Zadaća 2: FST i N-gramski modeli jezika

Zadaća se može raditi u grupi do 3 člana. Svi sudionici grupe moraju predati svoj uradak do naznačenog datuma. Datum predaje zadaće bit će objavljen na platformi na kojoj se dostavljaju zadaće.

Problem 1: Generator imena broja

Broj bodova: 10

U ovom zadatku potrebno je izgraditi FST koji za dani niz brojeva od 0-1000 generira zapis riječima. Dakle, za broj '123' vaš model treba producirati 'sto dvadeset i tri'.

Ovaj zadatak trebate riješiti koristeći Python biblioteku **openfst_python** [1] kao omotač za C++ biblioteku **openfst** [2]. Detaljnu dokumentaciju o tom alatu možete pogledati [3] ili proći tutorijal [4].

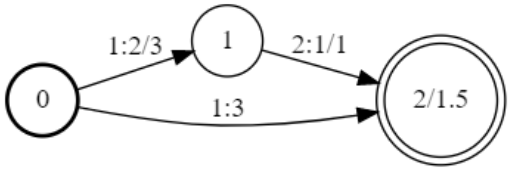
Vaš je zadatak sljedeći:

1. Predati FST unutar Python skripte ili ASCII datoteku koje se može proslijediti FST prevoditelju
2. Pripremiti ulazne i izlazne simbole za pretvaranje, vidi primjer ASCII simbola
3. Radni primjer

Na Jupyterhub platformi postavljene su sve potrebne biblioteke. Primjer jednog takvog jednostavnog modela (koji uključuje i težine):

```
import openfst_python as fst

f = fst.Fst()
s0 = f.add_state()
s1 = f.add_state()
s2 = f.add_state()
f.add_arc(s0, fst.Arc(1, 2, fst.Weight(f.weight_type(), 3.0), s1))
f.add_arc(s0, fst.Arc(1, 3, fst.Weight.One(f.weight_type()), s2))
f.add_arc(s1, fst.Arc(2, 1, fst.Weight(f.weight_type(), 1.0), s2))
f.set_start(s0)
f.set_final(s2, fst.Weight(f.weight_type(), 1.5))
```



Problem 2

Broj bodova: 10

U ovom zadatku potrebno je izgraditi n-gramski model za tekstove HR jezika koji su dani u referenci [4]. Tekstovi su dani u `epub` format što znači da ćete ih trebati pretvoriti u obradivi oblik (`txt` format).

Učinite sljedeće:

1. Izgradite skup za treniranje na kojem ćete naučiti model (80% rečenica korpusa). Evaluirajte model uz pomoć mjere perpleksnosti na skupu za testiranje (20%). Na tekstu primijenite neku od normalizacijskih tehnika kako bi dobili što bolju perpleksnost.
2. Pored MLE procjenitelja, koristite barem metodu zaglađivanja i interpolacije. Odaberite onu metodu koja daje najbolju perpleksnost. Za potrebe učenja hiperparametara interpolacije, možete uvesti novi skup za razvoj pa onda podjela na skupove učenja jest u omjeru 80:10:10
3. Najbolji N-gram model spremite u pickle objekt za ponovno korištenje. Na temelju tog modela generirajte nekoliko primjera rečenica.

Vašu zadaću dostavite kao Jupyter bilježnicu zajedno s potrebnim resursima.

Reference:

1. Openfst-python: https://pypi.org/project/openfst-python/ , osvježeno: 8.9.2020
2. OpenFST Python Extension: http://www.openfst.org/twiki/bin/view/FST/PythonExtension, osvježeno: 23.9.2020
3. OpenFst Library: http://www.openfst.org/twiki/bin/view/FST/WebHome, osvježeno: 27.8.2020
4. OpenFst Tutorial: http://www.openfst.org/twiki/bin/view/FST/FstHltTutorial, osvježeno: 6.2.2009
5. Bulaja zaklada: eLektire, donirano u sklopu Adris financiranja