Hibridni pristop za priporočanje vrstilcev univerzalne decimalne klasifikacije

Mladen Borovič, Sandi Majninger, Jani Dugonik, Marko Ferme, Milan Ojsteršek
Univerza v Mariboru, Fakulteta za elektrotehniko, računalništvo in informatiko Koroška cesta 46, 2000 Maribor
mladen.borovic@um.si, sandi.majninger@um.si, jani.dugonik@um.si, marko.ferme@um.si, milan.ojstersek@um.si

Izvleček

V prispevku predstavljamo hibridni pristop za priporočanje vrstilcev univerzalne decimalne klasifikacije. S pomočjo takšnega pristopa lahko knjižničarjem omogočimo polavtomatsko določanje vrstilcev univerzalne de- cimalne klasifikacije iz vsebine že obstoječih uvrščenih gradiv. Hibridni pristop deluje na podlagi združevanja rezultata metode BM25 in naivnega Bayesovega klasifikatorja, kjer oba pristopa vrneta seznam priporočenih vrstilcev. Oba seznama združimo v končni seznam priporočil z združevalno funkcijo. V prispevku podrobneje opišemo korpus, obliko podatkov, obliko vrstilcev univerzalne decimalne klasifikacije in delovanje posamezne metode znotraj hibridnega pristopa. Podamo tudi rezultate metrik natančnosti, priklica in Fß za sezname priporočil na korpusu besedil iz nacionalne infrastrukture odprtega dostopa.

Ključne besede: digitalne knjižnice, hibridni priporočilni sistemi, programska oprema v knjižnicah, Univer- zalna decimalna klasifikacija

Abstract

In this article we present a hybrid approach to recommending the Universal Decimal Classification notation for unclassified documents. By recommending Universal Decimal Classification notation to librarians, we can enable them to semi-automatically determine the notation using already classified documents. The hybrid approach combines the BM25 method and the naive Bayes classifier, where both methods return a list of recommended notations. Both lists are merged into a final recommendation list using a custom merge function. In detail we present the Universal Decimal Classification notation structure, the corpus of documents, the inputs to our methods and the inner workings of our hybrid approach consisting of both methods. We provide the measurement results of the recommendation lists for the corpus from the National Open-Access Infrastructure in the form of precision, recall and Fß metrics.

Keywords: digital libraries, hybrid recommender systems, library software, Universal Decimal Classification

1 UVOD

Z razvojem spletnih iskalnikov sta se področji računalništva in knjižničarstva združili v interdisciplinarno področje digitalnih knjižnic, ki se ukvarja predvsem z organizacijo, skladiščenjem, obdelavo in klasifikacijo dokumentov. Predvsem klasifikacija dokumentov je raziskovalno zelo aktivno področje. Kljub temu, da je na tem področju veliko različnih metod, ne obstaja veliko metod za avtomatizirano klasificiranje po knjižničarskih klasifikatorjih, kot so univerzalna decimalna klasifikacija (UDK) [Slavic, 2004], Deweyjeva decimalna klasifikacija (DDK) [Wang, 2009] in klasifikacija Library of Congress (LCC) [Godby & Stuler, 2003], [Frank & Paynter, 2004]. Obstajajo še drugi klasifikacijski sistemi, ki so ekskluzivno namenjeni določenim jezikom (npr. v Aziji obstajajo Kitajska, Japonska in Korejska knjižničarska klasifikacija). Ne glede na sistem klasifikacije se večina gradiv po svetu še vedno klasificira ročno – bodisi zaradi nezaupanja v avtomatizirano klasifikacijo, bodisi zaradi nezadovoljivega rezultata le-te.

Problem nezaupanja v avtomatizirano klasifikacijo je potrebno s stališča knjižničarjev razumeti, saj bodo ob napačni klasifikaciji imeli dodatno delo s popravljanjem zapisov v digitalnih knjižnicah, obenem pa takšni zapisi ne bodo zlahka dostopna, saj jih uporabniki ne bodo mogli najti s trenutnimi iskalnimi postopki. V prispevku se zavedamo tega problema in v želji po zmanjšanju nezaupanja, poskušamo knjižničarjem približati avtomatizirano klasifikacijo z uvedbo priporočanja ustreznih vrstilcev klasifikacije. Ker knjižničar dobi le priporočilo, katere vrstilce naj uporabi, se lahko še vedno odloči drugače - gre torej za polavtomatsko klasifikacijo.

V prispevku opisujemo hibridni pristop priporočanja vrstilcev univerzalne decimalne klasifikacije, ki uporablja uveljavljeno iskalno metodo BM25 in naivni Bayesov klasifikator. V drugem poglavju opišemo vrste priporočilnih sistemov in uporabo le-teh v digitalnih knjižnicah. Tretje poglavje opisuje univerzalno decimalno klasifikacijo. V četrtem poglavju opišemo obliko, pripravo in obdelavo podatkov korpusa besedil iz nacionalne infrastrukture odprtega dostopa. V petem poglavju opisujemo hibridni pristop k priporočanju z uporabo metode BM25 in naivnega Bayesovega klasifikatorja. Šesto poglavje vsebuje rezultate primerjave meritev metrik natančnosti, priklica in Fβ med metodo BM25, naivnim Bayesovim klasifikatorjem in predstavljeno hibridno metodo. V sedmem poglavju podamo zaključke in nekaj idej za izboljšavo hibridne metode.

2 PRIPOROČILNI SISTEMI V DIGITALNIH KNJIŽNICAH

V zadnjih letih smo lahko opazili razmah priporočilnih sistemov na veliko področij. Dandanes se najbolj uporabljajo v spletnih iskalnikih, družbenih omrežjih in raznih multimedijskih storitvah kot so YouTube, Netflix, Spotify in Last.fm. Priporočilni sistemi za svoje delovanje v glavnem uporabljajo dva tipa filtriranja podatkov. To sta vsebinsko filtriranje (angl. content-based filtering) in sodelovalno filtriranje (angl. collaborative filtering) [Melville & Sindhwani, 2017].

Vsebinsko filtriranje podatkov uporablja opis objekta priporočanja v nestrukturirani obliki, kot je recimo besedilo, ali pa v strukturirani obliki, kjer ima objekt vnaprej znane lastnosti, po katerih definiramo filtre. Ključnega pomena je torej opis objekta priporočanja, saj ta metoda z metrikami podobnosti išče podobne objekte priporočanja. Kadar imamo opravka s podatki v strukturirani obliki, so metrike podobnosti navadno kosinusna razdalja, Jaccardov indeks in Pearsonova korelacija [Lops et al., 2011]. Nestrukturirani podatki so ponavadi podani z besedilom zato so metrike podobnosti v tem primeru omejene na metrike podobnosti, ki jih uporabljamo v procesiranju naravnega jezika. Natančneje je v tem primeru zelo pogosta uporaba utežne sheme tf -idf v kombinaciji z razvrščevalno metodo BM25.

Sodelovalno filtriranje se v nasprotju z vsebinskim filtriranjem ne osredotoča na sam opis objekta priporočanja, temveč na uporabniško interakcijo z objekti priporočanja. Za ta tip filtriranja je pomembno, ali si je uporabnik objekt priporočanja ogledal, koliko časa ga je gledal in ali je opravil kakšno pomembnejšo interakcijo s tem objektom. V primeru spletnih trgovin je to nakup izdelka, v primeru digitalnih knjižnic pa prenos dokumenta na računalnik.

Tako vsebinsko kot sodelovalno filtriranje imata svoje slabosti. Glavna slabost sodelovalnega filtriranja je problem hladnega začetka. To je situacija, v kateri se znajdemo čisto na začetku, kadar še nimamo aktivnih uporabnikov in posledično nimamo podatkov o uporabniški interakciji z objekti priporočanja. Slabost vsebinskega priporočanja je prekomerna specializacija, kjer priporočilni sistem uporabniku priporoča zgolj eno vrsto objektov priporočanja, kar pa ni vedno zaželjeno. V tem primeru se poslužimo hibridnih priporočilnih sistemov, ki združujejo dve ali več metod filtriranja v eno samo z namenom izogibanja slabostim posamezne metode. Največkrat hibridni priporočilni sistemi združujejo sodelovalno in vsebinsko filtriranje, odvisno od ciljne uporabe priporočilnega sistema pa lahko združujemo tudi več tehnik sodelovalnega filtriranja oziroma več tehnik vsebinskega filtriranja. V splošnem poznamo več načinov hibridizacije [Burke, 2002]. Z utežno hibridizacijo sestavimo oceno podobnosti iz ocen vseh vključenih metod. Pri preklopni hibridizaciji sistem preklaplja med vključenimi metodami po potrebi ali glede na situacijo. Mešana hibridizacija rezultate vključenih metod prikaže skupaj v enem seznamu priporočil. Hibridizacija s kombinacijo značilk deluje tako, da so značilke iz več virov združene in se uporabijo kot vhod v eno tehniko priporočanja. Podobno deluje hibridizacija z obogatenjem značilk, kjer se ena metoda uporabi za pridobivanje značilk, ki so vhod drugi metodi. Kaskadna hibridizacija v delovanje vnaša zaporedje uporabe različnih metod. Naza-

dnje, hibridizacija na meta ravni deluje tako, da ena metoda zgradi model, ki je vhod naslednji metodi.

V digitalnih knjižnicah se priporočilni sistemi uporabljajo predvsem v namene priporočanja dokumentov in dru- gih gradiv, ki jih digitalne knjižnice ponujajo [Bai et al., 2019]. Priporočilni sistemi opisani v [Beel et al., 2017] in [Porcel et al., 2009] so bili zasnovani specifično za uporabo v digitalnih knjižnicah z namenom, da razi- skovalcem pomagajo najti zanimive publikacije. Podobno lahko takšne priporočilne sisteme zasledimo v akademskih družbenih omrežjih, kot je na primer Mendeley [Vargas et al., 2016]. V Sloveniji obstaja hibridni priporočilni sistem, ki deluje na nacionalni infrastrukturi odprtega dostopa in navzkrižno priporoča gradiva med digitalnimi knjižnicami in repozitoriji slovenskih univerz [Ojsteršek et al., 2014]. V tem primeru gre za kaskadno hibridizacijo z metodo vsebinskega filtriranja, ki ji sledi sodelovalno filtriranje.

3 UNIVERZALNA DECIMALNA KLASIFIKACIJA

Univerzalna decimalna klasifikacija (v nadaljevanju UDK) je knjižnični klasifikacijski sistem, ki služi kot orodje za vsebinsko označevanje dokumentov in iskanje po njih. Plačljiva licenca za UDK obsega več kot 70.000 vrstilcev.Obstaja tudi zastonjska različica, ki pa je močno okrnjena na okoli 2500 vrstilcev. Z uporabo tega klasifikacijskega sistema se lahko vsakemu dokumentu določi vrstilec, ki dokument uvršča v področje. UDK sestavljajo glavne tabele in pomožne tabele, kjer glavne tabele določajo področja človeškega znanja, pomožne pa dodatne informacije o področju (npr. čas, kraj, jezik in obliko). Izraz UDK je lahko preprost ali sestavljen. V slednjem primeru se uporabijo znaki za povezovanje, ki opisujejo tip povezave med vrstilci. Tako lahko z izrazom UDK opisujemo tudi interdisciplinarne dokumente. V tabelah 1-3 so podani zgledi vrstilcev in izrazov UDK.

Tabela 1: **Vrstilci vrhnjih področij univerzalne decimalne klasifikacije.**

Vrstilec	Področja
0	Znanost in znanje. Organizacije. Informacije. Dokumentacija. Bibliotekarstvo. Institucije. Publikacije.
1	Filozofija. Psihologija.
2	Teologija. Verstva.
3	Družbene vede. Politika. Ekonomija. Pravo. Izobraževanje.
5	Matematika. Naravoslovje.
6	Uporabne znanosti. Medicina. Tehnika.
7	Umetnost. Arhitektura. Fotografija. Glasba. Šport.
8	Jezik. Književnost
9	Geografija. Biografija. Zgodovina.

Tabela 2: **Hierarhična struktura vrstilcev univerzalne decimalne klasifikacije za področje Računalništvo (004), veja Računalniške komunikacije, Računalniška omrežja (004.7).**

Vrstilec	Opis področja
004	Računalniška znanost in tehnologija. Računalništvo. Obdelava podatkov
004.7	Računalniške komunikacije. Računalniška omrežja
004.73	Omrežja glede na prostranost
004.738	Medsebojno povezovanje omrežij. Medomrežanje

Tabela 3: **Primer preprostega in sestavljenega izraza UDK. Preprost izraz vsebuje splošni privesni vrstilec za obliko (043.2). Sestavljen izraz vsebuje enostaven odnos (znak „:“), zaporedno razširitev (znak „/“), splošni privesni vrstilec za obliko (043.2) in splošni privesni vrstilec za kraj (497.4).**

Vrstilec	Izkax UDK
Preprost	519.85(043.2)
Sestavljen	336.778(043.2):336.713/.717(497.4)

Za pridobitev izraza UDK je potrebna katalogizacija oziroma zahteva knjižničarjem v primerih, ko gre za zaključna dela. Knjižničarji z uporabo geslovnika ugotovijo, katere vrstilce naj dodajo v izraz UDK tako, da v geslovnik [Zalokar, Matjaž, 2002b], [Zalokar, Matjaž, 2002a] vnesejo ključne besede oziroma predmetne oznake. V primeru zaključnih del mora avtor knjižničarjem posredovati naslov, mentorja, ključne besede, povzetek in kazalo. Knjižničarji nato iz naslova in ključnih besed pridobijo vhod za geslovnik, na podlagi povzetka, kazala in mentorja pa se dokončno odločijo za primerne vrstilce UDK. Celoten proces pridobitve izraza UDK ponavadi traja do 2 dni. Kvaliteta izraza UDK je odvisna od geslovnika in števila vrstilcev UDK, ki jih imajo knjižničarji na voljo.

4 KORPUS BESEDIL IN OBDELAVA PODATKOV

V prispevku uporabljamo korpus besedil pridobljen iz nacionalne infrastrukture odprtega dostopa [Ojsteršek et al., 2014], ki se je izvedla v letu 2013 in obsega zaključna dela in znanstvene publika- cije iz vseh slovenskih univerz. Gre za obširen korpus besedil v slovenščini, ki obsega okoli 200.000 dokumentov in je segmentiran na ključne besede, naslove, povzetke, polno besedilo in vsebuje dodatne informacije o besedilih - med njimi tudi izraze UDK. Ker vsa besedila v korpusu nacionalne infrastrukture nimajo vseh informacij na voljo, smo uporabili filtrirano podmnožico 10.000 besedil, v kateri so vsa besedila, ki imajo podatek o naslovu, ključnih besedah, polnem besedilu in izrazu UDK. V nadaljnji obdelavi podatkov smo delali s polnimi besedili, kjer smo dodatno utežili besede v naslovih in ključnih besedah.

4.1 Predobdelava besedil

Iz vseh besedil smo najprej tvorili besedne uni-, bi- in tri-grame ter izvedli vse možne permutacije med njimi. Nad besednimi n-grami smo uporabili tudi postopek lematizacije tako, da smo hkrati hranili lematizirane in nelematizirane besedne n-grame. Nato smo za to množico izračunali uteži tf in idf . Utež tf predstavlja frekvenco določenega besednega n-grama v dokumentu, utež idf pa pomembnost besednega n-grama glede na celotno zbirko dokumentov. Tako smo dobili sezname vseh možnih besednih n-gramov in njihove pojavitve v dokumentih, kot tudi število dokumentov v katerih se pojavljajo. Z enoličnim identifikatorjem dokumenta smo lahko dostopali tudi do njegovega izraza UDK in s tem povezali izraze UDK s pripadajočimi besednimi n-grami.

4.2 Razpoznavalnik izrazov UDK

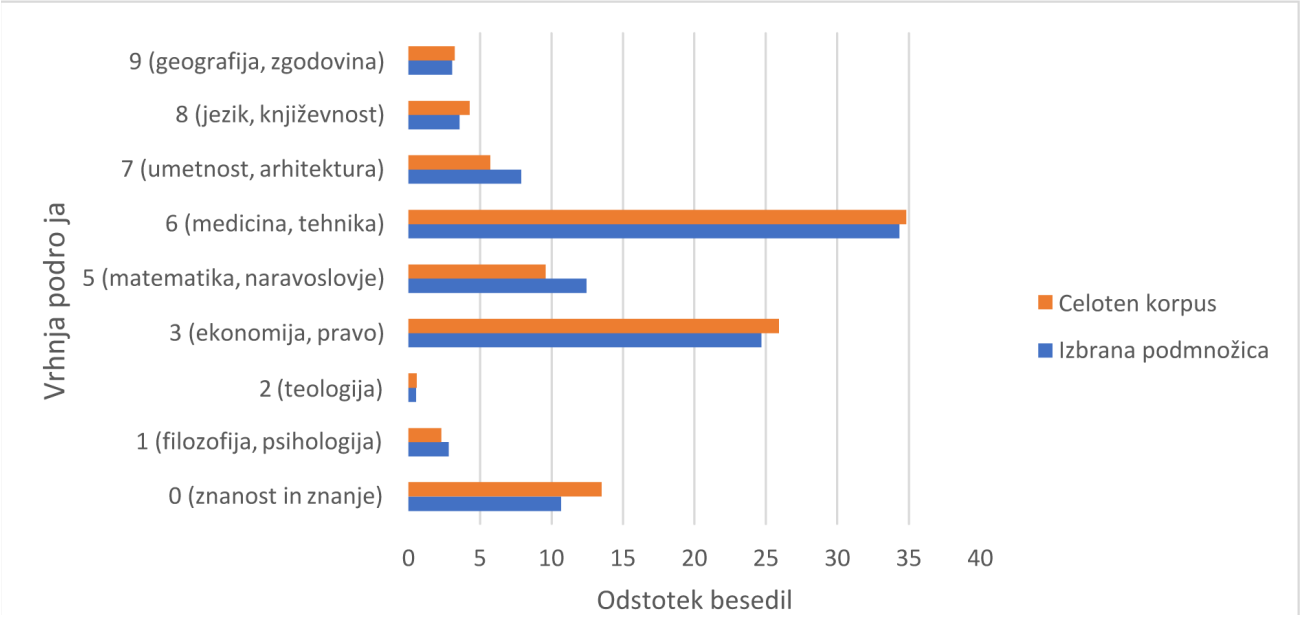
Ker je v korpusu besedil veliko takšnih, ki imajo sestavljen izraz UDK, smo zasnovali preprost razpoznavalnik izrazov UDK, ki zna iz sestavljenega izraza UDK vrniti vse vrstilce UDK. Pri tem smo upoštevali priredno in zaporedno razširitev, enostavne odnose, in podrobno delitev. Ostalih znakov za povezovanje nismo obravnavali, saj je bilo število dokumentov s temi znaki za povezovanje zanemarljivo. Prav tako nismo upoštevali splošnih privesnih vrstilcev.

Za povezavo z UDK smo uporabili brezplačno slovensko različico UDK v obliki povezanih odprtih podatkov (angl. linked open data) [UDC Consortium (UDCC), 2012]. Le-ta obsega 1445 vrstilcev UDK s slovenskim prevodom. Ta zbirka je v obliki parov (vrstilec, prevod). Zaradi omejenega števila brezplačnih vrstilcev je razpoznavanje v nekaterih primerih omejeno po globini univerzalne decimalne klasifikacije, kot je razvidno v tabeli 4.

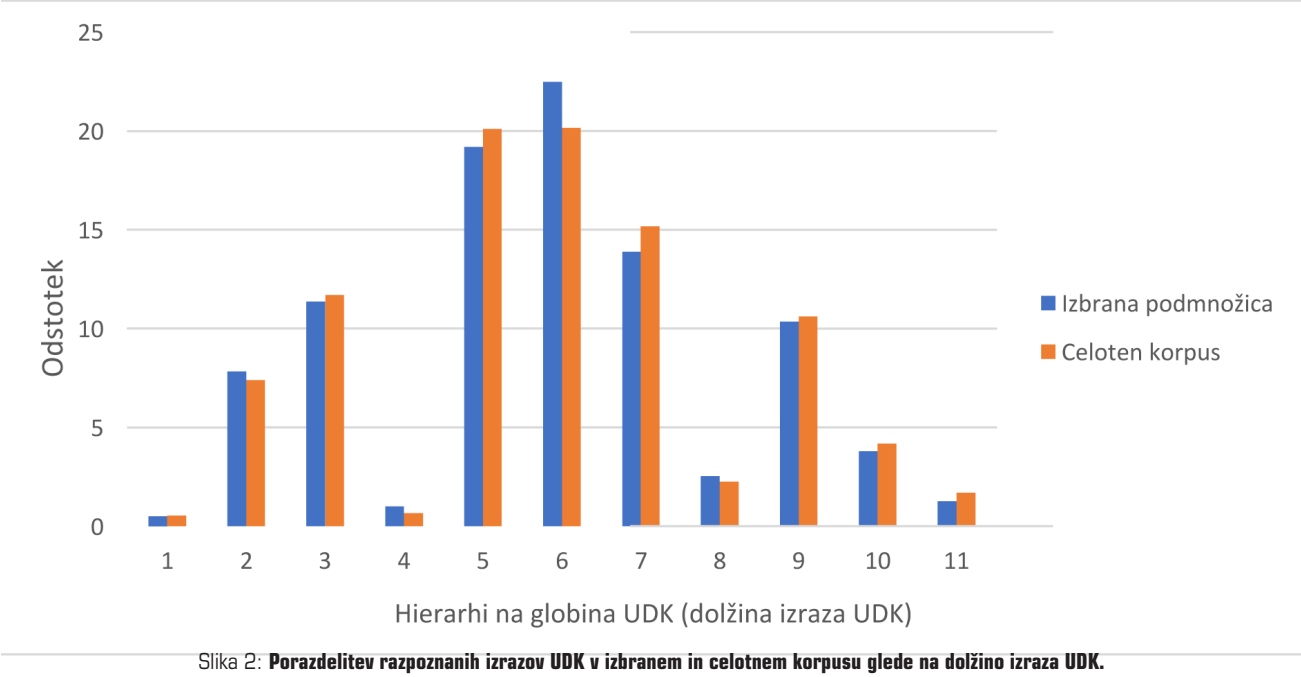
Tabela 4: **Primer delovanja razpoznavalnika izrazov UDK. Vrstilec 621.952.8 je bil razpoznan kot 621.9.**

Vhod	Izhod
(004.94:621.952.8)+658.8(043.2)	004.94 621.9 658. 003.63 8
711.4:711.1:158.937:003.63(497.4Slovenska Bistrica)(043.2)	711.4 711.1 158.937

Po obdelavi z razpoznavalnikom izrazov UDK smo preverili, kakšna je porazdelitev razpoznanih izrazov UDK v izbranem korpusu besedil. Preverili smo dolžino razpoznanih izrazov, saj dolžina izraza predstavlja globino v hierarhiji UDK in neposredno vpliva na specifičnost kategorizacije. Manjša dolžina izraza UDK pomeni splošnejšo kategorizacijo, večja dolžina izraza UDK pa specifično kategorizacijo (tabeli 1 in 2). Dolžino razpoznanega izraza UDK smo v meritvah uporabljali kot parameter. Tako smo lahko preverili, kako se uporabljene metode obnesejo na različnih nivojih specifičnosti hierarhičnih področij UDK. Slika 1 prikazuje odstotke razpoznanih izra-



Slika 1: **Porazdelitev razpoznanih izrazov UDK v izbranem in celotnem korpusu glede na vrhnja področja.**



Slika 2: **Porazdelitev razpoznanih izrazov UDK v izbranem in celotnem korpusu glede na dolžino izraza UDK.**

zov UDK v izbranem in celotnem korpusu glede na njihovo vrhnje področje. Slika 2 prikazuje odstotke razpoznanih izrazov UDK v izbranem in celotnem korpusu glede na njihovo dolžino.

5 HIBRIDNI PRISTOP K PRIPOROČANJU

V našem hibridnem pristopu uporabljamo dve metodi, ki ju uvrščamo med metode vsebinskega filtriranja. Uporabljamo metodo BM25 in naivni Bayesov klasifikator. Vhod v hibridno metodo je iskalni niz (tj. naslov, ključne besede, predmetne oznake), izhod pa je seznam najbolj ustreznih vrstilcev UDK, ki ga prikažemo knjižničarju. Ideja hibridnega pristopa je, da z obema metodama poiščemo k najbolj ustreznih vrstilcev UDK, nato pa rezultate združimo v končni seznam ustreznih vrstilcev UDK. BM25 in njene različice so že vrsto let najbolj uporabljene metode v implementaci-

jah iskalnikov (angl. full-text search) in se pojavljajo v različnih komercialnih rešitvah kot so Microsoft SQL Server, MySQL, Elasticsearch, Xapian, Solr in Lucene. Naivni Bayesov klasifikator je uveljavljena metoda na področju kategorizacije in klasifikacije besedil. V našem hibridnem pristopu ta metoda služi za uvrščanje določenih vrstilcev UDK v končni seznam priporočil, ki bi jih metoda BM25 morda izpustila.

5.1 BM25

BM25 (Best Match 25) [Robertson & Zaragoza, 2009] je metoda razvrščanja, ki omogoča razvrščanje dokumentov po podobnosti na podlagi besednih n-gramov, ki se pojavljajo v dokumentih. Začetki razvoja segajo med 1970 in 1980, ko sta avtorja začela razvijati ogrodje za pridobivanje informacij na podlagi verjetnosti. BM25 ni samo ena metoda temveč družina več metod, ki se razlikujejo po utežnih shemah in vrednostih parametrov pomembnosti za uteži. Največkrat se uporabljata uteži tf in idf . Danes obstaja veliko različic BM25, ki doprinesejo manjše izboljšave v specifičnih primerih [Trotman et al., 2014], [Lv & Zhai, 2011a], [Lv & Zhai, 2011b]. Različica BM25, ki jo uporabljamo se izračuna kot:

$$s(d, Q) = \sum_{i=1}^{||Q||} idf(q_i) \cdot \frac{tf(q_i, d) \cdot (k_1 + 1)}{tf(q_i, d) + k_1 \cdot B} \ , q_i \in Q, d \in D \qquad (1)$$

Za enačbo 1 velja:

- tf (qi, d) je utež tf v dokumentu d za besedni n--gram qi iskalnega niza Q. Vrednost je število pojavitev besednega n-grama qi v dokumentu d.
- k1 je parameter s privzeto vrednostjo k1 = 1.2. [Manning, Christopher D. and Raghavan, Prabhakar and Schütze, H
- idf (qi) je utež idf za besedni n-gram qi. Vrednost je število pojavitev besednega n-grama qi v celotnem korpusu D. Izračun uteži idf (qi) je podan z enačbo 2

$$idf(q_i) = \log \frac{||D|| - n(q_1) + 0{,}5}{n(q_1) + 0{,}5} \qquad (2)$$

  kjer je ||D|| število vseh dokumentov v korpusu D, n(qi) pa število dokumentov, ki vsebujejo besedni n-gram qi.
- B je normalizacijski faktor dan z enačbo 3

$$B = 1 - b + b \cdot \frac{l_d}{avgdl} \qquad (3)$$

  kjer ld predstavlja dolžino dokumenta d, avgdl pa povprečno dolžino dokumenta glede na celoten kor- pus D. Dolžina dokumenta je izražena s številom besed v dokumentu. Parameter b ima privzeto vre- dnost b = 0.75 [Manning, Christopher D. and Raghavan, Prabhakar and Schütze, Hinrich, 2008].

Ključno vlogo imata parametra k1 in b, ki uravnavata težo uteži tf in težo dolžine dokumentov v končnem izračunu. Dolžina dokumentov se meri s številom besednih n-gramov. Parametra upoštevata dve predpostavki o značilnostih, ki se pojavljajo pri pisanju dokumentov [He & Ounis, 2003]. Predpostavka o širini vsebine dokumenta (angl. verbosity hypothesis) govori o tem, da je lahko dokument daljši zaradi uporabe nepomembnih ali redundantnih besed, medtem ko predpostavka o obsegu dokumenta (angl. scope hypothesis) govori o daljših dokumentih zaradi uporabe več besed s kontekstom, ki tvorijo vsebino dokumenta. V praksi gre za kombinacijo teh dveh predpostavk, zato potrebujemo ustrezno normalizacijo. Dolžino vsakega dokumenta lahko normaliziramo s povprečno dolžino dokumentov. Nadalje lahko to normalizacijo reguliramo s parametrom b, kot kaže enačba 3, v enačbi 1 pa vidimo, da uporabimo funkcijo normalizacije B za normalizacijo uteži tf v navezi s parametrom k1.

Parameter k1 uravnava pomembnost uteži tf , parameter b pa pomembnost dolžine dokumentov. V interesu nam je, da sestavimo takšno funkcijo, ki bo delovala najbolje na različnih dokumentih v zbirki. To pomeni, da je treba ugotoviti katere vrednosti parametrov k1 in b so najboljše za dano zbirko [He & Ounis, 2005]. Vrednosti teh dveh parametrov niso strogo definirane, navadno pa se uporabijo vrednosti k1 [1.2, 2.0] in b = [0, 1] [Manning, Christopher D. and Raghavan, Prabhakar and Schütze, Hinrich, 2008].

Nad izbranim korpusom dokumentov smo izračunali uteži tf in idf ter za vsak par dokumentov

izračunali vrednosti BM25 z upoštevanjem privzetih vrednosti za parametra k1 in b. Z metodo BM25 nato poiščemo vhodnemu besedilu najbolj podobne dokumente, vzamemo njihove izraze UDK in z razpoznavalnikom pridobimo vrstilce UDK. Vrstilce nato uredimo v seznam po frekvenci pojavljanja in vrnemo prvih k elementov tega seznama (enačbi 4 in 5).

$$R = \arg\max_{k} \{s(d_j, Q)\}, j \in [1...||D||] \quad (4)$$

$$R_x = R_{BM25} = \{udk[r]\}, \forall\ r \in R \quad (5)$$

5.2 Naivni Bayesov klasifikator

Naivni Bayesov klasifikator smo naučili nad polnim besedilom s podatkom o enoličnem identifikatorju dokumenta in pripadajočih vrstilcih UDK. Izbran korpus, opisan v poglavju 4, smo naključno razdelili na učno množico, ki je obsegala 7.000 gradiv in testno množico, ki je obsegala 3.000 gradiv. Učna in testna množica sta imeli obliko trojic (identifikator, vrstilec, besedni n-gram). Vrstilci UDK predstavljajo razrede za klasifikacijo, saj želimo klasificirati nove primerke v vrstilce UDK. Pri izračunu verjetnosti uporabljamo metodo MLE (angl. maximum likelihood estimation) in Laplaceovo (znano tudi kot Add-one) glajenje (enačbi 6 in 7). Nc predstavlja število dokumentov, ki spadajo v razred c, N je število vseh dokumentov, Tct predstavlja število pojavljanj besednega n-grama t v dokumentih iz razreda c, V predstavlja množico vseh besednih n-gramov, m pa število vseh besednih n-gramov, ki se pojavijo v vhodnem nizu. Na koncu s pomočjo naučenega modela pridobimo seznam k najbolj verjetnih vrstilcev za dan vhod (enačba 8).

$$\hat{P}(c) = \frac{N_c}{N} \quad (6)$$

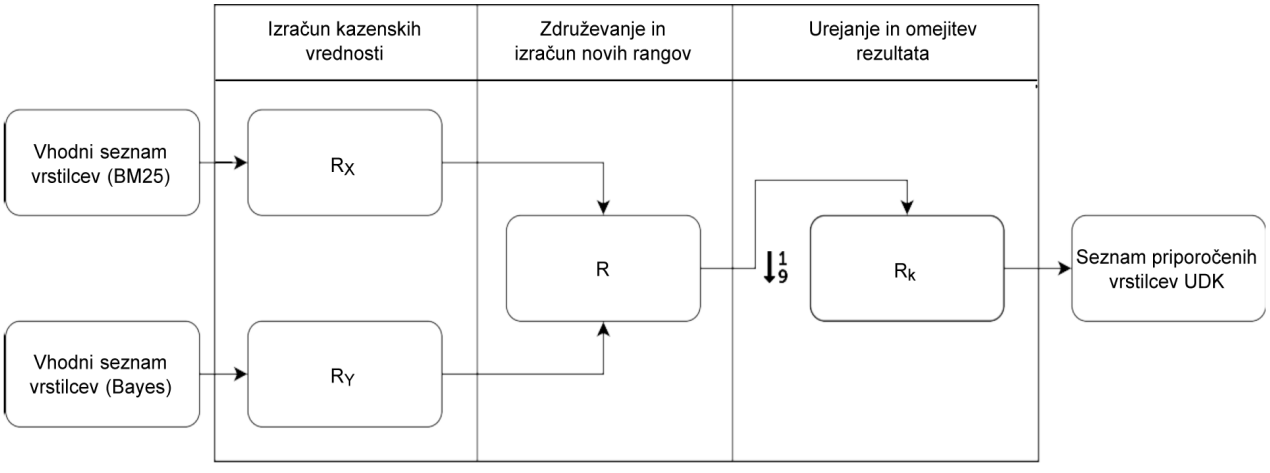
$$\hat{P}(t|c) = \frac{T_{ct} + 1}{\Sigma_{t' \in V} T_{ct'} + 1} \quad (7)$$

$$R_y = R_{Bayes} = \arg\max_{k} \{\log \hat{P}(c) + \sum_{i=1}^{m} \log \hat{P}(t_i|c)\} \quad (8)$$

5.3 Priporočanje z mešano hibridizacijo

V našem pristopu hibridnega priporočanja smo se odločili za tip mešane hibridizacije, ki združi rezultate dveh tehnik vsebinskega filtriranja (slika 3). Pristop mešane hibridizacije smo uporabili zato, ker želimo v končnem seznamu pridobiti čim več relevantnih vrstilcev UDK. ČC˘ e v skladu s pristopom mešane hibridizacije združujemo rezultate večih tehnik vsebinskega filtriranja, lahko v končnem seznamu pričakujemo vrstilce UDK, ki bi jih izpustili z uporabo zgolj ene metode vsebinskega filtriranja.

Gre torej za povečanje nabora priporočenih vrstilcev UDK v končnem seznamu priporočenih vrstilcev UDK. Seznama vrstilcev UDK, pridobljena z metodama BM25 in naivnim Bayesovim klasifikatorjem, združimo v končni seznam z združevalno funkcijo M , ki jo definiramo s psevdokodom 1.



Slika 3: **Shematika procesa priporočanja z mešano hibridizacijo.**

Algoritem 1: Združevalna funkcija M

Vhod : R_X - seznam vrstilcev UDK; rezultat prve metode vsebinskega filtriranja
Vhod : w_X - utež za enolični element v R_X; $w_X \in [0, 1]$ R
Vhod : R_Y - seznam vrstilcev UDK; rezultat druge metode vsebinskega filtriranja
Vhod : w_Y - utež za enolični element v R_Y; $w_Y \in [0, 1] \subset$ R
Vhod : k - omejitev števila elementov v izhodnem seznamu vrstilcev UDK; $k \in Z^+$
Izhod : R - seznam vrstilcev UDK, ki predstavlja združena seznama R_X in R_Y , omejen na k elementov

$R = \emptyset$
$R_k = \emptyset$
/* Element seznama je vektor r, ki vsebuje rang, vrstilec UDK in kazensko vrednost, ki je na začetku vedno enaka 0 */
$\mathbf{r} = (rank, class, penalty)$
/* Pridobimo velikosti seznamov R_X in R_Y */
$\lambda_X \leftarrow |R_X|$
$\lambda_Y \leftarrow |R_Y|$
/* Za vsak element $\mathbf{r}$ iz R_X , ki ni v R_Y izračunamo kazensko vrednost */
foreach $\mathbf{r} \in R_X \setminus R_Y$ **do**
    $\mathbf{r}.penalty \leftarrow (\lambda_X + \lambda_Y)\ w_X$
end
/* Za vsak element $\mathbf{r}$ iz R_Y , ki ni v R_X izračunamo kazensko vrednost */
foreach $\mathbf{r} \in R_Y \setminus R_X$ **do**
    $\mathbf{r}.penalty \leftarrow (\lambda_X + \lambda_Y)\ w_Y$
end
/* Združimo seznama R_X in R_Y v seznam R */
$R \leftarrow R_X \cup R_Y$
/* Za vsak element $\mathbf{r}$ iz R izračunamo nov rang na podlagi kazenskih vrednosti */
foreach $\mathbf{r} \in R$ **do**
    $\mathbf{r}.rank \leftarrow \frac{\mathbf{r}.rank + \mathbf{r}.penalty}{2}$
end
/* Seznam R uredimo naraščajoče po novih rangih */
$sort(R, \mathbf{r}.rank)$
/* Pridobimo seznam R_k, ki vsebuje prvih k elementov iz seznama R */
$R_k = \arg\max_k R$
return R_k

Ko sta na voljo seznama RX in RY , ki sta rezultat obeh metod vsebinskega filtriranja, ju je potrebno združiti z združevalno funkcijo M . Združevalna funkcija, ki jo uporabljamo, deluje na principu povprečnega ranga. V obeh seznamih iščemo enake vrstilce UDK in povprečimo njihove pozicije. Če se vrstilec pojavi v enem seznamu, v drugem pa ne, je njegov rang enak vsoti dolžin seznamov RX in RY. Takšna združevalna funkcija daje prednost tistim vrstilcem, ki so bili pridobljeni z obema metodama. Dodatno omogočimo tudi uteževanje kazenskih vrednosti na rang v primeru, da ena metoda vrne element, ki ga druga ne. Uteži kazenskih vrednosti wX in wY imata vrednosti med 0 in 1, kjer 0 ponazarja uteževanje brez vrednosti kazni, 1 pa uteževanje s polno vrednostjo kazni. Končno uteževanje lahko popolnoma spremenimo s spreminjanjem združevalne funkcije M .

6 EVALVACIJA IN REZULTATI

Merjenja uspešnosti priporočilnih sistemov se lahko lotimo na veliko načinov, saj ima vsak priporočilni sistem različen namen. Obstaja kar nekaj metod za evalvacijo priporočilnih sistemov [Pu et al., 2011], [Shani & Gunawardana, 2011], [Monti et al., 2019], [Bogaert et al., 2019], [Krauss et al., 2019]. Pred evalvacijo se moramo vprašati po rezultatu, ki ga želimo s priporočilnim sistemom doseči [Rendle et al., 2019].

V našem primeru gre za vsebinsko priporočanje, saj uporabljamo korpus besedil s katerim poskušamo najti vhodu podobne vrstilce UDK. Intuitivno lahko uporabljamo metrike kot sta natančnost in priklic, ki sta zelo znani na področjih iskalnikov in iskanju informacij [Hand & Christen, 2018], [Derczynski, 2016]. Čeprav ti dve metodi ocenjujeta uspešnost iskalnega sistema, vendarle nista zmožni oceniti uporabniške izkušnje, ki se pri priporočilnih sistemih ponavadi ocenjuje. Glavni problem knjižničarjev pri katalogiziranju je v tem, da je vrstilcev UDK veliko, hkrati pa je potrebno izbrati ustreznega. V veliki množici vrstilcev UDK je to lahko zahtevno in časovno potratno. Tako so knjižničarji zadovoljni že, če dobijo manjšo množico relevantnih vrstilcev UDK. Izmed vseh možnih vrstilcev UDK si želijo pridobiti torej samo najbolj ustrezne vrstilce UDK v pomoč, da kasneje ročno med njimi izberejo ustrezne. Zadovoljivo je tudi že, če dobijo na voljo vrhnje področje, od koder nato dalje samostojno določajo vrstilce UDK. Z vidika področja iskanja informacij gre pravzaprav za metriko priklica, ki v našem primeru meri razmerje moči množice preseka ustreznih vrstilcev UDK U in vseh vrnjenih vrstilcev UDK V , z močjo množice ustreznih vrstilcev UDK.

V našem primeru je torej metrika priklica pomembnejša od metrike natančnosti, saj gre za priporočilni sistem, ki nudi podporo pri polavtomatskem določanju vrstilcev UDK. Metrike, ki jih uporabljamo, zajemajo priklic (enačba 9), natančnost (enačba 10) in Fβ metriko (enačba 11) za vrednosti β = 1 in β = 50. Pri vrednosti β = 1 sta natančnost in priklic enakovredno uteženi, pri vrednosti β = 50 pa ima priklic 50-krat večjo težo kot natančnost.

$$R = \frac{|U| \cap |V|}{|U|} \tag{9}$$

$$P = \frac{|U| \cap |V|}{|V|} \tag{10}$$

$$F(\beta) = (1 + \beta^2) \frac{(PR)}{(\beta^2 P) + R} \tag{11}$$

Evalvacijo priporočanja vrstilcev UDK smo izvedli nad korpusom 10.000 besedil v slovenskem jeziku iz nacionalne infrastrukture odprtega dostopa, ki so imela podatek o klasifikaciji UDK. Pri tem smo izvzeli tista besedila, ki so bila v množici besedil, ki smo jih uporabili za učenje naivnega Bayesovega klasifikatorja in izračun uteži tf in idf . Meritve smo opravili za metodo BM25, naivni Bayesov klasifikator in hibridno metodo, ki združuje obe prej omenjeni metodi. Meritve smo ponovili pri različnih vrednostih za parameter kmax, ki predstavlja število vrnjenih vrstilcev. Pri tem smo se omejili na vrednosti kmax = [5, 10, 15]. V kombinaciji s parametrom kmax smo meritve ponovili tudi pri različnih vrednostih za globino hierarhije vrstilcev UDK. Globino hierarhije vrstilcev UDK udcp smo koračno po 2 znaka spreminjali na intervalu od 1 do 11 znakov. Dodatno smo v hibridni metodi spreminjali utež kazenskih vrednosti metode BM25 med 0.25 in 1 po koraku 0.25. Tabele 5, 6 in 7 vsebujejo rezultate meritev.

S hibridno metodo smo želeli povečati priklic ob predpostavki, da v našem scenariju uporabe metrika natanč- nosti ni pomembna za končnega uporabnika. Iz meritev je razvidno, da hibridna metoda v večini primerov dosega enake oziroma boljše vrednosti za metriko priklica in metriko Fβ=50 kot posamično uporabljeni metodi BM25 in Bayesov klasifikator. Opazimo, da je metoda BM25 tista, ki zagotavlja hkrati dobro natančnost in dober priklic, neodvisno od vseh preverjenih parametrov. Bayesov klasifikator je za vse preverjene vrednosti parametra kmax uporaben samo za vrhnja področja UDK (udcp = 1).

V scenariju, kadar vrnemo 5 priporočenih vrstilcev UDK (kmax = 5), hibridna metoda po metriki Fβ=50 dosega boljše vrednosti, kar je najbolj razvidno v primeru vrhnjih področij UDK (udcp = 1), za vse ostale preverjene globine hierarhije UDK pa je enakovredna metodi BM25. Največja izboljšava je pri vrhnjih področjih UDK. Kadar vrnemo 10 priporočenih vrstilcev UDK (kmax = 10) se hibridna metoda po metriki Fβ=50 znova obnese bolje kot metoda BM25. Izboljšava je vidna za vse preverjene globine hierarhije UDK, največja izboljšava pa je znova pri vrhnjih področjih UDK (udcp = 1). Kadar vrnemo 15 priporočenih vrstilcev UDK (kmax = 15), se po metriki Fβ=50 najbolje izkaže hibridna metoda na vseh globinah hierarhije UDK. Za vrhnja področja (udcp = 1) se tudi Bayesov klasifikator izkaže podobno dobro kot hibridna metoda.

Primerjali smo tudi delovanje hibridne metode ob različnih utežeh kazenskih vrednosti. V meritve in primerjavo smo vključili samo variante, kjer manj-

Tabela 5: **Rezultati meritev za uporabljene metode pri kmax = 5. Najvišje vrednosti so označene s krepko pisavo.**

kmax = 5	**Metoda**	***P***	***R***	$F_{\beta=1}$	$F_{\beta=50}$
udc_p = 1	BM25	**0.882**	0.852	**0.842**	0.852
	Bayes	0.248	0.836	0.371	0.835
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.267	**0.891**	0.399	**0.890**
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.267	**0.891**	0.399	**0.890**
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.267	**0.891**	0.399	**0.890**
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.267	**0.891**	0.399	**0.890**
udc_p = 3	BM25	**0.859**	0.908	**0.863**	0.908
	Bayes	0.097	0.343	0.147	0.343
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.281	0.912	0.416	0.911
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.284	0.916	0.420	0.915
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.286	**0.921**	0.422	**0.920**
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.286	**0.921**	0.422	**0.920**
udc_p = 5	BM25	**0.853**	**0.919**	**0.865**	**0.919**
	Bayes	0.032	0.105	0.048	0.105
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.277	0.903	0.411	0.902
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.286	0.918	0.423	0.917
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.287	**0.919**	0.424	0.918
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.287	**0.919**	0.424	0.918
udc_p = 7	BM25	**0.844**	**0.922**	**0.864**	**0.922**
	Bayes	0.049	0.154	0.072	0.154
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.279	0.904	0.414	0.903
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.289	**0.922**	0.426	0.921
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.289	**0.922**	0.426	0.921
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.289	**0.922**	0.426	0.921
udc_p = 9	BM25	**0.844**	0.922	**0.864**	0.922
	Bayes	0.051	0.161	0.075	0.161
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.281	0.906	0.416	0.905
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.290	**0.926**	0.427	**0.925**
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.290	**0.926**	0.427	**0.925**
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.290	**0.926**	0.427	**0.925**
udc_p = 11	BM25	**0.844**	0.922	**0.864**	0.922
	Bayes	0.050	0.156	0.073	0.156
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.280	0.905	0.415	0.904
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.290	**0.926**	0.427	**0.925**
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.290	**0.926**	0.427	**0.925**
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.290	**0.926**	0.427	**0.925**

šamo kazensko utež metodi BM25, ne pa tudi Bayesovemu klasifikatorju. Tako smo se odločili zato, ker manjšanje kazenskih uteži Bayesovemu klasifikatorju ne vodi v izboljšanje rezultatov metrik natančnosti, priklica, Fβ=1 in Fβ=50. Iz rezultatov meritev vidimo, da se manjšanje kazenskih uteži metodi BM25 splača vsaj do polovične vrednosti kazenske uteži (wBM25 = 0.5) za 5 vrnjenih zadetkov in vsaj do tričetrt vrednosti kazenske uteži (wBM25 = 0.75) za 10 in 15 vrnjenih zadetkov.

Tabela 6: **Rezultati meritev za uporabljene metode pri kmax = 10. Najvišje vrednosti so označene s krepko pisavo.**

kmax = 10	**Metoda**	***P***	***R***	$F_{\beta=1}$	$F_{\beta=50}$
$udc_p = 1$	BM25	**0.880**	0.852	**0.840**	0.852
	Bayes	0.146	0.902	0.245	0.900
	Hybrid $w_{BM25} = 1.0$, $w_{Bayes} = 1.0$	0.147	**0.906**	0.247	**0.904**
	Hybrid $w_{BM25} = 0.75$, $w_{Bayes} = 1.0$	0.147	**0.906**	0.247	**0.904**
	Hybrid $w_{BM25} = 0.5$, $w_{Bayes} = 1.0$	0.147	**0.906**	0.247	**0.904**
	Hybrid $w_{BM25} = 0.25$, $w_{Bayes} = 1.0$	0.147	**0.906**	0.247	**0.904**
$udc_p = 3$	BM25	**0.855**	0.914	**0.859**	0.914
	Bayes	0.062	0.439	0.107	0.438
	Hybrid $w_{BM25} = 1.0$, $w_{Bayes} = 1.0$	0.134	0.921	0.242	0.919
	Hybrid $w_{BM25} = 0.75$, $w_{Bayes} = 1.0$	0.144	0.923	0.243	0.921
	Hybrid $w_{BM25} = 0.5$, $w_{Bayes} = 1.0$	0.144	**0.927**	0.244	**0.925**
	Hybrid $w_{BM25} = 0.25$, $w_{Bayes} = 1.0$	0.144	**0.927**	0.244	**0.925**
$udc_p = 5$	BM25	**0.848**	0.920	**0.859**	0.920
	Bayes	0.032	0.212	0.055	0.212
	Hybrid $w_{BM25} = 1.0$, $w_{Bayes} = 1.0$	0.144	0.925	0.411	0.902
	Hybrid $w_{BM25} = 0.75$, $w_{Bayes} = 1.0$	0.145	**0.926**	0.245	0.923
	Hybrid $w_{BM25} = 0.5$, $w_{Bayes} = 1.0$	0.145	**0.926**	0.245	**0.924**
	Hybrid $w_{BM25} = 0.25$, $w_{Bayes} = 1.0$	0.145	**0.926**	0.245	**0.924**
$udc_p = 7$	BM25	**0.841**	0.925	**0.859**	0.925
	Bayes	0.035	0.217	0.059	0.217
	Hybrid $w_{BM25} = 1.0$, $w_{Bayes} = 1.0$	0.145	0.930	0.246	0.928
	Hybrid $w_{BM25} = 0.75$, $w_{Bayes} = 1.0$	0.146	0.932	0.247	0.930
	Hybrid $w_{BM25} = 0.5$, $w_{Bayes} = 1.0$	0.146	**0.933**	0.248	**0.931**
	Hybrid $w_{BM25} = 0.25$, $w_{Bayes} = 1.0$	0.146	**0.933**	0.248	**0.931**
$udc_p = 9$	BM25	**0.840**	0.925	**0.859**	0.925
	Bayes	0.033	0.209	0.056	0.209
	Hybrid $w_{BM25} = 1.0$, $w_{Bayes} = 1.0$	0.146	0.932	0.247	0.905
	Hybrid $w_{BM25} = 0.75$, $w_{Bayes} = 1.0$	0.128	0.824	0.217	0.822
	Hybrid $w_{BM25} = 0.5$, $w_{Bayes} = 1.0$	0.146	**0.933**	0.248	**0.931**
	Hybrid $w_{BM25} = 0.25$, $w_{Bayes} = 1.0$	0.146	**0.933**	0.248	**0.931**
$udc_p = 11$	BM25	**0.840**	0.925	**0.858**	0.925
	Bayes	0.032	0.203	0.055	0.203
	Hybrid $w_{BM25} = 1.0$, $w_{Bayes} = 1.0$	0.146	0.932	0.247	0.930
	Hybrid $w_{BM25} = 0.75$, $w_{Bayes} = 1.0$	0.146	0.932	0.247	0.930
	Hybrid $w_{BM25} = 0.5$, $w_{Bayes} = 1.0$	0.146	**0.933**	0.248	**0.931**
	Hybrid $w_{BM25} = 0.25$, $w_{Bayes} = 1.0$	0.146	**0.933**	0.248	**0.931**

Glede na porazdelitev razpoznanih izrazov UDK na hierarhično globino UDK (slika 2) smo ugotovili, da v primeru manjšega števila vrnjenih zadetkov ni bistvene razlike med uporabo BM25 in predlagane hibridne metode, kadar govorimo o odstotkovno največji pokritosti izbranega korpusa besedil, ki nastopi pri vrednostih parametra udcp = 5 in udcp = 7 ter metrikah priklica in Fβ=50. V splošnem smo ugotovili, da so vrednosti izbranih metrik približno enake za hierarhično globino UDK nad 7 znakov. Kadar pa se

Tabela 7: **Rezultati meritev za uporabljene metode pri kmax = 15. Najvišje vrednosti so označene s krepko pisavo.**

kmax = 15	**Metoda**	***P***	***R***	$F_{\beta=1}$	$F_{\beta=50}$
udc_p = 1	BM25	**0.880**	0.852	**0.840**	0.852
	Bayes	0.146	0.902	0.245	0.900
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.146	**0.906**	0.247	**0.904**
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.146	**0.906**	0.247	**0.904**
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.146	**0.906**	0.247	**0.904**
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.146	**0.906**	0.247	**0.904**
udc_p = 3	BM25	**0.854**	0.916	**0.857**	0.916
	Bayes	0.047	0.485	0.084	0.483
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.096	0.930	0.172	0.927
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.097	**0.931**	0.172	**0.928**
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.097	**0.931**	0.172	**0.928**
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.097	**0.931**	0.172	**0.928**
udc_p = 5	BM25	**0.846**	0.921	**0.857**	0.921
	Bayes	0.038	0.361	0.067	0.360
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.097	0.936	0.174	0.933
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.098	**0.938**	0.174	**0.935**
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.098	**0.938**	0.174	**0.935**
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.098	**0.938**	0.174	**0.935**
udc_p = 7	BM25	**0.839**	0.929	**0.857**	0.929
	Bayes	0.025	0.231	0.044	0.230
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.098	0.936	0.174	0.933
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.098	0.935	0.174	0.932
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.098	**0.939**	0.175	**0.936**
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.098	**0.939**	0.175	**0.936**
udc_p = 9	BM25	**0.838**	0.925	**0.856**	0.925
	Bayes	0.024	0.223	0.042	0.222
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.098	**0.936**	0.174	**0.933**
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.098	**0.936**	0.174	**0.933**
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.098	**0.936**	0.174	**0.933**
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.098	**0.936**	0.174	**0.933**
udc_p = 11	BM25	**0.838**	0.925	**0.856**	0.925
	Bayes	0.023	0.217	0.041	0.216
	Hybrid w_{BM25} = 1.0, w_{Bayes} = 1.0	0.098	**0.936**	0.174	**0.933**
	Hybrid w_{BM25} = 0.75, w_{Bayes} = 1.0	0.098	**0.936**	0.174	**0.933**
	Hybrid w_{BM25} = 0.5, w_{Bayes} = 1.0	0.098	**0.936**	0.174	**0.933**
	Hybrid w_{BM25} = 0.25, w_{Bayes} = 1.0	0.098	**0.936**	0.174	**0.933**

število vrnjenih zadetkov poveča, predlagana hibridna metoda konstantno vrača višje vrednosti izbranih metrik neodvisno od izbrane hierarhične globine UDK. Zaključujemo torej, da je uporaba predlagane hibridne metode ustrezna za polavtomatsko določanje vrstilcev UDK v obliki priporočilnega sistema, kjer knjižničarji dobijo predlagane vrstilce UDK na podlagi vhodnega besedila, med katerimi nato ročno izberejo ustrezne.

7 SKLEP

V članku smo predstavili hibridni pristop za priporočanje vrstilcev univerzalne decimalne klasifikacije. Opisali smo izbran korpus in predobdelavo besedil za uporabo v predlagani hibridni metodi. Prikazali smo kako z mešano hibridizacijo uporabimo metodi BM25 in naivni Bayesov klasifikator ter opisali preprosto združevalno funkcijo, ki oblikuje končni rezultat. Izvedli smo evalvacijo hibridne metode, metode BM25 in naivnega Bayesovega klasifikatorja, kjer smo ugotovili, da se hibridna metoda obnese bolje za metriki priklica in Fβ=50, ki sta bolj relevantni kot metrika natančnosti za scenarij uporabe sistema kot orodja za knjižničarje.

Predstavljen hibridni pristop lahko spreminjamo na več načinov in na več mestih. Ena izmed možnosti izboljšave je uporaba licenčne različice UDK vrstilcev, saj bi tako uspešno razpoznali večji delež izrazov UDK, še posebej na višji hierarhični globini UDK. Prav tako bi lahko izvedli optimizacijo metode BM25 za korpus, ki smo ga uporabljali, kjer bi z optimiziranjem parametrov k1 in b lahko iskali manjše izboljšave. Podobno bi lahko optimizirali vrednosti uteži kazenskih vrednosti. Hibridni pristop je vedno možno izboljšati s spreminjanjem združevalne funkcije M glede na potrebe končnega uporabnika ali pa z različnim načinom hibridizacije. Pri tem bi bila zanimiva predvsem utežni in kaskadni tip hibridizacije. Predstavljen hibridni pristop je prav tako ustrezen za uporabo pri določanju kandidatov dokumentov za podrobnejše preverjanje v sistemu za detekcijo podobnih vsebin. Nazadnje bi bilo zanimivo videti tudi, kako se na tem področju obnesejo nevronske mreže s povratno zanko, ki so v zadnjem obdobju zelo napredovale na področjih besedilnega rudarjenja in obdelave naravnega jezika.

LITERATURA

[1] Bai, X., Wang, M., Lee, I., Yang, Z., Kong, X., & Xia, F. (2019). *Scientific Paper Recommendation: A Survey*. IEEE Access, 7, 9324–9339.

[2] Beel, J., Aizawa, A., Breitinger, C., & Gipp, B. (2017). Mr. DLib: *Recommendations-as-a-Service (RaaS) for Academia*. In 2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL) (pp. 1–2).

[3] Bogaert, M., Lootens, J., den Poel, D. V., & Ballings, M. (2019). *Evaluating multi-label classifiers and recommender systems in the financial service sector.* European Journal of Operational Research, 279(2), 620 – 634.

[4] Burke, R. (2002). *Hybrid Recommender Systems: Survey and Experiments.* User Modeling and User-Adapted Interaction, 12(4), 331–370.

[5] Derczynski, L. (2016). *Complementarity, F-score, and NLP Evaluation. In Proceedings of the Tenth International Conference on Language Resources and Evaluation* (LREC 2016) (pp. 261–266). Portorož, Slovenia: European Language Resources Association (ELRA).

[6] Frank, E. & Paynter, G. W. (2004). *Predicting Library of Congress Classifications from Library of Congress Subject Headings.* J. Am. Soc. Inf. Sci. Technol., 55(3), 214–227.

[7] Godby, C. J. & Stuler, J. (2003). *The Library of Congress Classification as a Knowledge Base for Automatic Subject Categorization.* In Subject Retrieval in a Networked Environment: Proceedings of the IFLA Satellite Meeting held in Dublin, OH,14-16 August 2001 and sponsored by the IFLA Classification and Indexing Section, the IFLA Information Technology Section and OCLC (pp. 163–169).

[8] Hand, D. & Christen, P. (2018). *A note on using the F-measure for evaluating record linkage algorithms.* Statistics and Computing, 28(3), 539–547.

[9] He, B. & Ounis, I. (2003). *A Study of Parameter Tuning for Term Frequency Normalization.* In Proceedings of the Twelfth International Conference on Information and Knowledge Management, CIKM '03 (pp. 10–16). New York, NY, USA: ACM.

[10] He, B. & Ounis, I. (2005). Term Frequency Normalisation Tuning for BM25 and DFR Models. In D. E. Losada & J. M. Fernández-Luna (Eds.), Advances in Information Retrieval (pp. 200–214). Berlin, Heidelberg: Springer Berlin Heidelberg.

[11] Krauss, C., Merceron, A., & Arbanowski, S. (2019). *The Timeliness Deviation: A Novel Approach to Evaluate Educational Recommender Systems for Closed-Courses.* In Proceedings of the 9th International Conference on Learning Analytics & Knowledge, LAK19 (pp. 195–204). New York, NY, USA: ACM.

[12] Lops, P., de Gemmis, M., & Semeraro, G. (2011). *Content-based Recommender Systems: State of the Art and Trends,* (pp. 73–105). Springer US: Boston, MA.

[13] Lv, Y. & Zhai, C. (2011a). *Adaptive Term Frequency Normalization for BM25.* In Proceedings of the 20th ACM International Conference on Information and Knowledge Management, CIKM '11 (pp. 1985–1988). New York, NY, USA: ACM.

[14] Lv, Y. & Zhai, C. (2011b). *Lower-bounding Term Frequency Normalization. In Proceedings of the 20th ACM International Conference on Information and Knowledge Management,* CIKM '11 (pp. 7–16). New York, NY, USA: ACM.

[15] Manning, Christopher D. and Raghavan, Prabhakar and Schütze, Hinrich (2008). *Introduction to Information Retrieval.* New York, NY, USA: Cambridge University Press.

[16] Melville, P. & Sindhwani, V. (2017). *Recommender Systems*, (pp. 1056–1066). Springer US: Boston, MA.

[17] Monti, D., Palumbo, E., Rizzo, G., & Morisio, M. (2019). *Sequeval: An Offline Evaluation Framework for Sequence-Based Recommender Systems*. Information, 10, 174.

[18] Ojsteršek, M., Brezovnik, J., Kotar, M., Ferme, M., Hrovat, G., Bregant, A., & Borovič, M. (2014). *Establishing of a Slovenian open access infrastructure: a technical point of view.* Program, 48(4), 394–412.

[19] Porcel, C., Moreno, J., & Herrera-Viedma, E. (2009). *A multi-disciplinar recommender system to advice research resources in University Digital Libraries*. Expert Systems with Applications, 36(10), 12520–12528.

[20] Pu, P., Chen, L., & Hu, R. (2011). *A User-centric Evaluation Framework for Recommender Systems.* In Proceedings of the Fifth ACM Conference on Recommender Systems, RecSys '11 (pp. 157–164). New York, NY, USA: ACM.

[21] Rendle, S., Zhang, L., & Koren, Y. (2019). *On the Difficulty of Evaluating Baselines: A Study on Recommender Systems.* ArXiv, abs/1905.01395.

[22] Robertson, S. & Zaragoza, H. (2009). *The Probabilistic Relevance Framework. now.*
[23] Shani, G. & Gunawardana, A. (2011). *Evaluating Recommendation Systems*, (pp. 257–297). Springer US: Boston, MA.
[24] Slavic, A. (2004). *UDC implementation: From library shelves to a structured indexing language.* In International Cataloguing and Bibliographic Control., volume 33.3 (pp. 60–65).
[25] Trotman, A., Puurula, A., & Burgess, B. (2014). *Improvements to BM25 and Language Models Examined.* In Proceedings of the 2014 Australasian Document Computing Symposium, ADCS '14 (pp. 58:58–58:65). New York, NY, USA: ACM.
[26] UDC Consortium (UDCC) (2012). *Multilingual Universal Decimal Classifi- cation Summary* (UDCC Publication No. 088).
[27] Vargas, S., Hristakeva, M., & Jack, K. (2016). *Mendeley: Recommendations for Researchers.* In RecSys '16 Proceedings of the 10th ACM Conference on Recommender Systems (pp. 365–365). Boston, MA, USA.
[28] Wang, J. (2009). *An extensive study on automated Dewey Decimal Classification.* Journal of the American Society for Information Science and Technology, 60(11), 2269–2286.
[29] Zalokar, Matjaž (2002a). *Spletni splošni slovenski geslovnik*. http://old.nuk.uni-lj.si/ssg/geslovnik.html.
[30] Zalokar, Matjaž (2002b). *Splošni slovenski geslovnik.* Organizacija znanja, 7, 3–4.

▪

Mladen Borovič je doktorski študent in asistent na Fakulteti za elektrotehniko, računalništvo in informatiko na Univerzi v Mariboru. Njegovo raziskovalno delo obsega področja priporočilnih sistemov, iskalnih sistemov, porazdeljenih računalniških sistemov, odkrivanja podobnih vsebin, besedilnega rudarjenja in obdelave naravnega jezika. Še posebej se ukvarja s hibridnimi priporočilnimi sistemi in uporabo metod umetne inteligence v besedilnem rudarjenju.

▪

Sandi Majninger je doktorski študent in asistent na Fakulteti za elektrotehniko, računalništvo in informatiko na Univerzi v Mariboru. Raziskovalno je aktiven na področju obdelave naravnega jezika, odkrivanja podobnih vsebin ter ugotavljanju pomena iz besedil. Med drugim se ukvarja tudi z avtomatskim ocenjevanjem pomenske pravilnosti odgovorov na vprašanja odprtega tipa in avtomatskim ocenjevanjem daljših pisnih sestavkov ter esejev.

▪

Jani Dugonik je doktorski študent in asistent na Fakulteti za elektrotehniko, računalništvo in informatiko. Nje- gova raziskovalna področja vključujejo evolucijsko računanje, optimizacijske metode, procesiranje naravnega jezika in globoko učenje.
Marko Ferme je raziskovalec na Fakulteti za elektrotehniko, računalništvo in informatiko na Univerzi v Mariboru. Njegova raziskovalna področja obsegajo procesiranje naravnega jezika, sisteme za odgovarjanje na vprašanja v naravnem jeziku, ontologije in semantični splet, aktiven pa je tudi na več raziskovalnih in komercialnih projektih na področju digitalnih knjižnic.ziskovalnih projektih s področja strateškega planiranja, metodologij razvoja informacijskih sistemov, uporabe inteligentnih sistemov, avtomatizacije poslovnih procesov in obvladovanja ter porazdelitve velike količine podatkov.

▪

Milan Ojsteršek je raziskovalec na Fakulteti za elektrotehniko, računalništvo in informatiko na Univerzi v Mariboru. Njegova raziskovalna področja zajemajo heterogene računalniške sisteme, digitalne knjižnice, semantični splet in storitveno usmerjene arhitekture.Marko Ferme je raziskovalec na Fakulteti za elektrotehniko, računalništvo in informatiko na Univerzi v Mariboru. Njegova raziskovalna področja obsegajo procesiranje naravnega jezika, sisteme za odgovarjanje na vprašanja v naravnem jeziku, ontologije in semantični splet, aktiven pa je tudi na več raziskovalnih in komercialnih projektih na področju digitalnih knjižnic.ziskovalnih projektih s področja strateškega planiranja, metodologij razvoja informacijskih sistemov, uporabe inteligentnih sistemov, avtomatizacije poslovnih procesov in obvladovanja ter porazdelitve velike količine podatkov.