained, and this list begins with various punctuation symbols and continues with words starting with "A". It is essential that we should mention that all upper-case words precede lower-case words.

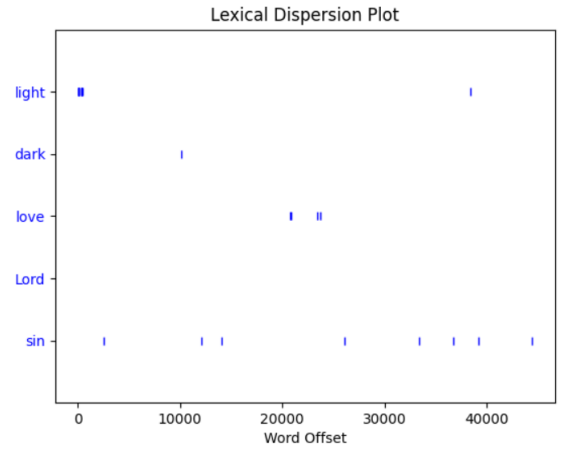


Figure 16: My screen capture of the actual lexical dispersion plot for the NLTK corpus 3 (i.e. The Book of Genesis) generated by the NLTK tool.

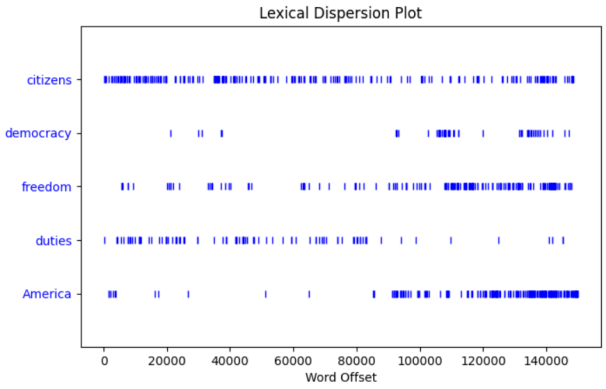


Figure 17: My screen capture of the lexical dispersion plot for the NLTK corpus 4 (i.e. Inaugural Address Corpus) generated by the NLTK tool.

```
>>>
>>> text4.dispersion_plot(["citizens", "democracy", "freedom", "duties", "America"])
```

Figure 18: My screen capture of the command line written in Python in order to obtain the lexical dispersion plot for the NLTK corpus 4 (i.e. Inaugural Address Corpus).

However, whilst working within the NLTK environment, one may also determine the location of a lexical item in the given corpus. This positional information can be displayed by means of a lexical dispersion plot. Each stripe represents an instance of an item, and each row represents the entire corpus. Let us consider the following example.

My illustrative examples (Figure 15 and Figure 16) capture the gist of the given lexical dispersion. However, in the pertinent literature, there are examples which perhaps illustrate the point in a more visually striking way [70]. Let us now consider them in turn in Figure 18 and Figure 17, respectively.

Other useful features pertain to parsing and part-of-speech tagging. In linguistics, one cannot avoid grappling with the notion from the traditional grammar, which treats parsing as the pedagogical exercise of labelling the grammatical elements of single sentences [77]. It is claimed that the criteria of analysis leading to the identification of grammatical elements seem to be more salient mainly owing to the ways in which speakers of a natural language (i.e. English, in this case) use these items to relate sentences in the language in its entirety. When it comes to part of speech, one encounters the fuzzy traditional notion, which refers to a grammatical class of words [77]. Due to their inexplicitness when defined, and the restricted nature of their definitions, some linguists tend to prefer alternative terms (for example, in the pertinent literature, the term 'parts of speech' is equated with the terms: 'word classes' [78], 'class of words' [79], and 'lexical category' [80], to mention but a few labels, among other things). However, since the terminological inconsistency is not the object of this study, I shall not broaden further that, no less interesting, topic.

On the one hand, one may spot the linguistically-oriented parsing task, which is mainly the task of assigning words to parts of speech. However, on the other hand, in the computationally-oriented analyses, this parsing process refers more to the assignment of syntactic structures to sentences, especially by parsing programmes, or, more precisely, parsers [74]. From a cognitively oriented perspective, however, parsing is treated as one of the mental processes involved in sentence comprehension, in which the listener determines the syntactic categories of words, joins them up in a tree, and ultimately, identifies the essential parts of that particular sentence [81]. Now, let us consider the basic use of tagger in computational linguistics.

```
>>> text = word_tokenize("Andrew Telfer is writing a note at his desk in one
corner of a big, book-lined room")
>>> nltk.pos_tag(text)
[('Andrew', 'NNP'), ('Telfer', 'NNP'), ('is', 'VBZ'), ('writing', 'VBG'), ('
a', 'DT'), ('note', 'NN'), ('at', 'IN'), ('his', 'PRP$'), ('desk', 'NN'), ('
in', 'IN'), ('one', 'CD'), ('corner', 'NN'), ('of', 'IN'), ('a', 'DT'), ('bi
g', 'JJ'), (',', ','), ('book-lined', 'JJ'), ('room', 'NN')]
>>>
```

Figure 19: My screen capture of the POS-tagger processing an illustrative utterance from my corpus (i.e. The Ninth Gate Corpus).

A part-of-speech tagger, or POS-tagger, processes a sequence of lexical items in a sentence from my corpus ("Andrew Telfer is writing a note at his desk in one corner of a big book-lined room."), and attaches a part of speech tag to each item (e.g. 'Andrew', 'NNP', 'Telfer', 'NNP', and so on). Briefly, my example (Figure 19) illustrates how the implementation of the NLTK tool may

be pretty convenient for a computational linguist wishing either to practice automatic tagging or to import a larger text portion in order to carry out a CTA. There are other, perhaps more attractive, possibilities, apart from the ones described, however, I feel this may suffice to illustrate the point under consideration. In the section that follows, I shall briefly mention certain potential benefits of utilising this free software tool from the point of view of a computational linguist.

Generally speaking, NLP with Python might be regarded as a promising field. It is in this sense that free software tools and libraries enabling such processing are most welcome as precious ingredients of any linguistically-motivated analysis. NLTK represents one such undertaking, which offers a multitude of features for linguists and computational linguists, alike. Corpus-based studies cannot avoid grappling with lexical items present in the naturally occurring language, such as English, French, Serbian, Croatian, etc. Not surprisingly, parsing, tokenisation and part-of-speech tagging have become highly explored possibilities of a computationally-oriented analysis within the current computational linguistic research.

I have shown some illustrative examples performed in the NLTK environment. One may notice potentially useful features, but also the lack of some clear-cut features for some language items, such as orthographic symbols and signs. Even though the part-of-speech tagging is sufficiently felicitous for the majority of computationally-driven analyses, it seems that in some cases, there are certain examples that cannot easily undergo this process. However, these instances seem to be rather rare. This issue poses some challenges, which might be accounted for by the still deeply rooted traditional parts of speech that are treated in terms of necessary and sufficient conditions salient for a lexical item to be included in a given class.

Next, the free software tool NLTK has shown that rather simple programming techniques could be combined so as to deal with large quantities of language material in the form of representative corpora. The features of NLTK abound in different parsing and tagging possibilities and may facilitate the automatic extraction of some key lexical items and/or phrases within a given corpus. Tools and techniques that the Python programming language provides for computational linguistics are numerous, and therefore might represent stimulating challenges posed by natural language processing.

Computing with language, if by this we refer to working with texts (i.e. spoken and written discourses) and words, seems to have been made easier by NLTK and its readily available corpora, followed by some additional features permitting the all-comprising analysis of the language material at hand. Furthermore, free software tool NLTK treats texts as lists of lexical items which may undergo analyses required by a computational linguist, and therefore, may offer help to those computational linguists that need a precise analysis.

III. The Comparison of the Selected Free Software Tools

In the following lines I shall briefly compare three free software tools. Let us briefly consider them in turn. The first free software tool in this analysis is Praat, which is highly functional from the point of view of acoustic phonetics and computational linguistics. Although it has certain limitations as to the duration of the spoken corpus chunk, it certainly represents a reliable resource offering diverse options for a plausible acoustic analysis. Praat has all the advantages of a free software tool and can be easily handled by computational linguists both beginners, and advanced researchers.

The second free software tool in this analysis is KH Coder, which can be used for treating text from a computational point of view, providing all sorts of statistical analyses, both qualitative and quantitative. Even though there are some challenges that should be responded to, such as spelling rules that influence the part of speech tagging and certain lexeme delimitation, these are forgivable weak points in such a multi-perspective analysis provided by KH Coder. This free software tool provides collocation patterns, multi-dimensional analyses and various visualisations which can help and complement the computational analysis of (mainly written) discourse.

The third free software tool in this analysis is NLTK, which is a Python-based natural language toolkit. It has a powerful corpus with the possibility of adding the language input data of one own and abounds in powerful features.

The feature shared by Praat and KH Coder is that there are some limitations with regard to input size. However, sometimes this does not affect performance. Also, one should add that there are some challenges in the NLTK working environment. Namely, when certain tasks are carried out, such as the tokenisation of a larger corpus, the task performance may slow down, and the data displayed after the executed command is not so clear whilst the data manipulation is not straightforward for a computational linguist who is not well-aware of all the possibilities of the Python programming language. However, this is not an insurmountable obstacle on the way paved by NLTK, since it provides other more appealing peculiarities.

Taking as a starting point the notion of performance, the following rough comparison of performance relations between Praat, KH Coder and NLTK might then be posited: NLTK and KH Coder share some features and functionalities (both tools have the possibility of generating and displaying concordances, visualisations, etc.), Praat has visualisation possibilities, but sometimes not of high picture quality. KH Coder and NLTK are mainly intended for the written medium (i.e. written corpus), whilst Praat is devoted to oral media (i.e. spoken corpus). All three software tools share one common feature, and this refers to the possibility of integrating

their tables and graphs readily into scientific papers, conference papers, books, and so on.

The strengths of Praat are to be found in the acoustic analysis of individual sounds, in the annotation of these sounds, and in browsing multiple sound and annotation files across the corpus. The strengths of KH Coder are the visualisations (particularly 3D) that can be further analysed, while the strengths of NLTK lie in its simplicity and elegance of data output display (however, this is in less attractive format than in the case of KH Coder). Pre-processing activities of the analysed software tools have been left aside, although they might also be indicators of certain advantages and disadvantages in raw data processing.

All three software tools have satisfactory output, at least for a user, who is a computational linguist, or a general linguist. It should be added that I have not considered the level of user-friendliness and successfulness from the point of view of a computer scientist, or an electrical engineer, for that matter, but solely from a perspective of a computational linguist. Limitations have been explored solely to a certain extent, since the author of the paper has attempted to perform an analysis by the described free software tools in fairly straightforward corpus-related contexts. Despite the described benefits, it has been noticed, however, that whilst working with large corpora some tools slow down (for instance, NLTK, and KH Coder, whilst processing the data and providing the output of the required feature). Yet, overall impression is that these analysed software tools seem to be irreproachable since they are free of charge and can be further modified and upgraded, which is not the case for proprietary software tools offering the ready-made templates and patterns that cannot be further modified according to one's needs. And this last remark is not insignificant in terms of the last parameter of performance.

The last parameter to be discussed is that pertaining to the user. Namely, certain linguistic research directions are still under the influence of the traditional non-Chomskyan linguistics, and therefore, utilise somewhat different terms and notations which may sometimes indirectly influence some aspects of the linguistic analysis. Praat and KH Coder do not require special programming skills and advanced programming knowledge, whilst NLTK requires sometimes even advanced knowledge of Python. Therefore, a user, who is most frequently a computational linguist, ought to know the fundamentals of this programming language. As regards the corpus-based analysis, it should be highlighted that I used my own corpora for the analyses carried out by means of Praat and KH Coder, whilst I used the ready-made and available corpora in the NLTK environment. Perhaps, this might be the reason for omitting some aspects of analysis since I relied on the previously prepared data. In the part that follows, some concluding and final observations are provided.

IV. Concluding Remarks

In the past six decades or so, we witnessed a rapid growth in the study of what is now well-known as Computational Linguistics. Nevertheless, unitary accounts have been scant. The aim of this investigation is to fill the lacuna in the current scholarship on free software tools in computational linguistics, at least from a descriptive point of view.

The first part of the paper provides introductory remarks and focuses on general observations concerning computers and computational linguistics. Additionally, certain theoretical underpinnings have been mentioned (namely, generative, optimality-driven, relevance-theoretic, and minimalist-motivated, among others). The second part presents free software tools in general, and then presents three free software tools in particular, which served as input to my subsequent argumentation and conclusions. This part is broken into subsections, each of which briefly presents the software tool in question, its performances and potential benefits. The third part is a sketchy comparative analysis which summarises the findings in connection with software tools performances intended for a specific user, i.e. a computational linguist. Some features and functionalities have been compared and a concise overview has been provided.

Although this paper is largely descriptive in its orientation, three case studies reflect the underlying assumptions of the theoretical frameworks in which they are to be found. Equally, this descriptive exploration was aimed at contributing to a better understanding of free software tools in the domain of computational linguistics. Burdened with an ill-famed and notorious reputation of having been persecuted by proprietary software tool creators and distributors, free software tools have not only resisted but are actually struggling for their own place in the realm of computational linguistics. This was illustrated by assessing and evaluating certain striking properties of three free software tools: Praat, KH Coder and NLTK. In the subsequent comparative analysis, these tools were juxtaposed and compared. From the point of view of the user, it has been claimed that expert users tend to operate these tools more easily when compared with linguists. Perhaps, the only exception might be a computational linguist with certain knowledge of programming languages. However, it has been assessed that all analysed tools are user-friendly and can be easily integrated into a linguistically-motivated study.

The analysed and described three software tools can generate graphs and tables and other visualisations that can support any undertaking concerned with linguistic analysis. These visualisations can further refine the analysis in terms of better understanding of relations between the tokens of lexical items. The main area of contention revolves around the questions of the speed of performing certain tasks (e.g. tokenisation of larger corpora, among other things). Another appealing challenge would refer to the semantic component, which sometimes might not be satisfactorily included in the CTA and NLP, but is, according to the literature, an important ingredient in automatic translation, particularly

in scientific fields [82]. Rather than posit these and similar challenges, I have considered the performance of three free software tools within a broader picture of its overall functionality and usefulness in the investigation carried out by a linguist. Therefore, some of my performance measurement results might exclude certain elements that are unimportant for the linguistic analysis.

In this rather brief and unpretentious study, I have reexamined the role of free software tools for computational linguistics from a comparative perspective. To this purpose, I have implemented and analysed three free software tools. My own corpora were used for analysis carried out by Praat and KH Coder, whereas I used the already available corpora and my own examples in the NLTK analysis. Consequently, perhaps this latter decision, to use the already existing language data, might have influenced certain results of the comparative analysis. My observations are not definitive, but rather constitute a tentative descriptive account, which can be further broadened by integrating diverse appealing dimensions of computational text analysis. Some future comparative investigation might significantly contribute not only to our understanding of the role of free software tools in computational linguistics in general, but also of the role of performance-measurement perceived similarities and differences. Needless to say, my tentative assumptions merit further elaboration.

Acknowledgements

I would like to express my gratitude to Professor Predrag Pejović (Faculty of Electrical Engineering, University of Belgrade) for his kindness, patience, expertise and wisdom. I am grateful to Professor Pejović for bringing the free software tools to my research attention. My gratitude goes to Professor Nadica Miljković (Faculty of Electrical Engineering, University of Belgrade) for inviting me to participate in the PSSOH Conference project whilst it was still in the making. Professor Miljković has inspired many parts of my research by generously sharing her experience and knowledge. Both Professor Pejović and Professor Miljković kindly smoothed the way of performing my modest computational analyses. Needless to say, the comments of the Anonymous Reviewer are highly appreciated.

References

[1] M. F. Bott, "Computational Linguistics," *New Horizons in Linguistics*, Harmondsworth, Penguin Books Ltd., 1971, pp. 215-228.

[2] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*, New York: Prentice Hall, 2000.

[3] M. Levy, *Computer-Assisted Language Learning: Context and Conceptualization*, Oxford: Clarendon Press, 1997.

[4] N. Smith, *The Twitter Machine: Reflections on Language*, Oxford: Basil Blackwell, 1989.

[5] D. Sperber and D. Wilson, *Relevance: Communication and Cognition*, Oxford: Blackwell, 1988.

[6] J. Lyons, *New Horizons in Linguistics*, Harmondsworth: Penguin Books Ltd., 1971.

[7] N. Chomsky, *Aspects of the Theory of Syntax*, Cambridge, Massachusetts: The MIT Press, 1965.

[8] J. Lieber, *Noam Chomsky: A Philosophic Overview*, New York: St. Martin's Press, 1975.

[9] B. L. Liles, *An Introductory Transformational Grammar*, Englewood Cliffs, New Jersey: Prentice-Hall Inc., 1971.

[10] R. Grishman, *Computational Linguistics: An Introduction*, Cambridge: Cambridge University Press, 1986.

[11] D. Wilson, "Relevance and Relevance theory," *MIT Encyclopedia of the Cognitive Sciences*, Cambridge, Massachusetts , The MIT Press, 1999, pp. 719-722.

[12] R. Kager, *Optimality Theory*, Cambridge: Cambridge University Press, 1999.

[13] D. Abercrombie, *Studies in Phonetics and Linguistics*, London: Oxford University Press, 1965.

[14] A. M. Di Sciullo, "Decomposing Compounds," *SKASE Journal of Theoretical Linguistics*, vol. 2-3, pp. 14-33, 2005.

[15] A. M. Di Sciullo, "Asymmetry and the Language Faculty," *Revista Linguistica*, vol. 13, no. 2, pp. 88-107, 2017.

[16] N. Smith and D. Wilson, *Modern Linguistics: The Results of Chomsky's Revolution*, Harmondsworth: Penguin Books Ltd., 1979.

[17] J. Pustejovsky, *The Generative Lexicon*, Cambridge, Massachusetts; London, England: The MIT Press, 1995.

[18] D. A. Reibel and S. A. Schane, *Modern Studies in English: Readings in Transformational Grammar*, Englewood Cliffs, New Jersey: Prentice-Hall Inc., 1969.

[19] R. A. Jacobs and P. S. Rosenbaum, *Readings in English Transformational Grammar*, Waltham, Massachusetts: Ginn and Company, A Xerox Company, 1970.

[20] N. Chomsky, *The Minimalist Program*, Cambridge, Massachusetts; London, England: The MIT Press, 1995.

[21] A. M. Di Sciullo and E. Williams, *On the Definition of Word*, Cambridge, Massachusetts: The MIT Press, 1987.

[22] D. Davidson, "Semantics for Natural Languages," *On Noam Chomsky: Critical Essays*, G. Harman, Ed., New York, Anchor Press, pp. 242-252, 1974.

[23] D. Burton, "Analysing Spoken Discourse," *Studies in Discourse Analysis*, M. Coulthard and M. Montgomery, Eds., London, Routledge and Kegan Paul Ltd., pp. 61-81, 1981.

[24] M. Coulthard, M. Montgomery and D. Brazil, "Developing a Description of Spoken Discourse," *Studies in Discourse Analysis*, M. Coulthard and M. Montgomery, Eds., London, Routledge and Kegan Paul Ltd., pp. 1-50, 1981,.

[25] D. Sperber, F. Clément, C. Heintz, O. Mascaro, H. Mercier, G. Origgi and D. Wilson, "Epistemic Vigilance," *Mind and Language*, vol. 25, no. 4, pp. 359–393, 2010.

[26] D. Mazzarella, *Inferential Pragmatics and Epistemic Vigilance*, London: University College London, 2015.

[27] J. Feller, B. Fitzgerald, S. Hissam and K. R. Lakhani, "Introduction," *Perspectives on Free and Open Source Software*, Cambridge, Massachusetts, The MIT Press, 2005, pp. xxvii-xxxi.

[28] H. T. Le and J. Brook, "Using Praat to Teach Intonation to ESL Students," *Hawaii Pacific University TESOL Working Paper* Series 9, vol. 9, no. 1-2, pp. 2-15, 2011.

[29] H. Buschmeier and M. Włodarczak, "TextGridTools: A TextGrid Processing and Analysis Toolkit for Python," *Proceedings der 27. Konferenz zur Elektronischen Sprachsignalverarbeitung*, Bielefeld, 2013.

[30] P. Boersma and D. Weenink, *Praat: Doing Phonetics by Computer*, 6 June 2020. [Online]. Available: https://www.fon.hum.uva.nl/praat/. [Accessed 6 July 2020].

[31] M. S. Suri, D. Setia and A. Jain, "PRAAT Implementation For Prosody Conversion," *Proceedings of the 4th National Conference; INDIACom-2010; Computing For Nation Development*, New Delhi, 2010.

[32] D. Loakes, "From IPA to Praat and Beyond," *The Oxford Handbook of the History of Linguistics*, Oxford, Oxford University Press, pp. 123-140, 2013.

[33] M. Magdin, T. Sulka, J. Tomanová and M. Vozár, "Voice Analysis Using PRAAT Software and Classification of User Emotional State," *International Journal of Interactive Multimedia & Artificial Intelligence*, vol. 5, no. 6, pp. 33-42, 2019.

[34] P. Boersma, "Optimality-Theoretic Learning in the Praat Program," *Institute of Phonetic Sciences Proceedings*, Amsterdam, 1999.
[35] J. Setter and J. Jenkins, "State-of-the-Art Review Article: Pronunciation," *Language Teaching*, vol. 38, no. 01, pp. 1-17, 2005.
[36] I. Wilson, "Using Praat and Moodle for Teaching Segmental and Suprasegmental Pronunciation," *TESOL Quarterly*, vol. 3, no. 1, pp. 33-43, 2005.
[37] B. Gorjian, A. Hayati and P. Pourkhoni, "Using Praat Software In Teaching Prosodic Features To EFL," *Procedia - Social and Behavioral Sciences*, Izmir, 2013.
[38] P. Boersma and V. v. Heuven, "Speak and unSpeak with PRAAT," *Glot International*, vol. 5, no. 9-10, pp. 341-347, 2001.
[39] J.-P. Goldman and A. C. Simon, "ProsoBox: a Praat Plugin for Analysing Prosody," *Acts of the 10th Speech Prosody Symposium*, Tokyo, 2020.
[40] W. Styler, "Using Praat for Linguistic Research," 25 December 2017. [Online]. Available: http://wstyler.ucsd.edu/praat/UsingPraatforLinguisticResearchLatest.pdf. [Accessed 6 June 2020].
[41] N. Chomsky and M. Halle, *The Sound Pattern of English*. *The First MIT Press Paperback Edition. Third Printing*, Cambridge, Massachusetts: The MIT Press, 1995.
[42] L. Bauer, "When is a Sequence of Two Nouns a Compound in English?," *English Language and Linguistics*, vol. 2, p. 65–86, 1998.
[43] L. Bauer, "Adjectives, Compounds and Words," *Nordic Journal of English Studies*, vol. 3.1, Special Issue: Worlds of Words: A Tribute to Arne Zettersten, p. 7–22, 2004.
[44] I. Plag, "The Variability of Compound Stress in English: Structural, Semantic, and Analogical Factors," *English Language and Linguistics*, vol. 10, no. 1, p. 143–172, 2006.
[45] L. Bauer, *Compounds and Compounding*, Cambridge: Cambridge University Press, 2017.
[46] M. Johnston and F. Busa, "Qualia Structure and the Compositional Interpretation of Compounds," *Proceedings of the ACL SIGLEX Workshop on Breadth and Depth of Semantic Lexicons*, Santa Cruz, California, 1996.
[47] I. Plag, G. Kunter and S. Lappe, "Testing Hypotheses about Compound Stress Assignment in English: A Corpus-Based Investigation," *Corpus Linguistics and Linguistic Theory*, vol. 3, no. 2, pp. 199-232, 2007.
[48] I. Plag, "Compound Stress Assignment by Analogy: The Constituent Family Bias," *Zeitschrift für Sprachwissenschaft*, vol. 29, no. 2, pp. 243-282, 2010.
[49] M. J. Bell and I. Plag, "Informativeness is a Determinant of Compound Stress in English," *Journal of Linguistics*, vol. 48, no. 3, pp. 485-520, 2012.
[50] M. Johnston, B. Boguraev and J. Pustejovsky, "The Acquisition and Interpretation of Complex Nominals," *Working Notes of AAAI Spring Symposium on the Representation and Acquisition of Lexical Knowledge*, Stanford, California, 1995.
[51] L. Polanyi, *Telling the American Story: A Structural and Cultural Analysis of Conversational Storytelling*, Cambridge, Massachusetts: The MIT Press, 1989.
[52] K. Higuchi, "Introduction to KH Coder," 6 June 2020. [Online]. Available: https://khcoder.net/en/. [Accessed 6 7 2020].
[53] S. A. García, N. O. Martínez, T. C. Carabel and I. F. Suárez, "Occupational Accidents and Their Prevention in the Spanish Digital Press," *Revista Latina de Comunicación Social*, vol. 72, pp. 1608-1625, 2017.
[54] K. Takamatsu, Y. Kozaki, A. Kishida, K. Bannaka, K. Mitsunari and Y. Nakata, "Analyzing Students' Course Evaluations Using Text Mining: Visualization of Open-Ended Responses in Co-Occurrence Network," *PEOPLE: International Journal of Social Sciences*, vol. 4, no. 3, pp. 142-153, 2018.
[55] N. Wang, "Potential for Teaching Materials for Advanced Reading Comprehension with the Use of Text Mining: A Report on the Practice of Using KH-Coder," *Senshu University Institute of Humanities Monthly Bulletin*, vol. 304, no. 1, pp. 19-29, 2020.
[56] R. Shineha and M. Tanaka, "Deprivation of Media Attention by Fukushima Daiichi Nuclear Accident: Comparison Between National and Local Newspapers," *Resilience: A New Paradigm of Nuclear Safety - From Accident Mitigation to Resilient Society Facing Extreme Situations*, J. Ahn, F. Guarnieri and K. Furuta, Eds., Cham, Springer International Publishing, pp. 111-125, 2017,.
[57] O. Ylijoki and J. Porras, "Conceptualizing Big Data: Analysis of Case Studies," *Intelligent Systems in Accounting, Finance and Management*, vol. 23, no. 4, pp. 295-310, 2016.
[58] P. Nattuthurai and A. Aryal, "Content Analysis of Dark Net Academic Journals from 2010-2017 Using KH Coder," *ACET Journal of Computer Education and Research*, vol. 12, no. 1, pp. 25-35, 2018.
[59] K. Benoit and A. Herzog, "Text Analysis: Estimating Policy Preferences from Written and Spoken Words," *Analytics, Policy and Governance*, J. Bachner, B. Ginsberg and K. Hill, Eds., New Haven, Connecticut, Yale University Press, pp. 137-159, 2017.
[60] S. Hori, "An Exploratory Analysis Of The Text Mining Of News Articles About "water And Society"," *WIT Transactions on The Built Environment*, vol. 168, pp. 501-508, 2015.
[61] S. Palmer and M. Campbell, " Text Analytics Visualisation of Course Experience Questionnaire Student Comment Data in Science and Technology," *AAEE 2015: Proceedings of the Australasian Association for Engineering Education 2015 Annual Conference*, Geelong, Victoria, 2015.
[62] M. D. Đurić, "Some Aspects of the Discourse Pertaining to Digital Musems of Digital Art," *MELISSA - Museums, Ethics, Library and Information Science, Studies, Archives*, vol. 16, no. 1, pp. 125-146, 2017.
[63] D. Biber, *Variation across Speech and Writing*, Cambdridge: Cambridge University Press, 1995.
[64] C. Krstev, *Processing of Serbian: Automata, Texts and Electronic Dictionaries*, Belgrade: Faculty of Philology, University of Belgrade, 2008.
[65] R. Carston, "Language and Cognition," *Linguistics: The Cambridge Survey, 3: Language: Psychological and Biological Aspects*, Cambridge, Cambridge University Press, pp. 38-68, 1989.
[66] D. Vitas, S. Koeva, C. Krstev and I. Obradović, "Tour du monde through the dictionaries," *Actes du 27eme Colloque International sur le Lexique et la Grammaire*, L'Aquila, 2008.
[67] E. Loper and S. Bird, "NLTK: The Natural Language Toolkit," *Proceedings of the ACL-02 Workshop onEffective tools and methodologies for teaching natural language processing and computational linguist*, Philadelphia, Pennsylvania, 2002.
[68] S.Vijayarani and R.Janani, "Text Minijng: Open Source Tokenization Tools - An Analysis," *Advanced Computational Intelligence: An International Journal (ACII)*, vol. 3, no. 1, pp. 37-47, 2016.
[69] S. Bird, E. Klein, E. Loper and J. Baldridge, "Multidisciplinary Instruction with the Natural Language Toolkit," *Proceedings of the Third Workshop on Issues in Teaching Computational Linguistics (TeachCL-08)*, Columbus, Ohio, 2008.
[70] S. Bird, E. Klein and E. Loper, *Natural Language Processing with Python*, Sebastopol, California & Cambridge: O'Reilly Media, Inc., 2009.
[71] J. Perkins, *Python Text Processing with NLTK 2.0 Cookbook*, Birmingham: Packt Publishing Ltd., 2010.
[72] S. Bromberger, "Types and Tokens in Linguistics," *Reflections on Chomsky*, A. George, Ed., Oxford, Basil Blackwell, pp. 58-89. 1990.
[73] S. Bromberger, *On What We Know We Don't Know: Explanation, Theory, Linguistics, and How Questions Shape Them*, Chicago; London; Stanford: The University of Chicago Press; Center for Study of Language and Information, 1992.
[74] P. H. Matthews, *The Concise Oxford Dictionary of Linguistics*, Oxford: Oxford University Press, 2005.
[75] H. G. Widdowson, *Linguistics*, Oxford: Oxford University Press, 1996.
[76] S. Bird, E. Klein and E. Loper, "Language Processing and Python," 2009. [Online]. Available: http://www.nltk.org/book_1ed/ch01.html. [Accessed 2 August 2020].
[77] D. Crystal, *A First Dictionary of Linguistics and Phonetics*, Second Impression, London: Andre Deutsch, 1983.
[78] F. Palmer, *Grammar*. Second Edition, Harmondsworth: Penguin Books Ltd.,1986.
[79] J. Aitchison, *Cassell's Dictionary of English Grammar*, London: Cassell & Co., 2001.

[80] R. L. Trask, *The Penguin Dictionary of English Grammar*, Harmondsworth: Penguin Group Ltd., 2000.
[81] S. Pinker, *The Language Instinct: How the Mind Creates Language*, New York: HarperPerennial, 1995.
[82] S. Thomas, *Computers: Their History, Present Applications, and Future*, New York: Holt, Rinehart and Winston, Inc., 1965.

Otvorena biblioteka

Andrija Sagić
Biblioteka „Milutin Bojić“, Beograd, Srbija
andrija.sagic@milutinbojic.org.rs

Rezime: U ovom radu ispitaće se značaj upotrebe FOSS alata u bibliotečko-informatičkoj delatnosti. S obzirom da postoje različiti tipovi biblioteka, pažnja će biti usmerena na delatnost javnih biblioteka. Kroz analizu rada javne biblioteke predstaviće se adekvatni FOSS alati koji pokrivaju specifičnu oblast rada biblioteke. Ispitaće se uloga zajednice koja stoji iza razvoja ovakvih tipova softvera i šta to može značiti za dalji razvoj biblioteka.

Ključne reči: FOSS, javna biblioteka, FOSS zajednica, digitalizacija kulturnog nasleđa

I. Uvod – Rad javne biblioteke

U zakonu o bibliotečko-informacionoj delatnosti („Sl. glasnik RS", br. 52/2011) definisani su tipovi biblioteka (Član 13) [1]:

„Biblioteke se prema sadržaju bibliotečko-informacione građe i izvora i prema profilu korisnika razvrstavaju na sledeće tipove: nacionalna biblioteka, javna biblioteka, školska biblioteka, visokoškolska biblioteka, univerzitetska biblioteka, biblioteka naučnoistraživačkog instituta i ustanove, specijalna biblioteka, kao i informacioni centri pri drugim ustanovama, organizacijama ili udruženjima.“ Predmet istraživanja u ovom radu biće javna biblioteka. Osnivač javne biblioteke, prema ovom zakonu, je lokalna samouprava (Član 14). Javna biblioteka je jedina otvorena za sve građane, osim izdavanja knjiga javne biblioteke organizuju kulturne programe, čuvaju i digitalizuju zavičajnu bibliotečku građu (knjige, časopise, štampu, stare fotografije...).

Organizacija rada javne biblioteke može se podeliti u nekoliko segmenata:

- administrativni poslovi
- bibliotečki poslovi
- digitalizacija kulturnog nasleđa
- organizacija kulturnih događaja

U svakom segmentu rada i aktivnosti prisutni su digitalni alati. Kroz opis zadataka svakog segmenata, predstaviće se FOSS (Free Open Source Software) alati kojima se mogu efikasno obavljati zadaci svakog navedenog segmenta rada javne biblioteke. Cilj ovog rada je da ispita mogućnost da li javna ustanova može svoju softversku infrastrukturu formirati koristeći FOSS.

II. Administrativni poslovi

U administrativne poslove spadaju računovodstvo, pravna služba, poslovna korespondencija, komunikacija sa lokalnim i državnim službama. Digitalni alati koji su potrebni za efikasno obavljanje navedenih poslova mogu se navesti ptema finkcijama, a to su: obrada teksta i tebela (takozvani Office paket), pregled i slanje elektronske pošte, internet pretraživač, planiranje resursa.

Programi za obradu teksta i tabela

Libre Office [2] je jedan od najzastupljenijih programa u FOSS zajednici i nalazi se u gotovo svim desktop Linux distribucijama. Sadrži sve neophodne alate za kancelarijsku upotrebu, program za obradu teksta (Writer), program za rad sa tabelama (Calc), a u okviru ovog paketa nalaze se još i programi za rad sa bazama podataka (Base), osnovne grafičke obrade (Draw), pripremu prezentacija, slajdova i veb stranica (Impress). Kompatibilan je sa svim popularnim formatima (npr. doc, xsl, ppt…) dok je osnovni format u kome se radi odf (eng. *Open Document Format*) [3]. Osnovni problem rada sa Libre Office paketom programa javlja se prilikom učitavanja nekih fajlova iz Windows-ovog Office paketa, ne dobija se identičan fajl prilikom otvaranja, npr. tabele u .doc dokumentu se ne otvaraju na željeni način, postoji razlika u fontovima, ponekad je i broj stranica različit. Ovaj problem je vidljiv prilikom preuzimanja konkursne dokumentacije za projekte Ministarstva Kulture i Informisanja Republike Srbije. U svetu postoji inicijativa da se omogući i odf standard prilikom preuzimanja zvanične dokumentacije. Libre Office paket programa moguće je instalirati na Linux, macOS i Windows sistemima. Objavljen je pod licencom Mozilla Public Licence v2. [4]

Program za poslovnu korespodenciju

Mozilla Thunderbird [5] je program namenjen radu sa elektronskom poštom i može se instalirati na svim desktop platformama. Rad u ovom programu je jednostavan a nudi dosta mogućnosti, jedna od najvažnijih koju treba pomenuti jeste integrisana podrška za OpenPGP [6] enkripciju pošte. Sva poslovna korespodencija se mora osigurati na neki način, a posebno kada se šalje važna dokumentacija. Mejl serveri među sobom komuniciraju enkriptovanom vezom, ali kada mi otvaramo poštu u našem programu ova veza nije osigurana pa postoji opasnost da se poverljivi podaci preuzmu. Komunikacija preko OpenPGP protokola oneomogućava pristup trećem licu da ima uvid u prepisku.

Thunderbird je objavljen pod Mozilla Public Licence v2. OpenPGP je IETF Proposed Standard RFC 4880 [7] protokol za enkripciju e-mail komunikacije.

Programi za računovodstvo

ERP (eng. *Enterprise Resource Planning*) sistemi, kao što je SAP [8], postoje i u FOSS verziji. Ovakav softver olakšava rad knjigovodstva i pravne službe. Axelor ERP [9] sadrži više od 20 integrisanih poslovnih aplikacija, među kojima su rad sa dobavljačima (telekomunikacija, struja, grejanje, oprema...), računovodstveni poslovi, ljudski resursi, inventar osnovnih sredstava, itd. Softver je objavljen pod GNU Affero General Public License v3.0 [10].

Rad sa državnim organima

Digitalizacija rada državnih službi olakšala je rad upravi biblilioteka (prijava/odjava zaposlenih, uplata i isplata, podnošenje završnih računa…) ali problem se javlja u podržanim sistemima – sve aplikacije za komunikaciju rade isključivo u Windows okruženju! Neke institucije čak zahtevaju pristup preko programa Internet Explorer. Ovo je

DOI: 10.5281/zenodo.4077400

jedini segment koji se ne može obavljati preko Linux distribucija.

III. Bibliotečki poslovi

U bibliotečke poslove spadaju nabavka publikacija, bibliotečka obrada (izrada metapodataka) i cirkulacija fonda (zaduženje građe), a softveri za ovu namenu zovu se ILS (eng. *Integrated Library System*).

Biblioteka „Milutib Bojić“ koristi sistem Bibliotečki informacioni sistem BISIS v5 [11] koji je razvijen na Fakultetu Tehničkih Nauka Univerziteta u Novom Sadu i objavljen je pod GNU General Public License v3.0. [12]

Katalogizacija u ovom sistemu moguća je putem uzajamnog kataloga preuzimanjem zapisa već obrađene građe, unošenjem novih zapisa, vođenje inventarnih knjiga za svako odeljenje i posebnih kolekcija, kao i povezivanje zapisa sa digitalnim izdanjem. Korisnici se unose u bazu preko koje se evidentira zaduženje građe (cirkulacija). Na ovaj način vidljivo je trenutno stanje fonda, zna se ko je zadužio publikaciju, do kog vremena je zadužena, prati se stanje članarine, mogu se štampati izveštaji na dnevnom, nedeljnom ili mesečnom nivu. Za korisnike postoji javna aplikacija, katalog kome se pristupa preko veb adrese https://opac.bisis.rs/lib/bmb .

Biblioteka „Milutin Bojić“ uvela je u upotrebu još tri zasebna javna kataloga:

- Katalog knjiga na stranim jezicima https://bookscat.milutinbojic.org.rs
- Katalog posebnih kolekcija https://kolekcije.milutinbojic.org.rs
- Katalog legata https://legati.milutinbojic.org.rs

Prva tri kataloga urađena su na platformi Librarian DB [13] koji je objavljen pod GNU GPL licencom. Ova platforma daje tabelarni pregled svih publikacija koje se mogu klasifikovati. Baza je smeštena u „SQL“ dok se komunikacija sa bazom vrši preko „PHP“ programskog jezika. U ovom sistemu vidljiva je i cirkulacija, ali samo na nivou zaduženosti. Katalog knjiga na stranim jezicima kao opciju nudi i prikaz na više stranih jezika (srpski, engleski, francuski, italijanski, španski i poljski). Postoje i instrukcije za pretragu koje olakšavaju korisniku da brže dođe do željenog rezultata.

Treći katalog urađen je na platformi SLiMS (eng. *Senayan Library Management System*) [14] koji je takođe objavljen pod GNU GPL licencom. Ima unapređen prikaz publikacija, opis je dostupan u MODS (eng. *Metadata Object Description Schema*) standardu [15] i JSON-LD (JavaScript Object Notation for Linking Data) [16] formatu. Svaka publikacija predstavljena je sa naslovnom koricom. Katalog je pretraživ po više parametara, velikim slovima su navedene opcije za pretragu:

BRZA PRETRAGA ujedno i najjednostavniji način pretrage, potrebno je uneti ključnu reč koja se nalazi u naslovu ili ime autora ili predmetnu odrenicu. Može se uneti i više ključnih reči čime se dobijaju precizniji rezultati.

DETALJNA PRETRAGA omogućava pretragu podataka po više parametara. Unesom ključne reči u određena polja (Naslov, Autor...) dobiju se rezultati prema definisanim specifičnim parametrima.

IV. Digitalizacija kulturnog nasleđa

Proces digitalizacije može se predstaviti kroz faze:

- prevođenje građe u digitalni format
- obrada digitalizovane građe
- publikovanje na internetu

Način prevođenja u digitalni format zavisi od vrste građe. Za potrebe ovog rada biće dovoljno samo nabrojati vrstu opreme i pomenuti neke programe koji će dovoljno ilustrovati FOSS rešenja. Sredstva za digitalizaciju kao što su skeneri, foto-aparati, kamere, audio i video oprema imaju podršku i u Linux okruženju.

Skeneri sa svetlosnom refleksijom su najzastupljeniji u bibliotekama jer je cena pristupačna, a podržavaju formate do A3. Svi skeneri su „vidljivi“ u Linuksu preko softverske podrške otvorenog koda SANE „Scanner Access Now Easy" [17] koja je publikovana pod GNU GPL licencom. SANE je zapravo API (eng. *Application Programming Interface*) odnosno skup jasno definisanih metoda komunikacije između različitih komponenti softvera koji omogućava standarizovani pristup bilo kom skeneru (flet skener, ručni skener, video kamere, film skeneri, itd). Rad sa skenerom može biti u grafičkom okruženju ili preko terminala. U grafičkom okruženju može se odlično raditi u programu Xsane [18] koji je objavljen pod GNU GPL licencom a moguće je instalirati na svim platformama. Građa koja se može digitalizovati na ovakvim skenerima: monografske i serijske publikacije, rukopisi, fotografije, foto negativi.

Foto aparati su pogodni za digitalizaciju većih formata, umetnina, stare rukopisne i štampane građe. Kvalitet dobijenog materijala zavisi od kvaliteta optike i samog aparata, dok je softverska podrška neophodna za kvalitetan prenos informacija na računar.

Odličnu podršku za daljinsko upravljanje foto aparatom nudi program darktable [19] koji sadrži kompletan proces fotografisanja, od kontole aparata preko samog slikanja do obrade foto materijala kroz tri programska rešenja koja kontrolišu svaki proces, kao što je obrada negativa i fotografija. Ovaj program objavljen je pod GNU GPL licencom i podržava Linux i Windows platformu.

Audio zapisi mogu biti zapisani na gramofonskim pločama, magnetofonskim trakama, kasetama ili voštanim cilindrima. Kvalitet reprodukcije zavisi od kvaliteta opreme sa koje se reprodukuje dok je za digitalizaciju ovakvih zapisa potrebna kvalitetna zvučna karta i/ili analogno – digitalni konverter. Jedan od najpoznatijih programa za snimanje i obradu audio zapisa je Audacity [20], u ovom programu moguće je obavljati sve neophodne zadatke, snimanje, filtriranje od šumova i pucketanja ploča, podešavanja nivoa i frekvencija. Objavljen je pod GNU General Public License v2.0 licencom i podržava sve platforme.

Video zapisi se mogu naći na filmskim trakama, VHS, BETA, Video8, DV i miniDV trakama. Ovaj tip digitalizacije je najkompleksniji i zahteva specijalizovanu opremu za određene formate uz koju dolazi fabrički softver koji se mogu instalirati na Linux računarima.

Obrada digitalizovanog materijala svodi se na tri formata: slika, zvuk i video.

Za obradu slika najviše se koristi GIMP [21] što je skraćenica od GNU *Image Manipulation Program*. Ovaj program se najviše koristi u FOSS zajednici, ima odlično grafičko okruženje, podržava dosta formata i daje odlične

rezultate. Objavljen je pod GNU GPL licencom i podržava sve platforme. Još jedan program koji vredi pomenuti jeste Image Magick [22], rad u ovom programu je uglavnom preko terminala, može koristiti za konverziju većeg broja fajlova. Scan Tailor [23] je još jedan program koji služi za obradu digitalizovane bibliotečke građe, sadrži korekcije iskrivljenih strana, margina, može podeliti strane ukoliko je urađen dvostrani sken, rotacija strana (u svim pravcima po 90 stepeni) i izbor sadržaja koji omogućava definisanje krajnje veličine obrađenog materijala. Objavljen je pod GNU GPL licencom. Postoje neke verzije ovog programa sa dodantim funkcijama, jedna od njih je Scan Tailor Advanced [24]. Program za obradu teksta Tesseract Open Source OCR Engine [25]. OCR (eng. *Optical Character Recognition* – Optičko Prepoznavanje Karaktera) je jedan od najvažnijih procesa u digitalizaciji tekstualne građe jer omogućava pretragu u tekstu. Program Tesseract je jedan od najefikasnijih alata koji daje odlične rezultate u obradi ćiriličnih tekstova. Objavljen je pod GNU GPL licencom.

Za obradu zvuka pored programa Audacity postoji i razvijenije rešenje Ardour [26] koje spada u grupu DAW (eng. *Digital Audio Workstation*) programa. Njagova prednost u odnosu na Audacity je u većoj ponudi dodataka (eng. *plug in*) za obradu audio materijala. Program je objavljen pod GNU GPL licencom i podržava sve platforme.

Za obradu video materijala dobre rezultate ima program Shotcut [27]. Program je objavljen pod GNU GPL licencom i podržava sve platforme.

Publikovanje na internetu je završni proces digitalizacije. Vrsta i količina informacija koja se plasira prevazilazi običan pregled slika uz kataloški opis. Sada je moguće pretraživati tekst i zvučni zapis, filtrirati umetničke slike, analizirati rukopise pregledom detalja i upoređivati različite verzije su samo neke od mogućnosti koje digitalizacija omogućava krajnjem korisniku.

Biblioteka „Milutin Bojić“ je svoju zavičajnu kolekciju prilagodila savremenim standardima prikaza digitalizovane građe, na adresi https://zavicajna.digitalna.rs dostupna je potpuno tekstualno pretraživa građa, jednostavnim unosom pojma ili samo tri početna slova, dobija se jedna ili više publikacija sa traženim pojmom, klikom na publikaciju traženi pojam se markira na određenoj strani u toj publikaciji, a moguće je pretraživati i celu frazu. Građa je pripremana i publikovana u programu The ResCarta Toolkit [28] koji sadrži kompletan set alata za obradu digitalizovane građe, sadrži odvojene aplikacije za unos metapodataka u xml formatu (Extensible Markup Language) [29], OCR (Optical character recognition), korekciju dobijenog teksta, korekciju transkribovanog zvučnog materijala, administraciju kolekcija sa mogućnošću izvoza podataka u Dublin Core standard, indeksaciju, proveru verodostojnosti (eng. *checksum*) i veb aplikacije. Ovaj skup alata se lako instalira i koristi, objavljen je pod Apache v2 [30] licencom, a dostupan je za Linux, macOS i Windows sisteme.

Digitalni repozitorijum biblioteke koji je dostupan na adresi https://milutinbojic.digitalna.rs baziran je na modularnom sistemu ISLANDORA [30] koji se sastoji od sledećih komponenti:

FEDORA (eng. *Flexible Extensible Digital Object Repository Architecture*) [31] je modularna platforma koja nudi pristup velikim i kompleksnim digitalizovanim materijalima, kolekcijama i naučnim materijalima, prema zvanično registrovanim korisnicima u upotrebi je u preko 300 institucija u 35 zemalja. Za rad sa ovim repozitorijumom potrebno je frontend rešenje.

Drupal CMS (eng. *Content Management System*) [32], sistem sa dostupnim modularnim rešenjima za određene vrste kolekcija (knjige, serijske publikacije, rukopisna građa…) to su aplikacije za prikaz, administracija je jednostavna a nudi dovoljno alata (obrada metapodataka, konfiguraciju pretrage, OCR, selekciju pristupa građi…).

Za ovaj vid predstavljanja građe neophodna je dobra hardverska infrastruktura: server, skladišni prostor, internet veza sa dovoljnom upload brzinom. Srećom, razvoj tehnologije i relativno niska cena komponenata omogućava čak i manjim bibliotekama da osposobe kvalitetni sopstveni digitalni sistem bez mnogo sredstava. Biblioteka „Milutin Bojić“ je uspela da obezbedi kompletan serverski sistem za digitalne sadržaje, optičku mrežu kapaciteta 20/20 Mb/s, samostalno napajanje i stabilizaciju mreže.

U serverskoj infrastrukturi najvažnije je konfigurisati stabilan i bezbedan sistem. CentOS [33] je jedan od najzastupljenijih otvorenih serverskih operativnih sistema, baziran je na otvorenom kodu Linux distribucije RedHat [34], ovaj sistem je stabilan, sa podrškom od deset godina i garantuje rad svih serverskih funkcija prilikom svakog unapređenja sistema.

Veoma je važno da se digitalizovana građa nalazi u vlasništvu institucije koja objavljuje digitalizovanu građu i da ima kontrolu nad celim sistemom. Razmotrimo softverska rešenja u serverskoj infrastrukturi i još neka rešenja za objavljivanje.

Važan segment publikovanja je i implementacija https protokola koja obezbeđuje enkriptovanu vezu veb stranice i korisnika, Linux Fondacija Error: Reference source not found razvila je otvoreno rešenje Let’s Encrypt [36] koje obezbeđuje enkriptovanu vezu, a može se instalirati na svim Linux serverskim sistemima.

Pored FOSS alata neophodno je pomenuti i Otvorene standarde interoperabilnosti OAI [37] i IIIF [38]. Inicijativa OAI razvila je protokol za deljenje metapodataka OAI-PMH [39] koji koriste sve relevantne naučne i kulturne ustanove. Za javne biblioteke ovakav protokol omogućava bolju vidljivst digitalnog kulturnog nasleđa. Digitalni sadržaji biblioteke „Milutin Bojić“ vidljivi su u katalogu worldcat [40] i u agregatoru CORE [41].

IIIF (eng. *International Image Interoperability Framework*) definiše veb prikaz digitalizovanog kulturnog nasleđa u API formi. Na ovaj način moguće je implementirati pretraživ tekstualni sloj, uporedni prikaz digitalnih objekata iz različitih digitalnih repozitorijuma u jednom pregledaču, koristiti specijalizovane pregledače sa dodatnim funkcijama za manipulaciju slika (kontrast, osvetljenje...), dodavati beleške, itd. Portal https://iiif.digitalna.rs nastao je kao inicijativa biblioteke „Milutin Bojić“ za implementaciju IIIF standarda u prikazu digitalizovanog kulturnog nasleđa u Srbiji, za sada su u ovom formatu dostupne kolekcije Biblioteke grada Beograda i Biblioteke „Milutin Bojić“. Portal IIIF Srbija ima integrisan IIIF pregledač Mirador v2 [42], video uputstvo i linkove prema IIIF kolekcijama velikih svetskih biblioteka koji se mogu uporedno pregledati na jednom mestu. Razvijeno je dosta FOSS alata za prikaz građe u IIIF standardu koji su navedeni na linku https://iiif.io/apps-demos. Razvoj IIIF standarda odvija se u zajednici koja je organizovana po principu FOSS zajednice.

V. ORGANIZACIJA KULTURNIH DOGAĐAJA

Jedna od delatnosti javnih biblioteka su i kulturni programi (promocije, tribine, predavanja, koncerti…) koji mogu biti propraćeni nekim vizuelnim materijalima poput ilustracija, video materijala i prezentacija. Sva dekstop okruženja na Linux distribucijama imaju svoje programe otvorenog koda za pregled fotografija i video materijala, PDF čitače dok je prezentacije moguće pregledati u standardnom OFIS paketu poput LibreOffice. Biblioteka „Milutin Bojić“ sve svoje programe prikazuje i uživo na svom Youtube kanalu. Softver OBS Studio (eng. *Open Broadcaster Software*) [43] veoma dobro može da manipuliše sa nekoliko kamera, dodatnim izvorima (softveri na računaru, dodatni izvor zvuka...). Sve ove funkcije se efikasno strimuju preko izabranog striming servisa YouTube, Twich. Softver OBS Studio objavljen je pod GPLv2 lincencom.

VI. FOSS ZAJEDNICA

Iza svakog FOSS alata stoji zajednica koja radi na razvoju i otklanjanju problema, zajednica ima svoju strukturu koju možemo najosnovnije prikazati ovako:

- koordinacija rada i odlučivanje o novim funkcijama alata
- interesne grupe (stakeholder) koje se bave određenom vrstom razvoja i namene
- testiranje koda.

Autor ovog teksta trenutno je uključen u rad zajednice koja radi na razvoju repozitorijuma ISLANDORA, IIIF zajednice i Mirador zajednice.

Ovakav princip rada je dobar način da se ljudi različitih obrazovnih profila okupi oko nekog projekta i da se kroz zajednički rad dođe do softverskog rešenja, u ovom slučaju nekog alata, koje će biti prilagođeno specifičnim potrebama korisnika.

VII. ZAKLJUČAK

Prikazan je delimičan spisak FOSS alata koji mogu biti korisni javnim bibliotekama. Takvi alati nisu samo besplatno rešenje kojim se može uštedeti već su vrlo funkcionalni i neophodni za razvoj bibliotečko – informacione delatnosti. Značaj upotrebe ovakvih alata, za nas u Srbiji, nije samo mogućnost adaptacije softvera već je pre svega priključenje svetskoj zajednici kroz razvoj i implementaciju standarda i rešenja. Potpuna kontrola sistema u svim segmentima korišćenja je stvar koja je od ključnog značaja za zaštitu kulturnog nasleđa a to je moguće samo uz pomoć FOSS alata.

Ovo je takođe i prilika da naša zajednica postane vidljiva u svetu i u segmentu razvoja alata i standarda za digitalizaciju a u bibliotečkoj delatnosti razvije specifične sisteme koji mogu biti svima od koristi. Biblioteka „Milutin Bojić“ vidljiva je na mapama Islandora [44], IIIF [45], ResCarta Toolkit [46]. Mirador v3 [47] sada ima zvaničan prevod na srpski jezik, Islandora8 u default verziji ima srpski jezik kao ponudu za OCR teksta [48].

Reference

[1] Pristupljeno: 27.09.2020. [Online]. Dostupno na adresi: https://www.kultura.gov.rs/extfile/sr/4841/-zakon-o-bibliotecko-informacionoj-delatnosti.doc

[2] Pristupljeno: 23.08.2020. [Online]. Dostupno na adresi: https://www.libreoffice.org

[3] Pristupljeno: 27.09.2020. [Online]. Dostupno na adresi: http://opendocumentformat.org

[4] Pristupljeno: 26.09.2020. [Online]. Dostupno na adresi: https://www.libreoffice.org/download/license/

[5] Pristupljeno: 23.08.2020. [Online]. Dostupno na adresi: https://www.thunderbird.net

[6] Pristupljeno: 26.09.2020. [Online]. Dostupno na adresi: https://www.openpgp.org

[7] Pristupljeno: 26.09.2020. [Online]. Dostupno na adresi: https://tools.ietf.org/html/rfc4880

[8] Pristupljeno: 27.09.2020. [Online]. Dostupno na adresi: https://www.sap.com/westbalkans/index.html

[9] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://www.axelor.com/erp

[10] Pristupljeno: 26.09.2020. [Online]. Dostupno na adresi: http://www.gnu.org/licenses/agpl-3.0.html

[11] Pristupljeno: 23.08.2020. [Online]. Dostupno na adresi: Pristupljeno: https://bisis.rs

[12] Pristupljeno: 26.09.2020. [Online]. Dostupno na adresi: http://www.gnu.org/licenses/gpl-3.0.html

[13] Pristupljeno: 23.08.2020. [Online]. Dostupno na adresi: Pristupljeno: https://sourceforge.net/projects/librariandb

[14] Pristupljeno: 23.08.2020. [Online]. Dostupno na adresi: Pristupljeno: https://slims.web.id/web

[15] Pristupljeno: 23.08.2020. [Online]. Dostupno na adresi: Pristupljeno: https://www.loc.gov/standards/mods

[16] Pristupljeno: 26.09.2020. [Online]. Dostupno na adresi: https://json-ld.org

[17] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: http://sane-project.org

[18] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: http://www.fifi.org/doc/xsane/html/sane-xsane-doc.html

[19] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: https://www.darktable.org

[20] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: https://www.audacityteam.org

[21] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: https://www.gimp.org

[22] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: https://imagemagick.org/index.php

[23] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: http://scantailor.org

[24] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: https://github.com/4lex4/scantailor-advanced

[25] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: https://github.com/tesseract-ocr/tesseract

[26] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: http://ardour.org

[27] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: https://www.shotcut.org

[28] Pristupljeno: 24.08.2020. [Online]. Dostupno na adresi: https://www.rescarta.org

[29] Pristupljeno: 26.09.2020. [Online]. Dostupno na adresi: https://www.w3.org/XML

[30] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://islandora.ca

[31] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://duraspace.org/fedora

[32] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://www.drupal.org

[33] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://www.centos.org

[34] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://www.redhat.com/en

[35] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://www.linuxfoundation.org

[36] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://letsencrypt.org

[37] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://www.openarchives.org

[38] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://iiif.io
[39] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://www.openarchives.org/pmh
[40] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://www.worldcat.org
[41] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://core.ac.uk
[42] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://projectmirador.org
[43] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://obsproject.com
[44] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://islandora.ca/islandora-installations
[45] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://iiif.io/community/consortium
[46] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://rescarta.org/index.php/sw/rescarta-web/web-links?task=weblink.go&id=24
[47] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://mirador-dev.netlify.app/__tests__/integration/mirador/
[48] Pristupljeno: 25.08.2020. [Online]. Dostupno na adresi: https://github.com/Islandora-Devops/ansible-role-crayfish/blob/main/vars/RedHat.yml

Dodatak 1
Tabela navedenih programa

Program	**Upotreba**	**Licenca**
Libre Office	U svim segmentima	Mozilla Public Licence v2
Mozilla Thunderbird	Administracija, bibliotečki poslovi	Mozilla Public Licence v2
Axelor ERP	Administracija	GNU Affero General Public License v3.0
BISIS v5	Bibliotečki poslovi, Digitalizacija kulturnog nasleđa	GNU General Public License v3.0
Librarian DB	Bibliotečki poslovi	GNU General Public License v2.0
SLiMS	Bibliotečki poslovi	GNU General Public License v3.0
Xsane	U svim segmentima	GNU General Public License v2.0
darktable	Digitalizacija kulturnog nasleđa	GNU General Public License v3.0
Audacity	Digitalizacija kulturnog nasleđa	GNU General Public License v2.0
GIMP	Digitalizacija kulturnog nasleđa	GNU General Public License v3.0
Image Magick	Digitalizacija kulturnog nasleđa	GNU General Public License v3.0
Scan Tailor	Digitalizacija kulturnog nasleđa	GNU General Public License v3.0
Scan Tailor Advanced	Digitalizacija kulturnog nasleđa	GNU General Public License v3.0
Tesseract	Digitalizacija kulturnog nasleđa	Apache License 2.0
Ardour	Digitalizacija kulturnog nasleđa	GNU General Public License v2.0
Shotcut	Digitalizacija kulturnog nasleđa	GNU General Public License v3.0
ResCarta Toolkit	Digitalizacija kulturnog nasleđa	Apache License 2.0
Islandora 7	Digitalizacija kulturnog nasleđa	GNU General Public License v3.0
Fedora repo	Digitalizacija kulturnog nasleđa	Apache License 2.0
Drupal CMS	Digitalizacija kulturnog nasleđa	GNU General Public License v2.0
CentOS	Digitalizacija kulturnog nasleđa	GNU General Public License v2.0
Mirador	Digitalizacija kulturnog nasleđa	Apache License 2.0
OBS Studio	Organizacija kulturnih događaja	GNU General Public License v2.0

Dodatak 2
Tabela dozvola, uslova i ograničenja korišćenja softvera u korišćenju i modifikovanju softvera

Naziv licence	**Dozvole**	**Uslovi**	**Ograničenja**
Mozilla Public Licence v2	Komercijalna upotreba Distribucija Modifikacija Upotreba patenta Lična upotreba	Objaviti kod Naznaka o licenci i kopirajtu Ista licenca (na nivou fajla)	Od odgovornosti Upotrebe zaštitnog znaka Garancije
GNU General Public License v2.0	Komercijalna upotreba Distribucija Modifikacija Lična upotreba	Objaviti kod Naznaka o licenci i kopirajtu Ista licenca Dokumentovanje promena	Od odgovornosti Garancije
GNU General Public License v3.0	Komercijalna upotreba Distribucija Modifikacija Upotreba patenta Lična upotreba	Objaviti kod Naznaka o licenci i kopirajtu Ista licenca Dokumentovanje promena	Od odgovornosti Garancije
GNU Affero General Public License v3.0	Komercijalna upotreba Distribucija Modifikacija Upotreba patenta Lična upotreba	Objaviti kod Naznaka o licenci i kopirajtu Mrežna upotreba je distribucija Ista licenca Dokumentovanje promena	Od odgovornosti Garancije
Apache License 2.0	Komercijalna upotreba Distribucija Modifikacija Upotreba patenta Lična upotreba	Naznaka o licenci i kopirajtu Dokumentovanje promena	Od odgovornosti Garancije

Objašnjenja:
Komercijalna upotreba – Licencirani materijal i derivati mogu se koristiti u komercijalne svrhe
Distribucija – Licencirani materijal može se dalje distribuirati
Modifikacija - Licencirani materijal može se modifikovati
Upotreba patenta – Izričito se odobravaju patentna prava
Lična upotreba – Licencirani materijal može se modifikovati i koristiti lično
Objaviti kod – Izvorni kod mora biti dostupan prilikom distribuiranja licenciranog materijala
Naznaka o licenci i kopirajtu – Kopija licence i naznaka kopirajta moraju biti uključeni u licencirani materijal
Ista licenca – Modifikacije moraju da se objave pod istiom licencom prilikom dalje distribucije licenciranog materijala. U nekim slučajevima može se koristiti slična licenca
Ista licenca (na nivu fajla) – Modifikacija postojećih fajlova moraju da se objave pod istom licencom prilikom dalje distribucije licenciranog materijala. U nekim slučajevima može se koristiti slična licenca, ili u slučaju kada se licencirani materijal koristi kao biblioteka ovaj uslov se ne mora primeniti.
Dokumentovanje promena – Mora se dokumentovati svaka promena u licenciranom materijalu.
Od odgovornosti – Licenca uključuje ograničenje od odgovornosti
Upotreba zaštitnog znaka – Ova licenca eksplicitno naglašava da NE PRENOSI prava na zaštitni znak, čak iako licence bez ovakve odredbe verovatno ne prenose bilo kakva implicitna prava na zaštitni znak.
Garancija – Ova licenca eksplicitno naglašava da NE daje nikakvu garanciju.

Izvor: https://choosealicense.com/licenses/

Slobodni i otvoreni softver u laboratorijskim i mrežnim istraživanjima u eksperimentalnoj psihologiji

Dušica Filipović Đurđević
Odeljenje za psihologiju, Filozofski fakultet, Univerzitet u Beogradu, Beograd, Srbija
Laboratorija za eksperimentalnu psihologiju, Filozofski fakultet, Univerzitet u Beogradu, Srbija
Laboratorija za eksperimentalnu psihologiju, Filozofski fakultet, Univerzitet u Novom Sadu, Srbija
dusica.djurdjevic@f.bg.ac.rs

Rezime: U ovom radu biće dat kratak prikaz najpoznatijih slobodnih i otvorenih softverskih aplikacija koje se koriste za prikazivanje stimulusa i prikupljanje podataka u eksperimentalno-psihološkim istraživanjima. Pored sažetog opisa osnovnih tehničkih preduslova koji treba da budu ispunjeni u eksperimentima u kognitivnoj psihologiji, biće dat uporedni prikaz stepena u kojem su oni ispunjeni u slučaju slobodnih i otvorenih softverskih rešenja. Konačno, biće upoređeni specifični tehnički preduslovi koje postavlja prikupljanje podataka preko mreže, a sve sa ciljem demonstriranja da sve brojnija i sve kvalitetnija kako slobodna, tako i otvorena softverska rešenja ispunjavaju stroge kriterijume savremenog eksperimentalnog istraživanja i velikom brzinom se prilagođavaju promenama i novonastalim potrebama savremenog istraživača.

Ključne reči: eksperiment, prikupljanje podataka, laboratorija, internet.

I. Uvod

U ovom radu bavićemo se softverom koji se koristi u eksperimentalnoj psihologiji – grani psihologije koju definiše metod koji koristi, a koja za temu može imati različite sfere ljudskog ponašanja, pa tako postoje eksperimentalna kognitivna, socijalna, razvojna, klinička psihologija itd. U današnje vreme upotreba eksperimentalnog softvera olakšava istraživački proces u svim pomenutim granama, jer olakšava proces izlaganja stimulusa koji čine deo eksperimentalne manipulacije i proces beleženja odgovora ispitanika. Međutim, najraniju primenu softver je našao u oblasti kognitivne psihologije, naročito u oblasti percepcije.

Neposredna potreba za računarski podržanim istraživanjima javila se u oblastima psihologije u kojoj je izlaganje stimulusa, s jedne i beleženje odgovora ispitanika s druge strane, bilo naročito zahtevno. Izlaganje stimulusa koje je zahtevalo tehničku podršku podrazumevalo je izlaganje komplikovanih stimulusa, ili izlaganje u veoma kratkim i precizno izmerenim vremenskim intervalima. Snimanje odgovora podrazumevalo je ne samo beleženje odgovora, već i precizno merenje vremena reagovanja. Pored toga, eksperimentalni nacrt ponekad bi podrazumevao i modifikovanje procesa izlaganja stimulusa u zavisnosti od odgovora ispitanika.

Precizna kontrola izlaganja stimulusa i merenja parametara odgovora ispitanika bila je moguća i pre nego što su računari postali dostupni široj javnosti i to zahvaljujući uređaju pod nazivom tahistoskop. Međutim, upravljanje tahistoskopom je bilo veoma zahtevno i proces izvođenja istraživanja veoma spor, a mogućnosti izlaganja stimulusa ograničene. Uprkos divljenju vrednoj posvećenosti i dovitljivosti koju su ispoljili istraživači iz predračunarske ere, primena računarske tehnologije donela je revolucionarnu promenu eksperimentalnoj psihologiji.

II. Pregled najčešće korišćenog softvera

Ovde će najpre biti opisan odabrani uzorak najčešće korišćenog slobodnog i otvorenog softvera koji se koristi u laboratorijskim uslovima, a potom i odabrani uzorak takvog softvera koji se koristi za sve popularnije prikupljanje podataka preko interneta.

A. Istraživanja koja se izvode u laboratorijskim uslovima

Eksperimentalni psiholozi za potrebe svojih istraživanja već decenijama koriste različite komercijalne softverske alate. Neki od najpoznatijih su **E-Prime**[1] [1], **Presentation**[2] [2], **SuperLab**[3] [3], **Experiment Builder**[4] [4], **Inquisite**[5] [5], **DirectRT**[6] [6]. Premda su njihove performanse bile i ostale na zavidnom nivou, licence nisu uvek bile dostupne svima, naročito istraživačima iz zemalja sa manje razvijenom ekonomijom. Pored toga, kao i većina komercijalnih softverskih aplikacija, mnogi od ovih programa patili su od nefleksibilnosti – ukoliko kreatori softvera nisu osmislili neku mogućnost, korisnik nije mogao da je doda. Eksplicitna poređenja su pokazala da svaki od njih ima neke prednosti i neke mane – npr. neki su fleksibilni, ali komplikovani za upotrebu, dok su neki drugi jednostavni za upotrebu, ali neprilagodljivi različitim potrebama [7]. Zbog toga su istraživači rano počeli da kreiraju slobodan softver za izvođenje eksperimenata u oblasti kognitivne psihologije.

Jedna od prvih slobodnih aplikacija bio je **DMDX**[7] [8], program koji je Kenet Forster počeo da razvija 1975. godine. Krajem osamdesetih godina pridružio mu se Džonatan Forster koji ga je krajem devedesetih prilagodio za rad u OS Windows. Program se redovno ažurira i postao je veoma popularan u istraživačkoj zajednici kognitivnih psihologa, prvenstveno zahvaljujući preciznosti u merenju vremenskih parametara. Naročito je cenjen od istraživača koji koriste paradigmu takozvanog maskiranog primovanja [9], gde je stimuluse potrebno izlagati u vrlo kratkim intervalima (desetine milisekundi). Jedan je od prvih programa koji je kognitivne psihologe upoznao sa pojmom vremena osvežavanja ekrana i jedan od retkih programa koji uz

1 https://pstnet.com/products/e-prime/
2 https://www.neurobs.com/
3 https://cedrus.com/superlab/index.htm
4 https://www.sr-research.com/experiment-builder/
5 https://www.millisecond.com/
6 https://www.empirisoft.com/directrt.aspx
7 http://www.u.arizona.edu/%7Ejforster/dmdx.htm

DOI: 10.5281/zenodo.4064302

instaliranje zahteva dodatno testiranje hardvera kako bi se omogućila kalibracija određenih parametara, što istraživačima daje naročitu sigurnost i preciznost merenja, ali i nelagodu zbog relativno komplikovane procedure inicijalnih testova. Eksperiment se kreira pomoću tekstualnog fajla, a nedavno je ponuđen i grafički interfejs koji olakšava ovu proceduru, **VisualDMDX**[8] [10].

PsychoPy[9] [11], [12], [13], [14] je trenutno najpopularniji i najnapredniji otvoreni softver za izvođenje eksperimenata u psihologiji i srodnim naukama, na tri platforme (Windows, Linux i macOS), koja mesečno ima na desetine hiljada korisnika i koju je do sada gradilo preko 90 volontera. Preciznije govoreći, potrebno je razlikovati *PsychoPy paket*, odnosno biblioteku koja je pisana u programskom jeziku Python i *PsychoPy Builder*, odnosno vizuelni interfejs koji olakšava pripremu eksperimenta tako što prevodi u kod ono što korisnik kreira pomoću grafičkog interfejsa. Rad na kreiranju PsychoPy biblioteke započet je 2002. godine, za potrebe eksperimenata koje je u to vreme izvodio njegov autor, a sa idejom da ukrsti prednosti OpenGL-a i jednostavnost sintakse u programskom jeziku Python. Pisana je sa ciljem da omogući istraživačima da imaju kontrolu nad komplikovanim procedurama i zahtevnim uslovima koje stimulusi treba da ispune. Ova biblioteka može se koristiti i tako što se piše kod koji upravlja eksperimentom, ali njenoj popularnosti je dodatno doprinelo to što je kreiran i pomenuti vizuelni interfejs koji prevodi instrukcije u kod. Na taj način, ovaj softver je objedinio napredne performanse sa korisničkim iskustvom. Ipak, kada se porede ta dva, PsychoPy daje prednost tehničkim aspektima.

OpenSesame[10] [15] je otvoreni softver koristi PsychoPy biblioteku kao back-end, a koji je od druge grupe autora razvijan paralelno sa PsychoPy Builder-om. Preciznije govoreći, OpenSesame je nezavisan od back-enda i može da koristi nekoliko biblioteka u te svrhe (PsychoPy, PyGame, PyOpenGL, Expyriment). Osnovna ideja njegovih autora bila je da korisnicima koji ne poznaju programski jezik Python približe vrhunske performanse PsychoPy biblioteke i da to učine na način koji omogućava jednostavnost i fleksibilnost. Premda zvanično ne pruža sve napredne mogućnosti koje sadrži PsychPy, autori su u svojoj zamisli u potpunosti uspeli, jer je OpenSesame izuzetno brzo postao jedan od najpopularnijih softverskih aplikacija u naučnoj zajednici. To je postignuto zahvaljujući ideji da se kombinuje jednostavnost koju daje grafički interfejs i fleksibilnost koju daje mogućnost unosa Python koda preko tzv. in-line skripta. Na taj način, korisnik štedi vreme koristeći već gotova, rutinska rešenja, ali zadržava i mogućost kreiranja sopstvenih novih i neuobičajenih rutina uz minimalno poznavanje programiranja. Pored toga, OpenSesame je pojednostavio način kreiranja vizuelnih stimulusa i zadržao dobre tehničke karakteristike koje omogućava biblioteka PsychoPy. Konačno, popularnosti je doprinela i velika i agilna grupa za podršku.

8 http://visualdmdx.com/
9 https://www.psychopy.org/
10 https://osdoc.cogsci.nl/

Pored ovde opisanih, postoji još čitav niz slobodnih ili otvorenih softverskih aplikacija, kao što su: **VideoToolbox** [16], **PsychToolbox/ MATLAB Psychophysics Toolbox**[11] [17], **PEBL**[12] [18], **PyScope**[13] za Mac OS [19], **PsyToolkit**[14] za Linux [20], **Tscope**[15] [21], **Vision Egg**[16] [22], **Tatool**[17], **ExpFactory**[18], **NodeGame**[19]. Konačno, postoji i domaća verzija slobodnog eksperimentalnog softvera – **UltraLab**[20] [23]. On je prvenstveno kreiran da studentima psihologije, kroz iskustvo iz prve ruke omogući lako usvajanje znanja i veština iz kognitivne psihologije. Osmišljen je sa idejom da bude jednostavan za upotrebu, tako da ne zahteva veliko iskustvo i da bude prenosiv, tako da ne zahteva instaliranje (pisan je u programskom jeziku Java). Zahvaljujući tome, lako može biti u upotrebi i u školskoj nastavi, a može biti od koristi i istraživačima početnicima. Pored samog programa, dostupni su i primeri nekih od eksperimentalnih paradigmi.

B. Istraživanja koja se izvode na mreži

U novije vreme, istraživači u oblasti psihologije počeli da primećuju da se njihova naučna saznanja zasnivaju dominantno na podacima koji potiču od tzv. WEIRD populacije: Western, Educated, Industrialized, Rich, and Democratic [24], jer su ispitanici u tim istraživanjima najčešće bili studenti. Zadovoljenje potrebe da se istraživanja izvode nad reprezentativnim uzorcima ponudilo je postojanje i raširenost upotrebe interneta. Ubrzo su istraživanja koja se izvode na mreži postala veoma učestala, a pojava fizičke izolacije zbog pandemije virusa SARS-COV2, učinila je da podaci koji se prikupe preko mreže mnogim istraživačima budu jedini podaci na koje mogu da računaju. Time je već započeti proces brzog prelaska na mrežna istraživanja dodatno ubrzan.

Prelazak na mrežu doneo je posebne izazove eksperimentalnim istraživanjima, jer je doneo nove zahteve za precizno merenje vremena izlaganja stimulusa i davanja odgovora. Ovi problemi rešeni su tako što se stimulusi prikazuju u pretraživaču, a eksperiment se izvršava lokalno.

Proces prikupljanja podataka preko interneta podrazumeva tri nivoa softverske podrške. Prvo, potrebno je okruženje u kojem će biti definisane instrukcije za izlaganje stimulusa i beleženje odgovora (*kreiranje eksperimenta*). Potom, potrebni su server i platforma koja će *upravljati eksperimentom*, odnosno koji će pohranjivati eksperimentalne fajlove koji su prethodno kreirani, formirati linkove za pristup eksperimentu i pohranjivati podatke koji su prikupljeni od ispitanika. Klikom na link, ispitanik preuzima eksperiment tako da se kod izvršava na lokalnoj mašini (čime se postiže preciznost izlaganja i merenja), a po završetku

11 http://psychtoolbox.org/
12 http://pebl.sourceforge.net/
13 https://github.com/portante/pycscope
14 https://www.psytoolkit.org/
15 https://expsy.ugent.be/tscope/index.html
16 http://visionegg.org/
17 http://www.tatool.ch/
18 https://expfactory.github.io/
19 https://nodegame.org/
20 https://osf.io/enwgm/?view_only=1104a5c416fd4dbe917e9b753e959c32

prikupljeni podaci šalju nazad na server. Linkove je moguće kreirati tako da se ograniči broj pristupa sa jedne IP adrese ili dozvoli neograničen broj pristupa. Konačno, potrebna je platforma za *regrutovanje ispitanika*, odnosno potreban je virtuelni prostor preko kojeg će link ka eksperimentu doći do ispitanika. U nastavku će svaki od nivoa biti zasebno opisan.

Kreiranje eksperimenta. Komercijalna softverska rešenja koja su najčešće u upotrebi u eksperimentalnoj psihologiji jesu **Qualtrics**[21], koji ima ograničene mogućnosti i pogodniji je za upitničke studije, dok se u hronometrijskim istraživanjima obično koriste **Gorilla**[22], koja je naročito popularna i **Inquisit Web**[23].

Slobodni ili otvoreni softver uspešno parira, a često i prevazilazi mogućnosti svojih komercijalnih parnjaka. Među najpopularnijim otvorenim softverskim rešenjima su **PsychoPy**[24] [13] i **OSWeb**[25] [15] koji pored kreiranja i zadavanja eksperimenta u laboratorijskim uslovima pružaju i mogućnost kreiranja eksperimentalnog fajla za mrežno zadavanje. Princip rada je isti kao za pripremu fajla za rad u lokalu, s tim što se umesto programskog jezika Python, za pisanje koda koristi programski jezik JavaScript. Za korisnike koji nisu vešti u programiranju, kreiranje eksperimentalnog fajla je identično kao prilikom rada u lokalu, s tim što se na kraju rada iskoristi opcija da program sam prevede instrukcije u JavaScript.

Veoma popularan je i otvoreni softver **jsPsych**[26] [25] koji predstavlja biblioteku gotovih rešenja napisanih u JavaScript programskom jeziku, kojima se upravlja različitim segmentima eksperimenta. Upotreba ovog softvera zahteva poznavanje ovog programskog jezika, s tim da postoji veoma razvijena mreža podrške i mnoštvo gotovih rešenja za različite eksperimentalne paradigme.

Tu su još i **Lab.js**[27] [26], **PsyToolkit**[28] [27] i sl.

Upravljanje eksperimentima: distribucija linkova i pohranjivanje podataka. Do nedavno slobodan za upotrebu (finansiran preko projekta), a od nedavno dostupan uz minimalnu finansijsku nadoknadu, najpoznatiji servis za upravljanje eksperimentima, **Pavlovia**[29] razvijan je od strane kreatora PsychoPy paketa. Pavlovia se koristi i kao mesto za razmenu eksperimenata pisanih uz pomoć PsychoPy biblioteke.

Od slobodnih rešenja trenutno je najpopularniji serverski sistem **JATOS**[30], koji je finansiran od strane Lajpciškog instituta za psihologiju (ZPID) i koji se za istraživanja manjeg obima može koristiti na adresi institutskog servera. Za istraživanja većeg obima poželjno je instalirati ga na lokalnom univerzitetskom serveru, pri čemu se kompletan softver može slobodno preuzeti. Pored toga, slobodni za korišćenje su i **psiTurk**[31], **Cognition.run**[32], **Pushkin**[33] itd.

Regrutovanje ispitanika. Konačno, potrebno je na neki način proširiti poziv ispitanicima i učiniti dostupnim linkove za pristup eksperimentima. Najpopularnije komercijalne platforme koje se koriste u te svrhe jesu **Amazon Mechanical Turk**[34], **Prolific Academic**[35], **Sona Systems**[36], **Figure Eight**[37]. Ovde se epitet komercijalno odnosi na činjenicu da ispitanici koji posećuju ove platforme, nakon učešća u eksperimentu dobijaju finansijsku nadoknadu za to, a odabrana platforma ima ulogu finansijskog i pravnog posrednika u tom procesu.

Do ispitanika se može doći i bez davanja finansijske nadoknade za učešće, a tada se linkovi najčešće distribuiraju preko društvenih mreža. Iskustva pokazuju da novac nije najbolji insentiv i da su ispitanici najposvećeniji kada znaju da će dobiti neku povratnu informaciju o sebi [28]. Neke laboratorije i neki projekti poseduju veb sajtove na kojima kontinuirano nude linkove za volontersko učešće u istraživanjima. Neki od njih su Word Likeness[38], Games with words[39], Small world of words[40], Lab in the wild[41], The music lab[42], Test my brain[43], itd.

III. Poređenje komercijalnog i slobodnog/otvorenog softvera

Poređenje dve grupe softverskih alata s obzirom na performanse pokazuje da cena nije prediktor kvaliteta. Slobodni, a naročito otvoreni softver po različitim tehničkim parametrima stoji rame uz rame, a često i pobeđuje tradicionalni, komercijalni softver [29], [30], [31], [32]. Premda neki od njih pokrivaju uzan domen, jer su pisani za potrebe specifične grupe istraživanja, oni koji su najrasprostranjeniji dovoljno su fleksibilni da mogu da ispune svaku potrebu: funkcionišu na različitim platformama, omogućavaju prikazivanje širokog spektra stimulusa (tekst, slika, zvuk, video, itd.) pod različitim uslovima, omogućavaju snimanje različitih vrsta odgovora, kao i njihovo kombinovanje (vreme reakcije, pritisak tastera, EEG, fMRI, očni pokreti itd). Otvoreni sofver neretko je i moćniji po svojoj prilagodljivosti, jer na njegovom razvoju radi armija „informisanih korisnika“ – naučnika koji taj sofver kroje upravo u skladu sa potrebama istraživačke zajednice.

Pored toga što je dostupan, otvoreni softver pruža naučnicima mogućnost da imaju uvid u način na koji je implementirano neko rešenje („da zavire ispod haube“), te na taj način mogu da imaju precizniju kontrolu nad svojim eksperimentima.

Konačno, slobodni i otvoreni softver ima veću zajednicu entuzijasta koji spremno i brzo pomažu u rešavanju problema.

21 https://www.qualtrics.com/
22 https://gorilla.sc/
23 https://www.millisecond.com/
24 https://www.psychopy.org/
25 https://osdoc.cogsci.nl/3.2/manual/osweb/
26 https://www.jspsych.org/
27 https://lab.js.org/
28 https://www.psytoolkit.org/
29 https://pavlovia.org/
30 https://www.jatos.org/
31 https://psiturk.org/
32 https://www.cognition.run/
33 https://languagelearninglab.gitbook.io/pushkin/
34 https://www.mturk.com/
35 https://www.prolific.co/
36 http://www.sona-systems.com/default.aspx
37 https://www.figure-eight.com/
38 https://worldlikeness.org/#
39 https://www.gameswithwords.org/
40 https://smallworldofwords.org/en/project
41 http://labinthewild.org/
42 https://www.themusiclab.org/
43 https://testmybrain.org/

IV. Zaključak

Razvoj informacionih tehnologija i preduzimljivost istraživača doveli su do toga da slobodni i otvoreni softver koji služi za prikazivanje stimulusa i prikupljanje podataka vrlo brzo po svojim performansama postane podjednako kvalitetan kao i postojeća komercijalna rešenja, kako u istraživanjima koja se sprovode u laboratorijskim uslovima, tako i u onima koja se odvijaju preko mreže. Pored toga, zahvaljujući entuzijazmu svojih tvoraca (koji su često i sami naučnici koji primenjuju taj sofver), neretko je brže davao odgovore na stroge i brzomenjajuće zahteve struke. Zahvaljujući tome, slobodni i otvoreni softver polako, ali sigurno preuzima dominaciju u oblasti eksperimentalno-psiholoških istraživanja.

Zahvalnica

Zahvaljujem se dr Đorđu Đurđeviću na kontinuiranoj pomoći i savetima u oblasti računarstva, softverskog inženjeringa i informacionih tehnologija. Hvala Kseniji Mišić na plodnim diskusijama i entuzijazmu koji delimo po pitanju prikupljanja podataka. Želela bih da izrazim zahvalnost i organizatorima besplatnih predavanja i radionica na temu prikupljanja podataka preko mreže, a to su: Stefanie Mueler i Leibniz Institute for Psychology[44] (ZPID) za organizovanje radionice o primeni OSWeb aplikacije, Python-a i server Jatos u procesu prikupljanja podataka, Joshua Hartshorne i njegov tim za organizaciju radionice Moving Research Online[45], konzorcijum projekta Words in the World za organizovanje serije predavanja pod nazivom WoW Open Office Hours[46], posvećenih prikupljanju podataka preko mreže, kao i Xenia Shmaltz sa Univerziteta u Minhenu za organizovanje serije predavanja pod nazivom Psycholinguistics and Research Methods Webinar Series[47]. Svi oni su, nakon održanih radionica, materijale učinili dostupnim preko različitih platformi (do kojih se može doći praćenjem ovde navedenih linkova).

Rad na ovoj publikaciji delimično je podržan sredstvima sa projekta Ministarstva prosvete, nauke i tehnološkog razvoja Republike Srbije: ON 179033.

Literatura

[1] Psychology Software Tools, Inc. [E-Prime 3.0]. 2016. Retrieved from https://www.pstnet.com

[2] Neurobehavioral Systems, Inc., Berkeley, CA. [Presentation 18.0]. 2020. Retrieved from www.neurobs.com

[3] Cedrus Corporation, San Pedro, CA. [SuperLab 5]. 2014. Retrieved from www.cedrus.com

[4] SR Research, Ottawa, Ontario, Canada [Experiment Builder] 2016. https://www.sr-research.com/

[5] Millisecond Software, Seattle, WA. Inquisit 5 [Computer software] 2015. Retrieved from https://www.millisecond.com

[6] Empirisoft, New York [DirectRT Version 2004.1.0.55]. 2004. http://www.empirisoft.com

[7] C. Stahl, „Software for generating psychological experiments“, *Experimental Psychology,* vol. 53, pp. 218–232, 2006. https://doi.org/10.1027/1618-3169.53.3.218

[8] K. I., Forster, & J. C., Forster, „DMDX: A Windows display program with millisecond accuracy“, *Behavior Research Methods, Instruments, & Computers*, vol. 35, pp. 116–124, 2003. https://doi.org/10.3758/BF03195503

[9] K. I., Forster, & C., Davis, C., „Repetition priming and frequency attenuation in lexical access“ *Journal of Experimental Psychoogy: Learniing, Memory, and Cognition,* vol. 10, pp. 680–698, 1984. https://doi.org/10.1037/0278-7393.10.4.680

[10] P., Garaizar, U. D., Reips, „Visual DMDX: A web-based authoring tool for DMDX, a Windows display program with millisecond accuracy“, *Behavior Research Methods*, vol. 47(3), pp. 620-631, 2015. https://doi.org/10.3758/s13428-014-0493-8

[11] J. W., Peirce, „PsychoPy - Psychophysics software in Python“, *Journal of Neuroscience Methods*, vol. 162(1-2), pp. 8-13, 2007. https://doi.org/10.1016/j.jneumeth.2006.11.017

[12] J. W., Peirce, „Generating stimuli for neuroscience using PsychoPy“, *Frontiers in Neuroinformatics*, vol. 2(10), pp. 1-8, 2009. https://doi.org/10.3389/neuro.11.010.2008

[13] J. W., Peirce, J. R., Gray, S., Simpson, M. R., MacAskill, R., Höchenberger, H., Sogo, E., Kastman, J., Lindeløv, „PsychoPy2: experiments in behavior made easy. *Behavior Research Methods*, vol. 51, pp. 195–203, 2019. https://doi.org/10.3758/s13428-018-01193-y

[14] J. W., Peirce, & M. R., MacAskill, „*Building Experiments in PsychoPy*“, London: Sage, 2018.

[15] S., Mathôt, D., Schreij, & J., Theeuwes, „OpenSesame: An open-source, graphical experiment builder for the social sciences“, *Behavior Research Methods*, vol. 44(2), pp. 314-324, 2012. https://doi.org/10.3758/s13428-011-0168-7

[16] D. G., Pelli, „The VideoToolbox software for visual psychophysics: Transforming numbers into movies“, *Spatial Vision*, vol. 10, pp. 437–442, 1997. https://doi.org/10.1163/156856897X00366

[17] D. H., Brainard, „The Psychophysics Toolbox“, *Spatial Vision*, vol. 10, pp. 433–436, 1997. https://doi.org/10.1163/156856897X0035

[18] S. T., Mueller, „The PEBL manual“, 2010. Retrieved from http://pebl.sourceforge.net/

[19] J. D., Cohen, B., MacWhinney, M., Flatt, & J., Provost, „PsyScope: An interactive graphic system for designing andcontrolling experiments in the psychology laboratory usingMacintosh computers“, *Behavior Research Methods, Instruments,& Computers*, vol. 25, pp. 257–271, 1993. https://doi.org/10.3758/BF03204507

[20] G., Stoet, „PsyToolkit - A software package for programming psychological experiments using Linux“, *Behavior Research Methods*, vol. 42(4), pp. 1096-1104, 2010. https://doi.org/10.3758/BRM.42.4.1096

[21] M., Stevens, J., Lammertyn, F., Verbruggen, A., Vandierendonck, „Tscope: A C library for programming cognitive experiments on the MS windows platform“, Behavior Research Methods, vol. 38(2), pp. 280-286, 2006. https://doi.org/10.3758/bf03192779

[22] A. D., Straw, „Vision Egg: An open-source library for realtimevisual stimulus generation“, *Frontiers in Neuroinformatics*, vol. 2, pp. 4, 2008. https://doi.org/10.3389/neuro.11.004.2008

[23] D., Filipović Đurđević, & Đ., Đurđević, „UltraLab – Jednostavan i slobodan softver za izvođenje eksperimenata u okviru praktične nastave psihologije“, under review.

[24] J. Henrich, S. J. Heine, A. Norenzayan. The weirdest people in the world? *Behavior and Brain Science,* 33(2-3), pp. 61-135, 2010. https://doi.org/10.1017/S0140525X0999152X

[25] J. R., de Leeuw, „jsPsych: A JavaScript library for creating behavioral experiments in a web browser“, *Behavior Research Methods*, vol. *47*(1), pp. 1-12, 2015. https://doi.org/10.3758/s13428-014-0458-y

[26] F., Henninger, Y., Shevchenko, U. K., Mertens, P. J., Kieslich, & B. E., Hilbig, „lab.js: A free, open, online study builder“, 2019. https://doi.org/10.31234/osf.io/fqr49

[27] G., Stoet, „PsyToolkit: A novel web-based method for running online questionnaires and reaction-time experiments“. *Teaching of Psychology*, vol. 44(1), pp. 24-31, 2017. https://doi.org/10.1177/0098628316677643

[28] M., Kosinski, S. C., Matz, S. D., Gosling, V., Popov, & D., Stillwell, „Facebook as a research tool for the social sciences: Opportunities, challenges, ethical considerations, and practical guidelines“, *American Psychologist*, vol. 70(6), pp. 543–556, 2015. https://doi.org/10.1037/a0039210

44 http://leibniz-psychology.org/opensesame/
45 http://www.movingresearchonline.info/
46 http://wordsintheworld.ca/home/open-office-hours/
47 https://osf.io/4gm65/

[29] D., Bridges, A., Pitio, M. R., MacAskill, J. W., Peirce, „The timing mega-study: comparing a range of experiment generators, both lab-based and online“, *PeerJ*, vol. 8, pp. e9414, 2020. https://doi.org/10.7717/peerj.9414

[30] J. R., de Leeuw, & B. A., Motz, „Psychophysics in a Web browser? Comparing response times collected with JavaScript and Psychophysics Toolbox in a visual search task“, *Behavior Research Methods*, vol. 8(1), pp. 1-12, 2016. https://doi.org/10.3758/s13428-015-0567-2

[31] J., Kim, U., Gabriel, P., Gygax, „Testing the effectiveness of the Internet-based instrument PsyToolkit: A comparison between web-based (PsyToolkit) and lab-based (E-Prime 3.0) measurements of response choice and response time in a complex psycholinguistic task“ PLoS ONE, vol. 14(9), pp. e0221802, 2019. https://doi.org/10.1371/journal.pone.0221802

[32] B.E., Hilbig, „Reaction time effects in lab- versus web-based research: Experimental evidence“, *Behavior Research Methods*, vol. 48(4), pp. 1718-1724, 2016. https://doi.org/10.3758/s13428-015-0678-9

Rodna raznolikost na Vikipediji – primeri iz prakse

Ivana Madžarević[1], Nebojša Ratković[2]
[1]*Menadžerka projekata i zajednice, Vikimedija Srbije*
[2]*Menadžer obrazovnog programa, Vikimedija Srbije*
ivana.madzarevic@vikimedija.org, nebojsa.ratkovic@vikimedija.org

Rezime: Budući da je Vikipedija našla široku primenu u svetskoj populaciji kao jedan od značajnih globalnih izvora informacija, obezbeđivanje i negovanje raznolikosti među njenim urednicima je ključno u izgradnji i raspodeli slobodnog znanja. Učešće različitih demografskih grupa u tom procesu je od velike važnosti, zbog čega pitanja ko piše sadržaj na Vikipediji i o čemu se piše postaju presudna u procesu demokratizacije stvaranja znanja. S toga, ovaj rad se fokusira na značaj rodne ravnopravnosti, veće zastupljenosti žena koje uređuju Vikipediju i više sadržaja koji se odnose na žene. Konceptualni okvir predstavljen u ovom radu deo je većeg napora i obuhvata različite projektne aktivnosti koje se realizuju od 2014. godine. Među njima, najznačajnije je globalna kampanja WikiGap koji Vikimedija Srbije realizuje od 2018. godine. Projekat ima za cilj osmišljavanje, realizaciju i procenu niza mera za smanjenje rodnog jaza na Vikipediji i veće aktivno učešće urednica Vikipedije. Smatramo da radovi ovog tipa daju dobre polazne tačke za dalja istraživanja koja imaju za cilj razvoj inovativnih rešenja za poboljšanje rodne ravnoteže na Vikipediji, stvaranje mogućnosti za prevazilaženje faktora koji izazivaju rodne razlike na Vikipediji i njihovih negativnih efekata na demokratiju slobodnog znanja.

Ključne reči: Vikipedija; rod; rodna analiza; rodna raznolikost; rodna ravnopravnost; rodni jaz.

I. Uvod

Otkad je napravljena Vikipedija 2001. godine, sa tada radikalnom idejom da svaki pojedinac koji ima pristup internetu, može da doprinese izgradnji ove besplatne i slobodne enciklopedije, Vikipedija nailazi na sve širu primenu u svakodnevnom životu, uprkos činjenici da informacije na njoj nisu revidirane u tradicionalnom akademskom smislu. Milioni ljudi [1] se na dnevnom nivou oslanjaju na informacije koje pronalaze na Vikipediji o različitim temama, čime je ona bez sumnje, postala uticajan izvor informacija na internetu, koji sadrži brojne podatke o različitim pojmovima i ličnostima, uključujući i podatke o uglednim ženama iz različitih zemalja. Iz navedenih razloga možemo reći da je Vikipedija vodeći primer istorijske promene koja se ogleda u načinu na koji znanje može da se deli sa drugima. Informacije koje se nalaze na Vikipediji su rezultat napora samoorganizovane grupe ljudi koji čine jedinstvenu zajednicu na internetu, koji znanje o različitim stvarima dele sledeći opšteprihvaćena pravila o verifikovanju informacija putem referenci, neutralnom stanovištu i nepristrasnosti.

Iako se u kontekstu Vikipedije, demokratičnost znanja ogleda u postojanju otvorene platforme koja svima garantuje pristup i uređivačka prava, uz princip otvorene saradnje i kolaborativnog rada, lično mišljenje autora ovog teksta je da otvorenost ne garantuje ravnopravnost. Uprkos otvorenosti na više nivoa, Vikipedija se bori sa nejednakom zastupljenošću različitih grupa urednika, pre svega u odnosu broja ženskih i muških urednika. Zbog široke upotrebe Vikipedije u svetskoj populaciji, obezbeđivanje i podsticanje reprezentativnog učešća žena čini se neophodnim za demokratizaciju prijema, izgradnje i preraspodele znanja. Iako problem rodne asimetrije na Vikipediji sve više privlači pažnju, mnogi korisnici sajta možda nisu svesni i nisu zabrinuti zbog ovog njenog nedostatka i kako ta činjenica utiče na sadržaj koji prihvataju kao činjeničan. U praksi, to je rezultiralo da zajednicu Vikipedijanaca uglavnom čine muškarci. [2]

Ako je znanje moć, moramo biti svesni ko preuzima odgovornost za uređivanje većine sadržaja na Vikipediji, kao i da li zbog toga ipak dolazi do određenog stepena pristrasnosti. Jedna od opasnosti lakog pristupa informacijama na Vikipediji ogleda se u tome, da ako ne postoji članak o nečemu, ljudi mogu pomisliti da je to nevažno. Zbog toga se na globalnom nivou javila potreba za većim prepoznavanjem žena o kojima nije postojao nikakav sadržaj na Vikipediji, ali i o njihovom doprinosu istoriji i društvu koji očigledno moraju biti vidljiviji. Zbog toga žene treba podstaći i da više uređuju Vikipediju kako bi se osiguralo da njihovi glasovi i interesovanja budu vidljiviji.

Početne analize o prirodi rodnih razlika na Vikipediji pružaju dobre polazne tačke za dalje istraživanje s ciljem razvoja inovativnih rešenja za poboljšanje rodne ravnoteže. Zbog toga ćemo se u nastavku ovog rada osvrnuti na podatke koji su o tome dostupni, a kasnije ćemo se fokusirati na podatke koji se odnose na Vikipediju na srpskom jeziku, kao i na napore koje Vikimedija Srbije[1] čini za aktivno učešće žena na ovoj jezičkoj varijante Vikipedije.

Kada se osvrnemo na rodni jaz na Vikipediji, moramo uzeti u obzir dva aspekta. Prvi je jaz u pogledu sadržaja o ženama na Vikipediji, dok se drugi tiče mnogo manjeg procenta urednica u odnosu na urednike. U oba slučaja se

1 Vikimedija Srbije je nevladino, nestranačko i neprofitno udruženje, čiji su ciljevi promocija i podržavanje stvaranja, sakupljanja i umnožavanja slobodnog sadržaja na srpskom jeziku isključivo na neprofitan način, kao i ideje da svi ljudi imaju jednak pristup znanju i obrazovanju. Zvanični je ogranak Zadužbine Vikimedija koja upravlja (rukovodi serverima) i pomaže razvoju projekata slobodnog znanja kao što su Vikipedija, Vikimedijina ostava, Vikiknjige, Vikicitat, Vikiizvornik i Vikiverzitet. Osnovna razlika između Vikimedije Srbije i Vikipedije na srpskom ogleda se u tome što je Vikimedija Srbije organizacija, dok je Vikipedija na srpskom onlajn, slobodna enciklopedija kojoj svako može doprinositi. Detaljnije na http://wikimedia.rs/O-nama

DOI: 10.5281/zenodo.4064634

može zaključiti da su žene zbog istorijskih barijera bile potisnute i to se odrazilo na vidljivost dostignuća žena u raznim oblastima.

II. Rodna struktura i Vikipedija zajednice

Studija o rodnoj strukturi iz 2009. godine utvrdila je da je na Vikipediji zastupljeno samo 13% urednica, a taj procenat se čak smanjio na 9% u 2011. [3] Bivša izvršna direktorka Vikimedija Zadužbine navodi devet razloga zašto je primetan ovakav disbalans na Vikipediji: 1.) slabo korisničko iskustvo u Vikipedijinom interfejsu za uređivanje; 2.) nedostatak slobodnog vremena žena; 3.) nedostatak samopouzdanja žena; 4.) averzija žena prema sukobu i nespremnost da učestvuju u dugim uređivačkim ratovima; 5.) verovanje da će njihov doprinos biti poništen ili obrisan; 6.) neki smatraju da je ukupna atmosfera na Vikipediji mizoginistička; 7.) kultura na Vikipediji je seksualna, pa je žene doživljavaju neprijatnom; 8.) obraćanje u muškom rodu nije primereno na jezicima koji imaju različite gramatičke oblike za različite polove i 9.) postoje manje mogućnosti za socijalne odnose i ton dobrodošlice u poređenju sa drugim veb lokacijama. [4]

U okviru korisničkih podešavanja na Vikipediji urednici mogu odabrati rod. To podešavanje u praksi utiče samo na ispis pojedinih rodno-osetljivih sistemskih poruka, ali osim toga ne nosi neko posebno značenje u Medijaviki softveru. U istraživanju na Vikipediji na srpskom obavljenom 2015. godine, uočeno je da je od ukupno 157.353 registrovanih korisnika, koliko ih je bilo u tom trenutku, 4.202 korisnika imaju uključeno podešavanje roda, od čega je odabrana opcija muškog roda za 3.409 korisnika, odnosno ženskog za 793. Dakle, prema tom podešavanju, broj žena je oko 18,9%. Takođe je primetno da su žene slabije aktivne u odnosu na muškarce kada je unos izmena[2] u pitanju. [5]

Kao odgovor na ovaj disparitet, na Vikipediji na srpskom je 2014. godine započet projekat Fem Viki sa ciljem da se, sa jedne strane, podstaknu žene na uređivanje Vikipedije, a sa druge strane da se doprinese kvalitetu i kvantitetu članaka na temu feminističke i rodne terminologije i biografija žena. [6] Projektu je prethodilo sagledavanje stanja na Vikipediji na srpskom jeziku. Iako primenljiv u srpskom jeziku, rodno senzitivan govor nije bilo moguće uočiti; članci o najpoznatijim feminističkim teoretičarkama nisu postojali; odrednica rod nije postojala kao poseban članak; u tekstu o diskriminaciji nije bilo ni reči o diskriminaciji na osnovu roda. Aktivnosti su podrazumevale ostvarivanje saradnji sa organizacijama koje se bave ovom tematikom i koje mogu pružiti priliku svojim članicama da kroz radionice steknu veštine uređivanja i da se oslobode straha od osuđivanja i potencijalnog brisanja sadržaja. Ovo je bio siguran prostor za doprinos Vikipediji i razmenu znanja. Posebno je bitno naglasiti značaj decentralizacije u različitim gradovima Srbije, u kojima postoji veća mogućnost za diskriminacijom na osnovu roda. Socijalni uslovi koji utiču na češće posvećivanje žene porodici, patrijahalna kultura, nedovoljna digitalna pismenost su izraženiji faktori u manjim i nerazvijenim gradovima. Zbog toga je jedan od ciljeva projekta bio i pružanje mogućnosti ženama u ovakvim okolnostima da prođu obuku i steknu samostalnost u uređivanju. Fem Viki je na Vikipediji na srpskom bio početak uočavanja problema rodnog jaza i početak rada kako bi se ovaj disbalans smanjio. [7]

Jedan od adekvatnih primera poduhvata da se broj žena koje uređuju Vikipediju poveća je Viki-bibliotekar. Viki-bibliotekar je projekat umrežavanja bibliotekara u celoj Srbiji kako bi stekli veštine uređivanja Vikipedije i uvećali slobodan sadržaj iz bibliotačke delatnosti. Iako prvobitni cilj nije fokusiran na povećanje urednica na Vikipediji, ipak su učesnici u velikoj meri bile bibliotekarke. Takođe, učesnici na ovom projektu su povezivali razne biblioteke, muzeje, arhive širom Srbije, pa su im izvori bili daleko dostupniji, naročito kada se radilo o ženama čiji rad nije vidljiv široj javnosti. U ovom slučaju primer iz prakse je dokazao da je potrebno ciljano se obratiti profesijama gde su žene zastupljeniji pol. Bibliotekarke su naročito bile spremnije da se priključe i drugim akcijama Vikimedije Srbije čiji je krajnji cilj postizanje balansa u rodnoj strukturi na Vikipediji i drugim Viki projektima.

Smanjivanje rodnog jaza na Vikipediji je sveobuhvatan proces i jednokratan uticaj nije održiv. Potrebne su sistemski organizovane akcije usmerene na jačanje i osamostaljivanje žena na Vikimedijinim projektima.

III. Sadržaj o ženama na Vikipediji

Istraživanje početkom 2020. godine je pokazalo da na Vikipediji na srpskom jeziku ima 35.410 članaka o muškarcima, dok je taj broj mnogo manji kada su u pitanju biografije žena — 8.303. Ako uporedimo statistike iz juna 2016, kada je procenat zastupljenosti biografija žena u odnosu na biografije muškaraca bio ~17.7%, može se uvideti da je odnos pozitivno promenjen za 1,29%. Pošto je ovo deo globalnih težnji, ako sagledamo rezultate na svetskom nivou, uočićemo da je najbolji uspeh u ovom pogledu imala Vikipedija na pandžabi jeziku, kod koje je ovaj odnos promenjen za 17,01%. [8] Ovaj problem je zastupljen na svim jezičkim verzijama Vikipedija, zbog čega u Vikimedijinom pokretu postoje korisničke grupe, organizacije, grupacije volontera, projekti koji su posvećeni uvećanju sadržaja o ženama iz raznih delova sveta. Neke od takvih su Wiki Donne[3], Women in Red[4], Wiki Loves Women[5], Art&Feminism[6], WikiGap[7], Wiki4Women[8].

U slučaju sadržaja je veoma bitno istaći i doživljaj značaja određene osobe na Vikipediji. Iako postoje propisana pravila o tome da li je neka osoba dovoljno relevantna da bi imala članak na Vikipediji [9], dogovor o

2 Izmena na Vikipediji podrazumeva bilo kakvo ažuriranje članaka - dodavanje sadržaja, slike, reference, ali i ispravka pravopisnih i gramatičkih grešaka.

3 https://meta.wikimedia.org/wiki/WikiDonne
4 https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Women_in_Red
5 https://meta.wikimedia.org/wiki/Wiki_Loves_Women/en
6 https://meta.wikimedia.org/wiki/Art%2BFeminism_User_Group
7 https://meta.wikimedia.org/wiki/WikiGap
8 https://meta.wikimedia.org/wiki/Wiki4Women

tome se takođe postiže konsenzusom zajednice, što u prevodu znači da zajednica urednika i urednica može da izglasa brisanje članaka ukoliko smatraju da nije pređen prag značaja. Ova činjenica, kao i često zapostavljen i nedovoljno prepoznat doprinos žena u istoriji, samo dodatno otežavaju postizanje rodne ravnopravnosti.

Kada govorimo o sadržaju o ženskim temama i biografijama žena, na Vikipediji na srpskom, akcije su bile fokusirane na takmičenja na Vikipediji, uređivačke maratone i WikiGap globalnu kampanju

Takmičenja na Vikipediji su jednomesečne akcije putem kojih urednici uređuju članke na određenu temu i tako šire zdrav takmičarski duh. Osim takmičenja u okviru Fem Viki projekta [10] koje se ticalo ženskih tema, na Vikipediji na srpskom su organizovana četiri takmičenja sa ciljem uvećanja biografija znamenitih žena. Prvo takmičenje je organizovano U okviru FemViki projekta 2014. godine, a zatim redom u 2017, 2019, 2020. Čitav koncept se zasniva na animiranju onih urednika i urednica kojima je ova tema bliska, a mala je verovatnoća da će ove biografije biti pokrivene na Vikipediji. Na ovaj način se čak povezuju urednici i urednice i međusobno razmenjuju informacije koje imaju, literaturu koja je dostupna ili čak upućuju na teme koje nisu pokrivene. Tokom tri takmičenja, od kojih je jedno organizovala Vikimedija Srbije, a jedno Zajednica Vikimedijanaca iz Republike Srpske, napisano je i dopunjeno 592 članka, što ukazuje na veliki značaj ovakvih poduhvata.

Uređivački maratoni su tematska okupljanja volontera koji tokom jednog ili više dana pišu i dopunjavaju članke na određenu temu. Za razliku od takmičenja, oni nemaju za cilj nadmetanje, već je prevashodno sve usmereno na uvećanje sadržaja. Zbog toga su ovakvi događaji veoma povoljni kada su u pitanju ženske teme. Jedan od najznačajnih svetskih pravaca kojima se Vikimedija Srbije priključila je WikiGap. Naime, Ministarstvo spoljnih poslova Švedske i Vikimedija Švedske pokrenule su inicijativu da se organizuje globalni uređivački maraton WikiGap i to neposredno pre i posle 8. marta, Međunarodnog dana žena. Ova globalna ideja uključivala je događaje u više od 50 zemalja od Švedske preko Indonezije, pa do Egipta i Kolumbije u saradnji sa lokalnim organizacijama. Veliki odziv dokaz je da je problem rodnog jaza na Vikipediji prisutan svuda u svetu. U Srbiji je WikiGap uređivački maraton okupio urednike i urednice koji su pisali o znamenitim ženama koje su svojim radom napravile promenu u društvu, ali ona do tada nije bila dovoljno vidljiva i prepoznata. Dobijanjem članka na najvećoj onlajn enciklopediji, rad ovih žena je dostupan ne samo za trenutne čitaoce, već i za urednike i urednice koji žele da nastave sa uređivanjem i dopunom informacija u budućnosti. Tokom tri uređivačka maratona, od kojih je jedan održan u Nišu, a dva u Beogradu, napisano je i dopunjeno 112 biografija značajnih žena. Poverenica za zaštitu rodne ravnopravnosti u Srbiji tokom događaja je istakla da su žene manje vidljive upravo zato što još uvek nisu osvojile javnu sferu, bez obzira na velike pomake i dostignuća. [11]

Osim Švedske ambasade, Vikimedija Srbije dobila je i veliku podršku Ministarstva za rad, zapošljavanje, boračka i socijalna pitanja Republike Srbije po pitanju smanjenja disbalansa u sadržaju o ženama u odnosu na sadržaj o muškarcima na Vikipediji. Ministarstvo i Vikimedija Srbije su organizovali uređivački maraton na kome su urednici i urednice pisali članke o srpskim znamenitim ženama iz Unesko oblasti - obrazovanje, nauka, kultura, umetnost, informisanje i komunikacija i tako zajedničkim snagama radili na povećanju prisustva žena na Vikipediji. [12]

Jedan značajan korak napravljen je u već pomenutom Fem Viki projektu. Naime, autorke knjige Rečnik rodne ravnopravnosti iz 2010. godine, Vesna Jarić i Nadežda Radović odobrile su da se sav sadržaj Rečnika postavi na Vikipediju, čime će se sadržaj na Vikipediji uvećati za više od 102 odrednice. Neke od odrednica su već postojale na Vikipediji, ali su bile proširene i dopunjene oslobođenim materijalom. [13] Ovo je, osim relevantnog sadržaja, značilo i proširenje ideje Fem Viki projekta van radionica.

IV. Zaključak

U ovom radu smo ukazali na različite aspekte rodne neravnopravnosti sa posebnim akcentom na Vikipediji na srpskom jeziku. Rodna neravnopravnost je važno pitanje budući da je upotreba Vikipedije sve veća, a sa tim i njen značaj kao centralnog skladišta znanja na internetu koje se koristi širom sveta, uključujući i obrazovne svrhe. Rezultati istraživanja predstavljenih u ovom radu otkrivaju značajne razlike među polovima na različitim nivoima upotrebe Vikipedije, koje se ne mogu pripisati činjenici da Vikipedija zapravo oslikava svet uopšte i da u tom smislu ona nije po nečemu specifična.

Na primer, razlike u pogledu sadržaja na Vikipediji se moraju pripisati isključivo onima koji ih pišu, a to su urednici Vikipedije. Verujemo da se razlike na Vikipediji u tom smislu, delom mogu objasniti načinom na koji je u našem društvu dokumentovan život značajnih muškaraca i žena, te da je rodna neravnopravnost slika dostupnosti odgovarajućih izvora. Pošto se urednici Vikipedije oslanjaju na kredibilne, dostupne, već objavljene izvore, to nam daje objašnjenje i odgovor na pitanje ko su žene koje su dovoljno znamenite da o njima postoji članak na Vikipediji. Osim toga, dobro je poznato iz stavova socijalne psihologije da ljudi uglavnom favorizuju onu grupu ljudi kojoj pripadaju u odnosu na druge ljude, te imajući u vidu odnos muških i ženskih urednika na Vikipediji, i ne treba da čudi nedovoljna brojnost članaka o ženama.

Ukratko, doprinosi ovog rada su dvostruki: (1) doprinos povećanju svesti o rodnim razlikama na Vikipediji na srpskom jeziku i o različitim načinima na koji se takve pristranosti mogu iskazati i (2) prikaz realizovanih akcija i potencijalna rešenja kako bi se smanjile razlike među polovima u budućnost.

Detaljnije i podrobnije istraživanje rodnih barijera na Vikipediji na srpskom jeziku ostavljamo za naš budući

rad koji će pružiti praćenje i vrednovanje ovih pitanja na Vikipediji.

Literatura

[1] Wikimedia Statistics, Monthly overview, [online], dostupno na https://stats.wikimedia.org/#/all-projects, posećeno 25.8.2020.

[2] N. Torres, Why Do So Few Women Edit Wikipedia?, [online], dostupno na https://hbr.org/2016/06/why-do-so-few-women-edit-wikipedia, posećeno 25.8.2020.

[3] N. Torres, Why Do So Few Women Edit Wikipedia?, [online], dostupno na https://hbr.org/2016/06/why-do-so-few-women-edit-wikipedia, posećeno 25.8.2020.

[4] S. Gardner, Nine Reasons Women Don't Edit Wikipedia (in their own words), [online], dostupno na https://suegardner.org/2011/02/19/nine-reasons-why-women-dont-edit-wikipedia-in-their-own-words/, posećeno 25.8.2020.

[5] F. Maljković, Izveštaj o rodnoj strukturi na Vikipediji na srpskom jeziku, [online], dostupno na https://commons.wikimedia.org/wiki/File:Rodna_struktura_na_Vikipediji_na_srpskom_jeziku.pdf, posećeno 25.8.2020.

[6] Stranica projekta Fen Viki, [online], dostupno na https://sr.wikipedia.org/wiki/%D0%92%D0%B8%D0%BA%D0%B8%D0%BF%D0%B5%D0%B4%D0%B8%D1%98%D0%B0:%D0%A4%D0%B5%D0%BC_%D0%92%D0%B8%D0%BA%D0%B8, posećeno 25.8.2020.

[7] Blog Vikimedije Srbije, FemViki 2014: Rekonstrukcija, [online], dostupno na http://blog.wikimedia.rs/?p=1229, posećeno 25.8.2020.

[8] Wikimedia - Meta, Content Gap, [online], dostupno na https://meta.wikimedia.org/wiki/User:Ijon/Content_gap, posećeno 25.8.2020.

[9] Vikipedija, Značaj (ljudi), [online], dostupno na https://sr.wikipedia.org/wiki/Vikipedija:Značaj_(ljudi), posećeno 25.8.2020.

[10] Stranica takmičenja u pisanju članaka o ženskim temama, [online], dostupno na https://sr.wikipedia.org/wiki/Vikipedija:Takmičenje_u_pisanju_članaka/Ženske_teme, posećeno 25.8.2020.

[11] Vikimedija Srbije, Dostignuća žena vidljivija na internetu zahvaljujući Vikigep inicijativi, [online], dostupno na http://wikimedia.rs/dostignuca-zena-vidljivija-na-internetu-zahvaljujuci-vikigep-inicijativi/, posećeno 25.8.2020.

[12] Stranica uređivačkog maratona povodom dana žena 2019. [online], dostupno na https://sr.wikipedia.org/wiki/Vikipedija:Uređivački_maraton_povodom_Dana_žena,_mart_2019, posećeno 25.8.2020.

[13] Blog Vikimedije Srbije, FemViki 2014: Rekonstrukcija, [online], dostupno na http://blog.wikimedia.rs/?p=1229, posećeno 25.8.2020.

Spisak autora(ki) / List of authors

B
Batanović Vuk, 29

C/Ć
Cvetanović Miloš, 7, 19
Ćirović Nataša, 46

Đ
Đurić D. Miloš, 57

F
Filipović Đurđević Dušica, 79

G
Gardašević Gordana, 24

K
Kosanović Biljana, 33
Krstonijević Sovjetka, 15

L/Lj
Lazarević B. Ljiljana, 51
Lazić Aleksandra, 51
Lekić Milica, 24
Ljubešić Nikola, 29

M
Madžarević Ivana, 84
Miličević-Petrović Maja, 29
Milić R. Aleksandar, 38
Miljković Nadica, 7

O
Otašević Vladimir, 33

P
Pejović Predrag, 7
Purić Danka, 51

R
Radivojević Zaharije, 19
Ratković Nebojša, 84

S/Š
Sagić Andrija, 71
Samardžić Tanja, 29
Stojanović Saša, 19
Ševkušić Milica, 7

T
Tošić Ivana, 42

V
Vignoli Italo, 4

Ž
Žeželj Iris, 51

CIP - Каталогизација у публикацији
Народна библиотека Србије, Београд

004(048)

НАЦИОНАЛНА конференције са међународним учешћем "Примена слободног софтвера и отвореног хардвера" (3 ; 2020 ; Београд)
Zbornik treće nacionalne konferencije sa međunarodnim učešćem pod nazivom Primena slobodnog softvera i otvorenog hardvera PSSOH 2020, u Beogradu, februara 2021. godine / [urednički i organizacioni odbor, editorial and organizational board Nadica Miljković, Predrag Pejović, Miloš Cvetanović]. - Beograd : Univerzitet, Elektrotehnički fakultet : Akademska Misao, 2021 (Beograd : Akademska Misao). - 88 str. : ilustr. ; 30 cm

Na spor. nasl. str.: Proceedings of the Third National Conference with International Participation titled Application of free software and open hardware PSSOH 2020, in Belgrade, February 2021. - Apstrakti na srp. I engl. jeziku. - Tiraž 50. - Bibliografija uz svaki apstrakt. - Registar.

ISBN 978-86-7466-872-6 (AM)

a) Računarstvo - Апстракти

COBISS.SR-ID 34069257

Konferenciju podržali

9 788674 668726