**RSSReader**

*+readRSS(String text)* – preuzima za konkretnog rss feed-a niz adresa(linkova) za clanke

**Web Scrapper**

**//**Koriscen je Selenium Web Driver za Eclipse

*+getText()* – metoda koja uzima text sa web stranice sadrzinu html elementa body, cuva svaki pojedinacno u direktorijumu fajlovi

*+napuniIVratiListu()* – metoda koja pravi listu RSS feedova

*+MAIN METODA* – generisanje getText() metode za svaki clanak

-SADA SMO DAKLE POKUPILI TEXTOVE SA LINKOV A- smatrao sam da je prihvatljivije ovakvo resenje nego da rucno da idem copy paste

**StartAnnotators**

*+openWebPage* – otvara URL (za sada nije iskoriscena metoda)

*+pokreniTagNadFajlom(String input, String output)* – Input ce biti string iz neke putanje, dok ce output biti naziv fajla koji ce dobiti ekstenziju .json – primena tagMe anotatora(fajloviJSON folder)

*+napraviTextFajloveKonacne(String putanjaDoFajla)* – pravi od json fajlova, text fajlove koji ce se koristiti pri primeni algoritma (fajloviText folder) –ISTOVREMENO PUNI RECNIK

*+pokreniTagNadSvima* – prethodno radi nad celim fajlovi direktorijumom

**Kontroler**

Sadrzi HashMapu key-value : putanja-tekst <String,String>

U konstruktoru metoda : *vratiFajlove(files)* – tekstualni fajlovi spremni za primenu algoritma

*+vratiStringFajla,PrintHashMap , izvuciNazivIzPutanje* – rade ono sto sam naziv metode kaze

***+iseciTextVratiPutanju(String putanja, int krajnjiIndeks)*** – Posto ce se desiti da textovi budu razlicite duzine, poredimo dva texta, I onda za parametar krajnjiIndeks stavljamo manji length od dva texta. Sa kraja isecemo visak da bi tekstovi bili iste duzine. Npr. Prvi text ima 102 reci, drugi 110, krajnjiIndeks ce biti 102, I od tog indeksa brisu se sve reci. Pravi se privremeni fajl koji ce se uvek menjati posto ce pri algoritmu u 99 posto slucajeva biti razlicite duzine, on se vraca I koristi za izracunavanje.

*+napuniRecnik* – prolazi fajloviText direktorijum I puni jedan fajl svim recima koji ce biti recnik I sluzice za poredjenje I primenu prilikom izracunavanja slicnosti textova.

**KosinusnaSlicnost**

*+vrati* –vraca slicnost izmedju dva teksta, rezultat ce biti od 0 – 1

Zakomentirao sam deo u metodu na pocetku *String reci = "me Julie loves Linda than more likes Jane"; -* u main metodi postoji jednostavan primer koji bi koristio ovaj recnik kada bi se oprobao. Mozete videti kako radi samo da iskoristite ovaj recnik I skinete komentare isto sa metode vrati(prvi,drugi) u main delu

*+sortByValues* – za dati tekst prodje kroz ceo direktorijum fajloviText I sortira mapu po slicnosti -🡪takva mapa ce biti potrebna jer u metodi HillClimbing treba da generisemo susede, a susedi ce biti slicni tekstovi.

**Algoritmi**

*+vratiMinHillClimbing()* – U zadatku je receno da treba da se nadje minimum fje. Minimum koji predstavlja najmanje odstupanje jednog texta od drugog..Posto similarity vraca 1 ukoliko su tekstovi isti, 0 ako su potpuno razliciti, izmedju vrednost onoliko koliko su slicni, ja sam ovde vracao maximum? Mogu da stavim primera radi similarity da vraca obrnutu vrednost? Tipa da ide similarity = |similarity-1|….?

*+vratiRandom* – vraca random element iz mape.

**TestCloudFrame**

Forma koja izlazi korisniku, skinuo sam biblioteku opencloud za prikaz tagclouda

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

to je to otprilike to, trebalo bi da se nalaze sve biblioteke u folderu koji sam vam poslao, one su ili u lib folderu ili u dodatneBiblioteke.

Mozete pokrenuti main metodu Algoritmi ili KosinusnaSlicnost klase da vidite kako program generalno radi.

Ono sto predstavlja problem jeste da program malo sporije radi kada ima vise reci, trenutno fajlovi u direktorijumu fajloviText su rucno napravljeni kao i recnik naravno, I gleda se samo primena algoritma. Nisam za svrhu koristio metodu da iseciTextVratiPutanju, jer sam je nesto I u medjuvremenu menjao, gledao sam da ubacujem upit koji ce imati isti broj reci kao svi tekstovi (10tak)

Mozete pokrenuti main metodu StartAnnotators koji bi trebalo da generise gotove kljucne reci iz fajlova da vidite njihovu sadrzinu. //samo imajte u vidu da ce se reci dodavati na fajl Recnik.txt koji trenutno ima samo reci iz 8 txt rucno generisanih fajlova.

Trebalo bi da napravim jos textove za upit I da konacno generisem pravi Recnik I textove. Bazu nisam koristio kao sto mozete, ne znam ima li potrebe radim preko kontrolera I txt fajlova?

Samo ce biti problem kod Vas da promenite svuda string koji sadrzi putanju do foldera, posto pise za mene user – Radomir, ako vam je workspace u documents folderu(mac racunar) samo promenite taj deo user, sa bibliotekama ne bi trebalo da bude problema.