

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 4909

**Analiza metagenomskog uzorka
dobivenog sekvenciranjem
koristeći uređaje treće generacije**

Marina Rupe

Zagreb, lipanj 2017.

Zagreb, 7. ožujka 2017.

ZAVRŠNI ZADATAK br. 4909

Pristupnik: **Marina Rupe (0036483027)**
Studij: Računarstvo
Modul: Računarska znanost

Zadatak: **Analiza metagenomskog uzorka dobivenog sekvenciranjem koristeći uređaje treće generacije**

Opis zadatka:

Brza i jeftina analiza metagenomskog uzorka može biti korisna za dijagnozu bolesti, kontrolu kvalitete hrane i utvrđivanje štetnih nametnika na biljkama. Tradicionalne laboratorijske metode su ili dugotrajne ili namijenjene za samo jednu vrstu. Za razliku od prijašnjih tehnologija, očitavanja dobivena uređajima treće generacije su puno dulja, ali sadrže i znatno veći postotak pogrešaka. Cilj ovog rada je izrada alata za analizu metagenomskog uzorka koji će moći koristiti duga očitavanja s velikim postotkom pogreške i koji će moći analizirati očitavanja kontinuirano, kako postanu dostupna. U prvom koraku postupka pronalaze se svi organizmi čije sekvence u svom dijelu imaju veliku sličnost s očitanim uzorcima, u drugom se uklanjaju sva ona očitavanja koja sadrže genetski materijal domaćina, a u trećem se utvrđuje koji organizmi su stvarno prisutni. U prvom koraku će se iskoristiti neko od postojećih rješenja poput Krakena ili Metaphlana, dok će se za drugi i treći korak razviti vlastito. Rješenje treba biti napisano u nekom od standardnih programskih jezika. Programski kod je potrebno komentirati i pri pisanju pratiti neki od standardnih stilova. Kompletanu aplikaciju postaviti na repozitorij Github.

Zadatak uručen pristupniku: 10. ožujka 2017.

Rok za predaju rada: 9. lipnja 2017.

Mentor:



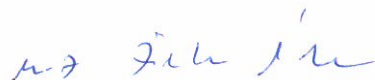
Izv. prof. dr. sc. Mile Šikić

Djelovođa:



Doc. dr. sc. Tomislav Hrkać

Predsjednik odbora za
završni rad modula:



Prof. dr. sc. Siniša Srbljić

SADRŽAJ

Popis slika	vi
Popis tablica	vii
1. Uvod	1
2. Pregled područja	2
2.1. Uređaji za sekvenciranje	2
2.2. Alati za analizu metagenomskog uzorka	2
3. Metode	3
3.1. Reduciranje baze	3
3.2. Maksimalna izglednost	3
3.3. Model mješavine	4
3.4. EM algoritam	5
3.4.1. Općeniti EM algoritam	5
3.4.2. EM algoritam korišten u metodi	6
4. Implementacija	9
4.1. Prvi korak	9
4.2. Drugi korak	11
4.3. Treći korak	12
5. Rezultati	14
5.1. <i>Klebsiella pneumoniae</i>	15
5.2. <i>Salmonella enterica</i>	16
5.3. <i>Staphylococcus aureus</i>	18
5.4. Mješavina	20
5.5. Osvrt na rezultate	20

6. Diskusija	22
7. Zaključak	24
Literatura	25

POPIS SLIKA

4.1. Organizacija modula	9
4.2. Prvi korak – reduciranje baze	11
4.3. Drugi korak – mapiranje	12
4.4. Treći korak – klasifikator	13

POPIS TABLICA

5.1. PBSIM parametri	14
5.2. Bakterije korištene za testiranje	14
5.3. Klebsiella pneumoniae za pokrivenost x1, rezultati nakon prvog potkoraka	15
5.4. Klebsiella pneumoniae za pokrivenost x1, rezultati nakon drugog potkoraka	15
5.5. Klebsiella pneumoniae za pokrivenost x5, rezultati nakon prvog potkoraka	15
5.6. Klebsiella pneumoniae za pokrivenost x5, rezultati nakon drugog potkoraka	16
5.7. Klebsiella pneumoniae za pokrivenost x10, rezultati nakon prvog potkoraka	16
5.8. Klebsiella pneumoniae za pokrivenost x10, rezultati nakon drugog potkoraka	16
5.9. Salmonella enterica za pokrivenost x1, rezultati nakon prvog potkoraka	17
5.10. Salmonella enterica za pokrivenost x1, rezultati nakon drugog potkoraka	17
5.11. Salmonella enterica za pokrivenost x5, rezultati nakon prvog potkoraka	17
5.12. Salmonella enterica za pokrivenost x5, rezultati nakon drugog potkoraka	17
5.13. Salmonella enterica za pokrivenost x10, rezultati nakon prvog potkoraka	18
5.14. Salmonella enterica za pokrivenost x10, rezultati nakon drugog potkoraka	18
5.15. Staphylococcus aureus za pokrivenost x1, rezultati nakon prvog potkoraka	18
5.16. Staphylococcus aureus za pokrivenost x1, rezultati nakon drugog potkoraka	19
5.17. Staphylococcus aureus za pokrivenost x5, rezultati nakon prvog potkoraka	19

5.18. Staphylococcus aureus za pokrivenost x5, rezultati nakon drugog potkoraka	19
5.19. Staphylococcus aureus za pokrivenost x10, rezultati nakon prvog potkoraka	19
5.20. Staphylococcus aureus za pokrivenost x10, rezultati nakon drugog potkoraka	20
5.21. Mješavina bakterija: Salmonella enterica (x15), Staphylococcus aureus (x10) i Klebsiella pneumoniae (x5), rezultati nakon drugog potkoraka	20

1. Uvod

Današnjica – vrijeme brzih, turbulentnih promjena. Vrijeme iznenadnih zbivanja kada su događanja oko čovjeka tako brza i kompleksna da zahtijevaju brze rezultate i odluke, jer se u protivnom negativna djelovanja reflektiraju na veliki broj ljudi. Jedan od bitnih, tj. ključnih faktora koji utječu na kvalitetu života ljudi su zdravstvena zaštita i ishrana. U medicini je vrlo bitna brza i točna dijagnoza bolesti radi adekvatnog liječenja pacijenata. Osim toga, u današnje vrijeme sve se više pažnje posvećuje kvaliteti hrane. Poljoprivreda se modernizirala i sve je veći naglasak na kontroli kvalitete hrane, utvrđivanju štetnih nametnika na biljkama kako bi se usavršila proizvodnja hrane, utvrđivanju bolesti u mesnim proizvodima i sl.

Za sve navedene primjere, a i razne druge, vrlo je bitno pronaći efikasnu metodu za određivanje organizama prisutnih u uzorku. Tradicionalne laboratorijske metode nisu se pokazale dovoljno dobrima. Obično su ili dugotrajne ili ograničene na jedan organizam, što predstavlja veliku prepreku u analizi. Osim vremenskog ograničenja i pokrivenosti različitih organizama, da bi se metoda široko koristila poželjno je da bude što jeftinija. S obzirom na važnost ovog problema, bilo je bitno potražiti bolji način za njegovo rješavanje.

Cilj ovog završnog rada bio je napraviti alat za analizu metagenomskog uzorka koji bi određivao koji su patogeni organizmi prisutni u uzorku. U Poglavlju 2 dan je pregled područja, u Poglavlju 3 objašnjene i izvedene su metode koje se koriste u rješenju i čija je implementacija kroz korake objašnjena u Poglavlju 4. Poglavlje 5 donosi prikaz rezultata dobivenima iz testiranja alata na primjerima uzoraka uz dodatne komentare. U Poglavlju 6 prokomentiran je alat, tj. poglavlje donosi diskusiju o uspješnosti implementiranog alata i cjelokupne ideje. Na kraju, Poglavlje 7 donosi zaključak završnog rada.

2. Pregled područja

2.1. Uređaji za sekvenciranje

Razvojem znanosti, s vremenom su se poboljšavali uređaji za sekvenciranje, tako da danas postoje tri generacije uređaja za sekvenciranje[9]. Za razliku od prve dvije generacije, sekvenciranjem koristeći uređaje treće generacije dobivaju se puno dulji sljedovi, no uz veći postotak pogrešaka tih uzorka. Na prvi pogled možda se čini da je to loš pristup, no postotak pogreške zapravo ne predstavlja velik problem.

Budući da je sekvenciranje postalo pristupačnije zbog manje skupe opreme, može se analizirati puno više uzoraka, a osiguranje točnosti za nesavršen sekvencirani uzorak prelazi na algoritme. Zbog toga je pažnja usmjerena na razvoj što boljih algoritama, dakle algoritama koji će što brže i što točnije identificirati organizme prisutne u uzorku.

U ovom završnom radu analiziraju se metagenomski uzorci dobiveni korištenjem upravo uređaja treće generacije, tzv. *nanopore* sekvenciranjem (Oxford Nanopore).

2.2. Alati za analizu metagenomskog uzorka

Otkad su mogućnosti sekvenciranja postale raširenije, počeli su se razvijati razni alati za analizu metagenomskog uzorka. Alati se međusobno razlikuju u brzini i točnosti te u onome što koriste za identificiranje organizama. Algoritmi za identifikaciju organizama (engl. *binning algorithms*) mogu se temeljiti ili na kompozicijskim značajkama ili na poravnanjima (sličnosti) sljedova, ili mogu biti kombinacija i jednog i drugog. Unutar tih kategorija opet postoje podjele. Alat koji je implementiran u svrhu ovog završnog rada koristi kompozicijske značajke, konkretno gene markere (engl. *marker genes*) za identifikaciju genoma prisutnih u uzorku. Neki od dostupnih alata su primjerice PathoScope[10] i MetaPhlAn[12].

3. Metode

3.1. Reduciranje baze

Potreba za reduciranom bazom organizama javlja se zbog zahtjeva brzine. Bilo bi vrlo sporo kad bi se mapiranje uzoraka s bazom (drugi korak) izvršavalo nad bazom svih organizama. Uz to, želimo izbjeći lažna pozitivna mapiranja jer smanjuju točnost algoritma[13]. Zbog toga koristimo gene markere te njima pridružimo odgovarajuće organizme koji im pripadaju da bi se kasnije metagenomski uzorci mapirali na gene markere, a ne na svaki organizam zasebno. Implementacija koraka reduciranja baze i korišteni izvori podataka detaljno su objašnjeni u Poglavlju 4.

3.2. Maksimalna izglednost

Maksimalna izglednost (engl. MLE – *maximum likelihood estimation*) je metoda za procjenu parametara statističkog modela na način da se pronađu parametri koji maksimiziraju funkciju izglednosti $L(\theta \mid D)$ gdje θ skup parametara koje treba odrediti, a D je skup podataka ($D = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)}\}$)[6][13].

Za početak potrebno je definirati funkciju gustoće zajedničke vjerojatnosti $p(\mathbf{x} \mid \theta)$ (engl. *joint probability density function*) za svako očitavanje. Uz pretpostavku da su podatci međusobno nezavisni i uz jednoliko raspodijeljen uzorak, ona iznosi:

$$p(D \mid \theta) = p(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)} \mid \theta) = p(\mathbf{x}^{(1)} \mid \theta)p(\mathbf{x}^{(2)} \mid \theta) \dots p(\mathbf{x}^{(n)} \mid \theta) \quad (3.1)$$

odnosno:

$$p(D \mid \theta) = \prod_{i=1}^N p(\mathbf{x}^{(i)} \mid \theta) \equiv L(\theta \mid D) \quad (3.2)$$

Funkcija se može promatrati i iz perspektive parametra θ uz fiksiranu vrijednost parametra D . Iz tog je pogleda ona zapravo ekvivalentna funkciji izglednosti $L(\theta \mid D)$

koja se traži. Cilj je procijeniti parametar $\hat{\theta}_{ML}$ koji maksimizira funkciju izglednosti.

$$\hat{\theta}_{ML} = \arg \max_{\theta} L(\theta \mid D) \quad (3.3)$$

U praksi je često korisno maksimizirati logaritam funkcije izglednosti umjesto da se maksimizira samu funkciju. Ta se funkcija naziva funkcija log-izglednosti (engl. *log-likelihood function*). Ako se logaritmiraju funkciju izglednosti, monotonost funkcije ostat će očuvana i zbog toga će izračun parametara biti točan. Također, logaritmiranjem funkcije $L(\theta \mid D)$ umjesto umnoška vjerojatnosti dobit će se suma njihovih logaritama što je mnogo jednostavnije pri računanju, a bitan je samo krajnji omjer tih vrijednosti[4].

$$L(\theta \mid D) = \ln \prod_{i=1}^N p(\mathbf{x}^{(i)} \mid \theta) = \sum_{i=1}^N \ln p(\mathbf{x}^{(i)} \mid \theta) \quad (3.4)$$

3.3. Model mješavine

U statistici, model mješavine (engl. *mixture model*) je vjerojatnosni model za predstavljanje prisutnosti subpopulacije unutar ukupne populacije bez da se izravno identificira koji uzorak pripada kojoj subpopulaciji[5].

$$p(\mathbf{x}) = \sum_{k=1}^K \pi_k p(\mathbf{x} \mid \theta_k) \quad (3.5)$$

Parametar π je koeficijent mješavine. Svaki pojedini parametar π_k predstavlja udio subpopulacije k u mješavini. Neka se broj očitavanja označi s N i broj genoma sa G . U tom slučaju, suma parametara π za svaki genom daje jedan, odnosno $\sum_{k=1}^G \pi_k = 1$.

Cilj je za svako očitavanje i odrediti vjerojatnost da pripada genomu j te na temelju tih vrijednosti odrediti parametar modela:

$$\theta = \{P(G_k), \theta_k\}_{k=1}^G \quad (3.6)$$

Ako se parametar θ_k označi sa G_k , a π_k s $P(G_k)$, vjerojatnost za mješavinu može se prikazati kao:

$$p(\mathbf{x}) = \sum_{k=1}^G P(G_k) p(\mathbf{x} \mid G_k) \quad (3.7)$$

Zatim treba odrediti vjerojatnost da \mathbf{x} potječe iz genoma k , odnosno treba odrediti *a posteriori* vjerojatnost parametara $P(G \mid \mathbf{x})$. Za to se koristi poznato Bayesovo pravilo:

$$P(G \mid \mathbf{x}) = \frac{P(G_k)p(\mathbf{x} \mid G_k)}{\sum_{j=1}^G P(G_j)p(\mathbf{x} \mid G_j)} = \frac{\pi_k p(\mathbf{x} \mid \boldsymbol{\theta}_k)}{\sum_{j=1}^G \pi_j p(\mathbf{x} \mid \boldsymbol{\theta}_j)} \quad (3.8)$$

Funkcija log-izglednosti za prethodni izraz:

$$\ln L(\boldsymbol{\theta} \mid D) = \sum_{i=1}^N \ln \sum_{k=1}^G \pi_k p(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}_k) \quad (3.9)$$

3.4. EM algoritam

3.4.1. Općeniti EM algoritam

EM algoritam (engl. *Expectation maximization algorithm*) je iterativna metoda za pronalaženje maksimalne izvjesnosti, odnosno maksimalne *a posteriori* (engl. MAP - *maximum a posteriori*) procjene parametara u statističkim modelima[2]. Primjenjuje se u slučajevima kad neki parametar modela nije poznat, tj. ne može se izravno promatrati. Takav se parametar naziva latentna varijabla (engl. *latent variable*).

EM algoritam pronalazi parametre $\boldsymbol{\theta}$ koji maksimiziraju prethodno definiranu funkciju log-izglednosti $\ln L(\boldsymbol{\theta} \mid D)$. Algoritam zadanom modelu dodaje skup latentnih varijabli \mathbf{Z} . Skup $\{D, \mathbf{Z}\}$ je potpun skup (engl. *complete*), dok je skup D nepotpun (engl. *incomplete*). Koristi se zajednička vjerojatnosna funkcija $p(D, \mathbf{Z} \mid \boldsymbol{\theta})$.

Funkciju $p(D \mid \boldsymbol{\theta})$ izražavamo kao:

$$p(D \mid \boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(D, \mathbf{Z} \mid \boldsymbol{\theta}) \quad (3.10)$$

Nepotpuna funkcija log-izglednosti računa se:

$$\ln L(\boldsymbol{\theta} \mid D) = \ln p(D \mid \boldsymbol{\theta}) = \ln \sum_{\mathbf{Z}} p(D, \mathbf{Z} \mid \boldsymbol{\theta}) \quad (3.11)$$

Potpuna funkcija log-izglednosti računa se:

$$\ln L(\boldsymbol{\theta} \mid D, \mathbf{Z}) = \ln p(D, \mathbf{Z} \mid \boldsymbol{\theta}) \quad (3.12)$$

Budući da skup \mathbf{Z} nije dostupan, ne može se izravno računati potpuna log-izglednost već je potrebno pronaći njenu očekivanu vrijednost, dakle $E(\ln L(\boldsymbol{\theta} \mid D, \mathbf{Z}))$.

Ovdje na red dolazi EM algoritam. Da bi se maksimiziralo očekivanje E , iterativno se provode dva koraka EM algoritma – očekivanje (engl. *expectation*) i maksimizacija (engl. *maximization*), odnosno skraćeno E-korak i M-korak. Pseudokod općenitog EM algoritma prikazan je na prikazu algoritma 1. Na početku se inicijaliziraju parametri θ^0 . Postupak se ponavlja do konvergencije parametara θ [8].

U E-koraku računa se očekivanje potpune log-izglednosti. Pri tome se fiksiraju parametri $\theta^{(t)}$:

$$Q(\theta \mid \theta^{(t)}) = E_{Z \mid D, \theta^{(t)}} [\ln L(\theta \mid D, Z)] = \sum_Z P(Z \mid D, \theta^{(t)}) \ln p(D, Z \mid \theta) \quad (3.13)$$

U M-koraku određuju se novi parametri $\theta^{(t+1)}$ koji maksimiziraju rezultat iz E-koraka:

$$\theta^{(t+1)} = \arg \max_{\theta} Q(\theta \mid \theta^{(t)}) \quad (3.14)$$

Algorithm 1 Općeniti EM algoritam

Ulaz: θ^0 – inicijalizirati parametre

$t \leftarrow 0$

while parametri θ nisu konvergirali **do**

E-korak: izračunati $P(Z \mid D, \theta^t)$

M-korak: $\theta^{(t+1)} \leftarrow \arg \max_{\theta} Q(\theta \mid \theta^t)$

$t \leftarrow t + 1$

end while

3.4.2. EM algoritam korišten u metodi

Pseudokod EM algoritma implementiranog u rješenju prikazan je na prikazu algoritma 2. Za početak, potrebno je prilagoditi algoritam zadanom problemu.

Algoritam kao ulaz dobiva rezultate mapiranja uzorka s reduciranom bazom i na temelju njih treba inicijalizirati parametre. Za rješavanje problema koristit će se prethodno definirani model mješavine. Svako očitavanje od njih R može biti mapirano na neki gen marker ili ne biti mapirano. U slučaju kad je očitavanje mapirano na gen marker, razlikuju se dva slučaja. Ako je genu markeru pridružen samo jedan genom, za očitavanje se kaže da je jedinstveno (engl. *unique read*) jer mu je pridružen samo jedan genom. U suprotnom, ako genu markeru pripada više genoma, očitavanju se pridružuje više genoma i takvo očitavanje je nejedinstveno (engl. *non-unique read*)[13].

Neka postoji vektor $\mathbf{Z} = (z_1, \dots, z_G)$ za koji vrijedi da je $z_j = 1$ ako očitavanje pripada genomu j , a u suprotnom neka je $z_j = 0$. Taj je vektor zapravo zastavica koja govori o pripadnosti očitavanja genomima te u skladu s tim poprima vrijednost 1 samo za jedan genom.

Vrijedi:

$$P(z_j = 1) = \pi_j \quad (3.15)$$

Odnosno:

$$P(\mathbf{z}) = \prod_{j=1}^G \pi_j^{z_j} \quad (3.16)$$

Za jedinstvena očitavanja poznat je parametar \mathbf{z} jer znamo kojem genomu pripada očitavanje. Problem se javlja kod nejedinstvenih očitavanja. Zbog toga, za svako očitavanje definiramo rezultat mapiranja $\mathbf{q}^{(i)} = (q_1^{(i)}, \dots, q_G^{(i)})$ nad svakim pojedinim genomom. Tim se parametrom označava nesigurnost mapiranja. Uz njega se definira i parametar $\boldsymbol{\delta}^{(i)} = (\delta_1^{(i)}, \dots, \delta_G^{(i)})$ koji predstavlja parametar za ponovnu raspodjelu (engl. *reassignment*) za svaki genom, odnosno koliko bi se nejedinstvenih očitavanja trebalo raspodijeliti tom genomu. Parametar \mathbf{y} za svako od R očitavanja govori je li jedinstveno ($y^{(i)} = 1$) ili nije ($y^{(i)} = 0$). Ako je očitavanje i jedinstveno, nije potrebno koristiti parametar $\boldsymbol{\delta}^{(i)}$. U skladu s tim, parametar $\boldsymbol{\theta}$ definira se kao $\delta_j^{(1-y^{(i)})} q_j^{(i)}$ [10][13]. Funkcija log-izglednosti:

$$p(x^{(i)} | \mathbf{z}, \boldsymbol{\theta}) = \prod_{j=1}^G p(x^{(i)} | \boldsymbol{\theta}_j)^{z_j} = \prod_{j=1}^G (\delta_j^{(1-y^{(i)})} q_j^{(i)})^{z_j} \quad (3.17)$$

Iz formula 3.16 i 3.17 dobije se zajednička distribucija:

$$P(\mathbf{z})p(x^{(i)} | \mathbf{z}, \boldsymbol{\theta}) = \prod_{j=1}^G \pi_j^{z_j} \prod_{j=1}^G (\delta_j^{(1-y^{(i)})} q_j^{(i)})^{z_j} = \prod_{j=1}^G \pi_j^{z_j} (\delta_j^{(1-y^{(i)})} q_j^{(i)})^{z_j} \quad (3.18)$$

Na temelju prethodno izračunatih formula može se izraziti potpuna funkcija log-izglednosti:

$$\ln L(\boldsymbol{\theta} | D, \mathbf{Z}) = \ln \prod_{i=1}^R \prod_{j=1}^G \pi_j^{z_j^{(i)}} p(x^{(i)} | \boldsymbol{\theta}_j)^{z_j^{(i)}} = \sum_{i=1}^R \sum_{j=1}^G z_j^{(i)} (\ln \pi_j + \ln p(x^{(i)} | \boldsymbol{\theta}_j)) \quad (3.19)$$

Uz pretpostavku da parametri $\boldsymbol{\pi}$ i $\boldsymbol{\theta}$ slijede Dirichletovu distribuciju, vrijedi:

$$p(\boldsymbol{\pi} \mid \mathbf{a}) \sim \prod_{j=1}^G \pi_j^{a_j-1} \quad (3.20)$$

$$p(\boldsymbol{\theta} \mid \mathbf{b}) \sim \prod_{j=1}^G \theta_j^{b_j-1} \quad (3.21)$$

Parametar a_j za genom j predstavlja broj jedinstvenih očitavanja za taj genom, dok parametar b_j predstavlja broj nejedinstvenih očitavanja za genom j .

Algorithm 2 EM algoritam korišten u metodi

Ulaz: inicijalizirati parametre: $\{\boldsymbol{\pi}_j, \boldsymbol{\theta}_j, \boldsymbol{\delta}_j\}_{j=1}^G$

$t \leftarrow 0$

while nisu konvergirali parametri $\boldsymbol{\pi}$ i $\boldsymbol{\delta}$ ili $L(\boldsymbol{\theta} \mid D)$ **do**

E-korak: izračunati $h_j^{(i)}$ za svako očitavanje $x^{(i)} \in D$ i svaki genom j koristeći trenutne vrijednosti parametara:

$$h_j^{(i)} = \frac{\pi_j \delta_j^{(1-y_i)} q_j^{(i)}}{\sum_{k=1}^G \pi_k \delta_k^{(1-y_i)} q_k^{(i)}}$$

M-korak: izračunati nove vrijednosti parametara $\boldsymbol{\pi}$ i $\boldsymbol{\delta}$ za svaki genom j koristeći izračunate vrijednosti h_j

$$\pi_j = \frac{\sum_{i=1}^R h_j^{(i)} + a_j}{N + \sum_{k=1}^G a_k}$$

$$\delta_j = \frac{\sum_{i=1}^R (1-y^{(i)}) h_j^{(i)} + b_j}{\sum_{i=1}^R (1-y^{(i)}) + \sum_{k=1}^G b_k}$$

Log-izglednost: izračunati trenutnu vrijednost:

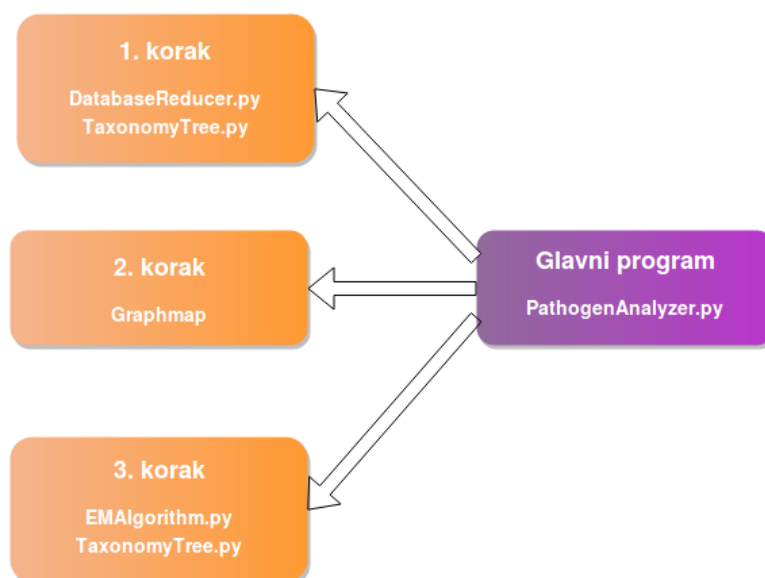
$$\ln L(\boldsymbol{\theta} \mid D) = \sum_{i=1}^R \ln \sum_{j=1}^G \pi_j p(x^{(i)} \mid \boldsymbol{\theta})$$

$t \leftarrow t + 1$

end while

4. Implementacija

Rješenje je napisano u programskom jeziku Python 3.5. Logika programa je podijeljena u module *DatabaseReducer.py*, *EMAlgorithm.py* i *TaxonomyTree.py* koji sadržavaju istoimene razrede.



Slika 4.1: Organizacija modula

4.1. Prvi korak

Prvi korak je kreiranje reducirane baze podataka i za njega je zadužen modul *DatabaseReducer.py*. Za izgradnju taksonomskog stabla poziva modul *TaxonomyTree.py*. Korišteni su geni markeri iz Metaphlana. Oni se nalaze u datoteci pod imenom *markers_info.txt* na web stranici Metaphlana[12]. Osim datoteke s genima markerima, korištena je datoteka *nodes.dmp* koja sadrži popis taksonomskih jedinki s njihovim taksonomskim brojevima (TI – engl. *taxonomy ID*), taksonomskim brojevima njio-

vih roditelja i drugim podacima vezanima za tu jedinku, i datoteka *names.dmp* koja sadrži ime za svaki taksonomski broj[7].

U prvom dijelu ovog koraka datoteka s podacima o taksonomskim jedinicama povezuje se s datotekom s imenima taksonomskih jedinica preko taksonomskog broja. Svi znakovi u imenima koji nisu alfanumerički trebaju se zamijeniti znakom "_" jer su imena organizama u datoteci s markerima u tom formatu. Također, u datoteci s imenima ponuđeno je više vrsta imena od kojih će se u rješenju koristiti znanstveno ime (engl. *scientific name*). Nakon toga izgradi se taksonomsko stablo jer su nam potrebni podatci o hijerarhiji organizama.

Nakon izgradnje stabla, parsira se datoteka s genima markerima. Svakom genu markeru pridružen je jedinstven identifikator (GI – engl. *gene ID*). Svaki redak datoteke predstavlja jedan gen marker i uz razne druge podatke sadrži dva bitna polja: "clade" polje koje sadrži oznaku taksonomske razine i ime kladusa¹ za taj marker razdvojene s "_" (npr. s__*Streptomyces_sp_KhCrAH_244*) i "ext" polje koji sadrži listu sojeva koji pripadaju tom markeru u obliku njihovih asemblija (npr. GCF_000024865), a ne pripadaju zadanom kladusu. Polje "ext" može biti i prazno.

Oznake taksonomskih razina kojima kladus može pripadati su:

a – sve taksonomske razine (engl. *all*);

k – carstvo (engl. *kingdom*);

p – koljeno (engl. *phylum*);

c – razred (engl. *class*);

o – red (engl. *order*);

f – porodica (engl. *family*);

g – rod (engl. *genus*);

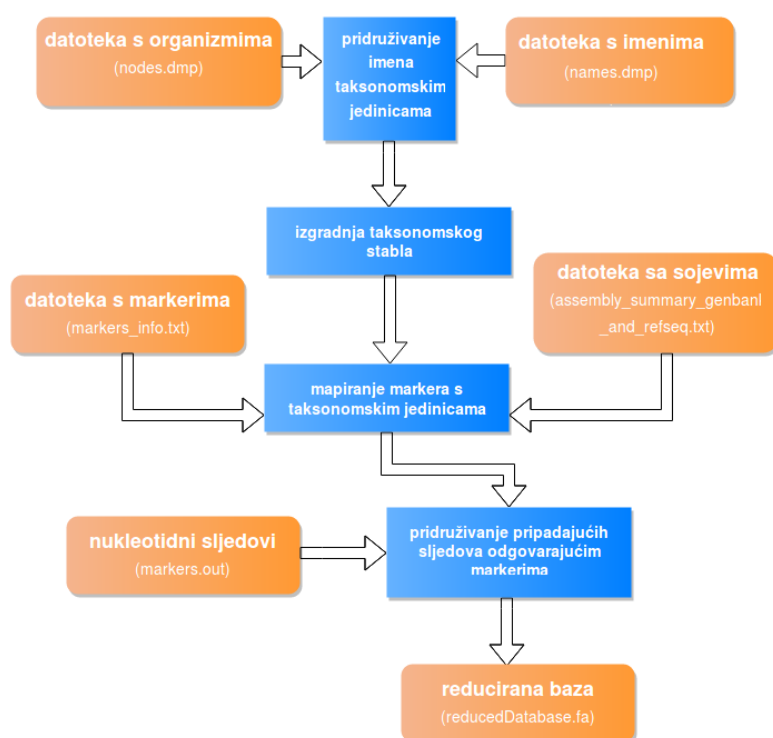
s – vrsta (engl. *species*).

Svakom markeru treba pridružiti listu taksonomskih brojeva organizama koji pripadaju tom markeru, odnosno trebamo dohvatiti taksonomske brojeve za kladus i sojeve. Svi pridruženi organizmi trebaju biti zadani na razini vrste (engl. *species*) zbog drugog koraka rješenja koji radi na razini vrste. Taksonomski broj kladusa dohvaćamo preko njegovog imena. U slučaju da je neki kladus iznad razine vrste, koristi se prethodno izgrađeno taksonomsko stablo kako bi se dohvatila sva djeca koja su na razini vrste. Dakle, spuštanjem po taksonomskom stablu do razine vrste dohva-

¹"Kladus (grč. *klados* = grana) ili *monophylum* je grana – grupa oblika života koja se sastoji od zajedničkog pretka i svih njegovih potomaka i predstavlja jedinstvenu granu na stablu života[3]."

timo odgovarajuće taksonomske brojeve koje pridružimo genu markeru. Da bismo dohvatili taksonomske brojeve za sojeve, potrebna je datoteka koja sadrži podatke o taksonomskim brojevima i imenima za svaki soj. Za tu namjenu, korištene su datoteke *assembly_summary_genbank.txt* i *assembly_summary_refseq.txt*[1] spojene u jednu zajedničku datoteku pod imenom *assembly_summary_genbank_and_refseq.txt*.

Nakon pridruživanja taksonomskih brojeva svakom markeru, markeri se na temelju svog identifikatora sparuju s pripadajućom nukleotidnom sekvencom te se kreira reducirana baza.

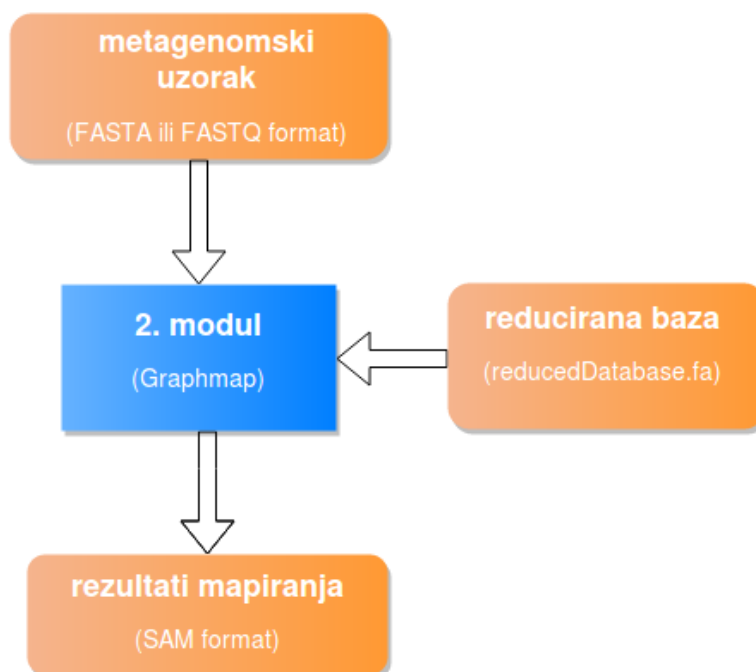


Slika 4.2: Prvi korak – reduciranje baze

4.2. Drugi korak

Drugi korak je mapiranje metagenomskih uzoraka s reduciranom bazom podataka. Za ovaj korak korišten je program Graphmap[11], no dozvoljeno je koristiti i neki drugi program iste namjene koji daje rezultat u SAM formatu. Ulazni podatci pohranjeni su u FASTA ili FASTQ formatu te se s pomoću Graphmapa poravnavaju s nukleotidnim sljedovima markera u reduciranoj bazi. Rezultat ovog koraka sprema se u datoteku u

SAM formatu koja se koristi u sljedećem koraku.



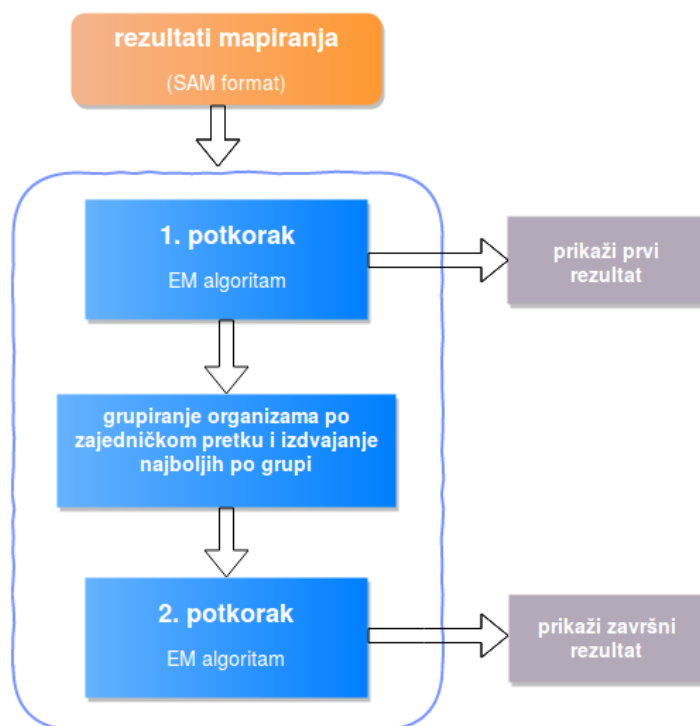
Slika 4.3: Drugi korak – mapiranje

4.3. Treći korak

Treći korak je klasifikator. To je zadnji i ključan dio rješenja zato što je njegova zadaća određivanje zastupljenosti pojedinih patogena u uzorku na temelju mapiranja iz prethodnog koraka. Za klasifikaciju je korišten EM algoritam koji je smješten u modulu *EMAlgorithm.py*. Ulaz klasifikatora je datoteka u SAM formatu koja sadrži potrebne podatke za daljnji algoritam. Iz njih se računaju i postavljaju inicijalni parametri EM algoritma.

Algoritam ima dva potkoraka. U prvom su potkoraku računa se zastupljenost svih organizama te se ispiše rezultat. Zbog velikog broja organizama, svakome će pripasti dio vjerojatnosti te će zbog toga i najveće vjerojatnosti biti relativno male. Da bi se izbjegla ta pojava, grupiramo organizme po zajedničkom pretku. Potom iz svake grupe biramo organizam koji je u prvom potkoraku imao najveću vjerojatnost kao predstavnika grupe i u drugom potkoraku računamo rezultat za samo te odabrane organizme. Pri grupiranju organizama koristi se taksonomsko stablo iz modula *TaxonomyTree.py*. Na kraju se prikaže rezultat drugog potkoraka algoritma. Pri grupiranju organizama po

zajedničkom pretku i za prikaz rezultata korištene su datoteke nodes.dmp i names.dmp.



Slika 4.4: Treći korak – klasifikator

5. Rezultati

Rješenje je testirano na sintetiziranom skupu bakterija kreiranim s pomoću PBSIM (PacBio simulatora sljedova). Parametri s kojima je alat pokrenut prikazani su u tablici 5.1[13].

Tablica 5.1: PBSIM parametri

Opcija	Vrijednost
data-type	CLR
depth	{ 1, 5, 10 }
length-mean	9753
length-sd	4260
length-min	5
length-max	100000
accuracy-mean	0.9
accuracy-sd	0.05
accuracy-min	0.7
difference-ratio	50:30:20

Za testiranje su se koristile bakterije iz tablice 5.2.

Tablica 5.2: Bakterije korištene za testiranje

Ti	TI vrste	Ime organizma
1263871	573	Klebsiella pneumoniae
1132507	28901	Salmonella enterica
282458	1280	Staphylococcus aureus

Svaka od navedenih bakterija testirana je s različitim pokrivenostima (engl. *coverage*): x1, x5 i x10. Dakle, za svaku bakteriju napravljena su tri testiranja. Za svako testiranje prikazane su dvije tablice. Prva tablica prikazuje najboljih pet rezultata dobivenih nakon prvog potkoraka (prvog izvođenja EM algoritma), dok druga tablica

prikazuje najboljih pet rezultata dobivenih nakon drugog potkoraka (nakon selekcije najboljih organizama po grupi i drugog izvođenja EM algoritma). Za svaki redak tablice prikazan je taksonomski broj organizma, ime organizma te njegova vjerojatnost u uzorku.

5.1. *Klebsiella pneumoniae*

Test je napravljen nad sintetičkim skupom kreiranim iz soja *Klebsiella pneumoniae* ATCC BAA-2146 s ciljem da se nakon drugog koraka kao rezultat dobije *Klebsiella pneumoniae*, dakle organizam na razini vrste.

Tablica 5.3: *Klebsiella pneumoniae* za pokrivenost x1, rezultati nakon prvog potkoraka

TI	Ime	Vjerojatnost
573	<i>Klebsiella pneumoniae</i>	0.002 35
749535	<i>Klebsiella</i> sp. MS 92-3	0.002 28
665944	<i>Klebsiella</i> sp. 4_1_44FAA	0.002 28
1182695	<i>Klebsiella</i> sp. KTE92	0.002 20
469608	<i>Klebsiella</i> sp. 1_1_55	0.002 16

Tablica 5.4: *Klebsiella pneumoniae* za pokrivenost x1, rezultati nakon drugog potkoraka

TI	Ime	Vjerojatnost
573	<i>Klebsiella pneumoniae</i>	0.432
720791	<i>Klebsiella</i> sp. enrichment culture clone SRC_DSB12	0.062
54291	<i>Raoultella ornithinolytica</i>	0.046
550	<i>Enterobacter cloacae</i>	0.043
548	<i>Enterobacter aerogenes</i>	0.040

Tablica 5.5: *Klebsiella pneumoniae* za pokrivenost x5, rezultati nakon prvog potkoraka

TI	Ime	Vjerojatnost
573	<i>Klebsiella pneumoniae</i>	0.002 41
749535	<i>Klebsiella</i> sp. MS 92-3_DSB12	0.002 35
665944	<i>Klebsiella</i> sp. 4_1_44FAA	0.002 31
1182695	<i>Klebsiella</i> sp. KTE92	0.002 17
469608	<i>Klebsiella</i> sp. 1_1_55	0.002 11

Tablica 5.6: *Klebsiella pneumoniae* za pokrivenost x5, rezultati nakon drugog potkoraka

TI	Ime	Vjerojatnost
573	<i>Klebsiella pneumoniae</i>	0.437
720776	<i>Klebsiella</i> sp. enrichment culture clone SRC_DSA21	0.061
54291	<i>Raoultella ornithinolytica</i>	0.049
548	<i>Enterobacter aerogenes</i>	0.048
550	<i>Enterobacter cloacae</i>	0.038

Tablica 5.7: *Klebsiella pneumoniae* za pokrivenost x10, rezultati nakon prvog potkoraka

TI	Ime	Vjerojatnost
573	<i>Klebsiella pneumoniae</i>	0.0025
749535	<i>Klebsiella</i> sp. MS 92-3	0.0024
665944	<i>Klebsiella</i> sp. 4_1_44FAA	0.0024
1182695	<i>Klebsiella</i> sp. KTE92	0.0022
469608	<i>Klebsiella</i> sp. 1_1_55	0.0022

Tablica 5.8: *Klebsiella pneumoniae* za pokrivenost x10, rezultati nakon drugog potkoraka

TI	Ime	Vjerojatnost
573	<i>Klebsiella pneumoniae</i>	0.442
1490266	<i>Klebsiella</i> sp. enrichment culture clone F-2	0.059
548	<i>Enterobacter aerogenes</i>	0.049
54291	<i>Raoultella ornithinolytica</i>	0.046
550	<i>Enterobacter cloacae</i>	0.044

5.2. *Salmonella enterica*

Test je napravljen nad sintetičkim skupom kreiranim iz soja *Salmonella enterica* subsp. *enterica* serovar *Typhi* str. *P-stx-12* s ciljem da se nakon drugog koraka kao rezultat dobije *Salmonella enterica*, dakle organizam na razini vrste.

Tablica 5.9: Salmonella enterica za pokrivenost x1, rezultati nakon prvog potkoraka

TI	Ime	Vjerojatnost
28901	Salmonella enterica	0.0099
54736	Salmonella bongori	0.0046
946044	Salmonella sp. enrichment culture clone CL107	0.0038
931991	Salmonella sp. ES-B43	0.0038
925972	Salmonella sp. 85MP	0.0038

Tablica 5.10: Salmonella enterica za pokrivenost x1, rezultati nakon drugog potkoraka

TI	Ime	Vjerojatnost
28901	Salmonella enterica	0.589
1308813	Salmonella sp. enrichment culture clone TB43_4	0.209
545	Citrobacter koseri	0.022
413503	Cronobacter malonaticus	0.019
564	Escherichia fergusonii	0.013

Tablica 5.11: Salmonella enterica za pokrivenost x5, rezultati nakon prvog potkoraka

TI	Ime	Vjerojatnost
28901	Salmonella enterica	0.0096
54736	Salmonella bongori	0.0049
946044	Salmonella sp. enrichment culture clone CL107	0.0038
931991	Salmonella sp. ES-B43	0.0038
925972	Salmonella sp. 85MP	0.0038

Tablica 5.12: Salmonella enterica za pokrivenost x5, rezultati nakon drugog potkoraka

TI	Ime	Vjerojatnost
28901	Salmonella enterica	0.577
688656	Salmonella sp. enrichment culture clone NJ-8	0.204
545	Citrobacter koseri	0.026
623	Shigella flexneri	0.024
546	Citrobacter freundii	0.020

Tablica 5.13: *Salmonella enterica* za pokrivenost x10, rezultati nakon prvog potkoraka

TI	Ime	Vjerojatnost
28901	<i>Salmonella enterica</i>	0.0089
54736	<i>Salmonella bongori</i>	0.0048
946044	<i>Salmonella</i> sp. enrichment culture clone CL107	0.0037
931991	<i>Salmonella</i> sp. ES-B43	0.0037
925972	<i>Salmonella</i> sp. 85MP	0.0037

Tablica 5.14: *Salmonella enterica* za pokrivenost x10, rezultati nakon drugog potkoraka

TI	Ime	Vjerojatnost
28901	<i>Salmonella enterica</i>	0.559
946044	<i>Salmonella</i> sp. enrichment culture clone CL107	0.208
545	<i>Citrobacter koseri</i>	0.022
623	<i>Citrobacter freundii</i>	0.018
623	<i>Shigella flexneri</i>	0.018

5.3. *Staphylococcus aureus*

Test je napravljen nad sintetičkim skupom kreiranim iz soja *Staphylococcus aureus subsp. aureus* MRSA252 s ciljem da se nakon drugog koraka kao rezultat dobije *Staphylococcus aureus*, dakle organizam na razini vrste.

Tablica 5.15: *Staphylococcus aureus* za pokrivenost x1, rezultati nakon prvog potkoraka

TI	Ime	Vjerojatnost
1280	<i>Staphylococcus aureus</i>	0.278
2130	<i>Ureaplasma urealyticum</i>	0.029
64160	<i>Desulfurobacterium thermolithotrophum</i>	0.017
1401027	<i>Flavobacterium limnosediminis</i>	0.017
1491	<i>Clostridium botulinum</i>	0.015

Tablica 5.16: Staphylococcus aureus za pokrivenost x1, rezultati nakon drugog potkoraka

TI	Ime	Vjerojatnost
1280	Staphylococcus aureus	0.551
2130	Ureaplasma urealyticum	0.058
64160	Desulfurobacterium thermolithotrophum	0.034
1401027	Flavobacterium limnosediminis	0.034
1491	Clostridium botulinum	0.029

Tablica 5.17: Staphylococcus aureus za pokrivenost x5, rezultati nakon prvog potkoraka

TI	Ime	Vjerojatnost
1280	Staphylococcus aureus	0.0745
2130	Ureaplasma urealyticum	0.0046
1491	Clostridium botulinum	0.0031
42422	Halobacteroides halobius	0.0028
169679	Clostridium saccharobutylicum	0.0028

Tablica 5.18: Staphylococcus aureus za pokrivenost x5, rezultati nakon drugog potkoraka

TI	Ime	Vjerojatnost
1280	Staphylococcus aureus	0.551
2130	Ureaplasma urealyticum	0.035
1491	Clostridium botulinum	0.024
42422	Halobacteroides halobius	0.022
5911	Tetrahymena thermophila	0.014

Tablica 5.19: Staphylococcus aureus za pokrivenost x10, rezultati nakon prvog potkoraka

TI	Ime	Vjerojatnost
1280	Staphylococcus aureus	0.0775
2130	Ureaplasma urealyticum	0.0044
5911	Tetrahymena thermophila	0.0027
1491	Clostridium botulinum	0.0023
42422	Halobacteroides halobius	0.0023

Tablica 5.20: *Staphylococcus aureus* za pokrivenost x10, rezultati nakon drugog potkoraka

TI	Ime	Vjerojatnost
1280	<i>Staphylococcus aureus</i>	0.505
2130	<i>Ureaplasma urealyticum</i>	0.029
5911	<i>Tetrahymena thermophila</i>	0.017
1491	<i>Clostridium botulinum</i>	0.015
42422	<i>Halobacteroides halobius</i>	0.015

5.4. Mješavina

Test mješavine napravljen je nad sintetičkim skupom kreiranim iz kombinacije spomenute tri bakterije: *Salmonella enterica* (pokrivenost x15), *Staphylococcus aureus* (pokrivenost x10) i *Klebsiella pneumoniae* (pokrivenost x5). Cilj je bio da se zadane bakterije pojave u rezultatu kao najzastupljenija tri organizma (redoslijedom od one s najvećom pokrivenosti prema onom s manjom pokrivenosti). Na tablici 5.21 prikazani su rezultati nakon drugog potkoraka algoritma.

Tablica 5.21: Mješavina bakterija: *Salmonella enterica* (x15), *Staphylococcus aureus* (x10) i *Klebsiella pneumoniae* (x5), rezultati nakon drugog potkoraka

TI	Ime	Vjerojatnost
28901	<i>Salmonella enterica</i>	0.309
573	<i>Klebsiella pneumoniae</i>	0.117
946044	<i>Salmonella</i> sp. enrichment culture clone CL107	0.113
1280	<i>Staphylococcus aureus</i>	0.107
763879	<i>Klebsiella</i> sp. enrichment culture clone SRC_DSD25	0.016

5.5. Osvrt na rezultate

Rezultati prvog koraka za pojedine bakterije daju točno rješenje, tj. na prvom mjestu doista se nalazi ispravan organizam. No, budući da su vjerojatnosti raspodijeljene nad svim mogućim organizmima, tj. mnogo organizama ima jako malu vjerojatnost pojavljivanja u uzorku, ni vjerojatnosti prvih pet najzastupljenijih organizama nisu velike.

Zbog toga u sljedećem koraku ostaju samo najbolji organizmi po grupi (oni s najvećom vjerojatnosti) te se još jednom ponovi algoritam. Rezultati drugog, odnosno

zadnjeg koraka daju također točno rješenje, no s puno većom vjerojatnosti za prvi organizam. To znači da se s većom pouzdanošću može reći da je taj organizam doista prisutan u uzorku, posebno zato što je njegova vjerojatnost i značajno veća od vjerojatnosti ostalih organizama.

Rezultati za mješavinu prikazuju točne organizme, no redoslijed nije sasvim dobar. *Staphylococcus aureus* pojavio se na četvrtom mjestu (umjesto na drugom), dok se *Klebsiella pneumoniae* pojavila na drugome mjestu (umjesto na trećem). Na trećem mjestu pojavila se još jedna vrsta *Salmonelle* što se može opravdati sličnosti te bakterije zadanoj bakteriji *Salmonella enterica*. Unatoč manjim odstupanjima, rezultat je i dalje dovoljno dobar jer doista prikazuje najzastupljenije bakterije u uzorku.

Dobiveni rezultati za pojedine bakterije dokazuju da je program točan za i da može s dovoljno velikom pouzdanošću utvrditi o kojem je organizmu riječ. U slučaju mješavine organizama može doći do manjih odstupanja u redoslijedu organizama, no očekivani organizmi će se pojaviti među najzastupljenijim organizmima.

6. Diskusija

Kao što se vidi iz rezultata, program daje točne rezultate. Na svim primjerima gdje se testirao jedan patogen točno je odredio o kojem je patogenu riječ (prvi se rezultat uz to znatno isticao sa svojim udjelom u uzorku u odnosu na ostale), dok je kod primjera s mješavinom pogodio najzastupljenije patogene, no s manjim odstupanjem u odnosu na njihove zastupljenosti u uzorku. Korištenje EM algoritma se pokazalo odličnim izborom za rješavanje zadanog problema. Program je ispunio i zadaću da izračuna rezultat u realnom vremenu i da pokriva različite vrste bakterija, a ne samo neki specifični skup. Reduciranje baze se pokazalo kao zaista zgodno rješenje za problem obuhvaćanja raznolikih organizama uz uštedu vremena zbog mapiranja na gene markere.

No, uvijek ima mjesta za poboljšanja. Za početak, treba poboljšati pouzdanost algoritma. Rezultat algoritma, odnosno organizam s najvećom vjerojatnosti trebao bi imati što veću vjerojatnost. Što je veća vjerojatnost, to je algoritam pouzdaniji (uz pretpostavku da radi točno). Trebalo bi usavršiti EM algoritam da već u prvom potkoraku daje rezultate s većim vjerojatnostima, odnosno kad bi se to ostvarilo, narasle bi vjerojatnosti i za drugi korak.

Osim toga, algoritam bi mogao raditi brže. Brzina je vrlo bitan kriterij i trebalo bi raditi na tome da je algoritam uz svoju točnost i dovoljno brz. Vremenski najzahtjevniji dio rješenja je drugi korak - mapiranje. Taj bi se korak trebao optimizirati. Veličina reducirane baze, kao što je već spomenuto, negativno utječe na trajanje mapiranja. Uz to, što je više organizama mapirano na uzorak, to će nam i vjerojatnosti u prvom potkoraku klasifikatora biti manje ili će se u krajnjem rezultatu pojaviti srodna bakterija s visokom vjerojatnosti kao što se vidjelo na primjeru mješavine. No, s druge strane, potrebno je podržati što veći broj organizama jer bez toga se smanjuje mogućnost prepoznavanja određenog patogena u uzorku. Treba pronaći balans između te dvije strane i generirati reduciranu bazu podataka u skladu s tim. Još jedna od mogućih opcija je korištenje nekog drugog skupa podataka umjesto onog od *Metaphlana*.

Mogao bi se poboljšati i treći korak, odnosno klasifikator. U trećem se koraku

gradi taksonomsko stablo i učitavaju se odgovarajuće strukture podataka koje se koriste za grupiranje organizama i traženje imena za svaki taksonomski broj. Kada bi se optimiziralo i paraleliziralo dohvaćanje tih podataka, uštedio bi se dio vremena.

Također, bilo bi idealno kada bi se drugi i treći korak rješenja mogli donekle paralelizirati, odnosno da se nakon svake određene količine mapiranja pokreće klasifikator i računa rezultate na temelju tih podataka. Time bi se rješenje prikazivalo dinamički i moglo bi se ranije vidjeti u kome se smjeru kreće.

7. Zaključak

Rezultati ovog završnog rada dokazali su da uz odabir pogodnih metoda i modela za rješavanje problema, zaista je moguće ostvariti brzu i jeftinu analizu metagenomskog uzorka. Za razliku od tradicionalnih laboratorijskih metoda koje rade dobro za samo jedan primjer ili su jako spore, metoda obrađena u ovom završnom radu radi u stvarnom vremenu i radi za raznolike organizme, a ne samo za ograničen skup organizama. I najvažnije – daje točne rezultate. Cilj završnog rada je dakle ostvaren. Alat radi vrlo precizno kada je u pitanju identifikacija jednog patogena, a u primjeru mješavine nekoliko patogena radi dovoljno precizno. U svakom slučaju, poboljšanja su uvijek dobrodošla i definitivno treba raditi na usavršavanju alata da bude još precizniji i brži. Motivacija za razvojem dolazi iz činjenice da je velika potražnja za ovakvim alatima za raznolike primjene, kao što su primjerice za dijagnostiku u medicini, za identifikaciju nametnika u poljoprivredi, kontrola kvalitete hrane i sl.

LITERATURA

- [1] Assembly summary files. ftp://ftp.ncbi.nlm.nih.gov/genomes/README_assembly_summary.txt. Datum nastanka: 12.04.2017. Datum pristupa: 30.05.2017.
- [2] Wikipedia – The Free Encyclopedia: Expectation–maximization algorithm. https://en.wikipedia.org/wiki/Expectation%E2%80%9393maximization_algorithm. Datum pristupa: 30.05.2017.
- [3] Wikipedija – Slobodna enciklopedija: Kladus. <https://bs.wikipedia.org/wiki/Kladus>. Datum pristupa: 30.05.2017.
- [4] Wikipedia – The Free Encyclopedia: Likelihood function. https://en.wikipedia.org/wiki/Likelihood_function. Datum pristupa: 30.05.2017.
- [5] Wikipedia – The Free Encyclopedia: Mixture model. https://en.wikipedia.org/wiki/Mixture_model. Datum pristupa: 30.05.2017.
- [6] Wikipedia – The Free Encyclopedia: Maximum likelihood estimation. https://en.wikipedia.org/wiki/Maximum_likelihood_estimation. Datum pristupa: 30.05.2017.
- [7] Extract Data from NCBI Taxonomy Files. <http://www.chnosz.net/manual/taxonomy.html>. Datum pristupa: 30.05.2017.
- [8] Do, C.B. i Batzoglou, S. What is the expectation maximization algorithm? *Nature Biotechnology*, 26(8):897–899, 2008.
- [9] Domazet-Lošo, M. i Šikić, M. Bioinformatika, 2013.
- [10] Francis, O.E., Bendall, M., Manimaran, S., Hong, C., Clement, N.L., Castro-Nallar, E., Snell, Q., Schaalje, G.B., Clement, M.J., Crandall, K.A., i journal =

Genome Research year = 2013 volume = "23" number = "10" pages = "1721—1729" Johnson, W.E., title = Pathoscope: Species identification and strain attribution with unassembled sequencing data.

- [11] Sović, I. Graphmap – A highly sensitive and accurate mapper for long, error-prone reads. <https://github.com/isovic/graphmap>. Datum nastanka: 26.05.2017. Datum pristupa: 30.05.2017.
- [12] Truong, D.T. i Segata, N. Metaphlan2 – Metagenomic Phylogenetic Analysis. <https://bitbucket.org/biobakery/metaphlan2>. Datum nastanka: 01.06.2017. Datum pristupa: 03.05.2017.
- [13] Vujević, I. *Real-Time Analysis of a Metagenomic Sample Obtained by Nanopore Based Sequencing Technology*. Diplomski rad, Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, 2016.

Analiza metagenomskog uzorka dobivenog sekvenciranjem koristeći uređaje treće generacije

Sažetak

Brza i jeftina analiza metagenomskog uzorka korisna je za kontrolu kvalitete hrane, dijagnozu bolesti i utvrđivanje štetnih nametnika na biljkama. Tradicionalne laboratorijske metode su ili dugotrajne ili namijenjene za samo jednu vrstu. Korištenjem uređaja treće generacije dobivaju se očitavanja koja su puno dulja, no sadrže i znatno veći postotak pogrešaka. Implementiran je alat koji utvrđuje koji su organizmi prisutni u metagenomskim uzorcima dobivenima koristeći upravo uređaje treće generacije. Alat u prvom koraku pronalazi sve organizme čiji genetski materijal nalikuje očitavanjima iz uzorka, a zatim u drugom koraku analizira rezultate iz prvog koraka i utvrđuje koji su organizmi prisutni u zadanom uzorku.

Ključne riječi: metagenomika, patogeni, dijagnostika

Analysis of a Metagenomic Sample Obtained by Third Generation Sequencing Technology

Abstract

A fast and inexpensive analysis of metagenomic samples is useful for food quality control, medical diagnosis and identifying harmful plant parasites. Traditional laboratory methods are either time-consuming or designed for a specific species. Third-generation devices produce reads that are much longer, but also have a significantly higher error rate. A tool is implemented to determine which organisms are present in metagenomic samples obtained via third-generation devices. In its first step, the tool finds all organisms whose genetic material resembles that of the sample's reads. Afterwards, in the second step, the tool analyzes the first step's results and determines which organisms are present in the given sample.

Keywords: metagenomics, pathogens, diagnostics