Generiranje imena naselja pomocu LSTM mreže

Antonio Čogelja Morena Granić Fran Lubina Iva Jurković Jakov Juvančić Matej Logarušić

***Sažetak*—Cilj projekta je LSTM rekurzivna neuronska mreža na razini znakova koja generira realistična imena hrvatskih naselja. Fokus projekta je treniranje i razvijanje neuronske mreže za generiranje realističnih imena hrvatskih naselja. Korištenjem LSTM mreže, koja je prilagođena za analizu sekvencijskih podataka, cilj je razviti model sposoban za učenje jezičnih obrazaca i struktura iz postojećih imena naselja. Svrha mreže je generiranje novih imena temeljenih na tim naučenim obrascima, pri čemu se zadržavaju jezične i strukturne zakonitosti specifične za taj kontekst. Željena točnost modela $\eta = 0.4$.**

***Index Terms*—Naselje, LSTM, rekurzivne mreže, neuronske mreže-**

I. Uvod

Ishod projekta je LSTM rekurzivna neuronska mreža na razini znakova koja generira realistična imena hrvatskih naselja. Mreža radi sa vektorima koji predstavljaju slova hrvatske abecede proširene specijalnim znakovima $\Sigma = \{\text{hrv. abeceda}\} \cup \{\langle start \rangle, \text{" } \backslash 0\text{"}\}$.
Ulaz mreže je one-hot vektor $\mathbf{x}^{(t)}$ dimezije $|\Sigma| = 30 + 2$.

$$\mathbf{x}_i^{(t)} = \begin{cases} 1, & \text{ako } i = j \\ 0, & \text{inače} \end{cases} \tag{1}$$

Izlaz dobiven na kraju pojedinog vremenskog koraka t je vektor vjerojatnosti pojave pojednog znaka abecende.

$$\hat{\mathbf{y}}^{(t)} = \begin{bmatrix} p(c_0) \\ p(c_1|c_0) \\ \vdots \\ p(c_{|\Sigma|-1}|\bigcap_{i=0}^{|\Sigma|-2} c_i) \end{bmatrix} \quad \text{Gdje} \quad c \in \Sigma \tag{2}$$

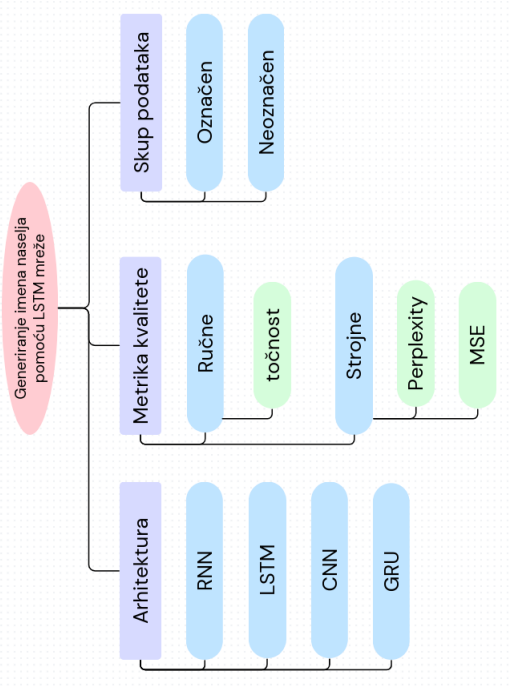
Vjerojatnosti su dobivene softmax funkcijom parametriziranom hiperparametrom temperature τ.
Na temelju tih vjerojatnosti se uzorkuje konačni izlazni vektor $\mathbf{y}^{(t)}$, odnosno t-ti znak u imenu naselja.

$$\mathbf{y}^{(t)} \sim \hat{\mathbf{y}}^{(t)} = \sigma_\tau(f(\mathbf{x}^{(t)}; \boldsymbol{\theta})) \tag{3}$$

$f(\mathbf{x}; \boldsymbol{\theta})$ predstavlja ukupno djelovanje ćelija modela nad njenim ulazom parametrizirano hiperparametrima modela $\boldsymbol{\theta} = \begin{bmatrix} |\mathbf{a}| & \mu & \tau \end{bmatrix}$
(opisani u poglavlju **??**) Temperaturno uzorkovanje je izabrano, jer omogućava eksperimentiranje i generiranje zanimljivih toponima. Izlaz mreže je niz znakova $\{\mathbf{y}^{(t)}\}\Big|_{t=0}^{T-1}$, odnosno ime naselja. Željena točnost modela η je $\lim_{\tau \to 0} \eta = 0.5$

II. Pregled literature

U ovom poglavlju dajemo kratki pregled postojeće literature na način kako je to učinjeno u [1]. Svrstavanjem radova na temelju kriterija: pristupa dubokom učenju, funkciji pogreške, primjeni, jeziku i skupu podataka.
Dobivamo taksonomiju na slici 1.
U konačnici odabiremo karakteristike našeg rješenja, navedene u tablici **??**.
Razlog za odabit pojedine karakteristike dajemo u dotičnom poglavlju.



Slika 1. taksonomija rješenja za generiranje teksta

Nas samo zanimaju radovi na jezicima spomenutim u cjelini I na razini riječi sa primjenom generiranja jezičnih konstrukta.
U konačnici odabiremo karakteristike našeg rješenja, navedene u tablici II.

Nakon pretraživanja ukupno smo naći 9 relevantnih radova sa tražilica:

1) Google (5 radova)
2) IEEE Xplore (2 rada)
3) pretraživanje literature (2 rada)

Tablica I
DETALJI PRETRAŽIVANJA

		Komentar
tražilice	Google, Google Scholar, BASE, CORE, Science.gov, Semantic Scholar, Baidu scholar, RefSeek, CiteSeerX, ScienceOpen, The Lens, arXiv, AMiner, ACM, IEEE Xplore, Science Direct, Springer Link, Web of Science	Samo smo na tražilici Google našli relevatne radove (njih 5)
traženi pojam	(LSTM OR GRU OR CNN OR RNN OR Neural Network) AND ((city AND generator AND name) OR (generator AND name))	Napisan sintaksom i operatorima koje koristi Google, ali na ostalim tražilicama je korišten prilagođeni izraz.

III. Opis implementirane LSTM mreže

Fokus projekta je treniranje i razvijanje neuronske mreže za generiranje realističnih imena hrvatskih naselja. Korištenjem LSTM mreže, koja je prilagođena za analizu sekvencijskih podataka, cilj je razviti model sposoban za učenje jezičnih obrazaca i struktura iz postojećih imena naselja. Svrha mreže je generiranje novih imena temeljenih na tim naučenim obrascima, pri čemu se zadržavaju jezične i strukturne zakonitosti specifične za taj kontekst.

Nadalje uz ćeliju je dodatno razvijeno GUI sučelje koje ima mogućnost generiranja 25 različitih imena gradova na 6 dostupnih jezike.
Težine jednom naučene mreže se pohranjuju u trajnu memoriju, odnosno u datoteke u za to predviđenom direktoriju: `./saved_models`.

A. Arhitektura

Teoretski modeli neuronskih mreža u kerasu mogu raditi sa ulaznim vektorima varijabilne dimenzionalnosti. U praksi, rad sa vektorima fiksne duljine poboljšava performanse, osobito vrijeme treniranja. Razlog tomu je to što ulazni vektori fiksne težine omogućavaju stvaranje tenzora fiksnih oblika, a posljedično i stabilne težine.

B. Ćelija

Ćelija izgledda ovako.

Nadalje slika V prikazuje unutarnju shemu ćelije.

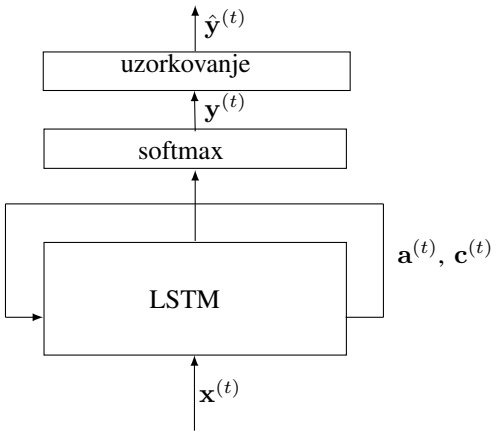
C. Treniranje

BPTT je korišten kao algoritam učenja. Kao funkcija gubitka koristi se kategorička unakrsna entropija.

$$L = -\sum_{t=0}^{|\Sigma|-1} \mathbf{z}_i^{(t)} \cdot log(\hat{\mathbf{y}}_i^{(t)}) \tag{4}$$

Tablica II
HIPERPARAMETRI NAŠE MREŽE

Hiperparametar	***Vrijednost***	***komentar***
broj jedinica po sloju	128	Broj jedinica po sloju određuje, i između ostalog, dimenziju skrivenog stanja ($\|\mathbf{a}\|$).
stopa učenja (μ)		
aktivacijska funckija	tanh	zadano
povratna akt. funkcija	σ_τ	zadano
bias	da	zadano
inicijalizator kernela	glorot jednoliki	zadano
inicijalizator povratne veze	glorot jednoliki	zadano
bias inicijalizator	zeros	zadano
forget bias	da	zadano
regularizacija kernela		zadano
regularizacija kernela povratne veze		zadano
bias regularizacija		zadano
kernel ograničenje		zadano
povratno ograničenje		zadano
bias ograničenje		zadano
dropout	0	zadano
povratni dropout	0	zadano



Slika 2. Arhitektura LSTM ćelije

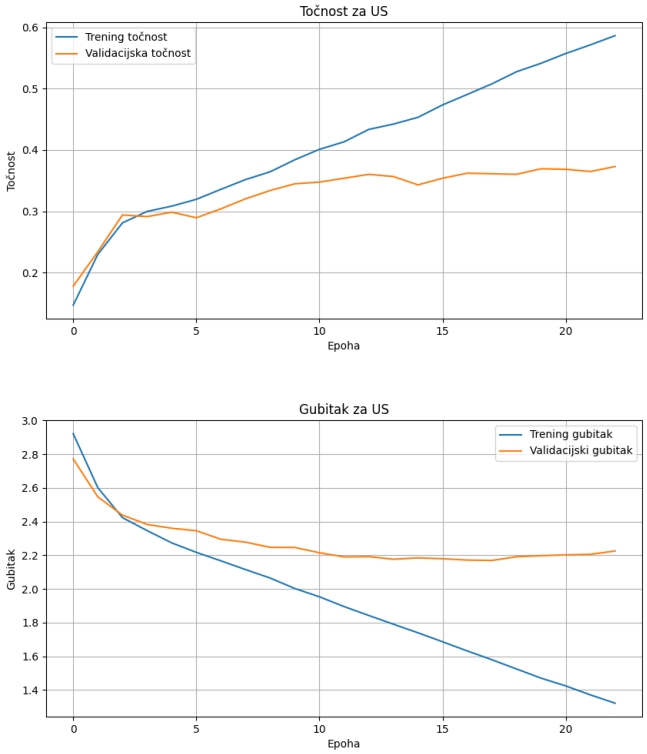
Gdje je $i \in [0, |\Sigma| - 1]$. $\hat{\mathbf{y}}^{(t)}$ je izlaz mreže, odnosno $\hat{\mathbf{y}}_i^{(t)}$ vjerojatnost da je idući znak i-ti znak abecede, a $\mathbf{z}^{(t)}$ je očekivani vektor. Inačica BPTT koju mi koristimo je u biti propagirani stohastički gradijentni spust.

Pri treniranju koristimo parametre navedene u tablici III.

Pri treniranju koristimo podjelu skupova na skup za treniranje i testiranje (validaciju) u omjeru 4:1.
Uz hiperparametre $\begin{bmatrix} |\mathbf{a}| & \mu \end{bmatrix} = \begin{bmatrix} 128 & 10^{-3} \end{bmatrix}$ dobivamo slijedeće vrijednosti funkcije gubitka i točnosti po epohama.

Tablica III
PARAMETRI PROCEDURE ZA TRENIRANJE

Parametar	***Vrijednost***
f-ja gubitka	kategorička unakrsna entropija
algoritam optimizacije	ADAM
metrika kvalitete	točnost



Slika 3. Američki gradovi

Grafovi za ostale jezike izgledaju gotovo identično, greška na skupu za testiranje počinje rasti nakon 20-30 epohe. Najveća točnost je uvijek $40\% \pm 5\%$.

IV. Opis eksperimentalnih rezultata

A. optimiranje hiperparametara

Pri optimiranju hiperparametra iz razmatranja je izuzet $|\mathbf{a}|$, jer se točnost monotono povećava sa rastućom dimenzijom skrivenog stanja, što pak nema nikakve veze sa stopom učenja. Prema tome za funkciju greške uvijek vrijedi $L(|\mathbf{a}|, \mu) = L(\mu)$.
Nadalje, zbog gornjega vrijedi: $\min_{|\mathbf{a}|,\mu} L = \min_{\mu}\{\min_{|\mathbf{a}|} L\} = \min_{\mu} L(\mu, \max|\mathbf{a}|)$.
Pri optimiranju hiperparametra iz razmatranja je izuzet $|\mathbf{a}|$. Koristimo heuristiku kako bi odredili vrijednost tog hiperparamtera ([2]).

$$|\mathbf{a}| = \frac{|\mathcal{D}_{train}|}{\alpha \cdot (N_i + N_o)} \tag{5}$$

B. Usporedba rezultata

–

V. Zaključak

–

Literatura

[1] Fatima, N., Imran, A S., Kastrati, Z., Daudpota, S M., Soomro, A (2022) "A Systematic Literature Review on Text Generation Using Deep Neural Network Models" IEEE Access, 10: 53490-53503, https://doi.org/10.1109/ACCESS.2022.3174108

[2] Eckhardt K. (2018, November 29). Choosing the right Hyperparameters for a simple LSTM using Keras. Towards data science. https://towardsdatascience.com/choosing-the-right-hyperparameters-for-a-simple-lstm-using-keras-f8e9ed76f046

[3] Karpathy A. (2015, May 21). The Unreasonable Effectiveness of Recurrent Neural Networks. Andrej Karpathy blog. http://karpathy.github.io/2015/05/21/rnn-effectiveness

[4] Randolph Z. (2020.) "Recursive Neural Network for Generating Novel Brand Names for Therapeutic Medicines". report. Department of Computer Science. Stanford. http://cs230.stanford.edu/projects_spring_2020/reports/38912979.pdf

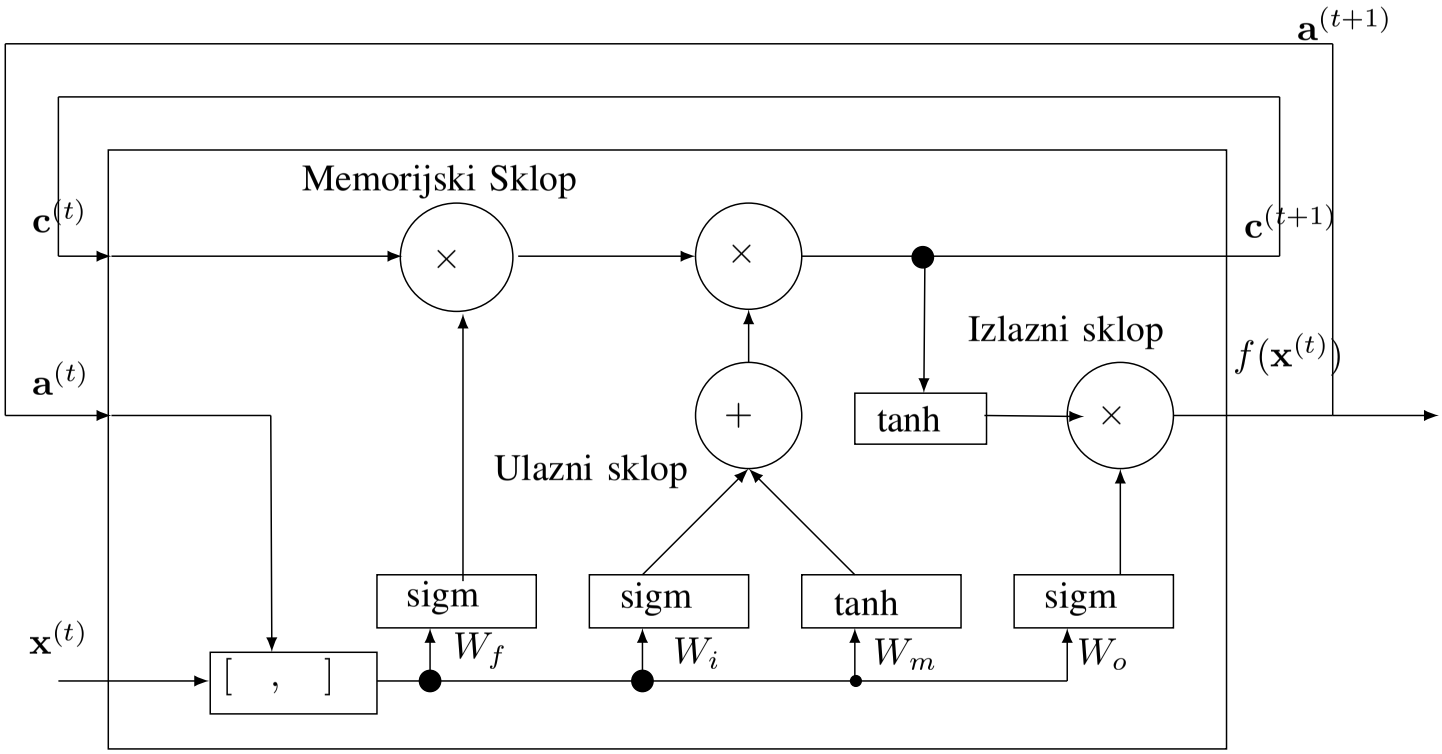
[5] Olah C. (2017, August 27). Understanding LSTM Networks. colah's blog. https://colah.github.io/posts/2015-08-Understanding-LSTMs/

[6] Dipanshu G. (2017, August 27). Master LSTM Networks With Python: A Guide Using TensorFlow and Keras. Medium. https://dipanshu10.medium.com/implementing-lstm-networks-with-python-a-guide-using-tensorflow-and-keras-915b58f502ce

[7] Rahalkar C. (2019, June 29). Name Generator Using Recurrent Neural Networks. Github. https://github.com/chaitanyarahalkar/Name-Generator-RNN

[8] Landy C. (2019, September 7). Look No More, The Data driven Baby Name generator. www.connorlandy.com. https://www.connorlandy.com/projects/rnn-name-generator

[9] Bosnali C. (2018, September 27). City-Name-Generation-using-LSTM-and-TF. Github. https://github.com/CihanBosnali/City-Name-Generation-using-LSTM-and-TF



Slika 4. Unutarnja arhitektura LSTM ćelije