Savezno učenje s unaprijed obučenim neuronskim mrežama temeljenim na transformerima

Borina, Mateo

Sveučilište Jurja Dobrile u Puli
Fakultet informatike u Puli

Mateo Borina

Savezno učenje s unaprijed obučenim neuronskim mrežama temeljenim na transformerima

Završni rad

Pula, 2022.

Sveučilište Jurja Dobrile u Puli
Fakultet informatike u Puli

Mateo Borina

Savezno učenje s unaprijed obučenim neuronskim mrežama temeljenim na transformerima

Završni rad

Ime Prezime studenta/studentice, JMBAG: Mateo Borina, 0303088140
Studijski smjer: preddiplomski sveučilišni studij informatika
Znanstveno područje: Društvene znanosti
Znanstveno polje: Informacijske i komunikacijske znanosti
Znanstvena grana: Informacijski sustavi i informatologija
Mentori: doc. dr. sc. Nikola Tanković, Robert Šajina, mag. inf.

Pula, rujan 2022.

IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani _____, kandidat za prvostupnika/ magistra _____ovime izjavljujem da je ovaj Završni/Diplomski rad rezultat isključivo mojega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Završnog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

_____

U Puli, _____, _____ godine

IZJAVA

o korištenju autorskog djela

Ja, _____ dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj završni rad pod nazivom

_____

_____

koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu

U Puli, _____, _____ godine

Potpis

_____

Sadržaj

1 Uvod

Federated learning (eng. *federated learning* - FL) je distribuirani okvir (eng. *framework*) za optimizaciju modela strojnog učenja (eng. *machine learning* - ML) (Konečný i ostali, 2015). U FL-u, podaci koji definiraju optimizaciju neravnomjerno su raspoređeni na nekoliko agenata. Cilj FL-a je istrenirati centralizirani model iz decentraliziranih podataka, gdje svaki segment decentraliziranih podataka predstavlja privatni segment podataka korisnika. Korisnički uređaji koriste se za lokalno treniranje modela preko korisničkih lokalnih podataka. Nakon završetka lokalnog treniranja, samo se ažuriranje modela šalje natrag na poslužitelj, negirajući potrebu za dijeljenjem podataka sa središnjim poslužiteljem. Nakon obuke, izračunate informacije se dijele s poslužiteljem. Zatim se na strani poslužitelja formira novi globalni model ažuriranjem prethodne globalne verzije modela s novim znanjem iz modela primljenih od korisnika. Globalni model se ažurira jednom u svakom krugu do konvergencije modela. Posljedično, model napreduje i postiže se distribuirana optimizacija.

Federalno učenje je iterativni proces (Kairouz i ostali, 2021), što znači da sadrži neke postupke gradnje, prerade i poboljšavanja. U FL-u se prerađuje i poboljšava središnji model. Iterativnost procesa osigurava dobru izvedbu i rezultate konačnog, središnjeg modela strojnog učenja. FL proces podijeljen je na runde (eng. *federated learning round*). Svaka runda procesa FL-a je skup interakcija klijent-poslužitelj. To uključuje prijenose trenutnog stanja središnjeg modela agentima, treniranja lokalnih modela na tim agentima, prikupljanja tih lokalnih ažuriranja, njihove obrade u jedinstveno globalno ažuriranje i ažuriranja globalnog modela. Svaka se runda FL-a stoga može podijeliti na pet dijelova (Bonawitz i ostali, 2019):

- Inicijalizacija - odabir modela strojnog učenja koji će se trenirati na agentima i aktivacija agenata koji čekaju zadatke
- Izbor klijenata - odabir dijela agenata za početak obuke na lokalnim podacima i prijenos trenutnog središnjeg modela na odabrane agente
- Konfiguracija - pokretanje treniranja modela na odabranim agentima s njihovim lokalnim podacima na unaprijed određen način
- Izvještavanje - slanje lokalnih modela sa agenata ka poslužitelju, agregacija primljenih modela, rješavanje problema i kvarova s ažuriranjima i slanje ažuriranja središnjeg modela agentima
- Završetak - prikupljanje ažuriranja i finalizacija središnjeg modela nakon zadovoljavanja unaprijed definiranog kriterija prekida

Neuralna mreža (eng. *neural network* - NN) je računalni sustav inspiriran biološkim neuronskim mrežama u mozgovima živih bića (Hopfield, 1982), sastavljen od umjetnih neurona ili čvorova. Neuronske mreže se koriste u mnogim suvremenim zadacima, kao što su prepoznavanje govora, prepoznavanje slike, obrada prirodnog jezika (eng. *natural language processing* - NLP) i sustavi preporuka.

Neuralne mreže koje se koriste u FL-u su umjetne neuralne mreže (eng. *artificial neural network*). Struktura umjetnih neuralnih mreža temelji se na umjetnim neuronima i vezama između tih umjetnih neurona (Yang & Yang, 2014), koje se nazivaju rubovi (eng. *edges*). Umjetni neuroni i rubovi su nelinearne matematičke funkcije (Alzahrani & Parker, 2020). Oni imaju jedan ili više ulaza i jedan izlaz. Izlazne vrijednosti umjetnih neurona i rubova ovise o težini (eng. *weight*) koja se prilagođava kako učenje napreduje. Težina mijenja podatke na spoju tako da ih smanjuje, povećava ili šalje u slučaju da je ispunjen uvjet. Umjetni neuroni u neuralnim mrežama agregirani su u slojeve. Slojevi neurona izvode različite transformacije nad podacima. Podaci putuju od ulaznog do izlaznog sloja, nakon čega se dobiva rezultat obrade.

NLP je potpodručje lingvistike i računalne znanosti koje se bavi programiranjem sustava koji mogu obraditi i analizirati velike količine podataka prirodnog jezika, poput teksta (Hirschberg & Manning, 2015). Cilj NLP-a je razumjeti sadržaj unutar podataka prirodnog jezika, što uključuje izvlačenje informacija iz skupa podataka poput teksta, organizaciju i kategorizaciju podataka te uočiti najvažnije informacije. Najveće koristi korištenja algoritama strojnog učenja i FL-a u NLP-u uključuju usredotočenost na najčešće slučajeve pri obradi, mogućnost primjene statističkog zaključivanja i povećavanje preciznosti zaključaka(Schank & Abelson, 1977).

Modeli koji se prvenstveno koriste u NLP-u su Transformeri ili modeli temeljeni na Transformerima (eng. *Transformer-based models*). Ovi modeli usvajaju mehanizme samopažnje, posebno određujući značaj svakog dijela ulaznih podataka (Vaswani i ostali, 2017). Dizajn Transformera prilagođen je obradi sekvencijalnih ulaznih podataka poput prirodnog jezika. Transformeri su namijenjeni prijevodu i sažimanju teksta. Svojstvo Transformera obrada je cijelog niza ulaznih podataka odjednom pri čemu mehanizmi samopažnje osiguravaju kontekst svim elementima u nizu ulaznih podataka. Arhitektura Transformera sastoji se od kodera (eng. *encoder*), jedinica pažnje (eng. *attention units*) i dekodera (eng. *decoder*). Funkcija svakog sloja kodera je generiranje koda koji sadrži informacije o međusobnoj relevantnosti dijelova ulaza. Svaki sloj dekodera služi za generiranje izlazne sekvence uz pomoć kontekstualnih informacija iz koda. Svaki sloj kodera i dekodera koristi jedinice pažnje kojima se važe relevantnost i proizvodi izlaz.

Prepoznavanje imenovanih entiteta (eng. *named entity recognition* - NER) samo je jedna od poddomena NLP-a. Cilj NER zadataka je klasifikacija rečeničnih riječi kao poznatih entiteta; kao što su osoba, lokacija i organizacija (Zitouni, 2014).

Treniranje u FL-u obično počinje s nasumično inicijaliziranim modelom. Međutim, tehnike prijenosa znanja pokazale su se vrlo učinkovitima u ubrzavanju procesa učenja, čak i u FL-u (Stremmel & Singh, 2020). Modeli koji se koriste u pristupima prijenosu znanja obično se treniraju na različitim skupovima podataka za različite zadatke, ali se mogu ponovno upotrijebiti kao dobra polazna točka za novi zadatak. Korištenje prethodno treniranog modela ubrzava proces treniranja i može rezultirati većom točnosti u usporedbi s nasumično inicijaliziranim modelom.

Ovaj rad će istražiti koje su veličine manjih unaprijed treniranih modela najbolje za zadatke imenovanih entiteta u FL-u i procijeniti ukupnu prikladnost FL-a za ovaj zadatak. Simulacije su izvedene s unaprijed treniranim modelima.

Modeli korišteni u eksperimentima temeljeni su na Transformerima, a simulacije su izvedene s različitim veličinama modela dvosmjernih enkodera iz Transformera (BERT) (Devlin i ostali, 2018), čiji je BERT sloj zamrznut u implementaciji. Posljedično, samo je izlazni sloj treniran i rezultati su kombinirani procesom usrednjavanja (eng. *averaging*).

Rezultati ovih simulacija su navedeni i objašnjeni. Zaključno, na temelju analize rezultata, cilj je istražiti i odgovoriti na utjecaj broja slojeva (L), veličina sakrivenih slojeva (H) i glava pozornosti (A) pri treniranju BERT jezičnih modela za NER zadatke u federalnom učenju.

Rad je organiziran na sljedeći način. U sljedećem odjeljku identificirani su postojeći radovi povezan s ciljevima ovog rada i navedeni su glavni nalazi u literaturi. Odjeljak 3. sastoji se od opisa i objašnjenja metoda kao i skupova podataka koji se koriste za treniranje modela u simulacijama. Odjeljak 4. opisuje i daje rezultate. Peti odjeljak sadrži raspravu o rezultatima, dok posljednji odjeljak zaključuje rad objašnjavajući učinke različite arhitekture BERT modela za modele trenirane u federalnom učenju. Izvorni kod se može pronaći na Githubu: https://github.com/fipu-lab/ner-federated.

2 Srodni radovi

Federalno učenje je posljednjih godina steklo velik istraživački interes. Mnogi objavljeni radovi povezani su s NLP-om, neuronskim mrežama, Transformerima i NER zadacima.

Iako je FL relativno nov koncept i još nije ušao u uobičajenu upotrebu, nedvojbeno bi se mogao primijeniti u praksi (McMahan i ostali, 2017). Svakako se može koristiti za probleme iz stvarnog svijeta. Korištenje FL-a za rješavanje problema iz stvarnog svijeta uključivalo bi dijeljenje podataka o klijentima i ogromne količine jezičnog modeliranja. Troškovi komunikacije između centraliziranog modela i klijenata također se smanjuju.

Federalno učenje primijenjeno na predviđanje riječi pokazuje stvarno dobre rezultate i već se komercijalno koristi (Hard i ostali, 2019). Model treniran federalnim učenjem u NLP-u može nadmašiti jače modele poput server trained Coupled Input-Forget Gates (CIFG) modela treniranih na serverima. Osim toga, istodobno pridonosi i privatnosti i kvaliteti modela. Primjena FL-a na NER zadatke u medicini pokazuje svoje najveće prednosti i doprinose (Ge i ostali, 2020). Osim što pomaže liječnicima prepoznavanjem nekih stvari koje su možda propustili, uvelike poboljšava privatnost pacijenata. Privatnost je poboljšana pomoću dva modula na različitim platformama. Privatni modul temelji se na podacima na svakoj platformi, dok zajednički modul sadrži znanje koje dijele različite platforme. Izvedba Transformera u FL-u nije na zadovoljavajućoj razini. To je posljedica činjenice da su Transformeri prilično veliki modeli, što znači da nisu namijenjeni za FL (Hong i ostali, 2021). Međutim, rješenje ovog problema mogli bi biti dinamički Transformeri. Dinamički Transformeri su modeli strojnog prevođenja koji skaliraju arhitekturu transformatora na temelju dostupnih resursa u određenom trenutku. Postizanje najsuvremenijih (SOTA) performansi na modelima temeljenim na transformatorima u FL-u još uvijek je teško zbog veličine tih modela (Stremmel & Singh, 2020). Međutim, vrlo je korisna primjena načela ovih velikih Transformera na manje modele. Dodatno, za određene zadatke uočeno je da modeli trenirani s različitim skupovima podataka mogu dati različite rezultate. Sigurnost FL-a također je poboljšana novim algoritmima agregacije (Fu i ostali, 2019). Čini se da su posebno otporni na backdoor napade. Također, njihova važna karakteristika je laka implementacija na postojeće FL okvire (eng. *frameworks*). Posljedično, očekuje se da će ovi novi algoritmi učiniti FL praktičnijim. Iako je posljednjih godina bilo mnogo novih istraživanja o federalnom učenju, još uvijek ima mnogo otvorenih pitanja (Kairouz i ostali, 2021). Treniranje modela temeljenih na Transformerima u FL-u je jedan od njih. Ima još puno prostora za poboljšanje algoritama i rješavanje problema koji uključuju decentralizirane skupove podataka, kao i optimizaciju koraka treniranja.

Korištenje FL-a za rješavanje NLP zadatka već je korišteno u stvarnom problemu predviđanja sljedećih riječi (eng. *next word prediction* - NWP) (Hard i ostali, 2019), (Wang i ostali, 2019). Međutim, u procesu treniranja korišteni su samo nasumično inicijalizirani modeli.

3 Metodologija

3.1 Model

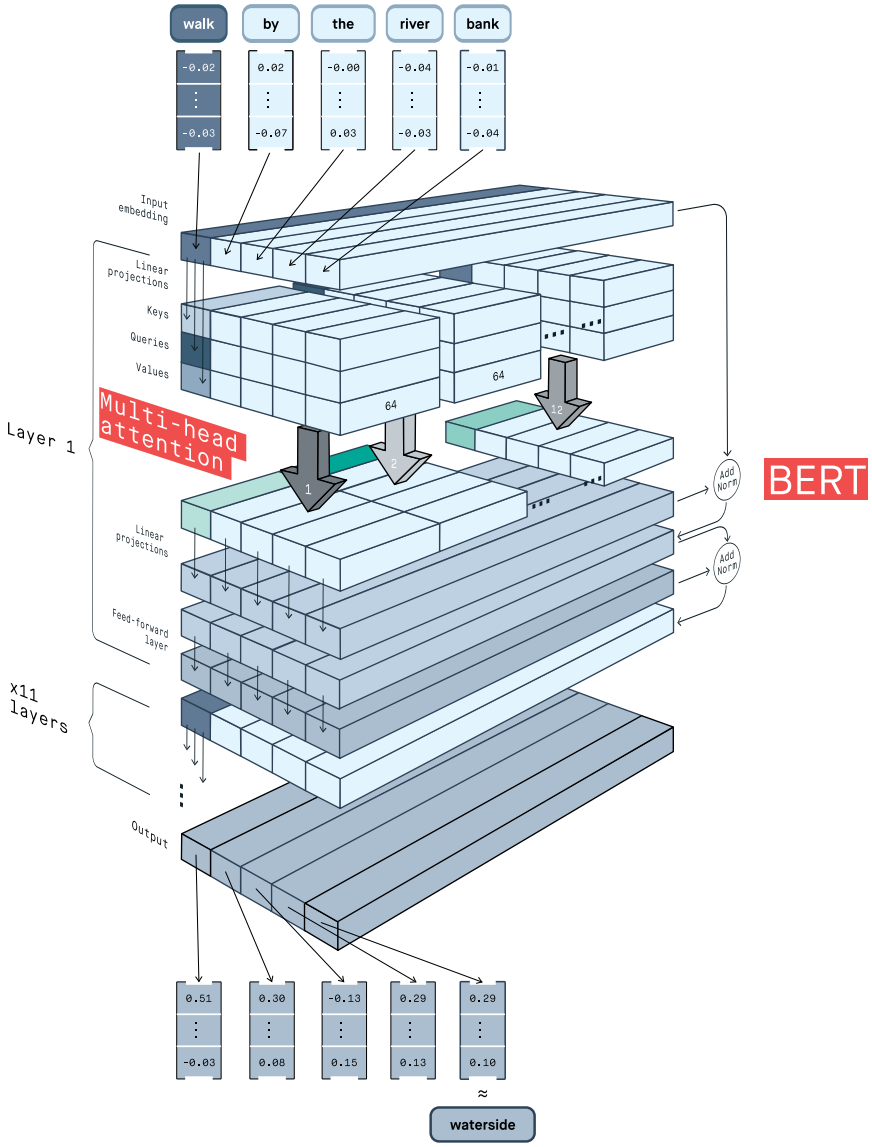
Unaprijed trenirani kompaktni BERT modeli (Turc i ostali, 2019) korišteni su u eksperimentima pomoću okvira (eng. *framework*) TensorFlow Federated (Abadi i ostali, 2015). Modeli su preuzeti s Google APIs Storage koji pripada Google Researchu (Turc i ostali, bez dat.) početkom svibnja 2022. Težine modela (eng. *weights*) pretvorene kako bi bile kompatibilne s TF2 tijekom rada (eng. workflow), koji je korišten u treniranju.

Korištena 24 BERT-a razlikuju se po broju skrivenih slojeva (eng. *hidden layers*) transformatorskog bloka L s veličinom skrivenih slojeva (eng. *hidden size*) H i broju glava pozornosti (eng. *attention heads*) A (pogledajte Tablicu 1. za više detalja). Arhitektura BERT modela prikazana je na Slici 1.

Tablica 1

Svih 24 kompaktnih BERT modela. L predstavlja skrivene slojeve, H predstavlja veličinu sakrivenih slojeva i A predstavlja glave pozornosti. Brojevi u okruglim zagradama su brojevi parametara modela (u milijunima).

	H=128, A=2	**H=256, A=4**	**H=512, A=8**	**H=768, A=12**
L=2	BERT-Tiny (4.39M)	2/256 (9.59M)	2/512 (22.47M)	2/768 (38.61M)
L=4	4/128 (4.78M)	BERT-Mini (11.17M)	BERT-Small (28.77M)	4/768 (52.79M)
L=6	6/128 (5.18M)	6/256 (12.75M)	6/512 (35.07M)	6/768 (66.96M)
L=8	8/128 (5.58M)	8/256 (14.33M)	BERT-Medium (41.38M)	8/768 (81.14M)
L=10	10/128 (5.97M)	10/256 (15.91M)	10/512 (47.68M)	10/768 (95.32M)
L=12	12/128 (6.37M)	12/256 (17.49M)	12/512 (53.99M)	BERT-Base (109.49M)



Slika 1

Arhitektura BERT modela (Futrzynski, 2020)

BERT model funkcionira koristeći Transformer, točnije njegov mehanizam pažnje koji uči kontekstualne odnose između riječi u tekstu. Mehanizam pažnje koristi se za izračunavanje vrijednosti izlaza, koji je vektor. Vrijednost izlaza jednak je ponderiranom zbroju vrijednosti. S obzirom na to da BERT generira jezični model, koristi se još samo mehanizam kodera u Transformeru. Funkcioniranje Transformera opisano je u uvodnom dijelu ovog rada.

Kompaktni BERT modeli prethodno su trenirani na velikom tekstualnom korpusu koji se sastojao od raznih knjiga kombiniranih u BookCorpus (Zhu i ostali, 2015) i neobjavljenog skupa tekstualnih podataka Wikipedije.

3.2 Podaci

Postojeći NER skupovi podataka CoNLL-2003 (Tjong Kim Sang & De Meulder, 2003) i Few-NERD (Ding i ostali, 2021) korišteni su u simulacijama. I CoNLL-2003 i Few-NERD sastoje se od podataka podijeljenih za treniranje, validaciju i testiranje.

Skup podataka CoNLL-2003 sastoji se od engleskih vijesti i članaka Reutersa u razdoblju od kolovoza 1996. do kolovoza 1997. godine. Izvorni skup podataka uključuje još jednu veliku datoteku s neoznačenim podacima, koji nisu korišteni u simulacijama ovog rada. Skup podataka sastoji se od četiri različita entiteta: osoba, lokacija, organizacija i imena različitih entiteta koji ne pripadaju niti jednoj drugoj kategoriji (MISC). CoNLL-2003 sadrži 14041 primjera za treniranje i 3453 primjera za testiranje. Skup podataka CoNLL-2003 preuzet je sa stranice autora (CoNLL-2003 dataset, bez dat.).

Skup podataka Few-NERD sastoji se od podataka podijeljenih u tri načina treniranja. Jedan od tih načina je učenje pod nadzorom (eng. *supervised learning*). Svaki način rada sadrži tri tekstualne datoteke: jednu za treniranje, jednu za testiranje i jednu za razvoj. Few-NERD sadrži 8 vrsta entiteta i 66 vrsta detaljnije opisanih entiteta (eng. fine-grained entities). U simulacijama se koriste samo grube vrste entiteta, a ti entiteti su: umjetnost, zgrada, događaj, lokacija, organizacija, osoba, proizvod i razno. Few-NERD sadrži 131767 primjera za treniranje i 37648 primjera za testiranje. Skup podataka Few-NERD preuzet je sa stranice autora (Few-NERD dataset, bez dat.).

U simulacijama, podaci za treniranje uniformno su podijeljeni na 100 korisnika, a testni podaci korišteni su za procjenu performansi točnosti globalnog modela.

3.3 Metode

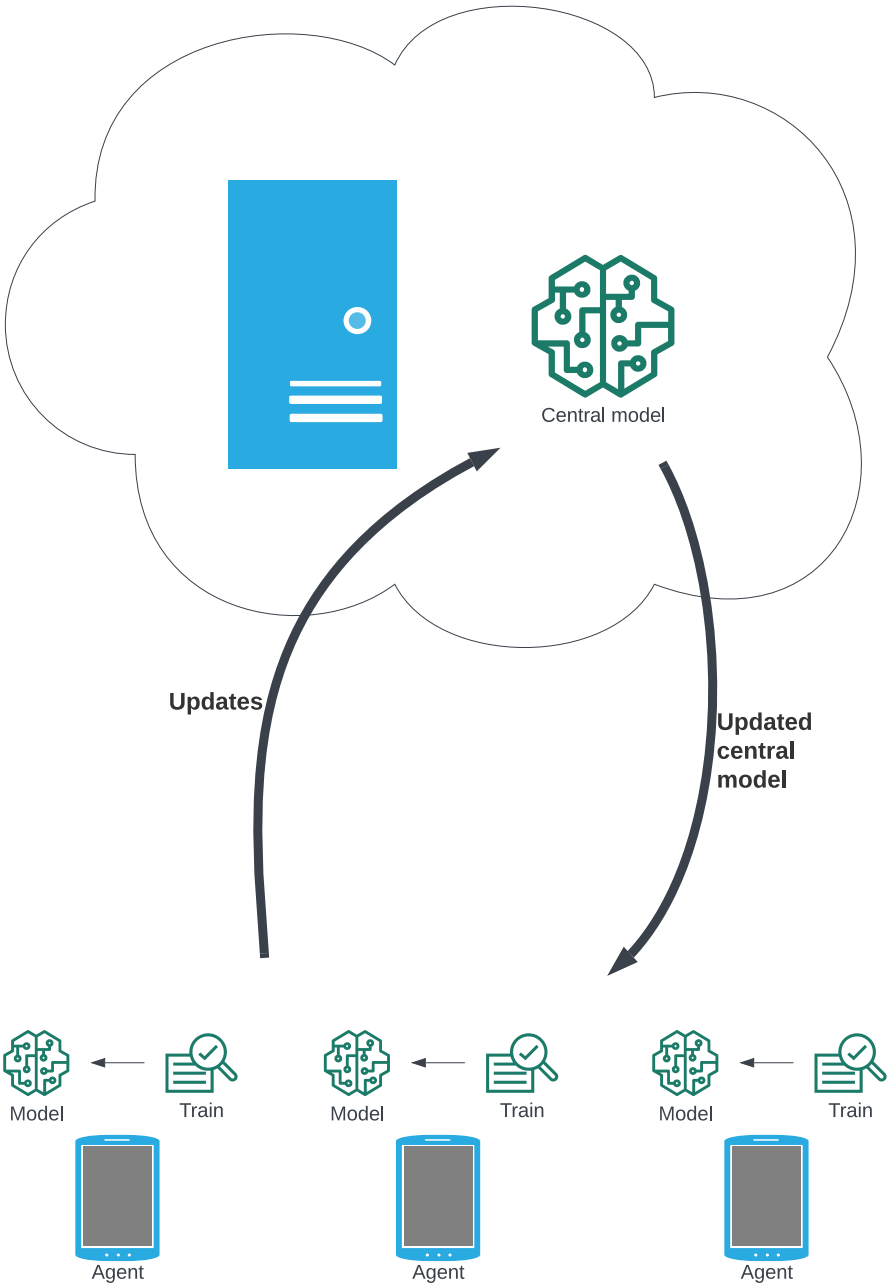
Sva 24 prethodno trenirana kompaktna BERT modela iz (Turc i ostali, 2019) korištena su za treniranje. Postojeći modeli koji se temelje na Transformerima modificirani su za podršku federalnom učenju. Prije treninga u implementaciji, slojevi BERT modela su zamrznuti i nisu trenirani pa je treniran samo izlazni sloj modela.

Predobrada podataka provodi se kako bi se osigurala kompatibilnost. Podatkovne točke su podijeljene u više skupova podataka nakon prethodne obrade kako bi se omogućila upotreba FL-a. Eksperiment treniranja za svaki par model-skup podataka izveden je tri puta. Nakon završetka treninga izračunati su prosječni rezultati iz tri eksperimenta s jednakim modelom i skupom podataka.

Podaci su pri treniranju podijeljeni na 100 korisnika, a testni podaci korišteni su za procjenu izvedbe točnosti globalnog modela za svaki eksperiment modela skupa podataka. Trenirano je 500 rundi, točnost globalnog modela računata je svakih 10 rundi. U svakom krugu FL procesa uzorkuje se deset korisnika. *Batch size* korišten kod treniranja postavljen je na 32, a stope učenja poslužitelja i klijenta postavljene su na 5e-3. Adamov optimizator korišten je za ažuriranje modela klijenta i poslužitelja, bez smanjenja stope učenja. Ovaj je *batch size*, koja je malo niža, korištena kako bi se minimizirao vremenski trošak uz pristojne rezultate na prosječnoj jačini hardvera. Navedena stopa učenja korištena je da se uravnoteži gubitak i brzinu učenja s obzirom na korištene skupove podataka.

Uzimajući u obzir samo zadnji sloj, tj. izlazni sloj je treniran, klijent u FL preuzima cijeli model, ali na poslužitelj vraća samo taj zadnji (izlazni) sloj. Nije potrebno da klijent učita cijeli model jer se mijenja samo jedan sloj. Zbog toga je opterećenje klijenta smanjeno jer je učitavanje samo zadnjeg sloja prilično jeftino.

Treniranjem se simulira način funkcioniranja FL-a u stvarnoj implementaciji. Slika 2. prikazuje način izvršavanja simulacija.



Slika 2

Grafički prikaz načina izvršavanja simulacija

4 Rezultati

Prije početka treniranja koje je rezultiralo konačnim rezultatima, obavljena su probna treniranja različitih modela s oba skupa podataka pri čemu su se prilagođavali hiper parametri *batch size* i stope učenja. To je učinjeno procesom pokušaja i pogreške (eng. trial-and-error). Rezultat toga su bolji rezultati treniranja uz primjereni vremenski trošak.

Rezultati simulacija sa svakim parom skupa podataka i modela prikazani su u Tablici 2. Tablica 2. prikazuje maksimalne (od prosječnih rezultata simulacija) f1-rezultate (eng. *f1-score*).

Općenito, veća točnost postignuta je skupom podataka CoNLL-2003 u usporedbi sa skupom podataka Few-NERD. To je vjerojatno zbog činjenice da je skup podataka CoNLL-2003 znatno manji i ima samo četiri različita tipa entiteta, u usporedbi s osam različitih tipova entiteta Few-NERD-a. Najmanje točnosti postignute su s modelima koji imaju najmanji broj glava pozornosti (A) i najmanjom veličinom sakrivenih slojeva (H). Točnost testa veća je s većim brojem glava pozornosti (A) i većim brojem veličina skrivenih slojeva (H). Ovaj je trend sličan za oba skupa podataka, što sugerira da povećanje broja glava pozornosti (A) i broja veličina skrivenih slojeva (H) donosi više koristi od jednostavnog povećanja broja skrivenih slojeva. Međutim, povećanje parametara H i A dolazi po cijenu većeg povećanja broja parametara modela u usporedbi sa samo povećanjem parametra L (pogledajte Tablicu 2.).

Tablica 2

Rezultati metrike f1-score kod treniranja modela sa CoNLL-2003, Few-NERD skupom podataka. Brojevi u zagradama predstavljaju standardne devijacije rezultata čiji je prosjek najveća točnost. L predstavlja skrivene slojeve, H predstavlja veličinu sakrivenih slojeva i A predstavlja glave pozornosti

Skupovi podataka		**Parametri**			
		H=128, A=2	**H=256, A=4**	**H=512, A=8**	**H=768, A=12**
CoNLL-2003	**L=2**	52.00% (0.24%)	63.57% (0.16%)	71.38% (0.2%)	*74.50% (0.09%)*
	L=4	51.88% (0.88%)	68.30% (0.08%)	75.85% (0.14%)	*78.89% (0.26%)*
	L=6	57.30% (0.29%)	68.70% (0.22%)	76.57% (0.02%)	***81.03% (0.07%)***
	L=8	**59.12% (0.38%)**	69.55% (0.36%)	76.01% (0.26%)	*80.63% (0.33%)*
	L=10	57.99% (0.42%)	**70.85% (0.55%)**	76.51% (0.25%)	*80.82% (0.28%)*
	L=12	58.47% (0.41%)	69.33% (0.56%)	**76.97% (0.26%)**	*80.07% (0.13%)*
Few-NERD	**L=2**	22.86% (0.13%)	31.55% (0.12%)	39.47% (0.14%)	*44.81% (0.04%)*
	L=4	25.06% (0.05%)	35.96% (0.02%)	45.57% (0.16%)	*49.33% (0.09%)*
	L=6	25.32% (0.14%)	37.80% (0.04%)	47.51% (0.12%)	*51.33% (0.08%)*
	L=8	25.67% (0.18%)	39.54% (0.07%)	47.35% (0.16%)	*51.28% (0.08%)*
	L=10	26.82% (0.24%)	**40.40% (0.22%)**	**48.28% (0.1%)**	***51.99% (0.05%)***
	L=12	**28.35% (0.21%)**	40.32% (0.1%)	47.59% (0.03%)	*51.14% (0.04%)*

Najmanje točnosti modela u treniranju s bilo kojim skupom podataka postignute su s modelima koji imaju najmanji broj transformatorskih slojeva. Samo model 4/128 kada je treniran sa skupom podataka CoNLL-2003 imao je nižu točnost od modela 2/128. Modeli za treniranje pomoću Few-NERD skupa podataka rezultirali su najvećim točnostima koje pripadaju modelima s 10 slojeva Transformera, osim kod modela s najmanjom veličinom sakrivenih slojeva

(H) i glava pozornosti, gdje su najveće točnosti postignute s modelom koji ima 12 slojeva. Najveće točnosti treniranja korištenjem skupa podataka CoNLL-2003, s obzirom na veličine sakrivenih slojeva (H) i broj glava pozornosti, pripadaju modelima s 8, 10, 12 i 6 slojeva Transformera, odnosno od najmanjeg do najvećeg broja veličina sakrivenih slojeva (H) i glava pozornosti (A).

Rezultati standardnih devijacija zadovoljavajući su. Sve standardne devijacije manje su od 1%. Primjećuje se da su standardne devijacije rezultata treniranja modela s Few-NERD skupom podataka manje od standardnih devijacija rezultata treniranja modela sa CoNLL-2003 skupom podataka. Treniranje modela skupom podataka CoNLL-2003 rezultiralo je povećanjem standardnih devijacija s povećanjem broja slojeva Transformera modela. Može se primijetiti trend gdje rezultati treniranja modela koji imaju manje veličine sakrivenih slojeva (H) i brojeve glava pozornosti (A) sa skupom podataka CoNLL-2003 imaju veće standardne devijacije. Veće standardne devijacije kod rezultata treniranja modela s Few-NERD skupom podataka mogu se pronaći duž druge dijagonale.

5 Rasprava

Neki od postignutih rezultata su očekivani, dok su neki vrlo zanimljivi. Rezultati treniranja modela s različitim brojem slojeva Transformera manje su konzistentni s obzirom na različite skupove podataka i mnogo manje očekivani. Postoji proporcionalna veza između nižih točnosti i manjih odstupanja jer su rezultati treniranja s Few-NERD skupom podataka imaju manja odstupanja.

Modeli trenirani s Few-NERD skupom podataka rezultirali su najvećom točnosti koja pripada modelima s 10 slojeva transformatora. Postoji odstupanje od tog trenda kod manjih modela. Najveće točnosti treniranja modela s CoNLL-2003 skupom podataka ne slijede nikakve posebne trendove. Međutim, rezultati treniranja modela na vrhuncu su kod srednje visokih slojevitih modela s najvećim i najmanjim veličinama skrivenih slojeva (H), kao i glavama pozornosti. Najveće točnosti modela pri treniranju sa srednjim veličinama skrivenih slojeva (H) i glavama pozornosti postignute su s modelima koji imaju veći broj slojeva.

Iz rezultata standardnih devijacija moguće je vidjeti da kod treniranja modela s manjim skupovima podataka, manji modeli daju dosljednije rezultate. Kod treniranja modela s velikim skupovima podataka, dosljednije rezultate može se očekivati kod najmanjih i najvećih modela.

6 Zaključak

Treniranje BERT modela u FL bila je prilično uspješna. Svi BERT modeli trenirani su s dva skupa podataka, jedan od njih je CoNLL-2003, a drugi Few-NERD. Iz postignutih rezultata izlučeni su važni trendovi i zakonitosti. Ti će trendovi pomoći budućim korisnicima i istraživačima kada treba odlučiti koji su modeli najbolji za određene svrhe i koje modele koristiti za dosljednije rezultate.

Uspješno je obavljeno treniranje svih 24 kompaktnih BERT modela. Uz rezultate se može zaključiti da su za treniranje najbolji modeli s najvećim veličinama sakrivenih slojeva (eng. *hidden sizes*) i najvećim brojevima glava pozornosti (eng. *attention heads*). Što se tiče odabira modela s obzirom na broj slojeva, ovdje su rezultati manje dosljedni i utječu o korištenom skupu podataka. Korištenjem većih skupova poput Few-NERD-a rezultati su dosljedniji. Pronađeno je da modeli s 10 slojeva daju najbolje rezultate, uz manja odstupanja kod treniranja pomoću većih skupova podataka i veća odstupanja pri treniranju s manjim skupovima podataka.

Popis literature i drugih izvora podataka koji su upotrijebljeni u izradi završnog rada

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., … Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems*. https://www.tensorflow.org/

Alzahrani, R. A., & Parker, A. C. (2020). Neuromorphic Circuits With Neural Modulation Enhancing the Information Content of Neural Signaling. *International Conference on Neuromorphic Systems 2020*, 1–8. https://doi.org/10.1145/3407197.3407204

Bonawitz, K. A., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konečný, J., Mazzocchi, S., McMahan, H. B., Overveldt, T. V., Petrou, D., Ramage, D., & Roselander, J. (2019). Towards federated learning at scale: System design. *CoRR*, *abs/1902.01046*. http://arxiv.org/abs/1902.01046

CoNLL-2003 dataset. (bez dat.). CNTS - Language Technology Group. http://www.cnts.ua.ac.be/conll2003/ner.tgz

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of deep bidirectional transformers for language understanding*. http://arxiv.org/abs/1810.04805

Ding, N., Xu, G., Chen, Y., Wang, X., Han, X., Xie, P., Zheng, H.-T., & Liu, Z. (2021). Few-NERD: A few-shot named entity recognition dataset. *CoRR*, *abs/2105.07464*. https://arxiv.org/abs/2105.07464

Few-NERD dataset. (bez dat.). Tsinghua University. https://ningding97.github.io/fewnerd/

Fu, S., Xie, C., Li, B., & Chen, Q. (2019). Attack-resistant federated learning with residual-based reweighting. *CoRR*, *abs/1912.11464*. http://arxiv.org/abs/1912.11464

Futrzynski, R. (2020). Getting meaning from text: Self-attention step-by-step video. U *Self-attention: step-by-step video | Peltarion*. https://peltarion.com/blog/data-science/self-attention-video

Ge, S., Wu, F., Wu, C., Qi, T., Huang, Y., & Xie, X. (2020). FedNER: Privacy-preserving medical named entity recognition with federated learning. *CoRR*, *abs/2003.09288*. https://arxiv.org/abs/2003.09288

Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., & Ramage, D. (2019). *Federated learning for mobile keyboard prediction*. https://arxiv.org/abs/1811.03604

Hirschberg, J., & Manning, C. D. (2015). Advances in natural language processing. *Science*, *349*, 261–266.

Hong, Z., Wang, J., Qu, X., Liu, J., Zhao, C., & Xiao, J. (2021). *Federated learning with dynamic transformer for text to speech*. https://arxiv.org/abs/2107.08795

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America*, *79*(8), 2554–2558. http://view.ncbi.nlm.nih.gov/pubmed/6953413]

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., D'Oliveira, R. G. L., Eichner, H., Rouayheb, S. E., Evans, D., Gardner, J., Garrett, Z., Gascón, A., Ghazi, B., Gibbons, P. B., … Zhao, S. (2021). *Advances and open problems in federated learning*. https://arxiv.org/abs/1912.04977

Konečný, J., McMahan, B., & Ramage, D. (2015). *Federated optimization:distributed optimization beyond the datacenter*. https://arxiv.org/abs/1511.03575

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & Arcas, B. A. y. (2017). *Communication-efficient learning of deep networks from decentralized data*. https://arxiv.org/abs/1602.05629

Schank, R. C., & Abelson, R. P. (1977). *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. L. Erlbaum Associates ; distributed by the Halsted Press Division of John Wiley; Sons.

Stremmel, J., & Singh, A. (2020). *Pretraining federated text models for next word prediction*. https://arxiv.org/abs/2005.04828

Tjong Kim Sang, E. F., & De Meulder, F. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, 142–147. https://doi.org/10.3115/1119176.1119195

Turc, I., Chang, M.-W., Lee, K., & Toutanova, K. (bez dat.). *All BERT models*. ver. 1. https://storage.googleapis.com/bert_models/2020_02_20/all_bert_models.zip

Turc, I., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Well-read students learn better: On the importance of pre-training compact models. *arXiv Preprint arXiv:1908.08962v2*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 5998–6008.

Wang, K., Mathews, R., Kiddon, C., Eichner, H., Beaufays, F., & Ramage, D. (2019). *Federated evaluation of on-device personalization*. https://arxiv.org/abs/1910.10252

Yang, Z. R., & Yang, Z. (2014). Artificial Neural Networks. U *Comprehensive Biomedical Physics* (str. 1–17). Elsevier. https://doi.org/10.1016/B978-0-444-53632-7.01101-1

Zhu, Y., Kiros, R., Zemel, R. S., Salakhutdinov, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *CoRR*, *abs/1506.06724*. http://arxiv.org/abs/1506.06724

Zitouni, I. (2014). *Natural language processing of semitic languages*. Springer.

Sažetak i ključne riječi (Abstract and keywords)

Federalno učenje je paradigma kolaborativnog učenja koja se brzo razvija i trenira modele neuronske mreže na uređajima korisnika. Federalno učenje se distribuira, a samo se rezultati treniranja dijele sa središnjim poslužiteljem, čuvajući privatnost. Modeli temeljeni na Transformerima su modeli strojnog učenja koji se koriste za predviđanje slijeda riječi u obradi prirodnog jezika (NLP). Ovaj rad istražuje ideju o treniranju Transformera u arhitekturi udruženog učenja na zadatku prepoznavanja imenovanih entiteta (NER). Transformeri za treniranje mogu se provesti korištenjem nasumično inicijaliziranih ili prethodno treniranih modela. Ovaj rad analizira rezultate treniranja prethodno treniranih BERT modela u federalnom učenju. Sve 24 manje veličine BERT modela koriste se u eksperimentima za istraživanje utjecaja slojeva i veličina sakrivenih slojeva (H) pri korištenju unaprijed treniranih modela. Uvježbani modeli se procjenjuju na dobro poznatim skupovima podataka, a konačni rezultati se navode i analiziraju. Izvorni kod se može pronaći na Githubu: https://github.com/fipu-lab/ner-federated

Abstract—Federated learning is a fast-emerging collaborative learning paradigm of training neural network models on users' devices. Federated learning is distributed, and only training results are shared with a central server, preserving privacy. Transformer-based models are machine learning models used for word sequence prediction in natural language processing. This paper explores the idea of training Transformers in federated learning architecture on a named-entity recognition (NER) task. Training Transformers can be conducted using randomly initialized or pre-trained models. This paper analyses results of training pre-trained BERT models in Federated learning. All 24 smaller BERT model sizes are used in the experiments to explore the impact of layers and hidden size when using pre-trained models. Trained models are evaluated on well-known datasets, and the final results are provided and analyzed. Source code can be found on Github: https://github.com/fipu-lab/ner-federated

Ključne riječi - federalno učenje; obrada prirodnog jezika; Transformeri; zadaci prepoznavanja imenovanih entiteta; BERT modeli; Skup podataka CoNLL-2003; Skup podataka Few-NERD

Keywords—federated learning; natural language processing; Transformers; named-entity recognition tasks; BERT models; CoNLL-2003 dataset; Few-NERD dataset

Popis tablica

Popis slika