Automated Analysis of Short Digital Messages Content with the Potential of Application in Business

Automatizirana analiza sadržaja kratkih digitalnih poruka s potencijalom primjene u poslovanju

Rok Grgec, Leo Mršić
Algebra University College
Visoko učilište Algebra
Gradišćanska 24, 10000, Zagreb, Croatia
{rok.grgec, leo.mrsic}@algebra.hr

Mladen Konecki
University of Zagreb
Sveučilište u Zagrebu
Faculty of Organization and Informatics
Fakultet organizacije i informatike
Pavlinska 2, 42000 Varaždin, Croatia
mladen.konecki@foi.hr

Abstract*. In this paper the potential business value derived from applying content analysis to short digital messages is presented and elaborated. The WhatsAppNLP library, developed for research purposes, facilitates machine learning model training and demonstrates chat log analysis. By employing natural language processing methods, the library simplifies the analysis process through method calls. WhatsAppNLP leverages the WhatsApp application's data, enabling users to freely download and save their conversations in text format. The Python library processes and analyzes the exported text, generating intuitive reports in the form of graphs, tables, and text messages. The approach allows for automated, anonymized, and ethical analysis of messages, enhancing work efficiency and enabling simultaneous analysis of a large message volume.*

Keywords. WhatsAppNLP, natural language processing, application of content analysis models in business, monetization of content analysis

Sažetak. *U ovom radu predstavljena je i razrađena potencijalna poslovna vrijednost proizašla iz primjene analize sadržaja na kratke digitalne poruke. Knjižnica WhatsAppNLP, razvijena u istraživačke svrhe, olakšava obuku modela strojnog učenja i demonstrira analizu dnevnika razgovora. Koristeći metode obrade prirodnog jezika, knjižnica pojednostavljuje proces analize putem poziva metoda. WhatsAppNLP koristi podatke aplikacije WhatsApp, omogućujući korisnicima da besplatno preuzmu i pohrane svoje razgovore u tekstualnom formatu. Python biblioteka obrađuje i analizira izvezeni tekst, generirajući intuitivna izvješća u obliku grafikona, tablica i tekstualnih poruka. Pristup omogućuje automatiziranu, anonimiziranu i etičku analizu poruka, čime se povećava učinkovitost rada i omogućuje istovremena analiza velike količine poruka.*

Ključne riječi. WhatsAppNLP, obrada prirodnog jezika, primjena modela analize sadržaja u poslovanju, monetizacija analize sadržaja

1 Introduction

Millions of users use some form of communication on the Internet every day. Whether it's using Facebook Messenger, Instagram Direct Messenger, WhatsApp, email etc. With every communication, users leave a trace of their conversation somewhere on the Internet. Most apps that support communication between two or more people store that data in certain databases to allow users to see their own conversations back in time. The same data says a lot about the characteristics of the conversation and users' "internet personality"

1 Uvod

Milijuni korisnika svakodnevno koriste neki od oblika komunikacije na internetu. Bilo da se radi o korištenju Facebook Messengera, Instagram Direct Messengera, WhatsAppa, e-pošte itd. Prilikom svake komunikacije korisnici ostavljaju trag svog razgovora negdje na internetu.

Većina aplikacija koje podržavaju komunikaciju između dvije ili više osoba pohranjuju te podatke u određene baze podataka kako bi korisnicima

that is expressed by using various additions to the conversation such as the use of emojis, the use of media records, the use of voice messages, the average conversation time, the most common conversation period of the day etc.

Users are often not fully aware of these characteristics or are aware to some extent, but in reality, they can be far from what they would tentatively expect. By analyzing these data, a clear and realistic idea of patterns of behavior and conversation can be obtained, which can help in trying to change some of the bad habits in the future or by analyzing, and the main goal is to try establishing facts that represent a business value (Mukhopadhyay, 2018) (Marasović, 2021) (Dinesh et al., 2021) (Sweigart, 2015). Focus of the research is to draw attention and provide platform for commercial application of the presented methods in various industries where data analysis is a message valuable such as: market research (companies can use the system to analyze social media conversations and know/get insights about consumer preferences, mood analysis and trends.

This data can inform the level of product development and be used to manage marketing strategies and manage customer relationships), customer support and service (companies can use the system to automatically analyze customer support records, identify common issues, and generate reports on customer satisfaction, response time, and agent performance; this allows companies to optimize the support process and improve the customer experience), social media tracking (the system can be used to analyze WhatsApp conversations related to brands, products, or specific topics, providing valuable insights into public opinion, mood trends, and customer feedback; this information can guide social media strategies, reputation management, and brand tracking), HR and employee engagement (HR departments can leverage the system to analyze internal employee interviews to assess employee mood, identify areas of improvement, and track engagement levels; this can help organizations increase employee satisfaction, teamwork and productivity) or education (researchers and educational institutions can use the system to analyze WhatsApp chat records in educational activities; this can provide insight into student engagement, learning patterns, and teacher-student interactions, helping to develop effective training strategies), to name some most important ones.

Machine learning is a branch of artificial intelligence that deals with shaping algorithms that improve their efficiency based on empirical data (Walkowska, 2010). A machine learning model defines a product created by programming a computer to adjust the parameters of machine learning based on the given examples. It can define the purpose of using machine learning for predictive purposes or descriptive purposes. Predictive models predict future

omogućile da vide svoje razgovore unatrag u prošlost. Isti podaci dosta govore o karakteristikama razgovora i "internetske osobnosti" korisnika koja se izražava korištenjem raznih dodataka razgovoru kao što su korištenje emojija, korištenje medijskih zapisa, korištenje glasovnih poruka, prosječna vrijeme razgovora, najčešće razdoblje razgovora u danu itd.

Korisnici često nisu u potpunosti svjesni ovih karakteristika ili su ih svjesni u određenoj mjeri, ali u stvarnosti one mogu biti daleko od onoga što bi okvirno očekivali. Analizom ovih podataka može se dobiti jasna i realna predodžba o obrascima ponašanja i razgovora, što može pomoći u pokušaju promjene neke od loših navika u budućnosti ili analizom, a glavni cilj je pokušati utvrditi činjenice koje predstavljaju poslovna vrijednost (Mukhopadhyay, 2018) (Marasović, 2021) (Dinesh et al., 2021) (Sweigart, 2015). Fokus istraživanja je privući pozornost i osigurati platformu za komercijalnu primjenu predstavljenih metoda u raznim industrijama gdje je analiza podataka vrijedna poruka kao što su: istraživanje tržišta (tvrtke mogu koristiti sustav za analizu razgovora na društvenim mrežama i saznati/dobiti uvide o preferencije potrošača, analiza raspoloženja i trendova.

Ovi podaci mogu informirati o razini razvoja proizvoda i koristiti se za upravljanje marketinškim strategijama i upravljanje odnosima s klijentima), korisničku podršku i usluge (tvrtke mogu koristiti sustav za automatsku analizu evidencije korisničke podrške, identificiranje uobičajenih problema i generiranje izvješća o zadovoljstvu kupaca, vrijeme odziva i učinak agenata; to omogućuje tvrtkama da optimiziraju proces podrške i poboljšaju korisničko iskustvo), praćenje društvenih medija (sustav se može koristiti za analizu WhatsApp razgovora povezanih s robnim markama, proizvodima ili određenim temama, pružajući dragocjene uvide u javnost mišljenja, trendovi raspoloženja i povratne informacije kupaca; ove informacije mogu usmjeravati strategije društvenih medija, upravljanje ugledom i praćenje brenda), HR i angažman zaposlenika (HR odjeli mogu iskoristiti sustav za analizu internih razgovora sa zaposlenicima kako bi procijenili raspoloženje zaposlenika, identificirali područja poboljšanja, i pratiti razine angažmana; to može pomoći organizacijama da povećaju zadovoljstvo zaposlenika, timski rad i produktivnost) ili obrazovanje (istraživači i obrazovne ustanove mogu koristiti sustav za analizu WhatsApp chat zapisa u obrazovnim aktivnostima; ovo može pružiti uvid u angažman učenika, obrasce učenja i interakcije učitelj-učenik, pomažući u razvoju učinkovitih strategija obuke), da spomenemo neke najvažnije.

Strojno učenje je grana umjetne inteligencije koja se bavi oblikovanjem algoritama koji poboljšavaju njihovu učinkovitost na temelju empirijskih podataka (Walkowska, 2010). Model strojnog učenja definira proizvod stvoren programiranjem računala za prilagodbu parametara strojnog učenja na temelju

values based on the data they received as a learning base. Descriptive models contain knowledge of data based on a large set, hard-to-read data for people, and generate readable reports based on these invisible trends (Vetulani et al., 2008). Natural language processing is a branch of machine learning that deals with the interaction between computers and humans using natural language.

As an example of the natural language processing, the following situation can be mentioned:

1. Person gives voice command to computer
2. Computer record audio command
3. Converting an audio command to a text command
4. Process data from a text command
5. The data is reconverted to an audio command, this time user-oriented
6. The computer responds to a person with a voice response

Other known examples of using natural language processing (Wonderflow, 2018):

7. Google Translate
8. Check spelling in Word
9. Applications like OK Google, Siri, Cortana, and Alexa

The ultimate goal of natural language processing is to read, decipher and understand human language in a valuable and usable way. Most natural language processing algorithms rely on machine learning to know the meaning derived from human language (Pribisalić, 2019).

The aim of the research presented in this paper was to define robust methodology for automated message analysis, to nominate value propositions to different (most importantly business) applications and to provide framework for future researchers in this domain. Scientific contribution of this paper includes presentation of methods and models used alongside with most important research milestone results. In presented form, research provides an insights related to future development and business users' support expectations management in the domain of of data science.

1.1 Related Work

Library developed for this project fits alongside other community and market libraries like NLTK, GenSim, SpaCy, CoreNLP, TextBlob, AllenNLP, polyglot or scikit-learn to name few examples (Kumar, 2020). However, to maintain simplicity, ease of use, focused applicability and lighweight, the WhatsAppNLP has been developed aiming towards researchers who will continue to use the research results for motivation and novel purposes. There are similar research paths available (Kelemen, 2015) (Rosen et al., 2011) as research papers, but those are not suitable for fast-track application and implementation.

danih primjera. Može definirati svrhu korištenja strojnog učenja u prediktivne ili deskriptivne svrhe. Prediktivni modeli predviđaju buduće vrijednosti na temelju podataka koje su primili kao osnovu za učenje. Deskriptivni modeli sadrže znanje o podacima temeljeno na velikom skupu podataka teško čitljivih za ljude i generiraju čitljiva izvješća na temelju tih nevidljivih trendova (Vetulani et al., 2008). Obrada prirodnog jezika je grana strojnog učenja koja se bavi interakcijom između računala i ljudi koristeći prirodni jezik.

Kao primjer obrade prirodnog jezika može se navesti sljedeća situacija:

1. Osoba daje glasovnu naredbu računalu
2. Računalno snimanje audio naredbe
3. Pretvaranje audio naredbe u tekstualnu naredbu
4. Obrada podataka iz tekstualne naredbe
5. Podaci se ponovno pretvaraju u audio naredbu, ovaj put orijentiranu na korisnika
6. Računalo odgovara osobi glasovnim odgovorom

Ostali poznati primjeri korištenja obrade prirodnog jezika (Wonderflow, 2018):

7. Google prevoditelj
8. Provjerite pravopis u Wordu
9. Aplikacije poput OK Google, Siri, Cortana i Alexa

Krajnji cilj obrade prirodnog jezika je čitanje, dešifriranje i razumijevanje ljudskog jezika na vrijedan i upotrebljiv način. Većina algoritama za obradu prirodnog jezika oslanja se na strojno učenje kako bi saznali značenje izvedeno iz ljudskog jezika (Pribisalić, 2019).

Cilj istraživanja predstavljenog u ovom radu bio je definirati robusnu metodologiju za automatiziranu analizu poruka, nominirati prijedloge vrijednosti za različite (najvažnije poslovne) aplikacije i pružiti okvir budućim istraživačima u ovoj domeni. Znanstveni doprinos ovog rada uključuje prezentaciju metoda i modela koji se koriste uz najvažnije rezultate istraživačke prekretnice. U predstavljenom obliku, istraživanja pružaju uvid u budući razvoj i upravljanje očekivanjima podrške poslovnih korisnika u domeni znanosti o podacima.

1.1 Pristup i dosadašnji radovi

Knjižnica razvijena za ovaj projekt uklapa se uz druge knjižnice zajednice i tržišta kao što su NLTK, GenSim, SpaCy, CoreNLP, TextBlob, AllenNLP, polyglot ili scikit-learn da navedemo nekoliko primjera (Kumar, 2020). Međutim, kako bismo održali jednostavnost, jednostavnost korištenja, usredotočenu primjenjivost i lighweight, razvijena je WhatsAppNLP knjižnica s ciljem prema istraživačima koji će nastaviti koristiti naša istraživanja u motivacijske i drugee svrhe. Postoje slični istraživački putovi dostupni (Kelemen, 2015) (Rosen et al., 2011) u obliku istraživačkih radova, ali isti nisu prikladni za brzu primjenu i rad u poslovnom okruženju.

2 Modeling Topics

In the form of natural language processing, the theme implies a set of words that "go together". This refers to words that associate the user when thinking about a topic, for example "sport", "food", "music". Each of these topics has its own set of words that define it: with "sport" come words such as "football, team play, win, player, swimming, success", along with the theme of "food" words such as "bread, fish, wine, salty, spoiled, eat, drink, cook" and with the theme "music" come words such as "loud, concert, listeners, singing, performance" etc. (Khurana et al., 2022) (Wang et al., 2019) (Vetulani et al., 2000) (Sworna et al., 2022).

A theme modeling algorithm is an algorithm that independently defines topics based on assigned data from a set of data intended for model training (usually a large set of text documents) and groups words that describe it under each defined topic. A new, unknown set of data (words) is then forwarded to the created model, on the basis of which the model independently categorizes the transmitted data into a particular set, i.e. determines which topic these words belong to (Walkowska, 2009) (Hult et al., 2006) (Jagota, 2020).

2.1 Latent Dirichlet Allocation

One of the applicable machine learning methods is Latent Dirichlet Allocation (LDA). The LDA treats the probability p(z|d) as a distribution that depends on a parameter. In other words, the LDA defines themes as the distribution of words from a fixed data set by adding to each topic the probabilities for each word from the submitted dataset. The aim of this method is to find significant word distributions across various topics and significant distributions of topics by various textual documents (textual data sources) and therefore the number of topics is a key parameter of this method of machine learning. LDA is also the most widely used method of machine learning used for the purpose of modeling topics (Hurlow et al., 2003) (Marasović, 2021).

The LDA method consists of two parts: (1) the content/words belonging to the document, which is already known, and (2) the content/words belonging to the topic or the probability of words belonging to the topic, which need to be calculated. The finding algorithm goes through each content and randomly assigns each word in the document to one of the "k" topics ("k" is selected in advance). For each document "d", each word is passed through and the formula p(theme "t" | document "d") calculates the proportion of words in the document "d" that are assigned to the topic "t". The algorithm tends to capture how many words belong to the topic "t" for the given document "d", excluding the current word.

The LDA presents documents as a mix of topics. Similarly, the topic is a mixture of words. If there is a

2 Modeliranje pojmova

U obliku obrade prirodnog jezika, tema podrazumijeva skup riječi koje "idu zajedno". Ovo se odnosi na riječi koje asociraju korisnika kada razmišlja o nekoj temi, na primjer "sport", "hrana", "glazba". Svaka od ovih tema ima svoj skup riječi koje je definiraju: uz "sport" dolaze riječi kao što su "nogomet, timska igra, pobjeda, igrač, plivanje, uspjeh", zajedno s temom riječi "hrana" kao što je "kruh", riba, vino, slano, pokvareno, jesti, piti, kuhati" a uz temu "glazba" dolaze riječi kao što su "glasno, koncert, slušatelji, pjevanje, nastup" itd. (Khurana et al., 2022) (Wang et al., 2019) (Vetulani et al., 2000) (Sworna et al., 2022).

Algoritam za modeliranje teme je algoritam koji neovisno definira teme na temelju dodijeljenih podataka iz skupa podataka namijenjenih obuci modela (obično veliki skup tekstualnih dokumenata) i grupira riječi koje ih opisuju pod svakom definiranom temom. Novi, nepoznati skup podataka (riječi) zatim se prosljeđuje kreiranom modelu, na temelju čega model samostalno kategorizira prenesene podatke u određeni skup, odnosno određuje kojoj temi te riječi pripadaju (Walkowska, 2009) (Hult et al., 2006) (Jagota, 2020).

2.1 Latentna Dirichletova alokacija

Jedna od primjenjivih metoda strojnog učenja je Latent Dirichlet Allocation (LDA). LDA tretira vjerojatnost p(z|d) kao distribuciju koja ovisi o parametru. Drugim riječima, LDA definira teme kao distribuciju riječi iz fiksnog skupa podataka dodavanjem svakoj temi vjerojatnosti za svaku riječ iz poslanog skupa podataka. Cilj ove metode je pronaći značajne distribucije riječi po raznim temama i značajne distribucije tema po različitim tekstualnim dokumentima (izvorima tekstualnih podataka) te je stoga broj tema ključni parametar ove metode strojnog učenja. LDA je također najraširenija metoda strojnog učenja koja se koristi u svrhu modeliranja tema (Hurlow et al., 2003) (Marasović, 2021).

LDA metoda se sastoji od dva dijela: (1) sadržaj/riječi koje pripadaju dokumentu, koji je već poznat, i (2) sadržaj/riječi koje pripadaju temi ili vjerojatnost riječi koje pripadaju temi, što treba biti izračunat. Algoritam za pronalaženje prolazi kroz svaki sadržaj i svaku riječ u dokumentu nasumično dodjeljuje jednoj od "k" tema ("k" je odabrano unaprijed). Za svaki dokument "d" prolazi svaka riječ i formula p(tema "t" | dokument "d") izračunava udio riječi u dokumentu "d" koje su dodijeljene temi "t". Algoritam nastoji uhvatiti koliko riječi pripada temi "t" za dati dokument "d", isključujući trenutnu riječ.

LDA predstavlja dokumente kao mješavinu tema. Slično tome, tema je mješavina riječi. Ako postoji velika vjerojatnost da će se riječ naći u temi, svi dokumenti koji imaju odabranu riječ bit će uže

high probability that the word will be in the topic, all documents that have a selected word will be more closely related to the topic as well. Similarly, if the selected word is not very likely to belong to a topic, documents containing the selected word will also be very unlikely to be in the topic, because the rest of the words in the document will belong to another topic. Example of LDA (Kelemen, 2015) is the following (see sentences below).

1. I love eating fruits and vegetables.
2. For breakfast, I eat an apple and drink fruit juice.
3. Small cats can be disobedient.
4. My sister loves little kittens.
5. Rabbits love to eat carrots.

Sentences 1 and 2: 100% belong to topic A
Sentences 3 and 4: 100% belong to topic B
Sentence 5: 50% belong to topic A, 50% to topic B
Topic A: 30% fruit, 15% vegetables, 10% eat, 10% drink etc. It is possible to conclude that this is about the topic of "food".
Topic B: 30% cats, 30% rabbit, 10% small etc. It is possible to conclude that this is about the topic of "animals".

2.2 Collection and Processing of Data

In order for the exported messages to continue to be processed, the "excess" that directly affects the speed and quality of work of the machine learning model itself has to be removed from them. Each message is written in the form of a string consisting of the date and time of sending the message, the name of the sender of the message and the text of the message itself.

Example of a message record:

```
['6/23/20, 22:38 - Rok: Hahahah\n',
 '6/23/20, 22:38 - Rok: 🤦\u200dđ\n',
 '6/23/20, 22:39 - Matea: 😏😏\n',
 '6/23/20, 22:39 - Matea: GLEDAJ sam si prizano\n',
 '6/23/20, 22:39 - Matea: *priznao\n']
```

Figure 1. Example of an exported chat log record

In the case when the data for further processing in a trained model for modeling topics have to be prepared, only the records of the content of the sent message are needed. The whole process of "cleaning" messages consists of several key steps before further processing:

1. Remove the part of the message that contains information about the time and date of the message and the name of the sender of the message
2. Remove English and Croatian stopping words
3. Remove emojis and media files
4. Remove punctuation marks
5. Convert a word to the same lowercase initial letter

povezani i s temom. Slično tome, ako nije vrlo vjerojatno da odabrana riječ pripada temi, dokumenti koji sadrže odabranu riječ također će vrlo vjerojatno biti u temi, jer će ostatak riječi u dokumentu pripadati drugoj temi. Primjer LDA (Kelement, 2015) je sljedeći (vidi rečenice u nastavku).

1. Volim jesti voće i povrće.
2. Za doručak pojedem jabuku i popijem voćni sok.
3. Male mačke mogu biti neposlušne.
4. Moja sestra voli male mačiće.
5. Kunići vole jesti mrkvu.

Rečenice 1 i 2: 100% pripadaju temi A
Rečenice 3 i 4: 100% pripadaju temi B
Rečenica 5: 50% pripada temi A, 50% temi B
Tema A: 30% voće, 15% povrće, 10% jesti, 10% piti itd. Može se zaključiti da se radi o temi "hrana".
Tema B: 30% mačke, 30% zečevi, 10% mali itd. Može se zaključiti da se radi o temi "životinje".

2.2 Prikupljanje i obrada podataka

Kako bi se eksportirane poruke nastavile obrađivati, iz njih se mora ukloniti "višak" koji izravno utječe na brzinu i kvalitetu rada samog modela strojnog učenja. Svaka poruka je zapisana u obliku niza koji se sastoji od datuma i vremena slanja poruke, imena pošiljatelja poruke i teksta same poruke.

Primjer zapisa poruke:

```
['6/23/20, 22:38 - Rok: Hahahah\n',
 '6/23/20, 22:38 - Rok: 🤦\u200dđ\n',
 '6/23/20, 22:39 - Matea: 😏😏\n',
 '6/23/20, 22:39 - Matea: GLEDAJ sam si prizano\n',
 '6/23/20, 22:39 - Matea: *priznao\n']
```

Slika 1. Primjer izvezenog zapisa dnevnika razgovora

U slučaju kada je potrebno pripremiti podatke za daljnju obradu u obučenom modelu za teme modeliranja, potrebni su samo zapisi sadržaja poslane poruke. Cijeli proces "čišćenja" poruka sastoji se od nekoliko ključnih koraka prije daljnje obrade:

1. Uklonite dio poruke koji sadrži podatke o vremenu i datumu poruke te ime pošiljatelja poruke
2. Uklonite engleske i hrvatske zaustavne riječi
3. Uklonite emojije i medijske datoteke
4. Uklonite interpunkcijske znakove
5. Pretvorite riječ u isto malo početno slovo

The kod metode koja izvodi sve gore navedene korake. Rezultat nakon metode provedene nad prvim dnevnikom chata:

```
['gledaj', 'prizano', 'priznao', 'cudno', 'imma']
```

Slika 2. Primjer 1, Rezultat funkcije čišćenja podataka

The code of the method that performs all of the above steps. The result after the method carried out over the first chat log:

```
['gledaj', 'prizano', 'priznao', 'cudno', 'imma']
```

Figure 2. Example 1, Result of the data cleanup function

The result after the method carried out over the first chat log:

```
['moze', 'medo', 'tkalci', 'isto', 'mogli']
```

Figure 3. Example 2, The result of the data cleanup function

2.3 Translation of Processed Data

After processing, the data should be translated into English because the machine learning model for modeling topics is based on words in English, i.e. it is based on the English dataset. The translation process takes place using Google Sheets technology. In combination with the Python programming language, Google Sheets allows to automate work with tables that possess a built-in method for translating data. An example of a method of translating data from a Google Sheets table cell = GOOGLETRANSLATE (A1, "en"; "en"). The method receives the cell, the source language of the text, and the desired output language of the translated word as input parameters. The translation process takes place in three steps: loading data into Google Sheets, applying the built-in Google Sheets method "googletranslate()" over data, and retrieving translated words from Google Sheet-s tables into a previously created field in the program itself. In order to automate the process of uploading, translating and retrieving translated data, Google requires its users to register and create a Google application programming interface (Google API). All data related to the granting of rights to the user, as well as the assignment of API tokens for access to the Google Sheet application, can be found in service_account.json. The file is used as an input parameter in the pygsheets.authorize (service_account_file = 'service_account.json') method that opens a link to the Google Sheets account where the data translation process will take place. In order to bring the data to Google Sheets, it must first be stored in a .csv format file in order to use the Pandas Python library method: pandas.read_csv(file). After that, using an open link to Google Sheets, the method set_dataframe is used, which receives as input parameters a previously read pandas file, and the coordinates of the fields in the Google Sheets table on which the data from the pandas file will start to be typed. After successfully loading the data, there is a translation method in the cells of the adjacent field "=googletranslate()" which automatically translates

Rezultat nakon metode provedene nad prvim dnevnikom chata:

```
['moze', 'medo', 'tkalci', 'isto', 'mogli']
```

Slika 3. Primjer 2, rezultat funkcije čišćenja podataka

2.3 Prevođenje obrađenih podataka

Nakon obrade podatke je potrebno prevesti na engleski jer se model strojnog učenja za teme modeliranja temelji na riječima na engleskom, odnosno temelji se na engleskom setu podataka. Proces prevođenja odvija se pomoću tehnologije Google Sheets. U kombinaciji s programskim jezikom Python, Google Sheets omogućuje automatizaciju rada s tablicama koje posjeduju ugrađenu metodu za prevođenje podataka. Primjer metode prevođenja podataka iz ćelije tablice Google tablica = GOOGLETRANSLATE (A1, " en "; " en "). Metoda prima ćeliju, izvorni jezik teksta i željeni izlazni jezik prevedene riječi kao ulazne parametre. Proces prevođenja odvija se u tri koraka: učitavanje podataka u Google tablice, primjena ugrađene metode Google tablica " googletranslate ()" nad podacima i dohvaćanje prevedenih riječi iz tablica Google tablica u prethodno kreirano polje u samom programu. Kako bi automatizirao proces učitavanja, prevođenja i dohvaćanja prevedenih podataka, Google od svojih korisnika zahtijeva da se registriraju i kreiraju Google sučelje za programiranje aplikacija (Google API). Svi podaci vezani uz dodjelu prava korisniku, kao i dodjelu API tokena za pristup aplikaciji Google Sheet nalaze se u service_account.json. Datoteka se koristi kao ulazni parametar u metodi pygsheets.authorize (service_account_file = ' service_account.json ') koja otvara vezu na račun Google tablica na kojem će se odvijati proces prevođenja podataka. Kako bi se podaci prenijeli u Google tablice, prvo moraju biti pohranjeni u datoteci.csv formata kako bi se koristila metoda biblioteke Pandas Python: pandas.read_csv (datoteka). Nakon toga se otvorenom poveznicom na Google Sheets koristi metoda set_dataframe koja kao ulazne parametre prima prethodno pročitanu pandas datoteku, te koordinate polja u Google Sheets tablici na kojima će se početi unositi podaci iz pandas datoteke. biti upisan. Nakon uspješnog učitavanja podataka, u ćelijama susjednog polja postoji metoda prevođenja "= googletranslate ()" koja automatski prevodi susjedna polja s prethodno unesenim podacima (Grossz et al., 1986). S obzirom da se radi o velikoj količini podataka koje je potrebno prevesti, potrebno je napraviti pauzu s preuzimanjem podataka kako bi se obavio posao prevođenja nad svim podacima koji su u redu za prevođenje. Određena je pauza od punih pet minuta, što je dovoljno vremena da se s hrvatskog na engleski prevede stotinjak tisuća riječi. Nakon prevođenja podaci se dohvaćaju metodom – get_col () koja kao ulazni parametar dobiva redni broj stupca Google Sheet tablice iz kojeg

adjacent fields with previously entered data (Grossz et al., 1986). Given that this is a large amount of data that needs to be translated, it is necessary to take a break with the download of data in order to perform the job of translation over all data in the queue for translation. The break is set for a full five minutes, which is enough time to translate about one hundred thousand words from Croatian to English. After translation, the data is retrieved by the method – get_col(), which receives as an input parameter the ordinal number of the Google Sheet table column from which all the data contained in that column will be retrieved and stored in the previously defined field "values_list". As a final step in the translation process, the translated data needs to be processed once again. This step is necessary because the translated data is no longer separate words. The "googletranslate()" method translates most Croatian words in such a way as to create unnecessary text, which directly acts on the method of machine learning of topic recognition because it creates a false impression of the topics present. An example of creating unnecessary text is translating the word "I run" from Croatian to English using the googletranslate() method. The word is translated as "I am running", which is originally correct, but in that newly created sentence, the word "I" and the word "am" represent an excess that needs to be removed. The process of cleaning translated words is due to the method of prepare_translated_words_for_model() which receives as an input parameter a field of previously retrieved translated words. The method performs several operations on words: a "lemmatisis" operation that reduces the word to its shape in the dictionary, an operation to remove words with less than three letters, an operation to remove punctuation marks, and an operation to remove English stopping words. After processing the translated data, we get a field of translated words, which is further used as the input parameter of the method for identifying topics of conversation and as an input parameter of the method for obtaining statistical data on the number of most represented words in a conversation.

This approach of converting text documents into a field of separate words is called the "sack of words" approach.

An example of the translated data on the first chat log:

```
['lapsuz', 'free', 'friday', 'tomorrow', 'long']
```

Figure 4. Example 1. translated words

An example of translated data on another chat log:

```
['drink', 'account', 'drink', 'dump', 'team']
```

Figure 5. Example 2. translated words

će se dohvatiti svi podaci sadržani u tom stupcu i pohraniti u prethodno definirano polje " popis_vrijednosti ". Kao posljednji korak u procesu prevođenja, prevedene podatke potrebno je još jednom obraditi. Ovaj korak je neophodan jer prevedeni podaci više nisu zasebne riječi. Metoda " googletranslate ()" prevodi većinu hrvatskih riječi na način da stvara nepotreban tekst, što izravno djeluje na metodu strojnog učenja prepoznavanja tema jer stvara lažni dojam o prisutnim temama. Primjer stvaranja nepotrebnog teksta je prevođenje riječi "I run" s hrvatskog na engleski metodom googletranslate (). Riječ je prevedena kao "trčim", što je izvorno točno, ali u toj novostvorenoj rečenici riječ "ja" i riječ "jesam" predstavljaju višak koji treba ukloniti. Proces čišćenja prevedenih riječi je zahvaljujući metodi priprave_translated_words_for-_model() koja kao ulazni parametar prima polje prethodno dohvaćenih prevedenih riječi. Metoda izvodi nekoliko operacija na riječima: operaciju " lemmatisis " koja reducira riječ na njen oblik u rječniku, operaciju za uklanjanje riječi s manje od tri slova, operaciju za uklanjanje interpunkcijskih znakova i operaciju za uklanjanje engleskih zaustavnih riječi. Nakon obrade prevedenih podataka dobivamo polje prevedenih riječi koje se dalje koristi kao ulazni parametar metode za identifikaciju tema razgovora i kao ulazni parametar metode za dobivanje statističkih podataka o broju najzastupljenijih riječi u razgovor.

Ovaj pristup pretvaranja tekstualnih dokumenata u polje zasebnih riječi naziva se pristup "vreće riječi".

Primjer prevedenih podataka na prvom dnevniku chata:

```
['lapsuz', 'free', 'friday', 'tomorrow', 'long']
```

Slika 4. Primjer 1. prevedene riječi

Primjer prevedenih podataka na drugom dnevniku chata:

```
['drink', 'account', 'drink', 'dump', 'team']
```

Slika 5. Primjer 2. prevedene riječi

2.4. Model strojnog učenja

Postoji nekoliko načina za treniranje modela strojnog učenja. Svaki pristup ima svoju upotrebu, prednosti i nedostatke. Kao metoda uvježbavanja modela strojnog učenja identificiranja tema razgovora korištena je prethodno spomenuta metoda latentne Dirichletove alokacije (LDA). Kao skup tekstualnih podataka (eng. Data set) korišten je skup podataka "20 newsgroup". Skup podataka "20 newsgorup " zbirka je tekstualnih dokumenata koji se sastoje od 18.828 tekstualnih članaka, poruka i e-mailova, od kojih je svaki kategoriziran pod jednim od 20 naslova.

Ovaj skup podataka jedan je od najčešće korištenih skupova podataka u obradi prirodnog jezika, namijenjen eksperimentiranju s metodama strojnog

2.4 Machine Learning Model

There are a number of ways to train machine learning models. Each approach has its own uses, pros and cons. As a method of training machine learning models of identifying topics of conversation, the previously mentioned method of latent Dirichlet allocation (LDA) has been used. As a set of textual data (eng. Data set) the "20 newsgroup" dataset has been used. The "20 newsgorup" dataset is a collection of text documents consisting of 18,828 text articles, messages and e-mails, each of which is categorized under one of 20 headings. This dataset is one of the most commonly used datasets in natural language processing, intended to experiment with machine learning methods of categorizing and grouping textual data. In the case of training the model of identifying topics of conversation, it is one of the best possible choices because it covers a variety of topics that include politics, religion, sports, space and medicine, which directly affects the range of topics that can be recognized by the model in a given set of text messages from the WhatsApp application. The dataset as well as its processing methods and training methods of LDA models are found in the publicly available "Genism" and "Sklearn" Python libraries. The first step in training the LDA theme recognition model is to retrieve the "20newsgroup" dataset. Example:

import gensim from gensim.models import *LdaModel*
from sklearn.datasets import *fetch_20newsgroups*
newsgroups_train =
fetch_20newsgroups(subset='train', shuffle = ***True)***

Given that this is uncleaned data containing text unusable parts, the data needs to be cleaned using the preprocess() method, which receives a text document from a "20newsgroup" file as an input parameter. Pre-processing example of "20newsgroup" dataset:

processed_docs = []
for doc **in** newsgroups_train. Data:
processed_docs. append(preprocess(doc))

The "preprocess()" method performs several operations on data. An operation to remove punctuation marks, an operation to remove stopping words in English, an operation to convert a word into a lowercase initial letter, and a "leminizing" operation. Lemmatizing – converting words to dictionary form). After processing, each sentence in the data set is stored in the form of a separate word field, that is, in the form of a sentence field, each of which is a word field. Example of a field obtained using the preprocess():
The created field is additionally filtered by the extreme filtering method filter_extremes() which receives as input parameters the number of minimal repetitions of a single word, the percentage of the number of words that appear in all documents and the number of words that are retained after the filtering has been done. After učenja kategorizacije i grupiranja tekstualnih podataka. U slučaju treniranja modela identificiranja tema razgovora, to je jedan od najboljih mogućih izbora jer pokriva niz tema koje uključuju politiku, religiju, sport, svemir i medicinu, što izravno utječe na raspon tema koje se mogu koje model prepoznaje u zadanom skupu tekstualnih poruka iz aplikacije WhatsApp. Skup podataka kao i njegove metode obrade i metode obuke LDA modela nalaze se u javno dostupnim Python bibliotekama "Genism" i " Sklearn ". Prvi korak u obuci modela prepoznavanja tema LDA je dohvaćanje skupa podataka "20newsgroup". Primjer:

import gensim iz gensim.modeli import *LdaModel*
iz sklearn.datasets import *fetch_20newsgroups*
newsgroups_train =
fetch_20newsgroups(subset='train', shuffle = ***True)***

S obzirom da se radi o neočišćenim podacima koji sadrže tekst neupotrebljivih dijelova, podatke je potrebno očistiti metodom preprocess(), koja prima tekstualni dokument iz datoteke "20newsgroup" kao ulazni parametar. Primjer prethodne obrade skupa podataka "20newsgroup":

obrađeni_dokumenti = []
za doc _ newsgroups_train. Podaci:
obrađeni_dokumenti.
dodati(pretproces(doc))

Metoda "preprocess()" izvodi nekoliko operacija na podacima. Operacija uklanjanja interpunkcijskih znakova, operacija uklanjanja zaustavnih riječi na engleskom jeziku, operacija pretvaranja riječi u početno malo slovo i operacija " leminizacije ". Lematiziranje – pretvaranje riječi u oblik rječnika). Nakon obrade svaka rečenica u skupu podataka pohranjuje se u obliku zasebnog polja riječi, odnosno u obliku polja rečenice od kojih je svako polje riječi. Primjer polja dobivenog korištenjem preprocess():
Kreirano polje se dodatno filtrira metodom ekstremnog filtriranja filter_extremes () koja kao ulazne parametre prima broj minimalnih ponavljanja pojedine riječi, postotak broja riječi koje se pojavljuju u svim dokumentima i broj riječi koje se zadržavaju nakon filtriranje je obavljeno. Nakon filtriranja, pomoću metode doc2bow(), polje se pretvara u vreću riječi. Bag of words), te je spreman kao ulazni parametar LDA metode za razvoj modela strojnog učenja.

```
[['lerxst', 'thing', 'subject', 'nntp', 'post', 'host', 'organ', 'univers', 'maryland', 'colle
g', 'park', 'line', 'wonder', 'enlighten', 'door', 'sport', 'look', 'late', 'earli', 'call', 'br
icklin', 'door', 'small', 'addit', 'bumper', 'separ', 'rest', 'bodi', 'know', 'tellm', 'model',
'engin', 'spec', 'year', 'product', 'histori', 'info', 'funki', 'look', 'mail', 'thank', 'brin
g', 'neighborhood', 'lerxst'], ['guykuo', 'carson', 'washington', 'subject', 'clock', 'poll', 'f
inal', 'summari', 'final', 'clock', 'report', 'keyword', 'acceler', 'clock', 'upgrad', 'articl',
'shelley', 'qvfo', 'innc', 'organ', 'univers', 'washington', 'line', 'nntp', 'post', 'host', 'ca
rson', 'washington', 'fair', 'number', 'brave', 'soul', 'upgrad', 'clock', 'oscil', 'share', 'ex
peri', 'poll', 'send', 'brief', 'messag', 'detail', 'experi', 'procedur', 'speed', 'attain', 'ra
t', 'speed', 'card', 'adapt', 'heat', 'sink', 'hour', 'usag', 'floppi', 'disk', 'function', 'flo
ppi', 'especi', 'request', 'summar', 'day', 'network', 'knowledg', 'base', 'clock', 'upgrad', 'h
aven', 'answer', 'poll', 'thank', 'guykuo', 'washington']]
```

Slika 6. Polje dobiveno metodom preprocess().

filtering, using the doc2bow() method, the field is converted into a sack of words . Bag of words), and is ready as an input parameter of the LDA method for the development of machine learning models.

```
[['lerxst', 'thing', 'subject', 'nntp', 'post', 'host', 'organ', 'univers', 'maryland', 'colle
g', 'park', 'line', 'wonder', 'enlighten', 'door', 'sport', 'look', 'late', 'earli', 'call', 'br
icklin', 'door', 'small', 'addit', 'bumper', 'separ', 'rest', 'bodi', 'know', 'tellm', 'model',
'engin', 'spec', 'year', 'product', 'histori', 'info', 'funki', 'look', 'mail', 'thank', 'brin
g', 'neighborhood', 'lerxst'], ['guykuo', 'carson', 'washington', 'subject', 'clock', 'poll', 'f
inal', 'summari', 'final', 'clock', 'report', 'keyword', 'acceler', 'clock', 'upgrad', 'articl',
'shelley', 'qvfo', 'innc', 'organ', 'univers', 'washington', 'line', 'nntp', 'post', 'host', 'ca
rson', 'washington', 'fair', 'number', 'brave', 'soul', 'upgrad', 'clock', 'oscil', 'share', 'ex
peri', 'poll', 'send', 'brief', 'messag', 'detail', 'experi', 'procedur', 'speed', 'attain', 'ra
t', 'speed', 'card', 'adapt', 'heat', 'sink', 'hour', 'usag', 'floppi', 'disk', 'function', 'flo
ppi', 'especi', 'request', 'summar', 'day', 'network', 'knowledg', 'base', 'clock', 'upgrad', 'h
aven', 'answer', 'poll', 'thank', 'guykuo', 'washington']]
```

Figure 6. Field obtained by preprocess() method

2.5 LDA Machine Learning Method

Training models using the LDA machine learning method makes it much easier for to use the "Gensim" Python library. Model training requires the gensim.models.LdaMulticore() method. The method receives several input parameters: bow_corpus, Num_topic, Id2word, Passes and Workers. The bow_corpus parameter represents a sack of words that serves as a training data set. The num_topic parameter represents the number of topics that will be categorized by words from a given bag of words. The id2word parameter is used to map each word to its integer. This parameter was used in the later scrolling of categorized topics. The pass parameter is the number of repetitions through a set of data when training a model. It is advisable to put this parameter on a number greater than one, so that the trained model has as accurately categorized words by topic as possible. The workers parameter defines the number of additional processor cores used to train the model, with the aim of training the model as quickly as possible. An example of modelmaking using the LdaMulticore() method:

lda_model=
gensim.models.LdaMulticore(bow_corpus,
num_topics = 8, id2word = dictionary, passes = 10,
workers = 2)

After completing the model training process, it is possible to get an insight into the result of the model by scrolling through categorized words, i.e. scrolling through defined topics.

for idx, topic **in** lda_model. print_topics(-1):
print("Topic: **{}** **n**Words: **{}**". format(idx, topic)) print("**n**")

This loop shows a categorized topic and its words that describe it. Next to each word in a particular topic is a numerical record of the weight of a word that represents the most common word in that category. The more represented a word in a particular categorized topic, when processing a bag of unknown words (words that come as an input parameter for

2.5 LDA metoda strojnog učenja

Modeli obuke pomoću metode strojnog učenja LDA znatno olakšavaju korištenje biblioteke " Gensim " Python. Obuka modela zahtijeva metodu gensim.models.LdaMulticore (). Metoda prima nekoliko ulaznih parametara: bow_corpus, Num_topic, Id2word, Passes i Workers. Parametar bow_corpus predstavlja vreću riječi koja služi kao skup podataka za obuku. Parametar num_topic predstavlja broj tema koje će biti kategorizirane prema riječima iz zadane vrećice riječi. Parametar id2word koristi se za mapiranje svake riječi u njezin cijeli broj. Ovaj je parametar korišten u kasnijem listanju kategoriziranih tema. Parametar prolaza je broj ponavljanja kroz skup podataka prilikom treniranja modela. Preporučljivo je ovaj parametar staviti na broj veći od jedan, kako bi obučeni model imao što točnije kategorizirane riječi po temi. Radni parametar definira broj dodatnih procesorskih jezgri koje se koriste za treniranje modela, s ciljem što bržeg treniranja modela. Primjer izrade modela korištenjem metode LdaMulticore ():

lda_model = gensim.models.LdaMulticore
(bow_corpus, num_topics = 8, id2word = rječnik,
propusnice = 10, radnici = 2)

Nakon završetka procesa obuke modela, moguće je dobiti uvid u rezultat modela skrolanjem kroz kategorizirane riječi, odnosno skrolanjem kroz definirane teme.

za idx, tema **u** lda_model. print_topics (-1):
print("Tema: **{}** \ **n** Riječi : **{}** ". format(idx, tema)) ispis(" **n** ")

Ova petlja prikazuje kategoriziranu temu i riječi koje je opisuju. Uz svaku riječ u pojedinoj temi nalazi se brojčani zapis težine riječi koja predstavlja najčešću riječ u toj kategoriji. Što je neka riječ zastupljenija u određenoj kategoriziranoj temi, prilikom obrade baga nepoznatih riječi (riječi koje dolaze kao ulazni parametar za identifikaciju tema, u našem slučaju WhatsApp poruka), to je veća vjerojatnost prepoznavanja tema na temelju dodijeljenog baga riječi s nekategoriziranim riječima.

Primjer kategoriziranih tema dobivenih strojnim učenjem:

Tema:1
Riječi: 0.018"window" + 0.017*"file" +*
0.011"program" + 0.007*" slika " +*
0.007"version" + 0.006*"graphic" + 0.006*"avail"*
+ 0.006" softwar " + 0,006*"boja" +*
0,005"poslužitelj"*

Slika 7. Primjer kategorizirane teme dobivene strojnim učenjem

identifying topics, in our case WhatsApp messages), the greater the probability of recognizing topics based on an assigned bag of words with uncategorized words.

Example of categorized topics obtained by machine learning:

Topic:1
Words: 0.018"window" + 0.017*"file" + 0.011*"program" + 0.007*"imag" + 0.007*"version" + 0.006*"graphic" + 0.006*"avail" + 0.006*"softwar" + 0.006*"color" + 0.005*"server"*

Figure 7. Example of categorized topic obtained by machine learning

3 Model Testing

The model has been tested on an unknown text record consisting of several related sentences (sentences of the same subject matter). Given that the trained model knows how to recognize the topic of religion, as a test text a textual record of religious themes has been used.

The text record should initially be converted into a bag of words, after which we bring it as an input parameter to the trained model. The result is presented in the form of the achieved representation of the results (engl. score) determining the topic and the words that describe it (engl. topic). An example of testing a created model for identifying topics:

testing_topic = 'One of the key points of Jesus' ministry was how the Kingdom of God is accessible to all people. This underlying theme is especially evident in parables like "The Good Samaritan", "A Lost Sheep", and "A Lost Son". These parables are used to explain how Samaritans, the lost, and sinners all can find their way back to God, and how they will be greeted and embraced by a loving Jesus. These stories illustrate one of Jesus' main teachings, that all people who have faith in Him, will be accepted and embraced by God in heaven.'

Data preprocessing step for the unseen document:
bow_vector = dictionary.
doc2bow(preprocess(testing_topic))
for index, score **in** sorted(lda_model[bow_vector], key=**lambda** tup: -1*tup[1]): print("Score: **{}\t** Topic: **{}**". format(score, lda_model.
print_topic(index, 5)))

Output parameter:
Score: 0.9686757922172546
Topic: 0.013"christian" + 0.008*"jesus" + 0.007*"exist" + 0.005*"bibl" + 0.005*"morality"*

From the above example, we were convinced that the model is capable (with a representation of 97%),

3 Testiranje modela

Model je testiran na nepoznatom tekstualnom zapisu koji se sastoji od više povezanih rečenica (rečenica iste tematike). S obzirom na to da obučeni model zna prepoznati temu vjere, kao ispitni tekst korišten je tekstualni zapis religijske tematike.

Tekstualni zapis treba inicijalno pretvoriti u torbu riječi, nakon čega ga unosimo kao ulazni parametar u obučeni model. Rezultat se prikazuje u obliku postignutog prikaza rezultata (engl. score) određujući temu i riječi koje je opisuju (engl. topic). Primjer testiranja izrađenog modela za identifikaciju tema:

testing_topic = 'Jedna od ključnih točaka Isusove službe bila je kako je Kraljevstvo Božje dostupno svim ljudima. Ova temeljna tema posebno je vidljiva u parabolama kao što su "Dobri Samaritanac", "Izgubljena ovca" i "Izgubljeni sin". Ove se prispodobe koriste da objasne kako Samarićani, izgubljeni i grešnici mogu pronaći svoj put natrag do Boga i kako će ih pozdraviti i zagrliti Isus pun ljubavi. Ove priče ilustriraju jedno od Isusovih glavnih učenja, da će svi ljudi koji imaju vjere u Njega, biti prihvaćeni i zagrljeni od Boga na nebu.'

predobrade podataka za nevidljivi dokument:
luk_vektor = rječnik.
doc2bow(pretproces(test_topic))
za indeks, rezultat **u** sorted(lda_model [bow_vector], key= **lambda** tup: -1*tup[1]): print("Rezultat: **{}\t** Tema: **{}** ". format(rezultat, lda_model.
print_topic (indeks, 5)))

Izlazni parametar:
Ocjena: 0,9686757922172546
Tema: 0,013" kršćanin " + 0,008*" isus " + 0,007*"postojati" + 0,005*" bibl " + 0,005*"moral"*

Iz gornjeg primjera smo se uvjerili da je model sposoban (s zastupljenošću od 97%), na temelju vreće riječi kao ulaznog parametra, prepoznati predmetnu temu.

3.1 Testiranje modela na stvarnim primjerima zapisa chata WhatsApp

WhatsAppNLP biblioteka ima metodu koja izvršava cijeli, prethodno spomenuti i opisani proces obrade, prevođenja i obrade prevedenih podataka, modela obuke i ispisa prepoznatih tema na temelju proslijeđenih podataka iz WhatsApp chat dnevnika.

Metoda analyize_chat_topics () radi upravo to, na temelju proslijeđene eksportirane datoteke sa zapisom svih poruka u tekstualnom formatu i brojem željenih tema na temelju kojih će se kreirati LDA model, prepoznaje teme prisutne u proslijeđenoj Dnevnik chata WhatsApp. Demonstracija na dva stvarna primjera WhatsApp dnevnika razgovora:

based on the sack of words as the input parameter, to recognize the topic in question.

3.1 Testing Models on Real Examples of WhatsApp Chat Logs

WhatsAppNLP library has a method that executes the entire, previously mentioned and described process of processing, translating and processing translated data, training models, and printing recognized topics based on forwarded data from the WhatsApp chat log.

The method analyize_chat_topics() does just that, based on the forwarded exported file with a record of all messages in text format and the number of desired topics on the basis of which the LDA model will be created, it recognizes the topics present in the forwarded WhatsApp chat log. A demonstration on two real examples of WhatsApp chat logs:

```
from data_preprocessing import *
file = 'chat_with_person1.txt'
analyize_chat_topics(file, 3)
```

The result of the function over the first chat log:

```
Score: 0.6965230107307434
Topic: 0.006*"drive" + 0.005*"game" +
0.004*"team" + 0.004*"card" + 0.004*"play"
```

From the first obtained result it is possible to conclude that the top 3 topics of conversation from the attached chat log, topics related to "team games", "religion" and "technology" with a representation of 70%, 16% and 14%.

```
from data_preprocessing import *
file = 'chat_with_person2.txt'
analyize_chat_topics(file, 3)
```

The result of the function over the second chat log:

```
Score: 0.4966925084590912
Topic: 0.012*"armenian" + 0.007*"english" +
0.007*"turkish" + 0.004*"live" + 0.004*"greek"
```

As a result of the analysis over the second chat log, it can be concluded that these are more diverse words and that the topic recognition algorithm predicts the result with less representation (more less representation implies more topics, not less accuracy). From the attached we can conclude that the top 3 topics are related to "nationalities", "universe" and "team games". As is evident from the attached, the model is not able to title categorized topics. The logical reasoning process remains with the user of the WhatsAppNLP library.

```
from data_preprocessing import *
file = 'chat_with_person1.txt'
analyize_chat_topics (datoteka, 3)
```

Rezultat funkcije u prvom dnevniku razgovora:

```
Rezultat: 0,6965230107307434
Tema: 0,006*"drive" + 0,005*"game" +
0,004*"team" + 0,004*"card" + 0,004*"play"
```

Iz prvog dobivenog rezultata moguće je zaključiti da su top 3 teme razgovora iz priloženog chat dnevnika, teme vezane uz "timske igre", "religiju" i "tehnologiju" sa zastupljenošću od 70%, 16% i 14%.

```
from data_preprocessing import *
file = 'chat_with_person2.txt'
analyize_chat_topics (datoteka, 3)
```

Rezultat funkcije preko drugog dnevnika razgovora:

```
Rezultat: 0,4966925084590912
Tema: 0,012*" armenski " + 0,007*" engleski " +
0,007*" turski " + 0,004*"uživo" + 0,004*" grčki "
```

Kao rezultat analize drugog chat dnevnika može se zaključiti da se radi o više različitih riječi i da algoritam za prepoznavanje tema predviđa rezultat s manjom zastupljenošću (više manje zastupljenosti implicira više tema, a ne manju točnost). Iz priloženog možemo zaključiti da su top 3 teme vezane uz "nacionalnosti", "svemir" i "timske igre". Kao što je vidljivo iz priloženog, model nije u mogućnosti nasloviti kategorizirane teme. Logički proces zaključivanja ostaje na korisniku biblioteke WhatsAppNLP.

3.2 Statistika odabranih poruka

Osim identificiranja tema, biblioteka WhatsAppNLP sadrži metode matematičkog izračunavanja statistike poruka koje se odnose na dijelove zapisa poruka kao što su datum i vrijeme poslane poruke, najčešće korištene riječi po sudioniku u izvezenom razgovoru, najčešća razdoblja razgovora u dan, tjedan i mjesec, najčešće korišteni emoji, oblaci riječi i slično.

Statistika se prikazuje u grafičkom, numeričkom ili tekstualnom prikazu, na jednostavan i intuitivan način. Sve metode zahtijevaju samo priloženi put do eksportiranih podataka u tekstualnom formatu kao ulazni parametar. Svaka od ovih metoda radi na sličnim principima.

Iz proslijeđenog dnevnika razgovora stvara se "Pandas DataFrame " u kojem se podaci sortiraju, filtriraju i prikazuju u obliku grafikona, matematičkih izračuna i tekstualnih izvješća pomoću "Pandas"

3.2 Selected Message Statistics

In addition to identifying topics, the WhatsAppNLP library contains methods of mathematically calculating message statistics related to parts of message records such as the date and time of message sent, the most commonly used words per participant in the exported conversation, the most common conversation periods in the day, week and month, the most commonly used emojis, word clouds and the like.

Statistics are displayed in graphical, numerical or textual representations, in a simple and intuitive way. All methods require only the attached path to the exported data in text format as an input parameter. Each of these methods works on similar principles.

From the forwarded chat log, a "Pandas DataFrame" is created in which data is sorted, filtered and displayed in the form of graphs, mathematical calculations and text reports using "Pandas" built-in methods. List and demonstration of methods on two analyzed chat logs.

Description and name of the method messages_per_user (file_path) is a method used to analyze the number of messages per participant in an exported conversation. The sum of all sent messages is displayed in the form of a text report.

```
name
Matea    22103
Rok      17896
Name: name, dtype: int64
```

Figure 8. Examples of the messages_per_user method()

From the pictures it can be see the total number of messages sent by participants of two different chat logs. From the first example, it can be concluded that Rok in the above example sends fewer messages in both exported conversations, and Antonio and Matea more. Description and name of the method: media_per_user(file_path) is a method similar to the previous method, which as an output parameter generates the sum of all media records per user in the exported chat log.

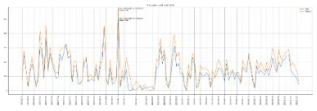


Figure 9. Example of the messages_over_time() method

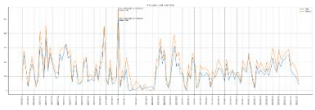
The figure shows the total number of messages sent in a given time period. The first example shows a continuous conversation through almost all dates with minor deviations in the middle of the conversation. The second example shows a graph from which can be concluded that this is an occasional conversation with deviations in regularity and the number of messages sent at the very end. The values on the x-axis represent ugrađenih metoda. Popis i demonstracija metoda na dva analizirana dnevnika razgovora.

Opis i naziv metode messages_per_user (file_path) je metoda koja se koristi za analizu broja poruka po sudioniku u izvezenom razgovoru. Zbroj svih poslanih poruka prikazuje se u obliku tekstualnog izvješća.

```
name
Matea    22103
Rok      17896
Name: name, dtype: int64
```

Slika 8. Primjeri metode messages_per_user ()

Na slikama se može vidjeti ukupan broj poruka koje su poslali sudionici dva različita chat dnevnika. Iz prvog primjera se može zaključiti da Rok u gornjem primjeru šalje manje poruka u oba izvezena razgovora, a Antonio i Matea više. Opis i naziv metode: media_per_user (file_path) je metoda slična prethodnoj metodi, koja kao izlazni parametar generira zbroj svih medijskih zapisa po korisniku u izvezenom chat logu.



Slika 9. Primjer metode messages_over_time ().

Slika prikazuje ukupan broj poruka poslanih u određenom vremenskom razdoblju. Prvi primjer prikazuje kontinuirani razgovor kroz gotovo sve datume s manjim odstupanjima u sredini razgovora. Drugi primjer prikazuje grafikon iz kojeg se može zaključiti da se radi o povremenom razgovoru s odstupanjima u redovitosti i broju poslanih poruka na samom kraju. Vrijednosti na x-osi predstavljaju kontinuirane vrijednosti, dok je konačni izgled vizualizacije posljedica fragmentacije načina komunikacije koja nije u potpunosti u realnom vremenu već ovisi o responzivnosti sudionika u razgovoru. Opis i naziv metode: emojis_per_sender (file_path) je metoda koja analizira prisutne emojije pronađene u izvezenom dnevniku chata i ispisuje broj korištenja određenog emojija s prikazom samog emojija po korisniku koji ga je koristio.

3.3 Ograničenja

Predložena ideja također ima određena ograničenja koja treba priznati. Prvo, biblioteka WhatsAppNLP oslanja se na to da korisnici ručno spremaju svoje razgovore u tekstualnom formatu, što može dovesti do potencijalnih pogrešaka ili propusta. Štoviše, analiza koju provodi knjižnica ovisi o točnosti i mogućnostima korištenih metoda obrade prirodnog jezika, što može utjecati na kvalitetu generiranih uvida. Nadalje, analiza knjižnice ograničena je na

continuous values, while the final appearance of the visualization is due to the fragmentation of the mode of communication that is not completely in real time but depends on the responsiveness of the participants in the conversation. Description and name of the method: emojis_per_sender(file_path) is a method that analyzes the emojis present found in the exported chat log and prints the number of uses of a particular emoji with the display of the emoji itself per user who used it.

3.3 Limitations

The proposed idea also has certain limitations that should be acknowledged. Firstly, the WhatsAppNLP library relies on users manually saving their conversations in text format, which may introduce potential errors or omissions. Moreover, the analysis performed by the library is dependent on the accuracy and capabilities of the natural language processing methods employed, which can impact the quality of insights generated. Additionally, the library's analysis is limited to the content of the messages and does not consider other contextual factors that may influence the interpretation of the data.

Furthermore, the automated analysis provided by the library leaves the task of data interpretation solely to the user, potentially leading to subjective or biased conclusions. Finally, while the approach enables analysis according to security protocols, privacy concerns and ethical considerations should be carefully addressed to ensure compliance with data protection regulations and user consent.

3.4 Business Value Proposition

The business value proposition from this research and the WhatsAppNLP library is to offer businesses and organizations a powerful tool for understanding and extracting meaningful information from short digital messages. It not only enhances work efficiency but also facilitates data-driven decision-making, allowing businesses to make informed choices and stay competitive in their respective industries. Additionally, the library's focus on ethical analysis and privacy protection ensures that it aligns with responsible data usage practices, increasing trust and credibility among users. Library provides a streamlined approach to analyzing short digital messages. By employing natural language processing methods and offering simple method calls, the analysis process is made more accessible to users, even those without extensive expertise in data science or NLP. This ease of use enhances the overall efficiency of analyzing the messages. It takes advantage of the data generated from the WhatsApp application. Since users can freely download and save their conversations in text format, the library can efficiently process and analyze this data, saving time and resources that would otherwise be spent on manual data collection. It also sadržaj poruka i ne uzima u obzir druge kontekstualne čimbenike koji mogu utjecati na tumačenje podataka.

Nadalje, automatizirana analiza koju pruža knjižnica prepušta zadatak tumačenja podataka isključivo korisniku, što potencijalno dovodi do subjektivnih ili pristranih zaključaka. Konačno, dok pristup omogućuje analizu u skladu sa sigurnosnim protokolima, brige o privatnosti i etička pitanja treba pažljivo razmotriti kako bi se osigurala usklađenost s propisima o zaštiti podataka i pristankom korisnika.

3.4 Poslovna vrijednost

Prijedlog poslovne vrijednosti iz ovog istraživanja i WhatsAppNLP biblioteke je ponuditi tvrtkama i organizacijama moćan alat za razumijevanje i izvlačenje smislenih informacija iz kratkih digitalnih poruka. Ne samo da poboljšava radnu učinkovitost, već i olakšava donošenje odluka na temelju podataka, omogućujući poduzećima da donesu informirane odluke i ostanu konkurentna u svojim industrijama. Osim toga, usredotočenost knjižnice na etičku analizu i zaštitu privatnosti osigurava da je u skladu s odgovornim praksama korištenja podataka, povećavajući povjerenje i vjerodostojnost među korisnicima. Knjižnica pruža pojednostavljeni pristup analizi kratkih digitalnih poruka. Korištenjem metoda obrade prirodnog jezika i ponudom jednostavnih metodskih poziva, postupak analize postaje dostupniji korisnicima, čak i onima bez opsežne stručnosti u znanosti o podacima ili NLP-u. Ova jednostavnost korištenja povećava ukupnu učinkovitost analize poruka. Iskorištava podatke generirane iz aplikacije WhatsApp. Budući da korisnici mogu slobodno preuzimati i spremati svoje razgovore u tekstualnom formatu, knjižnica može učinkovito obrađivati i analizirati te podatke, štedeći vrijeme i resurse koji bi se inače potrošili na ručno prikupljanje podataka. Također generira intuitivna izvješća u različitim formatima, kao što su grafikoni, tablice i tekstualne poruke. Ova vizualno privlačna i lako razumljiva izvješća omogućuju korisnicima da brzo shvate ključne uvide iz podataka. Ova je značajka posebno vrijedna za tvrtke i organizacije koje traže korisne informacije iz velike količine poruka. Nadalje, značajno povećava učinkovitost u obradi podataka. Ova automatizacija omogućuje tvrtkama da uštede vrijeme i resurse, omogućujući im da se usredotoče na druge kritične zadatke i strateško donošenje odluka. Konačno, privatnost i etika bitna su razmatranja kada se radi o podacima koje su izradili korisnici. WhatsAppNLP osigurava da se analiza poruka provodi anonimizirano i etički, štiteći privatnost pojedinaca, pružajući vrijedne uvide.

4. Zaključak

Stvaranje modela strojnog učenja zahtijeva zahtjevna znanja iz područja programiranja i podatkovne

generates intuitive reports in various formats, such as graphs, tables, and text messages. These visually appealing and easy-to-understand reports enable users to quickly grasp key insights from the data. This feature is especially valuable for businesses and organizations seeking actionable information from a large volume of messages. Furthermore, it significantly increases efficiency in data processing. This automation allows businesses to save time and resources, enabling them to focus on other critical tasks and strategic decision-making. Finally, privacy and ethics are essential considerations when dealing with user-generated data. WhatsAppNLP ensures that the analysis of messages is conducted in an anonymized and ethical manner, safeguarding the privacy of individuals while providing valuable insights.

4 Conclusion

Creating a machine learning model requires demanding knowledge in the field of programming and data science. The quality of the model depends on the quality of the data we use to learn the machine learning model and nowadays there are countless different sources of quality data. Although one of the most commonly used datasets was chosen, whose application is directly related to the categorization of text documents, and consists of more than 18,000 text documents with a million number of words, there are still shortcomings in the obscurity of categorization of topics on the example of a single conversation with text messages. This is best demonstrated by the complexity of creating a model that is able to recognize any topic that two or more users can talk about.

On the bright side, the WhatsAppNLP library proves that this is very possible and with the expansion of datasets with which the model will be taught to identify a larger number of topics, it gives an optimistic picture for the future of natural language processing development. Such a system can be used in real time, but the real value is in the analysis of communication files that subsequently need to be analyzed in large quantities for the needs of business research. The paper shows that using WhatsAppNLP methods of analyzing WhatsApp messages, it is possible to come to knowledge about patterns and behavior trends. WhatsAppNLP library has plenty of room for improvement and refinement like: multilingual support, real time analysis, integration with other chat applications, customizable analysis parameters and community support. By incorporating these improvements, the WhatsAppNLP library can evolve into a more powerful and versatile tool, increasing its appeal to businesses, researchers, and data analysts seeking to derive valuable insights from short digital messages.

znanosti. Kvaliteta modela ovisi o kvaliteti podataka koje koristimo za učenje modela strojnog učenja, a danas postoji bezbroj različitih izvora kvalitetnih podataka. Iako je odabran jedan od najčešće korištenih skupova podataka čija je primjena izravno povezana s kategorizacijom tekstualnih dokumenata, a sastoji se od više od 18.000 tekstualnih dokumenata s milijunskim brojem riječi, još uvijek postoje nedostaci u nejasnoći kategorizacije tema na primjer jednog razgovora s tekstualnim porukama. To najbolje pokazuje složenost izrade modela koji je u stanju prepoznati bilo koju temu o kojoj mogu razgovarati dva ili više korisnika.

Sa vedrije strane, biblioteka WhatsAppNLP dokazuje da je to vrlo moguće i uz proširenje skupova podataka s kojima će se model naučiti identificirati veći broj tema, daje optimističnu sliku za budućnost razvoja obrade prirodnog jezika. Takav sustav može se koristiti u realnom vremenu, ali prava vrijednost je u analizi komunikacijskih datoteka koje je naknadno potrebno analizirati u velikim količinama za potrebe poslovnih istraživanja. U radu se pokazuje da je pomoću WhatsAppNLP metoda analize WhatsApp poruka moguće doći do saznanja o obrascima i trendovima ponašanja. Knjižnica WhatsAppNLP ima puno prostora za poboljšanje i doradu, poput: višejezične podrške, analize u stvarnom vremenu, integracije s drugim aplikacijama za chat, prilagodljivih parametara analize i podrške zajednice. Uključivanjem ovih poboljšanja, WhatsAppNLP biblioteka može se razviti u moćniji i svestraniji alat, povećavajući svoju privlačnost tvrtkama, istraživačima i analitičarima podataka koji nastoje izvući vrijedne uvide iz kratkih digitalnih poruka.

U ovom radu predstavljena je učinkovita metoda analize sadržaja poruka kako bi se istaknuo potencijal primjene takvih i sličnih tehnika u poslovanju modernih, digitalno orijentiranih organizacija kako je opisano u rubrici prijedloga poslovne vrijednosti.

Kada je riječ o obradi prirodnog jezika, nadogradnja modela strojnog učenja znači bolje prepoznavanje tema kao i prepoznavanje šireg spektra tema.

Nadogradnja u tom smislu zahtijeva prikupljanje i kategorizaciju većeg broja tekstova u zajednički skup podataka koji će se koristiti kao izvor podataka strojnog učenja. Posljednjih godina bilježi se porast trenda implementacije sadržaja društvenih mreža, konkretno digitalnih platformi za komunikaciju i razmjenu poriva, na niz novih načina za stvaranje poslovne vrijednosti. Društveni mediji, naposljetku, sadrže mnoštvo informacija i dezinformacija o pojedinačnim korisnicima i njihovim mrežama, a više zakona ograničava što tijela za provođenje zakona mogu učiniti s podacima društvenih medija.

In this paper, an efficient method of analyzing the content of messages has been presented in order to highlight the potential of applying such and similar techniques in the business of modern, digitally oriented organizations as described in the business value proposition section.

When it comes to natural language processing, upgrading machine learning models means better recognition of topics as well as recognizing a wider range of topics.

Upgrading in this sense requires the collection and categorization of a larger number of texts into a common dataset to be used as a source of machine learning data. In recent years, there has been an increase in the trend of implementing social media content, in the specific case of digital platforms for communication and exchange of urges, in a number of new ways to create business value. Social media, after all, contains a wealth of information and misinformation about individual users and their networks, and more laws limit what law enforcement can do with social media data.

References

Mukhopadhyay, S. (2018). Advanced Data Analytics Using Python. Apress, ISBN-13 (pbk): 978-1-4842-3449-5 ISBN-13 (electronic): 978-1-4842-3450-1

Marasović, A. (2021) Latent semantic analysis, variants and applications. *Faculty of Science*, Retrieved from https://repozitorij.pmf.unizg.hr/islandora/object/pmf%3A5264/datastream/PDF

Dinesh K., Sunil A., Sorab K., & Mohanty N.M. (2021). Sentiment Study Approach Based on Chat Summarization, Retrieved from: https://www.researchgate.net/publication/334096166_Sentiment_Study_Approach_Based_on_Chat_Summarization

Sweigart A. (2015). Automate the Boring Stuff with Python, 2nd Edition: Practical Programming for Total Beginners, No Starch Press, ISBN-13: 978-1593279929,

Walkowska, J. (2010). *An NLP-Oriented Analysis of the Instant Messaging Discours.* In Sojka, P., Horák, A., Kopeček, I., & Pala, K. (Eds) (2010) *Text, Speech and Dialogue. TSD 2010. Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, vol 6231. https://doi.org/10.1007/978-3-642-15760-8_73

Vetulani, Z., Marciniak, J., Konieczka, P., & Walkowska, J. (2008). An SMS-Based System Architecture (Logical Model) To Support Management of Information Exchange in Emergency Situations POLINT-112-SMS project.

Reference

Mukhopadhyay, S. (2018). Advanced Data Analytics Using Python. *Apress*, ISBN-13 (pbk): 978-1-4842-3449-5 ISBN-13 (electronic): 978-1-4842-3450-1

Marasović, A. (2021) Latent semantic analysis, variants and applications. *Faculty of Science*, Retrieved from https://repozitorij.pmf.unizg.hr/islandora/object/pmf%3A5264/datastream/PDF

Dinesh K., Sunil A., Sorab K., & Mohanty N.M. (2021). Sentiment Study Approach Based on Chat Summarization, Retrieved from: https://www.researchgate.net/publication/334096166_Sentiment_Study_Approach_Based_on_Chat_Summarization

Sweigart A. (2015). Automate the Boring Stuff with Python, 2nd Edition: Practical Programming for Total Beginners, No Starch Press, ISBN-13: 978-1593279929,

Walkowska, J. (2010). *An NLP-Oriented Analysis of the Instant Messaging Discours.* In Sojka, P., Horák, A., Kopeček, I., & Pala, K. (Eds) (2010) *Text, Speech and Dialogue. TSD 2010. Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, vol 6231. https://doi.org/10.1007/978-3-642-15760-8_73

Vetulani, Z., Marciniak, J., Konieczka, P., & Walkowska, J. (2008). An SMS-Based System Architecture (Logical Model) To Support Management of Information Exchange in Emergency Situations POLINT-112-SMS project. In: *Intelligent Information Processing IV - 5th IFIP International Conference on Intelligent Information Processing* (pp. 240–253). Springer, Boston

Khurana, D., Koli, A., Khatter, K. et al. (2022). Natural language processing: state of the art, current trends and challenges. Multimed Tools Appl. https://doi.org/10.1007/s11042-022-13428-4

Wang Z., Ng P., Ma X., Nallapati R., Xiang B. (2019). Multi-passage bert: A globally normalized bert model for open-domain question answering, arXiv preprint arXiv:1908.08167

Vetulani, Z., Marciniak, J. (2000). Corpus Based Methodology in the Study and Design of Systems with Emulated Linguistic Competence, In: Christodoulakis, D.N. (Ed.) *NLP 2000*, LNCS (LNAI), vol. 1835. (pp. 346–357) Springer, Heidelberg

Sworna Z.T., Mousavi Z, Babar M.A. (2022). *NLP methods in host-based intrusion detection*

In: *5th IFIP International Conference* (pp. 240–253). Springer, Boston

Khurana, D., Koli, A., Khatter, K. et al. (2022). Natural language processing: state of the art, current trends and challenges. Multimed Tools Appl. https://doi.org/10.1007/s11042-022-13428-4

Wang Z., Ng P., Ma X., Nallapati R., Xiang B. (2019). Multi-passage bert: A globally normalized bert model for open-domain question answering, arXiv preprint arXiv:1908.08167

Vetulani, Z., Marciniak, J. (2000). Corpus Based Methodology in the Study and Design of Systems with Emulated Linguistic Competence, In: Christodoulakis, D.N. (Ed.) *NLP 2000*, LNCS (LNAI), vol. 1835. (pp. 346–357) Springer, Heidelberg

Sworna Z.T., Mousavi Z, Babar M.A. (2022). *NLP methods in host-based intrusion detection Systems: A systematic review and future directions*, arXiv preprint arXiv:2201.08066

Walkowska, J. (2009). Gathering and Analysis of a Corpus of Polish SMS Dialogues. In: *Challenging Problems of Science. Computer Science/Recent Advances in Intelligent Information Systems*, (pp. 145–157). Academic Publishing House EXIT, Warsaw

Hult, C.A., Richins, R. (2006). The Rhetoric and Discourse of Instant Messaging, In: *Computers and Composition Online*, Retrieved from: http://www.bgsu.edu/cconline/hultrichins_im/hultrichins_im.htm

Hurlow, C., Brown, A. (2003). Generation Txt? The Sociolinguistics of Young People's Text-Messaging, In: *Discourse Analysis Online*, *Department of Communication*, University of Washington, Box 353740, Seattle, WA 98195, USA, Retrieved from: http://extra.shu.ac.uk/daol/articles/v1/n1/a3/thurlow2002003-01.html

Grossz, B.J., Sidner, C.L. (1986). Attention, Intentions and the Structure of Discourse, *Computational Linguistics* 12(3), 175–204

Pribisalić, E. (2019). Obrada prirodnog jezika, *Sveučilište u Osijeku*, Retrieved from: http://www.mathos.unios.hr/~mdjumic/uploads/diplomski/PRI14.pdf

Wonderflow (2018), 2 NLP Examples: How Natural Language Processing is Used, https://www.wonderflow.ai/blog/natural-language-processing-examples

Jagota, A. (2020) Topic Modeling In NLP with a focus on Latent Dirichlet Allocation, https://towardsdatascience.com/topic-modeling-in-nlp-524b4cffeb68

Chen, E, (2011) Introduction to Latent Dirichlet Allocation, http://blog.echen.me/2011/08/22/introduction-to-latent-dirichlet-allocation/

Kelemen, Zádor Dániel. (2015). Fundamental Analysis of a Developer Support Chat Log for Identifying Process Improvement Opportunities. 10.13140/RG.2.1.1465.0407

Rosen, Devan & Miagkikh, Victor & Suthers, Dan. (2011). Social and semantic network analysis of chat logs. ACM IC Proceeding Series. 134-139. 10.1145/2090116.2090137.

Kumar T. Sandeep. (2020). 8 great Python libraries for natural language processing, Pulse, Retreived from: https://www.linkedin.com/pulse/8-great-python-libraries-natural-language-processing-t-sandeep-kumar/

https://towardsdatascience.com/topic-modeling-in-nlp-524b4cffeb68

Chen, E, (2011) Introduction to Latent Dirichlet Allocation, http://blog.echen.me/2011/08/22/introduction-to-latent-dirichlet-allocation/

Kelemen, Zádor Dániel. (2015). Fundamental Analysis of a Developer Support Chat Log for Identifying Process Improvement Opportunities. 10.13140/RG.2.1.1465.0407

Rosen, Devan & Miagkikh, Victor & Suthers, Dan. (2011). Social and semantic network analysis of chat logs. ACM IC Proceeding Series. 134-139. 10.1145/2090116.2090137.

Kumar T. Sandeep. (2020). 8 great Python libraries for natural language processing, Pulse, Retreived from: https://www.linkedin.com/pulse/8-great-python-libraries-natural-language-processing-t-sandeep-kumar/