

Tehnički izveštaj o projektu iz Deep Learninga

Studenti:

Nikola Stevanović, br. Indeksa 1281

Danilo Veljović, br. Indeksa 1120

Projekat:

Chatbot za generisanje web strana

Izveštaj

Uvod:

Prototip Chatbota je kreiran kombinacijom pristupa bag-of-words i dubokog učenja. Četbot može da prepozna i generiše sitnije HTML elemente, poput div, span, textarea itd i neke predefinisane krupne elemente koji su interno definisani, poput grida, headera, footera, koji su kombinacija više osnovnih elemenata. Četbot može istovremeno da generiše i više elemenata i da ih ubacuje na proizvoljna mesta. Četbot takođe može da radi ubacivanje elemenata jedan u drugi, kao i da ubacuje elemente jedan iznad drugog i jedan ispod drugog i takođe može da ubaci određeni, najčešće tekstualni sadržaj koji se nalazi između zagrada `< tekst >`. Sve vreme rada postoji prikaz koji u realnom vremenu prikazuje generisane elemente.

Opis korišćenih tehnika i princip rada

Projekat je razvijan u programskom jeziku Python i pomoću Tensorflow biblioteke.

Projekat koristi dva modela neuronskih mreža. Jedan model služi da prepozna html elemente koje je potrebno ubaciti, a drugi model prepoznaje prostorne odrednice (before, after, inside). Oba modela su feed-forward modeli, koji kao aktivacione funkcije koriste relu, tanh i sigmoid. Izlazni neuroni koriste sigmoid aktivacionu funkciju, da bi mogli da rade multilabel klasifikaciju, odnosno da prepoznaju više klasa istovremeno.

Aplikacija prvo prepoznaje klase kojima unos odgovara. Primera radi, ako korisnik unese *add a span and a div*, aplikacija će prepoznati klase span i div. Ovo se postiže tako što se prvo bag-of-words pristupom nađu elementi koji su prisutni, a zatim se feed-forward mrežom nalazi html kod prepoznatih klasa koje treba ubaciti. Drugi model prepoznaje prostorne odrednice, tj i koristi se na sličan način kao prvi model. Nakon što se detektuju prostorne odrednice, koje su iskorišćene u rečenici i nakon što se detektuju svi elementi koji su prisutni oni se ubacuju u postojeću web stranu.

Dataset koji je korišćen za treniranje mreža je elements.json, za model koji treba da vrati elemente za ubacivanje, i location.json, kojim se trenira model koji prepoznaje prostorne odrednice.

Takođe je moguće dodati i složene html elemente poput tabele. Kada četbot detektuje da korisnik pokušava da doda tabelu, on postavlja pitanje korisniku koliko kolona i koliko redova treba da ima ta tabela i onda joj dinamički generiše sadržaj. Implementacija koja je pružena treba da posluži kao polazna osnova za mnogo složeniji model u kome bi se na primer mogla koristiti mašina stanja za postavljanje pitanja, i proveriti da li je četbot dobio sve informacije o elementu koji treba da generiše.

Na slici 1 je dat prikaz jednog od složenih elemenata koje četbot može da generiše

```

{
  "tag": "grid",
  "patterns": [
    "i need a grid",
    "put a grid",
    "give me a grid",
    "add a grid"
  ],
  "responses": [
    "<BEFORE><div class=\"container\" id = ?>\n <div class=\"row\" id = ?>\n <div class=\"col-sm-3\" id = ?>
<CONTENT><INSIDE> </div>\n <div class=\"col-sm-6\" id = ?> <CONTENT><INSIDE> </div>\n <div
class=\"col-sm-3\" id = ?> <CONTENT><INSIDE> </div>\n </div>\n <div class=\"row\" id = ?>\n <div
class=\"col-sm-3\" id = ?> <CONTENT><INSIDE> </div>\n <div class=\"col-sm-6\" id = ?>
<CONTENT><INSIDE> </div>\n <div class=\"col-sm-3\" id = ?> <CONTENT><INSIDE> </div>\n </div>\n <div
class=\"row\" id = ?>\n <div class=\"col-sm-3\" id = ?> <CONTENT><INSIDE> </div>\n <div class=\"col-sm-6\"
id = ?> <CONTENT><INSIDE> </div>\n <div class=\"col-sm-3\" id = ?> <CONTENT><INSIDE> </div>\n
</div>\n</div><AFTER>"
  ]
}

```

Slika 1 - Složeni element grid

Na slici 1 se takođe vidi oznaka (tag) koji identifikuje element. Patterns sekcija ukazuje na moguće načine na koje sve korisnik može da zatraži taj element, u ovom slučaju grid I odgovor koji se generiše u tom slučaju odnosno sam element.

Slično za prostorne odrednice, jedan primer je dat na slici 2. Identifikator elementa u ovom slučaju je after, u patterns delu su dati svi sinonimi koje korisnik može da upotrebi, a da misli da treba da se neki element doda nakon nekog elementa. Kada se prepozna after tag, nema odgovora, u našem slučaju je jedino važno da se on prepozna.

```

{
  "tag": "after",
  "patterns": [
    "after",
    "behind",
    "next to",
    "posterior to",
    "following",
    "beside"
  ]
}

```

Pre treniranja mreže se korisnikov ulaz prečisti od znakova interpunkcije I čitav ulaz se tokenizuje I standardizuje (prevodi u small caps) I stavlja u niz kojim se mreža trenira.