Prevođenje teksta sa engleskog jezika na nemački jezik

Seminarski rad
Matematički fakultet,
Univerzitet u Beogradu

Dijana Zulfikarić, Stefan Pantić

January 20, 2019

Sadržaj

1 Opis problema

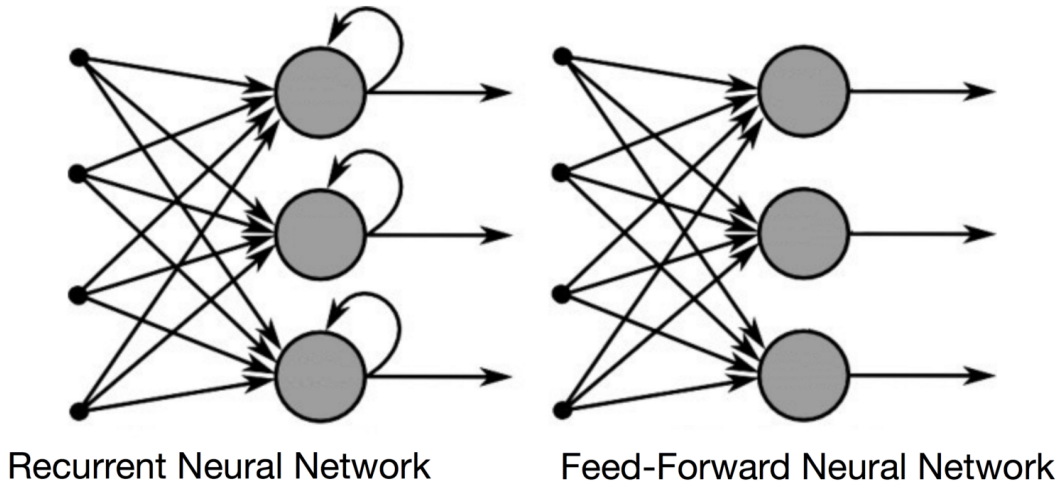
Problem kojim smo se bavili u ovom radu jeste generisanje prevoda engleskih rečenica na nemački jezik. Kao jedan od problema kojem je bilo pristupa na različite načine, postojalo je više algoritama kojima smo se mogli baviti u ovom slučaju. Za nas je najinteresantniji pristup bio pristup korišćen u radu Attention is all you need.

2 Zašto rešenja korišćenjem rekurentnih neuronskih mreža nisu dovoljno dobra?

Jedan od najkorišćenijih pristupa prilikom rešavanja ovakvih problema jeste korišćenje *rekurentnih neuronskih mreža*. Ovaj pristup, bez obzira na svoje mogućnosti, ima i određenih nedostataka koji će biti navedeni u narednom delu.

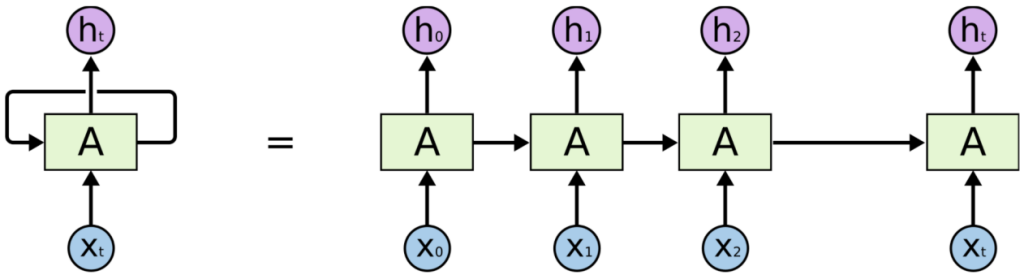
2.1 Kako funkcionišu rekurentne neuronske mreže?

Rekurentne neuronske mreže su trenutno industrijski standard za obradu sekvencijalnih podataka iz razloga što predstavljaju prvi algoritam koji pamti pređašnja stanja i uzima ih u obzir za generisanje novih izlaza. Bez obzira na to što sam algoritam nije toliko nov, svoj pun potencijal mogao je iskazati tek u nekoliko prethodnih godina zbog značajnog pojeftinjenja hardverskih komponenti. Ideja rekurentnih neuronskih mreža jeste da pamte značajne informacije koje su dobile, što omogućava povećanje preciznosti prilikom predikcija narednog izlaza. Upravo zbog toga su preferirani algoritam koji se koristi za obradu sekvencijalnih podataka kao što su vremenske serije, obrada govora, teksta, zvuka itd.
Za razliku od običnih neuronskih mreža sa propagacijom unapred kod kojih se informacija prenosi samo u jednom pravcu (od ulaznog sloja, preko skrivenih sloja do izlaznog sloja) tako da informacija nikad ne dolazi do istog čvora dva puta, kod rekurentnih neuronskih mreža informacija se ciklično vraća u čvorove. Zbog ovoga obične mreže ne mogu da se "sete" nijedne informacije iz prošlosti, osim onih koje su dobijene prilikom treninga. Na sledećoj slici može se videti poređenje strukture ove dve vrste neuronskih mreža.



Obične rekurentne neuronske mreže imaju kratkotrajno pamćenje, dok u kombinaciji sa LSTM-ovima dobijaju i dugotrajno pamćenje. Još jedan slikovit primer rada ovih mreža u poređenju sa običnim mrežama sa propagacijom unapred može biti generisanje reči "neuron". Naime, u trenutku kada mreža sa propagacijom unapred dođe do karaktera "u", ona je već zaboravila pređašnje karaktere "n" i "e", što čini zadatak predviđanja narednog karaktera gotovo nemogućim. Rekurentna neuronska mreža zahvaljujući svojoj unutrašnjoj memoriji ove informacije zadržava, pa je i rešavanje ovakvog problema znatno lakši.

Ovakva arhitektura neuronske mreže poseduje dva ulaza - prvi predstavlja sadašnjost, a drugi prošlost. Rekurentne neuronske mreže računaju i ažuriraju težine na oba ulaza korišćenjem gradijentnog spusta ili neke modifikacije gradijentnog spusta. Na slici ispod grafički je razmotan sloj rekurentne mreže. **X** predstavlja ulaznu sekvencu, a **H** predstavlja izlaznu sekvencu. Možemo videti da je izlaz iz prvog čvora takođe jedan od ulaza u sledeći čvor.



2.2 Šta mogu biti problemi u ovakvom pristupu?

Problemi prilikom korišćenja rekurentnih neuronskih mreža mogu biti nestajući ili eksplodirajući gradijenti, koji se rešavaju uvođenjem LSTM-a. LSTM (*Long Short Term Memory*) predstavlja poboljšanje rekurentnih mreža uvođenjem dužeg "pamćenja". Sama LSTM ćelija sadrži tri tzv. *kapije* - ulazna, izlazna i zaboravljanje. Ulazna kapija odlučuje da li je novi ulaz potreban za dalju propagaciju, izlazna kapija odlučuje da li je potrebno da izlaz iz trenutnog čvora utiče na izlaz iz trenutnog vremenskog trenutka, i kapija zaboravljanja odlučuje koliko je prethodnih informacija relevantno za dalji tok učenja u trenutnom vremenskom trenutku.
Jasno je da uvođenjem LSTM-a možemo rešiti neke od problema koji nastaju zbog same arhitekture rekurentnih neuronskih mreža, ali i da to znatno usložnjava i usporava učenje. Takođe, mogu se javiti problemi ukoliko su sekvence koje se koriste za učenje predugačke, jer se ipak gube informacije zbog nestajućih gradijenata, zbog čega se predugačke zavisnosti ipak ne mogu naučiti.

3 Attention is all you need

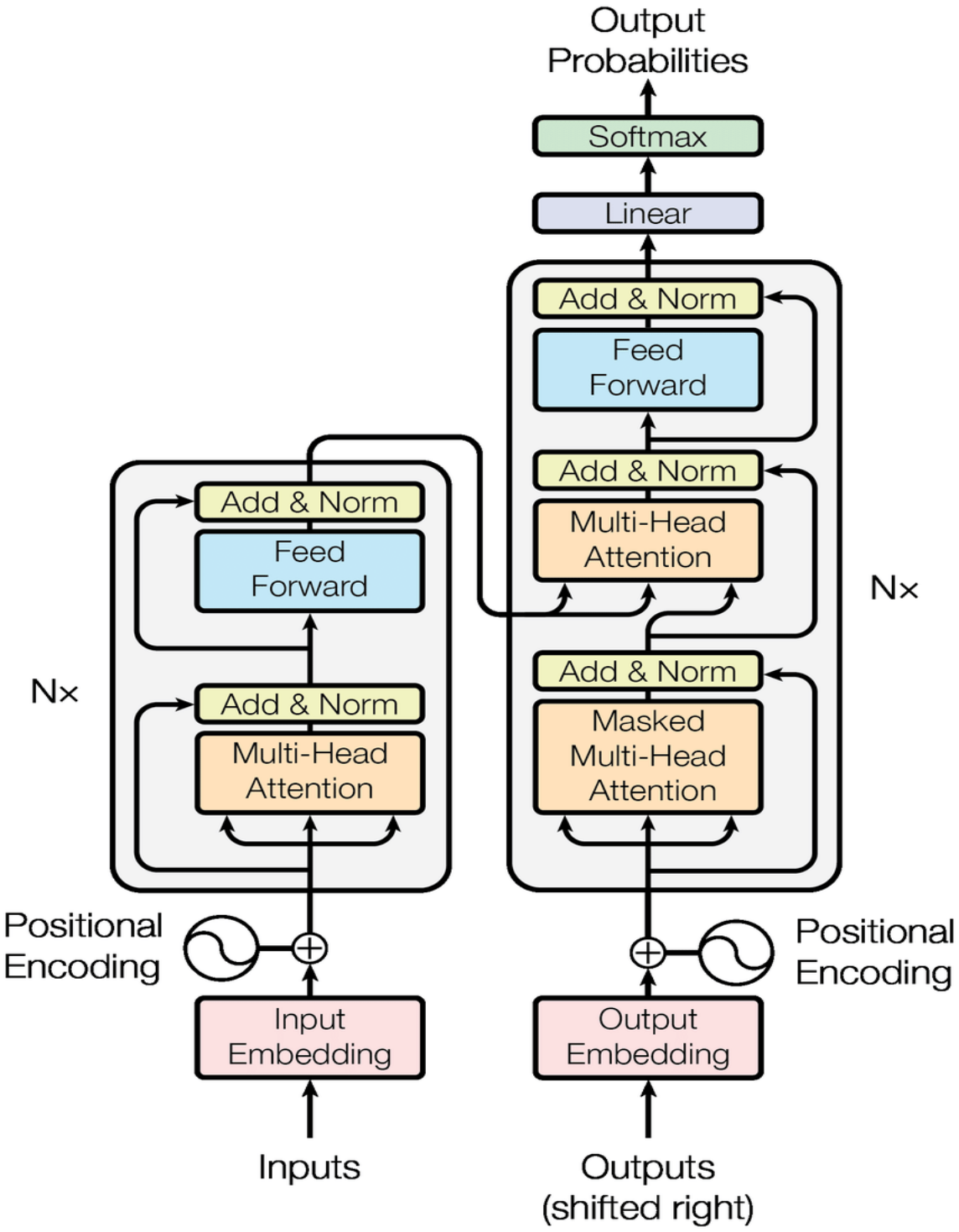
U narednom delu pozabavićemo se opisom nove arhitekture koja se može koristiti prilikom analize sekvencijalnih podataka bez korišćenja rekurentnih neuronskih mreža.

3.1 Šta je osnovna ideja ovakvog pristupa?

Umesto korišćenja standardnog RNN pristupa, arhitektura *Transformer* koristi *Attention* mehanizme kako bi procesirao ulazne sekvence i učio semantička mapiranja između izvorne rečenice i njenog prevoda. Ovakva arhitektura može se posmatrati kao dvoslojna arhitektura u kojoj prvi sloj predstavlja **enkoder** a drugi **dekoder**. Oba sloja vrše isto procesiranje ulaza, imaju *embedding* sloj koji je praćen pozicionim enkodiranjem. Zbog toga što se ne koristi klasičan RNN pristup koji prirodno enkodira sekvencijalne zavisnosti, potrebno je da ih ručno uvedemo. Ovo je urađeno korišćenjem sinusoida različitih frekvencija koje uvode instinktivnu vremensku komponentu podacima, ali i dopuštaju da se podatak procesuira u celosti u svakom trenutku.

3.2 Pristup koji je korišćen u rešavanju ovog problema

Arhitektura **enkodera** sastoji se od proizvoljnog broja *Multi-Head Scaled Dot-Product Attention (MHDPA)* blokova koji uzimaju poziciono-enkodirane podatke i iz njih izvlače *key, query, value* trojku, enkodira relacije korišćenjem skalarnog proizvoda i proizvodi izlaznu sekvencu. Arhitektura **dekodera** sastoji se od proizvoljnog broja *Masked MHDPA* i *MHDPA* blokova i vrši slične operacije enkoderu. Razlika je u tome što se ključ i vrednost enkodera prosleđuju dekoderu, dok je query koji ulazi u dekoder izlaz iz *Masked MHDPA* bloka. Izlaz dekodera propagira se kroz jedan linearni sloj koji ima *softmax* aktivacionu funkciju i proizvodi izlaznu sekvencu. Na narednoj slici predstavljena je arhitektura modela ***Transformer***:



4 Pravljenje modela

4.1 Dobijanje podataka

Podaci predstavljaju fajlove u kojima se nalaze rečenice na engleskom jeziku i korespondirajuće rečenice na nemačkom jeziku. Mogu se preuzeti pokretanjem sledeće komande:

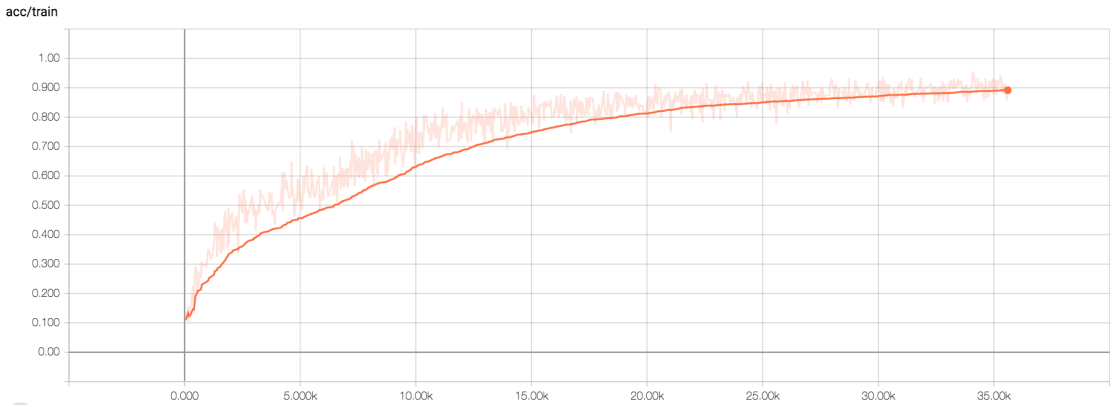
```
wget -qO- --show-progress https://wit3.fbk.eu/archive/2016-01//texts/de/en/de-en.tgz
tar xz; mv de-en data
```

4.2 Implementacija modela

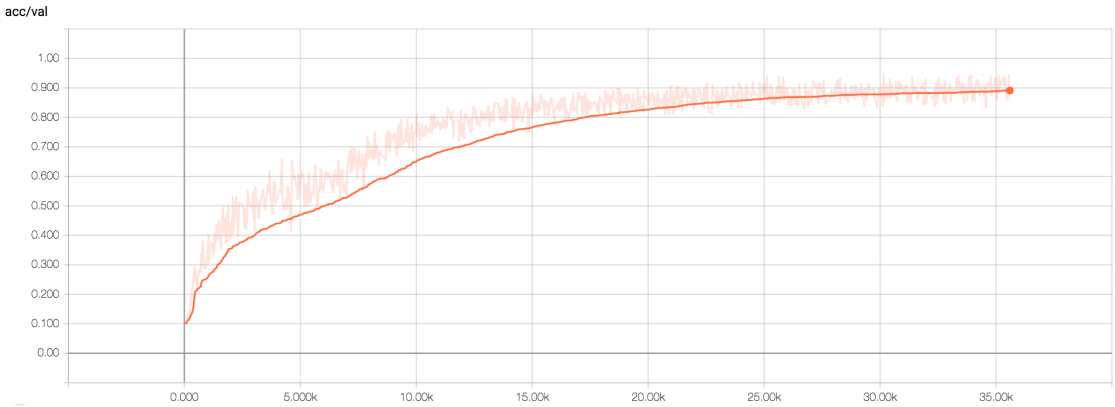
Prilikom rada osnovna biblioteka koja je bila korišćena je biblioteka ***tensorflow***, dok su ostale pomoćne biblioteke navedene u datoteci requirements.txt. Ostali implementacioni detalji i kodovi mogu se pogledati na GitHub repozitorijumu.

5 Analiza treninga i testa

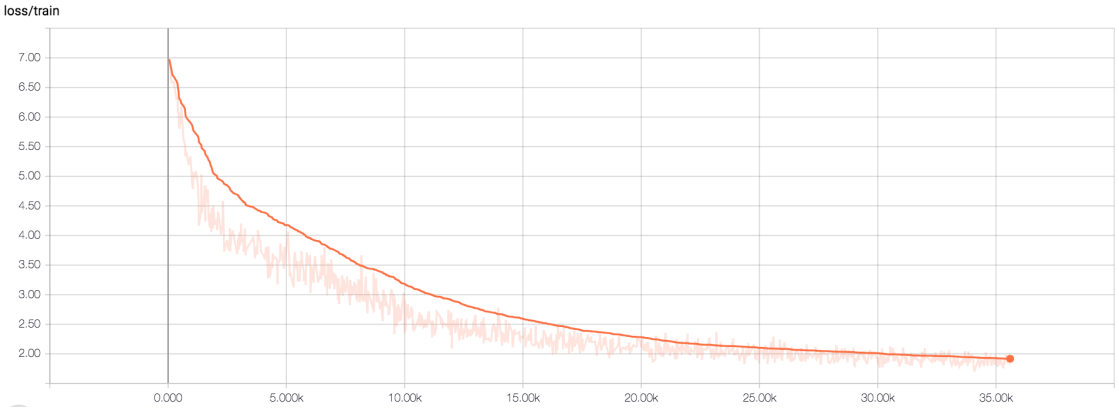
Prilikom procesa učenja pratili smo preciznost i funkciju gubitka koja je bila optimizovana korišćenjem tehnike gradijentnog spusta (kao funkcija greške korišćena je *categorical cross-entropy*). Slede slike grafika dobijene prilikom učenja na trening skupu i skupu za validaciju sa korišćenjem alata ***tensorboard***:



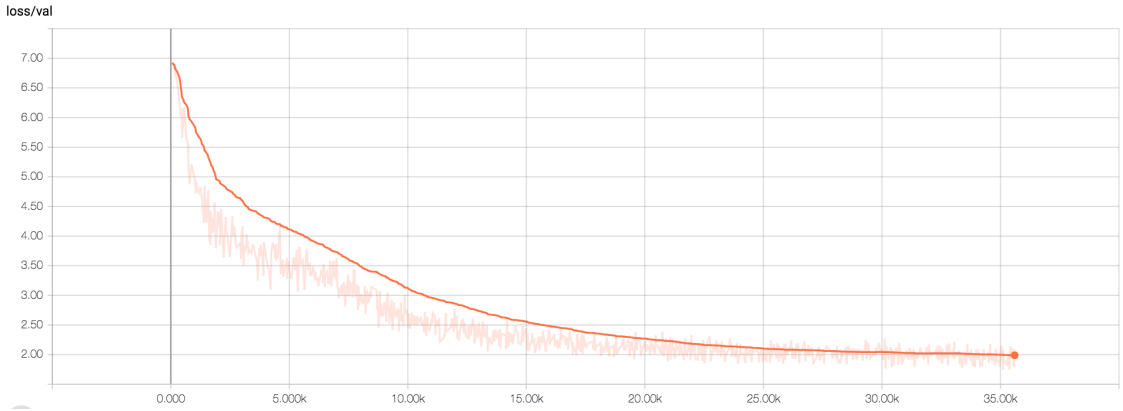
Preciznost na trening skupu.



Preciznost na skupu za validaciju.



Funkcija gubitka na trening skupu.



Funkcija gubitka na skupu za validaciju.

6 Zaključak

Analizirajući dobijene rezultate došli smo do sledećih zaključaka:

- Model je dobro razumeo povezanost između podataka, što se može zaključiti iz gorenavedenih slika grafika dobijenih prilikom izvršavanja na trening i validacionom skupu.
- Rezultati dobijeni prilikom testa su takođe zadovoljavajući.
- Bez obzira na drugačiju arhitekturu i izostavljanje rekurentnih neuronskih mreža prilikom rešavanja ovog tipa problema, treniranje i pokretanje ovog modela je još uvek izuzetno zahtevno kako vremenski, tako i memorijski.