Uočavanje Long Method smell-a

Zdravko Dugonjić
zdravko.dugi@gmail.com | Soft Kompjuting 2021

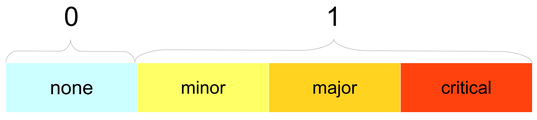
Long Method smell predstavlja jednu od loših praksi koja dovodi do degradacije kvaliteta programskog koda. Projekat demonstrira rešenje koje automatski pronalazi ovaj problem u kodu oslanjajući se na znanje sa StackOverflow platforme i tehnika mašinskog učenja.

Problem

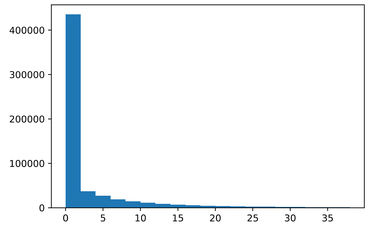
Cilj projekta je da se za datu metodu odredi da li je prisutan *Long Method smell*. Ulaz u sistem predstavlja telo metode, a izlaz da li je *smell* uočen.

Skupovi podataka

MLCQ skup sadrži Java metode sa anotacijama koje indikuju prisustvo *smell-a*. Anotacije su binarizovane.



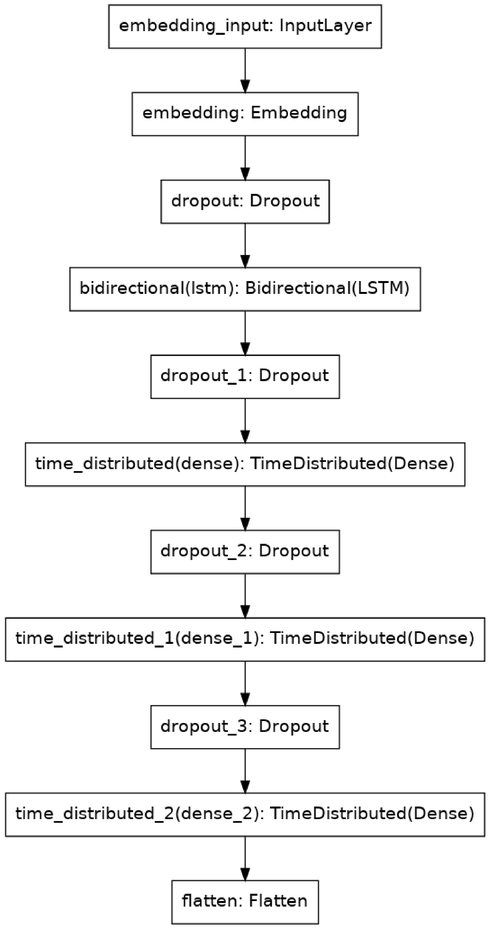
Drugi skup je generisan izdvajanjem *snippet-a* iz StackOverflow *post-ova*. Odbačeni su *snippet-i* koji sadrže manje od 4 linije koda.

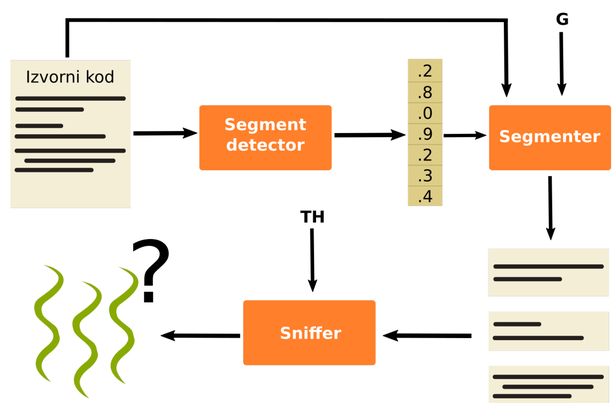


S*nippet-i* su konkatenirani, a znakovi za novi red na mestu spajanja *snippet-a* označeni kao početak novog logičkog segmenta (1) dok su ostali karakteri označeni negativno (0).

Metodologija

Segment detector kao ulaz prima izvorni kod tela metode i za svaki karakter vraća verovatnoću da li predstavlja početak novog logičkog bloka.
Iz tela metode se uklanjaju znakovi koji predstavljaju uvlačenje, a višestruki znakovi za novi red se menjaju jednim. Pretprocesuirani kod se prosleđuje neuronskoj mreži u vidu okvira dužine 100 karaktera.





Segmenter prima pretprocesuirano telo metode i vektor verovatnoća. Na osnovu tela metode i parametra za granularizaciju (**G**) generiše se izlaz koji predstavlja vektor logičkih blokova.

U poslednjoj fazi, *Sniffer* posmatra vektor logičkih blokova i u slučaju da postoje dva bloka sa sličnošću ispod zadatog praga (**TH**), konstatuje prisustvo *Long Method smell-a.* Logički blok reprezentuje niz identifikatora, izuzimajući ključne reči programskog jezika Java.

```
int user = new Admin();
```

↓

user, admin

Sličnost blokova se ustanovljava korišćenjem *Latent Semantic Indexing*.

Eksperiment

Evaluacija je izvršena nad MLCQ skupom. Izabrana kombinacija parametara je 0.4 za granularnosti i 0.4 za prag sličnosti.

Segment detector je zasebno obučavan nad StackOverflow generisanim skupom. Mreža je postigla F1-skor od 93% i tačnost od 99.97%.

Rezultati i diskusija

Predloženi metod je uspeo da postigne rezultat **F1-skor** od **59%** na MLCQ testnom skupu.

	precision	recall	f1
0	0.96	0.93	0.94
1	0.54	0.65	0.59

Kako *Segmenter* prima vektor verovatnoća i na osnovu praga granularnosti vrši segmentaciju koda, uočeno je da bi ovo moglo da predstavlja usko grlo arhitekture rešenja. Naime, gube se ostale verovatnoće i njihova distribucija u daljem procesuiranju.

Kako bi se potkrepila hipoteza o uskom grlu, izvršena je zamena *Segmenter-a* i *Sniffer-a* korakom koji implementira *Random Forest*. Dobijen je **F1-skor** od **70%** za pozitivnu klasu, što otvara put ka daljem unapređenju predloženog rešenja.