

Sadržaj

1. Uvod.....	1
2. Predviđanje cijena dionica.....	2
2.1 Problem trgovanja dionicama	2
2.2 Pristup problemu	4
2.3 Strojno učenje	7
3. Izrada modela neuronske mreže	9
3.1 LSTM mreže	10
3.2 Ulazni podaci neuronske mreže	16
3.3 Arhitektura neuronske mreže	18
4. Implementacija modela	20
4.1 Treniranje i testiranje modela neuronske mreže	21
4.2 Izrada skripte za predikciju cijena dionica	21
4.2.1 Implementacija modela neuronske mreže u skriptu	22
4.2.2 Dohvaćanje aktualnih podataka	22
4.2.3 Obrada podataka.....	23
4.2.4 Prikazivanje rezultata.....	24
5. Analiza rezultata.....	26
5.1 Ispitivanje modela	26
5.2 Primjer korištenja skripte za predikciju cijena dionica	30
6. Budući rad	34
7. Zaključak	35
Literatura	36

1. Uvod

Dionica je oblik vrijednosnog papira koji ukazuje na to da dioničar (vlasnik dionice) ima razmjerno vlasništvo u korporaciji koja ga izdaje [1]. Još od 12. stoljeća, dionice se koriste kao dokaz vlasništva u različitim fondovima ili tvrtkama.

Cijena dionice proporcionalna je kapitalu i vrijednosti tvrtke na koju se odnosi. Zbog fluktuacije njihove vrijednosti, dionice uglavnom služe kao sredstvo za trgovanje s ciljem stjecanja kratkoročne zarade (princip „kupi nisko, prodaj visoko“) ili kao imovina koja dugoročno čuva vrijednost uloženog kapitala.

Razvoj internetskih burzi učinio je trgovanje dionicama vrlo pristupačnim ozbiljnim investitorima, ali i subjektima koji se u tome tek žele okušati. Razvoj algoritama za trgovanje dionicama, učinio je proces postanka uspješnim investitorom još jednostavnijim, jer umjesto samostalnog iscrpnog pretraživanja trendova i ekonomsko-političkih zbivanja, pojedinci se mogu osloniti na algoritme koji istraživanja i donošenje financijskih odluka vrše umjesto njih.

Zbog nedeterminističkog kretanja cijena dionica, nije moguće izravno definirati stanja koja je moguće dodijeliti pojedinim vrijednostima, već se problem pronalazaka pravilnosti među podacima mora svesti na traženje međusobnih veza između vrijednosti, zbog čega su neuronske mreže savršen alat za pristup rješavanju problema. Vrsta neuronskih mreža korištena za konstrukciju predikcija je LSTM mreža, čija je glavna značajka slijedna analiza podataka i pamćenje stanja.

Cilj rada je ispitati različite modele neuronskih mreža na različitim ulaznim podacima te implementacija mreže s najboljim rezultatima u skriptu za predikciju cijena dionica. Skripta služi kao korisničko sučelje, kako bi korisnici bez znanja iz područja tehničke domene mogli jednostavno koristiti model na aktualnim podacima.

2. Predviđanje cijena dionica

Pojava interneta i globalizacija omogućili su da, neovisno o geografskom položaju, dioničar (osoba koja trguje dionicama ili posjeduje dionice) ima uvid u sve ekonomske ili političke informacije vezane uz poslovanje tvrtke čiji udio posjeduje. Uz aktualne podatke, vrlo je lako pronaći i podatke o poslovanju iste tvrtke čak i prije pedeset godina (ako je, naravno, tada postojala).

Veliki korak u povećanju pristupačnosti trgovanja dionica bio je razvoj internetskih burzi, kao što je „Robinhood“. Za početak trgovanja na „Robinhood-u“, potrebno je imati uređaj koji može pristupiti servisu (računalno, mobitel, tablet...) i internetsku vezu. Nakon što je dioničar potvrdio svoj identitet, može ući na burzu.

Jedna od barijera za početak trgovanja dionicama u prošlosti bila je i mogućnost kupnje isključivo cijele dionice. Cijena jedne dionice može iznositi i nekoliko tisuća eura, što većini ne-investitora nije pristupačno i predstavlja prevelik financijski rizik. Trgovanje na burzama se na tom principu vršilo sve do 2017. godine, kada je dio posrednika omogućio kupnju djelomičnih dionica [2]. Time je omogućena kupnja dijela dionice proporcionalna uloženom kapitalu, koji može iznositi svega nekoliko centa.

Pristupačnost podacima, burzama i proizvoljni iznosi sredstva za ulaganje učinili su uspješno trgovanje dionicama lakšim no ikad, ali unatoč tome, u prosjeku se preko 80% dioničara, nakon pretrpljenih novčanih gubitaka, povuče s tržišta [3].

2.1 Problem trgovanja dionicama

Još u ranim 2000-tima, aktualne informacije o poslovanju neke tvrtke bilo je moguće pronaći u poslovnim novinama ili pozivom na burzu. Također, iako su postojale konkretne informacije, njihova opsežnost bila je ograničena. Novine su izlazile svaki dan ili tjedan, stoga je količina pruženih informacija bila ograničena.

Kako bi stvorili uspješan portfelj koji donosi profit, investitori su, uz aktualno stanje tržišta, morali znati velik broj informacija i tehnika vezanih uz upravljanje portfeljem. Takve informacije uglavnom su pohranjene u knjigama zbog svoje opsežnosti i kompleksnosti.

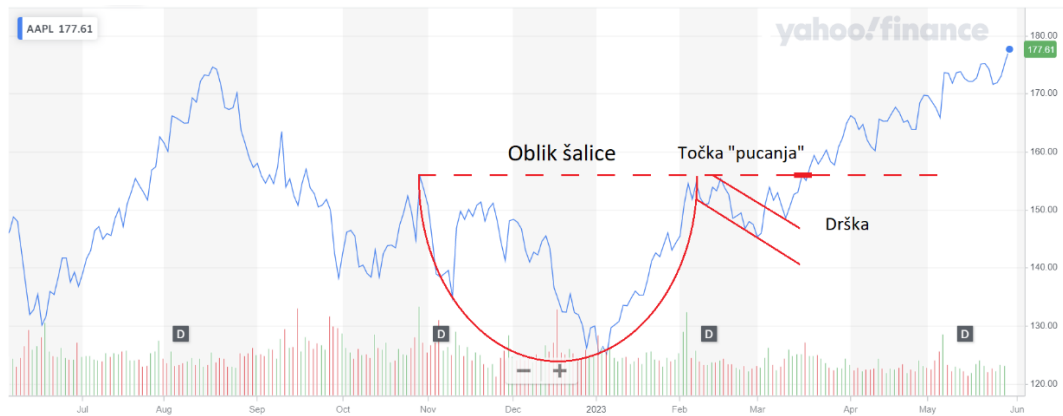
Ovisno o financijskim ciljevima, agresivnosti ulaganja, vremenu koje želi provesti na tržištu i rizicima koje je spreman podnijeti, investitor ima nekoliko opcija pri izgradnji portfelja.

Investitori čiji je cilj očuvati vrijednost uloženog kapitala s niskim godišnjim profitom fokusirat će se na dugoročno ulaganje u dionice stabilnih tvrtki čija vrijednost kontinuirano raste kroz godine. Dugoročno ulaganje uglavnom je niskog rizika, jer profit nije podložan utjecaju trenutnog stanja tržišta, već se očekuje dobit u vremenskom okviru duljem od npr. deset godina. Investitor u tom slučaju treba diversificirati portfelj, kako uloženi kapital ne bi bio ovisan o uspjehu nekoliko pojedinačnih tvrtki.

Suprotno od dugoročnog ulaganja, investitori se mogu fokusirati na kratkoročno ulaganje. Vremenski okvir kratkoročnog ulaganja u pojedinačnu dionicu može trajati svega nekoliko minuta. Cilj takvog ulaganja jest prodaja dionice po cijeni višoj od kupovne u što kraćem vremenskom razmaku između kupnje i prodaje. Iako profit kratkoročnim ulaganjem kroz nekoliko sati može biti jednak profitu koji dugoročni portfelj akumulira kroz nekoliko godina, ova metoda ulaganja je mnogo riskantnija. Razlog tome je izloženost portfelja aktualnim ekonomskim i političkim zbivanjima. Investitori čija namjera nije dugoročno posjedovanje dionice, uslijed kratkoročnog pada cijene dionice, prisiljeni su prodati je po cijeni nižoj od kupovne, čime je dobiveni kapital niži od uloženog. Iako je tržište podložno različitim vanjskim utjecajima, pokazalo se da postoje određene pravilnosti u kretanju cijena dionica zbog reakcija tržišta na različite okolnosti.

Proučavanje grafova cijena te potraga za pravilnostima u istima s ciljem uočavanja ulagačkih prilika predmet je tehničke analize cijena dionica. Iako se nije moguće u potpunosti pouzdati u tehničku analizu pri trgovanju dionicama zbog podložnosti tržišta na vanjske utjecaje, tehnička analiza je čest alat korišten među investitorima kao pomagalo pri donošenju odluka o trgovanju. Tehnička analiza sadrži brojne tehnike, od jednostavnih kao što su linije otpora i podrške, koje je moguće razumjeti kroz nekoliko minuta učenja, do kompleksnih tehnika poput „Ichimoku oblaka“, koji sadrži nekoliko komponenata koje je moguće različito interpretirati. Jedno od područja tehničke analize jest utvrđivanje pravilnosti u kretanju cijena, odnosno uočavanja oblika grafova.

Na slici 2.1 prikazana je cijena dionice tvrtke „Apple“ u 2022. i 2023. godini, gdje je crvenom bojom istaknut „oblik šalice“, koji se koristi kao indikator rasta cijene. Sastoji se od šalice i drške. Kada cijena (nakon oblika drške) naraste iznad razine šalice, pravilo nalaže da slijedi period kontinuiranog rasta cijene, što je slučaj u primjeru.



Slika 2.1: oblik šalice [4]

Problem kod pronalaženja oblika jest što su rezultati analize subjektivni, odnosno dok će neki investitori uočiti „oblik šalice“ u navedenom primjeru, drugi se neće složiti i odbacit će navedeni oblik kao indikator. Ako se radi o trenutnim cijenama, a ne onima iz prošlosti, nemoguće je znati koja je skupina investitora bila u pravu. Također, sam pojam kontinuiranog rasta cijene subjektivan je u očima investitora. Rast cijene može iznositi manje od 1%, što će u pravilu zadovoljiti ulagače s visokim ulaznim kapitalom i niskim rizikom koji su spremni prihvatiti, dok će agresivniji ulagači taj rast smatrati zanemarivim i nedovoljnim za stjecanje željenog profita.

Razvoj računalne snage i algoritama strojnog učenja potaknuo je razvoj algoritama koji vrše tehničku analizu umjesto investitora s ciljem minimizacije faktora ljudskih grešaka kod provođenja postupka. Tema ovog rada je izrada sličnog algoritma s ciljem predviđanja cijene dionice za kratkoročno ulaganje, točnije za ulaganje kroz nadolazeća dvadeset i četiri sata. Zbog apstrakcije pojma rasta ili pada cijene, to jest subjektivne razine prihvatljivog iznosa rasta, u ovom radu, fokus je isključivo na prikazivanju konkretnih vrijednosti cijena i na neoporecivih činjenica, koje nisu otvorene subjektivnim interpretacijama.

2.2 Pristup problemu

Za „uspješnu“ izradu algoritma za predikciju cijena dionica, potrebno je upoznati se sa samim problemom kretanja cijene dionica kako bi uopće mogli ispravno pristupiti rješavanju istog.

Cijena dionice tvrtke ovisi o vrlo velikom broju faktora, od kojih velik dio nije direktno povezan s poslovanjem tvrtke i odlukama koje vodstvo tvrtke donosi, zbog čega je neophodno pratiti zbivanja i odnose u ekonomskom „ekosustavu“ u kojem se tvrtka nalazi. Pogledajmo primjer kojim ćemo ilustrirati ovisnost poslovanja tvrtke o ekonomskim zbivanjima na tržištu. Uzmimo kao primjer tvrtke X i Y koje proizvode i prodaju mobitele slične cijene te su konkurenti na tržištu. Tvrtka X materijale nabavlja od tvrtke A , dok tvrtka Y iste nabavlja od tvrtke B . Ako tvrtka A cijenu materijala poveća 10%, cijena uređaja kojeg proizvodi tvrtka X , kako bi tvrtka X ostvarila profit, mora porasti 5%, zbog čega će cijena uređaja kojeg proizvodi tvrtka Y biti pristupačnija. Time će tvrtka Y ostvariti natjecateljsku prednost u odnosu na tvrtku X , što će dovesti do veće prodaje uređaja tvrtke Y te posljedično rasta cijene njezine dionice. Iako tvrtka Y nije radila promjene u vlastitom poslovanju, zbog ekonomskih okolnosti na tržištu je ostvarila prodajnu prednost nad konkurentom.

Velik faktor u uspješnom poslovanju tvrtke imaju i politički odnosi unutar države u kojoj ista posluje, ali i odnosima vanjskih zemalja s matičnom zemljom. Danas, kada su društvene mreže i mediji postali moćan prodajni i marketinški alat, politički stavovi predstavnika (ili zaposlenika) mogu uvelike utjecati na poslovanje tvrtke. Za ilustraciju, ponovo ćemo se poslužiti primjerom. Ako direktor tvrtke X izrazi svoje konzervativne stavove u intervju, postoji mogućnost da će tvrtka izgubiti dio liberalno orijentiranih klijenata.

Uz navedene primjere, postoji vrlo velik broj situacija koje mogu utjecati na poslovanje i vrijednost tvrtke, koje su van njene kontrole. Sam proizvod ili usluge koje tvrtka nudi nisu dovoljne da bi joj osigurale „uspjeh“. Svaka odluka koju profesionalni investitori donesu, utvrđena je uzvši u obzir što veći broj faktora koji se odnose na samo poslovanje tvrtke, ali i tržište u kojem se ista nalazi. Ako investitor u vijestima pročita novosti koje bi mogle utjecati na poslovanje neke tvrtke, pokušat će novostečeno znanje iskoristiti za donošenje odluka pri budućim trgovanjima.

Primarni indikatori vrijednosti tvrtke su unutarnji faktori koji se odnose na samo poslovanje tvrtke. Broj prodanih proizvoda, sklapanje novog partnerstva, zamjena vodstva tvrtke i povećanje budžeta za oglašavanje samo su neki od čimbenika koji mogu pozitivno ili negativno utjecati na poslovanje tvrtke. Moguće je lako uočiti utjecaj određenih čimbenika na vrijednost tvrtke. Ako neka tvrtka proda dvostruko više proizvoda nego prethodne godine, lako je pretpostaviti da je vrijednost tvrtke porasla. S druge strane, utjecaj nekih promjene nije moguće lako interpretirati. Ako dođe do zamjene direktora

tvrtke, nemoguće je unaprijed znati kakve će promjene u poslovanju ta promjena donijeti. Novi direktor može nizom dobrih odluka u konačnici poboljšati poslovanje tvrtke. Također, niz loših odluka može narušiti partnerstva koja je tvrtka stekla ili smanjiti pristupačnost proizvoda, što dugoročno nosi loše posljedice.

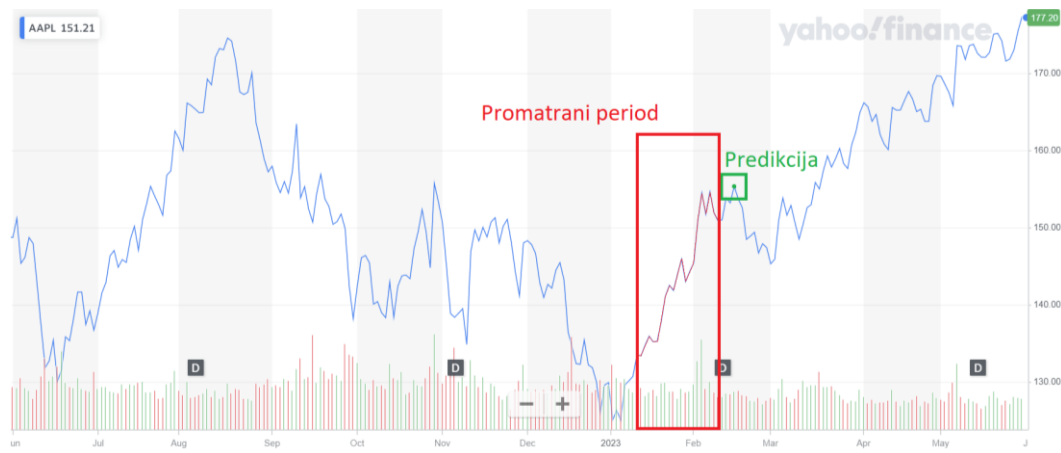
Za razliku od arhetipskog investitora koji uspješno može tumačiti zbivanja u poslovnom, ekonomskom i političkom svijetu, ali i unutar same tvrtke te razumjeti njihov utjecaj, isto je vrlo teško isto reći za algoritam. Razvoj takvog algoritma s trenutnim znanjima nije moguć, ili barem takav algoritam nije predstavljen javnosti. Stvaranje znanja koje računalo može koristiti iz ekonomsko-političkih zbivanja vrlo je kompleksan problem, a i zaključivanje na temelju istog, zahtijevao bi razumijevanje utjecaja pojedinih događaja na tržište. Prethodno spomenuti arhetipski investitori, također su vrlo rijetki u stvarnome svijetu. Kupovne odluke za tvrtke koje su specijalizirane za trgovanje dionicama donose timovi stručnjaka iz različitih grana na području financija. Unatoč velikoj količini znanja i iskustva na temelju kojih se odluke donose, profitabilno trgovanje dionicama i dalje je vrlo težak problem i nemoguće je sa sto-postotnom sigurnošću donositi „ispravne“ kupovne odluke.

Uklonivši sve vanjske i unutarnje utjecaje na poslovanje tvrtke, fokus algoritamske implementacije stavlja se na tehničku analizu. Iako je i sama težak problem i nije u potpunosti pouzdana, tehnička analiza sama može biti koristan alat u donošenju odluka investitora. Ako investitor na temelju vlastitog znanja i iskustva zaključi da trenutna ekonomska klima ima pozitivan utjecaj na poslovanje neke tvrtke, optimistični rezultati tehničke analize mogu dodatno poduprijeti donesenu odluku. Zbog „pravilnih“ reakcija tržišta na pojedina stanja, tehnička analiza može nam pomoći primijeniti iskustva iz prošlosti kako bi donijeli ispravne odluke.

Tehnička analiza implicira rad s isključivo numeričkim vrijednostima. Neke od vrijednosti kojima je moguće definirati kretanje dionice i jednom danu su prosječna cijena, cijena pri otvaranju i zatvaranju burze. Radi lakše analize podataka, bavit ćemo se isključivo vrijednostima u vrijeme zatvaranja tržišta, jer je to efektivna vrijednost kojoj bi cijena dionice, ako je predikcija točna, trebala težiti. Budući da je kretanje cijene dionice u pravilu nedeterminističko, ovaj problem nije moguće promatrati kao problem pretraživanja prostora stanja. Vrijednosti dionice nije moguće podijeliti u stanja koja se povezuju u sljedove. Najispravniji pristup problemu izrade algoritma za vršenje tehničke analize tržišta jest korištenje strojnog učenja.

2.3 Strojno učenje

Strojno učenje je oblik umjetne inteligencije koji koristi podatke i algoritme kako bi oponašao ljudski način učenja, postupno povećavajući točnost naučenog [5]. Ideja je na temelju dovoljno velikog skupa podataka „naučiti“ računalo uočavati pravilnosti u kretanju cijena dionica, kako bi na temelju aktualnih podataka računalo moglo predvidjeti cijenu dionice u budućnosti. Na slici 2.2 ilustrirana je predikcija cijene na temelju analiziranog vremenskog perioda.

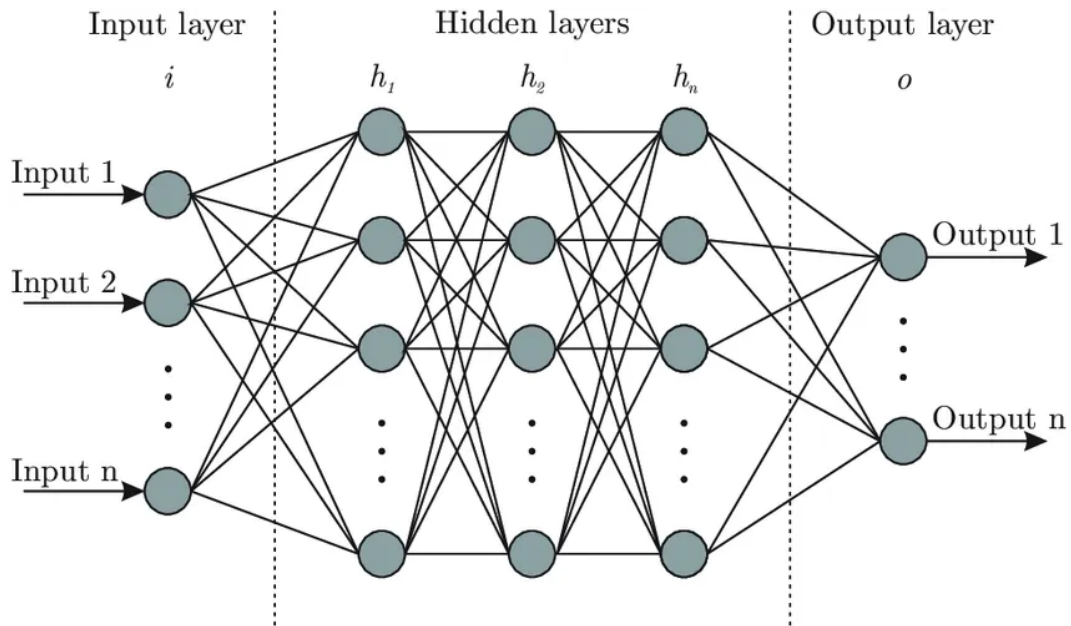


Slika 2.2: formiranje predikcije [4]

Ovakav pristup odstupa od tradicionalne tehničke analize, budući da se ne primjenjuju tipične tehnike i pravila. U konačnici, predikcija će biti donesena na temelju cijena u prethodno definiranom vremenskom okviru, npr. na temelju posljednjih mjesec dana. Za analizu podataka i predikciju rezultata korištene su neuronske mreže.

Neuronska mreža je model koji nastoji prepoznati osnovne veze u skupu podataka putem procesa koji oponaša način rada ljudskog mozga [6].

Zbog načela prepoznavanja veza u skupu podataka, neuronske mreže su prikladan alat za pristup problemu pronalaska uzoraka i pravilnosti u kretanju cijena dionica.



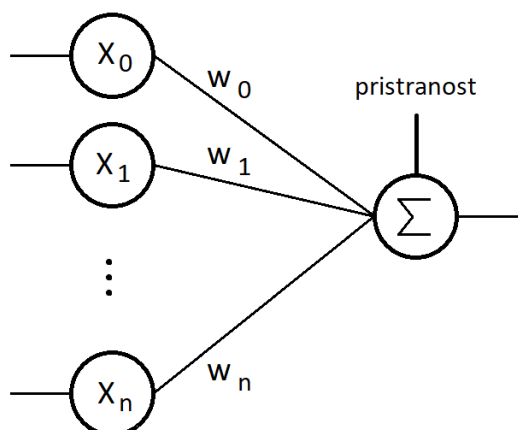
Slika 2.3: struktura neuronske mreže [7]

Postoji nekoliko vrsta neuronskih mreža, stoga je ključno odabrati vrstu prikladnu za analizu

nizova podataka, zbog čega su za analizu podataka korištene povratne neuronske mreže (engl. *recurrent neural networks*, skraćeno RNN). Povratne neuronske mreže koriste se za sekvencijsku analizu niza podataka i također ih je moguće podijeliti u nekoliko podvrsta. Vrsta povratnih neuronskih mreža najprikladnija za analizu vrijednosti cijena dionica je ćelija s dugoročnom memorijom (engl. *long short-term memory*, skraćeno LSTM) mreža. LSTM mreže pri analizi podataka uzimaju u obzir njihov redosljed, što je od presudne važnosti za mogućnost pronalaska pravilnosti u kretanju vrijednosti dionica.

3. Izrada modela neuronske mreže

Osnovna gradivna jedinica neuronskih mreža je, kao i u životinjskom mozgu, neuron. Umjetni neuroni međusobno su povezani težinskim vezama. Za razliku od životinjskog živčanog sustava, umjetne neuronske mreže ne mijenjaju arhitekturu, odnosno ne nastaju nove veze među neuronima prilikom učenja. Kod umjetnih neuronskih mreža, učenje se ostvaruje optimizacijom vrijednosti težina koje povezuju neurone što rezultira ojačavanjem ili oslabljivanjem međusobnih veza. Umjetni neuroni služe kao zbrajala umnoška težina veza s povezanim neuronima i izlaznih vrijednosti istih, uz nadodanu pristranost (engl. *bias*). Pristranost je vrijednost koja je dodijeljena svakom neuronu.



Slika 3.1: umjetni neuron

Umjetni neuron, kao jedinka, nema značajnu sposobnost učenja. Samostalno može naučiti isključivo linearne funkcije, što je dovoljno za rješavanje vrlo jednostavnih (linearnih) problema.

Umjetni neuroni svoj maksimalni potencijal ne ostvaruju kao izolirane jedinice, već kao dio cjeline sastavljene od više međusobno povezanih neurona. Osnovna cjelina grupiranih neurona naziva se sloj neuronske mreže. Iako neuroni u slojevima nisu međusobno povezani, veze ostvaruju s neuronima (uglavnom) susjednih slojeva. Više međusobno povezanih slojeva čini neuronsku mrežu.

3.1 LSTM mreže

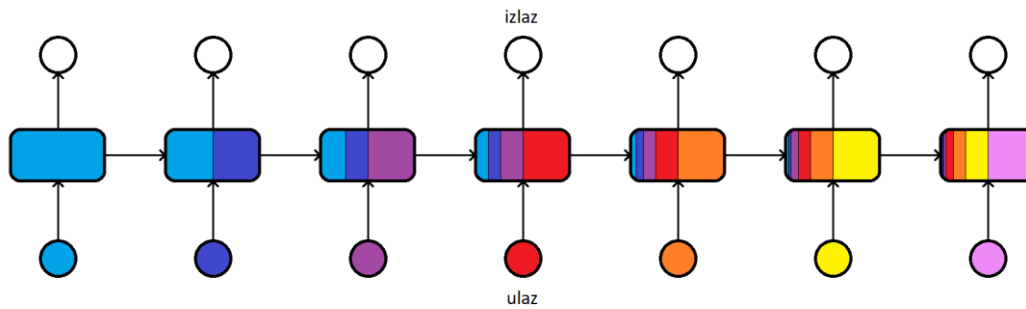
Kako bi bile ispravno ukomponirane u arhitekturu neuronske mreže, potrebno je dublje razumijevanje principa rada LSTM slojeva. Izraz LSTM sloj nije u potpunosti precizan. Pri implementaciji arhitekture modela, LSTM je moguće gledati kao cjelinu, odnosno sloj, iako se sastoji od nekoliko različitih slojeva, od kojih svaki ima zadaću nužnu za sekvencijsku analizu podataka. Niz slojeva koji kombinirani čine LSTM sloj, naziva se LSTM ćelija. Za lakše razumijevanje LSTM, potrebno je razumjeti što povratnim neuronskim mrežama omogućuje obradu niza podataka.

Povratne neuronske mreže, uz ulazne podatke, koriste „povratnu petlju“, odnosno izlaz jednog neurona prosljeđuje se na ulaz drugog neurona na sloju.



Slika 3.2: koncept povratnih neuronskih mreža

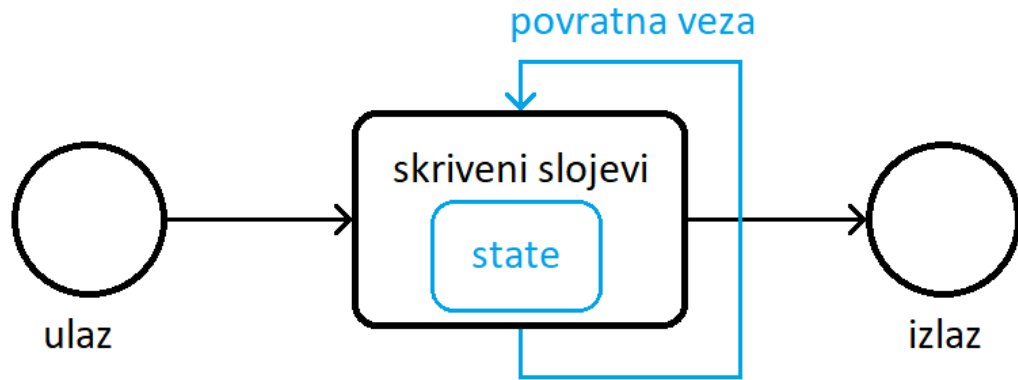
Na slici 3.2 ilustriran je koncept ponovnog korištenja podataka u povratnim neuronskim mrežama. U „klasičnim“ neuronskim mrežama, jedine ulazne vrijednosti koje neuroni primaju su izlazi prethodnog sloja pomnoženi s odgovarajućim težinama i vlastite pristranosti. Kod povratnih neuronskih mreža, nadolazeći slojevi primaju informacije o ulaznim podacima iz prethodnih slojeva, čime je omogućena obrada podataka na temelju ostalih ulaznih vrijednosti, što je upravo osnovno svojstvo povratnih neuronskih mreža. Prosljeđivanje informacija u nadolazeće slojeve prikazano je na slici 3.3.



Slika 3.3: prosljeđivanje ulaznih podataka u RNN

Kod obrade podataka u srednjem sloju, rezultat ovisi o prethodnim ulaznim podacima, ali samo do određene granice. Na slici 3.3, prvi ulazni podatak označen je plavom bojom. U svakom nadolazećem sloju, dio skrivenog sloja koji zauzima plava informacija je sve manji. Iako se radi o ilustraciji, isti efekt pojavljuje se kod implementacije povratnih neuronskih mreža. Efekt kod kojeg s rastom broja propagacija informacije slabi njen utjecaj te se posljedično skraćuje period pamćenja informacije naziva se nestajući gradijent (*engl. the vanishing gradient*).

Nestajući gradijent česta je pojava u povratnim neuronskim mrežama, koja ograničava njihovu primjenu. Opće povratne neuronske mreže iz tog razloga nije moguće izravno koristiti za probleme gdje tražena vrijednost ili klasa ovise o velikom broju slijednih ulaznih parametara, kao što je slučaj kod predikcije cijena dionica. Promjenom arhitekture povratnih neuronskih mreža, moguće je konstruirati mreže koje imaju mogućnost pamćenja većeg broja podataka. Takve mreže nazivamo mreže s dugim kratkoročnim pamćenjem (LSTM mreže). LSTM mreže su podskup povratnih neuronskih mreža i obje arhitekture obuhvaćaju međusobnu propagaciju ulaznih podataka među slojevima, međutim dodani slojevi neurona omogućuju LSTM mrežama čuvanje stanja (*engl. state*).



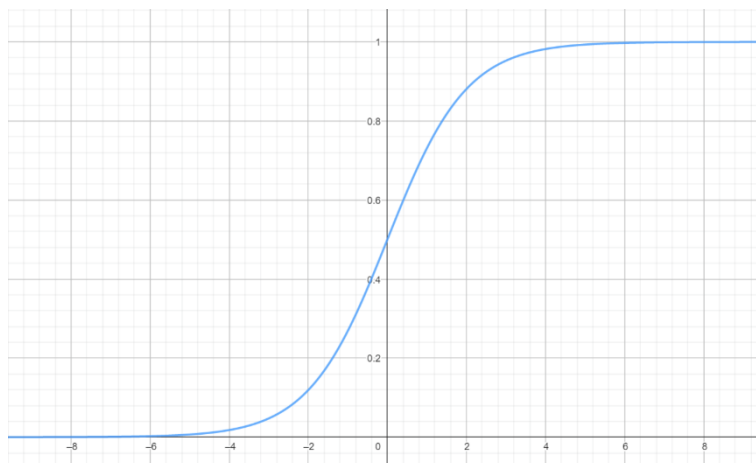
Slika 3.4: koncept ćelije s dugoročnom memorijom

Za razumijevanje arhitekture LSTM mreža, nužno je poznavanje pojma aktivacijske funkcije. Aktivacijska funkcija spaja se na izlaz neurona i služi za manipulaciju (preslikavanje) izlaznog signala. Ako je izlaz neurona označen simbolom x , aktivacijska funkcija primjenjuje se kao $f(x)$.

Jedna od najčešće korištenih aktivacijskih funkcija je funkcija sigmoide (*sig*, σ), koja izlaz preslikava u vrijednosti iz intervala $< 0, 1 >$.

$$f(x) = \frac{1}{1 + e^{-x}}$$

Jednadžba 3.1: funkcija sigmoide

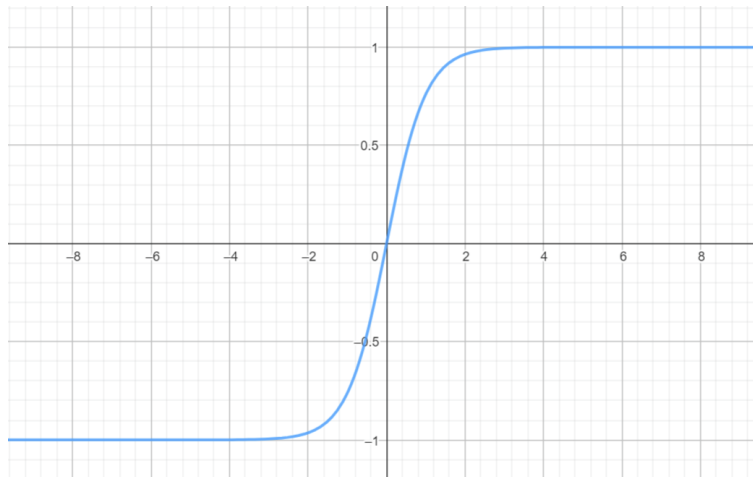


Slika 3.5: graf funkcije sigmoide

Model LSTM također koristi i funkciju tangens hiperbolni (\tanh), čija je kodomena interval $\langle -1, 1 \rangle$.

$$f(x) = \tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1}$$

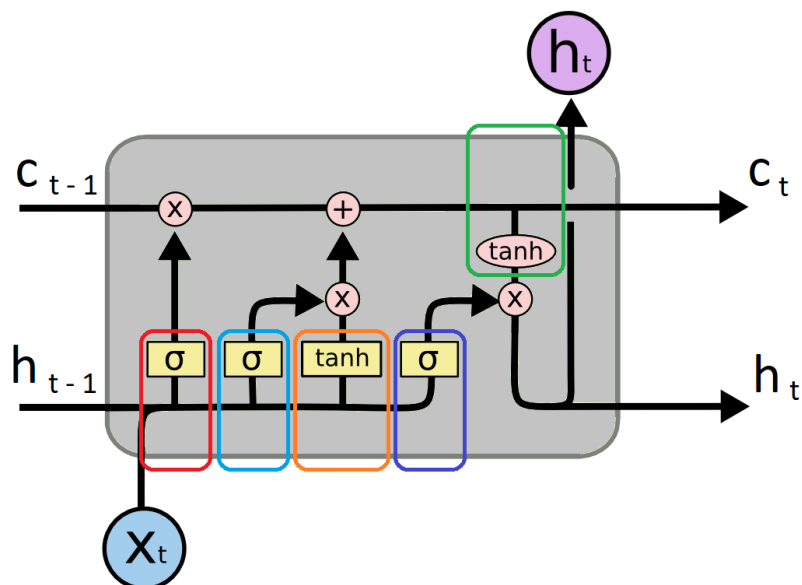
Jednadžba 3.2: funkcija tangens hiperbolni



Slika 3.6: graf funkcije tangens hiperbolni

Svrha aktivacijskih funkcija je skaliranje izlaznih vrijednosti neurona na male vrijednosti, kako ne bi došlo do nekontroliranog rasta vrijednosti. Ako neuronska mreža ima tri skrivena sloja koji se sastoje od svega jednog neurona i neka su vrijednosti svih težina jednake deset, za ulaznu vrijednost jednaku jedan, izlaz je reda veličine deset tisuća. Izlazi neuronskih mreža koje imaju vrlo velik broj neurona i skrivenih slojeva, neovisno o ulaznim vrijednostima, bili bi nezamislivo veliki brojevi, zbog čega je nužno izlaze skalirati na razumne vrijednosti.

Česta primjena funkcije sigmoide jest skaliranje vrijednosti kod klasifikacije ili donošenja binarne odluke (da / ne). Svaki član LSTM mreže koristi tri aktivacijske funkcije sigmoide kako bi donio odluke količini o podataka koji će se spremati, obrisati ili proslijediti na izlaz. U LSTM slojevima, funkcija $\tanh(x)$ koristi se za skaliranje izlaznih vrijednosti. Arhitektura LSTM sloja prikazana je na slici 3.7.



Slika 3.7: ćelija LSTM mreže [8]

Usredotočimo se na sliku 3.7. Simbol c označava liniju spremljenog stanja. Spremljeno stanje s indeksom c_t odnosi se na stanje koje će biti spremljeno na izlazu iz trenutne ćelije, dok je stanje c_{t-1} spremljeno stanje na ulazu u trenutnu ćeliju. Simbolom h označeni su izlazni podaci, što oznaku h_t čini izlazom iz trenutne ćelije, dok je h_{t-1} oznaka izlazne vrijednosti prethodne ćelije. Simbol X_t odnosi se na ulazni podatak trenutne ćelije (engl. *input*). Linija spremljenog stanja ključan je element LSTM mreža, jer se u nju spremaju ulazni podaci iz prethodnih ćelija, čime se uklanja efekt nestajućeg gradijenta. Najjednostavnije rečeno, linija spremljenog stanja ima ulogu „memorije“ LSTM mreže. Sve međusobno povezane LSTM ćelije dijele istu liniju spremljenog stanja.

Ulaz u svaku LSTM ćeliju čine ulazni podatak ćelije X_t i izlazni podatak iz prethodne LSTM ćelije h_{t-1} . Funkcija sigmoide označena crvenom bojom odlučuje u kojoj mjeri će se spremljeni podaci iz c_{t-1} odbaciti. Izlaz aktivacijske funkcije jednak nuli povlači potpuno odbacivanje do sada spremljenih podataka, dok izlazna vrijednost s iznosom jedan označava potpuno čuvanje podataka. Primijetimo da vrijednost funkcije sigmoide asimptotski pada prema vrijednosti nula i asimptotski raste prema vrijednosti jedan, što implicira da podaci nikada neće biti u potpunosti sačuvani ili obrisani. Opisano svojstvo zajedničko je svim aktivacijskim funkcijama sigmoide, tangensa hiperbolnog, ali i svih drugih funkcija s asimptotskim približavanjem konačnim vrijednostima.

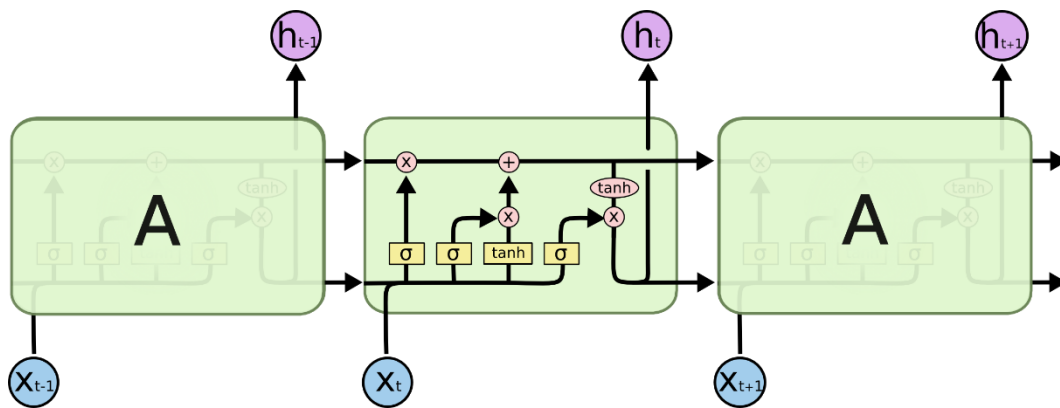
Plavo označena funkcija sigmoide i funkcija tangens hiperboli označena narančastom bojom zajedno utječu na podatke koji će u trenutnoj LSTM ćeliji biti spremljeni u liniju

spremljenog stanja. Funkcija sigmoide donosi odluku o udjelu ulaznih podataka koji će biti spremljeni u liniju spremljenog stanja. Ovisno o izlaznoj vrijednosti aktivacijske funkcije, podaci mogu biti u potpunosti odbačeni (vrijednost jednaka nuli), potpuno spremljeni (vrijednost jednaka 1) ili djelomično spremljeni (vrijednosti izlazne funkcije veće su od nula i manje od jedan, gdje je udio spremljenih podataka jednak vrijednosti izlazne funkcije). Nakon donošenja odluke o količini podataka koji će biti spremljeni, kako bi spremljene vrijednosti bile proporcionalne prethodno spremljenima, skaliranje podataka vrši funkcija tangensa hiperbolnog.

Zadnja aktivacijska funkcija sigmoide označena ljubičastom bojom filtrira podatke koji se prosljeđuju na izlaz trenutne ćelije h_t . Kriterij odabira količine spremljenih podataka jednak je onome iz prethodnog primjera (odabir količine podataka spremljenih u sabirnicu).

Funkcija tangensa hiperbolnog označena zelenom bojom skalira izlazne podatke na vrijednosti prikladne za obradu s ulaznim vrijednostima iduće LSTM ćelije. Objašnjenje principa rada LSTM ćelija inspirirano je člankom [9].

Poput neurona, LSTM ćelije se uglavnom ne koriste zasebno, već se međusobno nadovezuju. Primjer povezivanja više LSTM ćelija prikazan je na slici 3.8.



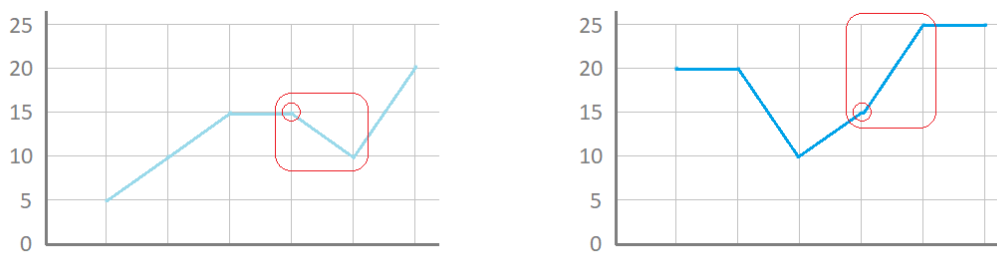
Slika 3.8: nadovezivanje ćelija LSTM mreže [8]

Zbog sposobnosti spremanja ulaznih podataka u zajedničku liniju spremljenog stanja c , LSTM mreže savršene su za analiziranje slijednih skupova podataka, kao što su cijene dionica, bez gubitka utjecaja vrijednosti s početka niza na krajnji rezultat.

3.2 Ulazni podaci neuronske mreže

Neuronske mreže imaju veliku prilagodljivost ulaznim podacima. Moguće je izgraditi arhitekturu neuronske mreže za proizvoljnu vrstu i oblik ulaznih podataka. Neuronska mreža na ulaznom sloju može primiti jedinstvenu numeričku vrijednost, ali i višedimenzionalne matrice, ovisno o potrebama problema.

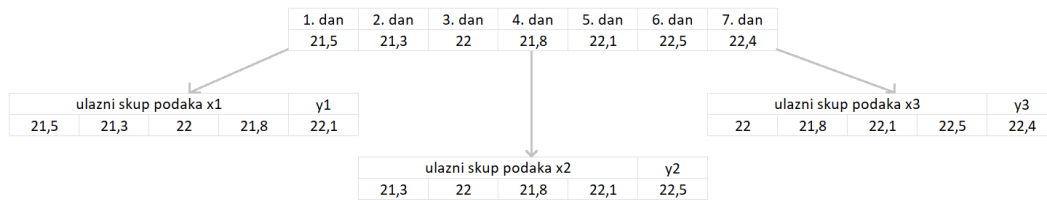
Prethodno je spomenuto da LSTM mreže kao ulaznu vrijednost primaju nizove podataka. Vrijednosti dionica prikazane su kao par numeričke vrijednosti (iznosa cijene) i datuma za koji je ta vrijednost vezana, odnosno kao cijena i trenutak u kojem je ta cijena bila aktualna. Jedan uređeni par cijene i datuma bez konteksta ne nosi nikakvu informaciju kod predviđanja cijene za idući dan. Na slici 3.9, možemo vidjeti dva slučaja gdje je u nekom trenutku cijena dionice bila jednaka 15, no u prvom slučaju (s lijeve stranice) vrijednost nastavlja padati, dok u drugom slučaju (s desne strane), iako je referentna vrijednost bila jednaka, vrijednost raste.



Slika 3.9: primjer manjka informacije koju nosi isključivo jedna vrijednost

Na slici 3.9 nisu navedeni datumi, ali primijetimo da u kontekstu problem, datumi nisu važni. Općenito u problemu predviđanja vrijednosti dionica, nećemo se oslanjati na datume, zbog toga što datum sam po sebi ne nosi relevantne informacije. Jedino važno svojstvo datuma jest pružanje informacije o redoslijedu vrijednosti cijena. Ako znamo redoslijed cijena, datum kao informaciju možemo zanemariti.

Slijed podataka odnosit će se na cijene dionica kroz jednako dugačak niz dana. Ako kao parametar mreži pružimo niz od deset vrijednosti, mreži smo pružili deset uzastopnih cijena iste dionice u vrijeme zatvaranja tržišta. Na slici 3.10 prikazan je primjer podjele skupa podataka za slučaj u kojem mreža kao ulazni parametar prima vrijednosti dionice kroz četiri različita dana.



Slika 3.10: podjela podataka ulaznog skupa

Nakon niza od četiri ulazna podatka, idući dan u nizu odnosi se na ciljnu (izlaznu vrijednost), odnosno na temelju četiri uzastopne vrijednosti, model će pokušati predvidjeti petu, izlaznu vrijednost.

Kod treninga modela, potrebno je odrediti duljinu niza ulaznih podataka koji će se uzimati u obzir pri predikciji cijene dionice na kraju nadolazećeg dana. Savršenu duljinu intervala (duljinu intervala koja će davati najbolje rezultate) nije moguće odrediti teorijski, jer nije poznato kako će različite arhitekture mreža reagirati na različite duljine ulaznih intervala te kako će se mijenjati odstupanje izlaznih vrijednosti mreže od očekivanih (stvarnih) vrijednosti. Jedini način kako pronaći optimalnu duljinu ulaznih vrijednosti jest provesti učenje mreže za različite duljine ulaznih nizova te usporediti dobivene rezultate.

Kompletan skup podataka (*engl.* dataset) koji će se koristiti za trening i obradu modela, potrebno je podijeliti u 3 dijela:

- podaci za trening (često 70% ukupnog skupa podataka)
- podaci za validaciju (često 20% ukupnog skupa podataka)
- podaci za testiranje (često 10% ukupnog skupa podataka)

Podaci za trening koriste se isključivo u periodu treninga modela, točnije koriste se za optimizaciju težina neuronske mreže. Podaci za validaciju koriste se za kontinuirane provjere napretka modela, dok se testni podaci koriste isključivo za provjeru finalnih rezultata modela i oni su mjerodavni pokazatelji točnosti modela. Od presudne je važnosti da model za vrijeme treninga ne koristi podatke za validaciju i testiranje. Ako dođe do suprotnog, model će imati priliku naučiti testne vrijednosti, zbog čega se gubi svojstvo generalizacije koja se ispituje na nepoznatim podacima i više neće biti moguće ispitati stvarne performanse modela.

3.3 Arhitektura neuronske mreže

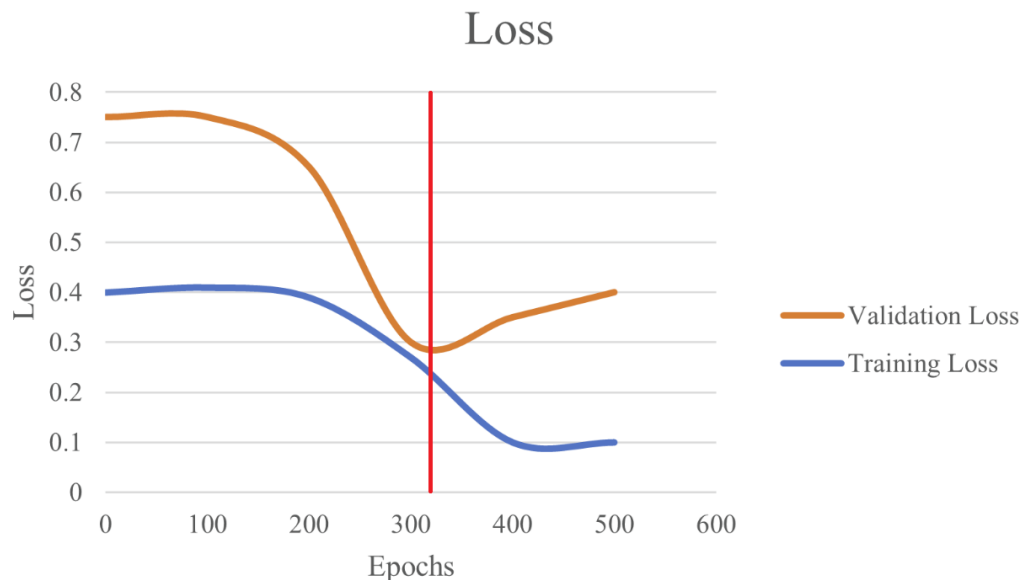
Neuronska mreža sastoji se od povezanih slojeva neurona. Neurone grupiramo i povezujemo kako bi povećali kapacitet neuronske mreže, odnosno njenu sposobnost učenja. Iako je intuitivna pomisao da je veća sposobnost učenja mreže poželjna, u praksi nije tako. Iznimne sposobnosti učenja neuronske mreže su ujedno najveća prednost, ali i najveći problem neuronskih mreža, jer kapacitet veći od zahtjevnog za rješavanje problema vodi do prenaučnosti.

Za razumijevanje pojma prenaučnosti, potrebno je upoznati se s pojmom treninga neuronske mreže. Fokus će biti isključivo na nadziranom učenju neuronskih mreža. Kod nadziranog učenja neuronskih mreža, uz ulazne podatke, mreži prosljeđujemo i očekivani izlaz. Time mreža može izračunati grešku nastalu propagacijom ulazne vrijednosti te na temelju dobivene greške, prilagoditi vrijednost težina. Neuronska mreža kao ulaz prima podatke oblika (x, y) gdje je x vektor ulaznih vrijednosti, a y je očekivana izlazna vrijednost. Primjer uređenog para za primjer učenja kvadratne funkcije $f(x) = x^2$ pruženog mreži je $(2, 4)$, jer je $4 = 2^2$. Trening neuronske mreže je proces optimizacije vrijednosti neuronske mreže na temelju skupa ulaznih podataka koji su namijenjeni za učenje (trening), s ciljem postizanja generalizacije mreže za rješavanje problema. Kao i prevelika složenost mreže, trening također može izazvati prenaučnost.

Prenaučnost je stanje pri kojem neuronska mreža previše prilagodi težine ulaznim podacima na kojima je trenirana, što posljedično vodi do loše generalizacije. Odsustvo generalizacije implicira nemogućnost rješavanja problema na neviđenim podacima, čime neuronska mreža postaje neprimjenjiva.

Ovisno o vrsti algoritma koji se primjenjuje pri strojnom učenju, postoje različite metode sprječavanja prenaučnosti. Metoda koja je univerzalna za sve algoritme, tako i neuronske mreže, je sprečavanje prenaučnosti validacijskim skupom podataka. Svakom epohom treninga, model će se sve više prilagoditi podacima za trening. Do određene granice, optimizacija težina vodi poboljšanju generalizacije modela. Nakon određenog broja epoha treninga (prolaskom modela kroz cijeli skup podataka za trening), model se više ne prilagođava općenitim podacima, već isključivo podacima za trening, što je signal prenaučnosti. Kako bismo utvrdili tu granicu, koristimo validacijski set. Ideja sprečavanja prenaučnosti provođenjem validacije je, nakon svake epohe treninga, model testirati na validacijskom skupu podataka. Za vrijeme prve faze treninga (inicijalnom optimiziranju težina), greške, odnosno odstupanje modela, smanjivat će se na podacima za testiranje, ali i

na skupu za validaciju. Kada model krene gubiti sposobnost generalizacije, greška na podacima za validaciju krenut će rasti, dok će se greška na testnim podacima nastaviti smanjivati, što je posljedica prenaučivosti. Na slici 3.11, crvenom bojom, označena je granica prenaučivosti.



Slika 3.11: granica prenaučivosti modela [10]

Granica prenaučivosti indikator je za zaustavljanje treninga. Kada je greška na validacijskom skupu najmanja, model najbolje generalizira. Trening modela uz paralelno testiranje točnosti na validacijskom setu naziva se metoda unakrsne provjere.

Kod konstrukcije arhitekture neuronske mreže, poželjno je voditi se načelom Occamove britve i konstruirati najjednostavniju mrežu pogodnu za rješavanje problema. Uz sprečavanje prenaučivosti, jednostavnije mreže koriste manje računalnih resursa i lakše ih je implementirati u druge sustave, kao što su mikrokontroleri.

Kod modela neuronske mreže, nije moguće znati njegove performanse, prije no što ga ispitamo. S namjerom pronalaska najbolje arhitekture mreže za zadani problem, uz testiranje modela s različitom duljinom ulaznih parametara, ispitano je nekoliko različitih arhitektura neuronskih mreža. Sve arhitekture izvedene su iz inicijalne arhitekture, ali međusobno se razlikuju u broju slojeva ili broju neurona u slojevima.

4. Implementacija modela

Modeli neuronskih mreža programirani su programskim jezikom „Python“ uz primjenu biblioteke „Tensorflow“ za implementaciju samih neuronskih mreža. Kao referentna arhitektura mreže, korištena je neuronsku mrežu iz članka [11].

```
model = keras.models.Sequential()
model.add(keras.layers.LSTM(units=64,
                             return_sequences=True,
                             input_shape=(x_train.shape[1], 1)))
model.add(keras.layers.LSTM(units=64))
model.add(keras.layers.Dense(32))
model.add(keras.layers.Dropout(0.5))
model.add(keras.layers.Dense(1))
```

Slika 3.12: referentna arhitektura modela neuronske mreže [11]

Slijedi objašnjenje poziva funkcija sa slike 3.12. Neuronska mreža konstruirana je pozivom funkcije *keras.models.Sequential()*, koji stvara sekvencijsku neuronsku mrežu u kojoj se propagacija podataka vrši sloj po sloj, redosljedom jednakim redosljedu dodavanja slojeva.

Svaki poziv funkcije *model.add(...)* označava dodavanje novog sloja u mrežu. Dva početna sloja neuronske mreže su LSTM slojevi, kod kojih argument *units=64* označava 64 LSTM prethodno opisane ćelije u LSTM sloju. *Dense* slojevi neuronske mreže su slojevi „običnih“ potpuno povezanih umjetnih neurona, kod kojih pruženi argument (31 i 1) označava broj neurona u sloju. Posljednja vrsta dodanog sloja je *Dropout* sloj, koji služi za dodatno sprečavanje prenaučivosti modela. Isključivo za vrijeme treninga, s frekvencijom jednakom argumentu (u ovom slučaju 0.5), *Dropout* postavlja ulazne vrijednosti sloja na 0, a ostale vrijednosti skalirane su s $\frac{1}{1-rate}$, gdje je *rate* argument. Skaliranje održava konačnu sumu izlaznih vrijednosti konstantnom, kako *Dropout* sloj ne bi imao utjecaj na konačne izlazne vrijednosti. Korištenje validacijskog seta i metode unakrsne provjere je snažna mjera zaštite od prenaučivosti, stoga u implementaciji modela nije korišten *Dropout* sloj, budući da nema značajan utjecaj na prenaučivost, čime se također smanjuje kompleksnost modela. Objašnjenja slojeva preuzeta su sa službene stranice „Keras-a“ [12].

4.1 Treniranje i testiranje modela neuronske mreže

U potrazi za modelom s najboljim performansama, konstruirana su četiri različita modela sličnih arhitektura. Preostala tri modela, bazirana su na nultom, početnom modelu, koji je vrlo sličan modelu iz članka [11], ali s odsustvom *Dropout* sloja.

U prvom modelu, prva dva sloja su LSTM slojevi, koji sadrže šezdeset i četiri LSTM ćelije. Nakon LSTM slojeva, slijede dva Dense sloja, prvi s trideset i dva, a idući s jednim neuronom u sloju.

Arhitektura drugog modela je vrlo slična arhitekturi prvog, ali umjesto jednog, druga mreža ima dva Dense sloja s trideset i dva neurona.

Treća LSTM mreža sadrži jednak broj i redoslijed slojeva kao i prva mreža, ali je broj neurona u svakom sloju dvostruko manji, osim u zadnjem Dense sloju, gdje je i dalje jedan neuron.

Četvrta mreža, poput treće, također dijeli broj i raspored slojeva s prvom mrežom, ali je broj neurona po sloju dvostruko veći za sve slojeve, osim posljednjeg Dense sloja, koji i dalje sadrži jedan neuron.

Svaki od modela treniran je i testiran s tri različite duljine niza ulaznih podataka. Korišteni su vremenski okviri duljine osam, šesnaest te trideset i dva podatka, točnije dana. Cilj testiranja različitih arhitektura modela na različitoj duljini ulaznih podataka je pronaći kombinaciju koja će dati model s najboljim performansama. Taj model biti će spremljen te implementiran u skriptu za predviđanje cijene dionica u stvarnom vremenu.

4.2 Izrada skripte za predikciju cijena dionica

Prethodno trenirani modeli testirani su isključivo na pripremljenim testnim podacima. Cilj izrade skripte je omogućiti korištenje modela u „stvarnom svijetu“ na aktualnim podacima. Modeli mogu analizirati pripremljene ulazne podatke i na temelju njih pokušati predvidjeti cijenu dionice za nadolazeći dan, ali za ispravan rad, modeli moraju biti u kontroliranom okruženju s obrađenim podacima. Zadatak skripte je omogućiti korisnicima koji ne razumiju rad i arhitekturu modela interakciju s istim preko jednostavne aplikacije komandnog retka.

4.2.1 Implementacija modela neuronske mreže u skriptu

Za svaku različitu duljinu niza ulaznih podataka (osam, šesnaest, trideset i dva) provodi se učenje četiri opisana modela neuronskih mreža. Po završetku učenja neuronskih mreža, točnost modela ispituje se na validacijskom skupu. Za svaku duljinu ulaznog niza, odabire se model s najboljim performansama. Točnost modela mjeri se srednjom kvadratnom pogreškom. Izračun srednje kvadratne pogreške (engl. *mean squared error*, skraćeno *mse*) prikazan je jednadžbom 4.1, gdje je n broj primjera u validacijskom skupu, $y_{ispravno}$ stvarna izlazna vrijednost za dani ulaz, a $y_{dobiveno}$ izlazna vrijednost neuronske mreže.

$$mse = \frac{1}{n} * \sum_{i=1}^n (y_{ispravno_i} - y_{dobiveno_i})^2$$

Jednadžba 4.1: izračun srednje kvadratne pogreške

Model s najmanjom srednjom kvadratnom pogreškom ispituje se na skupu podataka za testiranje. Srednja kvadratna pogreška modela na skupu podataka za testiranje odredit će koji model će biti implementiran u skripti. Greška na testnom skupu računat će se za model s najboljim performansama za određenu duljinu ulaznog niza podataka. Provođenjem navedenih testova, pronaći ćemo model koji daje najbolje rezultate s obzirom na arhitekturu mreže i duljinu ulaznih podataka.

4.2.2 Dohvaćanje aktualnih podataka

Polazni cilj izrade modela bio je izrada modela koji se može koristiti u stvarnom vremenu na aktualnim podacima, a ne isključivo na podacima iz prošlosti. Kako bi bilo moguće analizirati aktualne podatke, potrebno ih je dohvatiti. Aktualne podatke o cijenama dionica i fondova moguće je preuzeti s [13]. Podatke je moguće dohvatiti iz željenog vremenskog period i moguće ih je preuzeti u „csv“ obliku, što ih čini savršenima za analizu.

Kako korisnik skripte ne bi morao samostalno preuzimati tekuće cijene dionice ili fonda čiju bi cijenu htio predvidjeti, napravljena je zasebna skripta čija je zadaća preuzimanje i učitavanje tih podataka. Za izradu skripte, korištena je biblioteka „Selenium“, koja pruža

funkcionalnosti potrebne za otvaranje web mjesta i interakciju sa sučeljem umjesto korisnika. Kako bi usmjerio skriptu na ispravno web mjesto, korisnik mora unijeti simbol dionice ili fonda čija ga cijena zanima. Tablica 4.1 prikazuje imena i simbole nekih od najpoznatijih svjetskih tvrtki.

IME	Apple	Tesla	Amazon	Microsoft
SIMBOL	AAPL	TSLA	AMZN	MSFT

Tablica 4.1: imena tvrtki i njihovi simboli

Na temelju simbola, skripta izrađuje URL, koji otvara web mjesto na kojem je moguće preuzeti podatke o vrijednostima tvrtke ili fonda. Preuzimaju se podaci o cijenama iz prethodne godine.

4.2.3 Obrada podataka

Način obrade preuzetih podataka sličan je obradi ukupnog skupa podataka, no razlika je što kod preuzetih podataka ne postoji skup podataka za testiranje, jer model pokušava predvidjeti vrijednosti koje trenutno ne postoje.

Budući da je kretanje cijena dionica nedeterministički proces, model je potrebno upoznati s aktualnim kretanjem vrijednosti dionice, zbog čega se vrši trening modela na preuzetim podacima. Niz podataka sastavljen od vrijednosti dionice u posljednjih x dana, gdje x označava duljinu ulaznog niza za koju će modeli postići najbolje rezultate, koristit će se kao ulazni podatak za predviđanje cijene dionice na kraju nadolazećeg dana. S pretpostavkom da pogreška predikcije modela neće premašiti srednju apsolutnu pogrešku modela na validacijskom skupu, očekivana pogreška predikcije bit će izražena kao srednja apsolutna pogreška koju model ostvaruje na validacijskim podacima. Izračun srednje apsolutne pogreške (engl. *mean absolute error*, skraćeno *mae*) prikazan je jednadžbom 4.2, gdje je n broj primjera u validacijskom skupu, $y_{ispravno}$ stvarna izlazna vrijednost za dani ulaz, a $y_{dobiveno}$ izlazna vrijednost neuronske mreže.

$$mae = \frac{1}{n} * \sum_{i=1}^n |y_{ispravno_i} - y_{dobiveno_i}|$$

Jednadžba 4.2: izračun srednje apsolutne pogreške

4.2.4 Prikazivanje rezultata

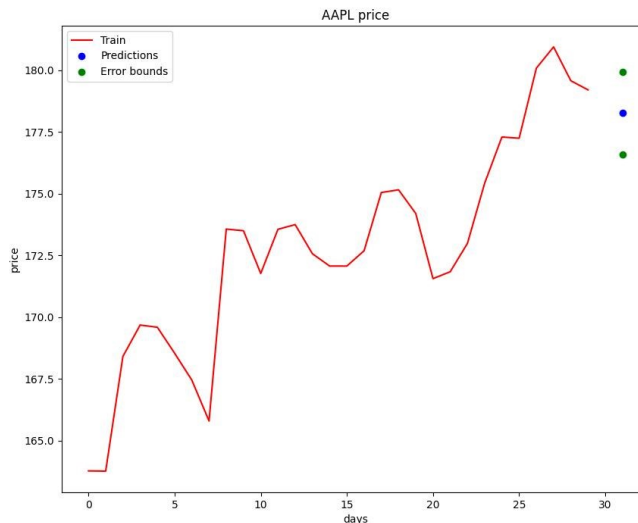
Dobivene rezultate potrebno je prikazati u korisniku razumljivom obliku. Skripta će rezultate prikazati numerički, ali i grafički.

Numerički prikaz rezultata sastojat će se od dobivene predikcije i srednje apsolutne pogreške izračunate na podacima za validaciju. Jednadžba 4.3 prikazuje oblik ispisa predikcije i očekivane apsolutne greške, gdje je $y_{dobiveno}$ predikcija neuronske mreže za nadolazeći dan, mae srednja apsolutna pogreška, a USD oznaka valute (američki dolar).

$$predikcija = y_{dobiveno} \pm mae \text{ USD}$$

Jednadžba 4.3: oblik prikaza predikcije i očekivane apsolutne greške

Grafički prikaz rezultata također prikazuje stvarne vrijednosti i predikcije, ali za razliku od numeričkih vrijednosti, pruža više informacija, kao što su vrijednosti cijene dionice u posljednjih trideset dana.



Slika 4.1: grafički prikaz predikcije cijene dionice

Nadolazeći opis odnosi se na sliku 4.1, koja prikazuje grafički prikaz cijene dionice. Os apscisa (x -os) označava dan na koji se vrijednost dionice odnosi, a os ordinata (y -os) prikazuje cijenu dionice u američkim dolarima (USD). Crvenom bojom označene su preuzete vrijednosti cijene dionice koje su prikazane za posljednjih trideset dana. Plavom

točkom prikazana je predviđena vrijednost cijene dionice za nadolazeći dan. Zelene točke označavaju najvišu (gornja točka) i najnižu (donja točka) očekivanu vrijednost. Najviša očekivana vrijednost dobivena je dodavanjem srednje vrijednosti apsolutne pogreške na vrijednost predikcije cijene dionice, dok je kod izračuna najniže očekivane vrijednosti pogreška oduzeta od vrijednosti predikcije. Budući da u trenutku pokretanja model pokušava predvidjeti cijenu dionice na kraju sutrašnjeg dana, trenutno ne postoji stvarna vrijednost s kojom je moguće usporediti predikciju, što je razlog ispisa očekivane pogreške.

Grafički prikaz daje uvid u odnos stvarnih cijena i vrijednost predikcije, no s grafa je nemoguće očitati točnu vrijednost predikcije za nadolazeći dan, zbog čega numeričkom vrijednošću nadopunjujemo grafički prikaz kako bismo dobili egzaktne vrijednosti greške i predikcije.

5. Analiza rezultata

Nakon što je svaki model prošao proces učenja, potrebno je usporediti rezultate dobivene na validacijskim setovima. Ispitivanje točnosti modela na podacima za validaciju i trening bit će mjerodavno za odabir najboljeg modela. Korišteni su jednaki skupovi podataka za učenje i ispitivanje svih modela, čime se iz procesa ispitivanja uklanja faktor sreće, koji bi postojao kod nasumične raspodjele podataka. Ako na istom testnom skupu prvi model griješi manje od drugog, prvi model je nedvojbeno i neporecivo bolji od drugog.

5.1 Ispitivanje modela

Za pronalazak modela neuronske mreže s najboljim performansama, točnije s najmanjom greškom, potrebno je međusobno usporediti rezultate dobivene učenjem modela za svaku veličinu skupa ulaznih podataka zasebno. Greška modela računa se kao srednja kvadratna pogreška na validacijskom skupu podataka. Kod prikazivanja rezultata, vrijednosti srednje kvadrante pogreške bit će prikazane s tri decimalna mjesta, čime se ne gubi preciznost rezultata, jer zaokružene decimale, u ovom slučaju, nisu značajne za odabir najboljih modela.

srednja kvadratna pogreška		duljina niza ulaznih podataka		
		8 dana	16 dana	32 dana
arhitektura modela	model 1	15767.153	411.827	391.917
	model 2	15709.884	409.73	396.294
	model 3	15792.945	409.442	404.455
	model 4	15815.532	408.399	406.822

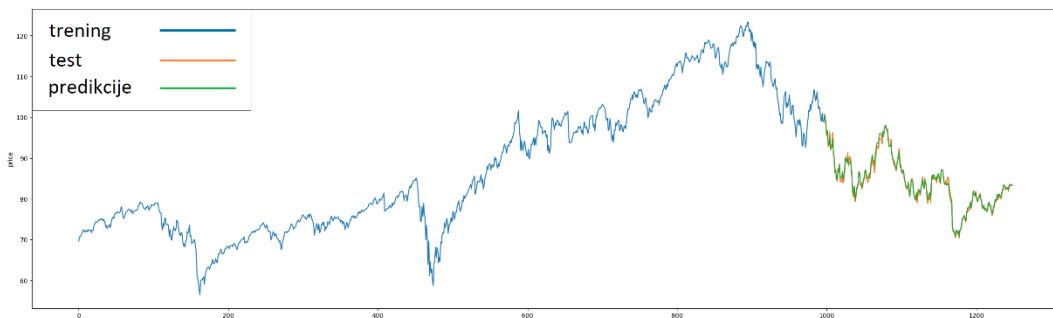
Tablica 5.1: prikaz iznosa greške modela na skupu podataka za validaciju

Rezultati ispitivanja mreža na skupu podataka za validaciju prikazani su u tablici 5.1. Zanimljivo je primijetiti da ne postoji općenito najbolja arhitektura neuronske mreže za ovaj problem, već rezultati svake arhitekture modela pojedinačno ovise o duljini ulaznog niza. Jedini model koji niti za jednu duljinu ulaznog niza imao najbolje rezultate je model broj tri, odnosno model s dvostruko manjim brojem neurona po sloju u odnosu na početni

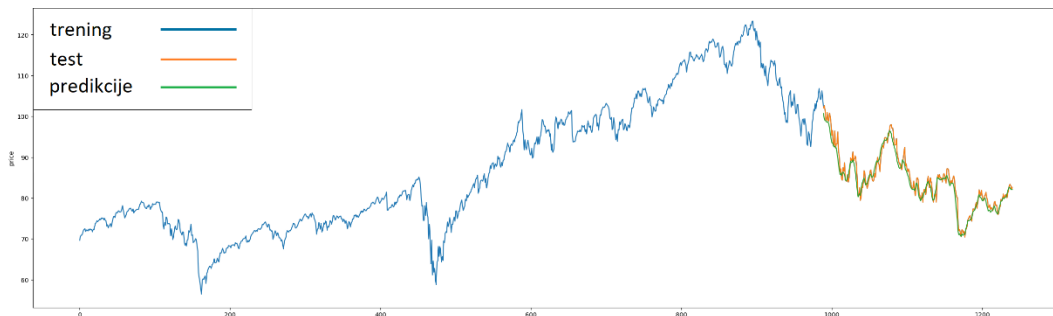
model, iz čega možemo povući zaključak da je arhitektura mreže prejednostavna za problem analize kretanja cijena dionica.

Nakon pronalaska najboljeg modela za svaku od tri veličine skupa ulaznih podataka, ispitat ćemo modele na ispitnim primjerima te usporediti dobivene rezultate kako bismo odabrali najbolji model.

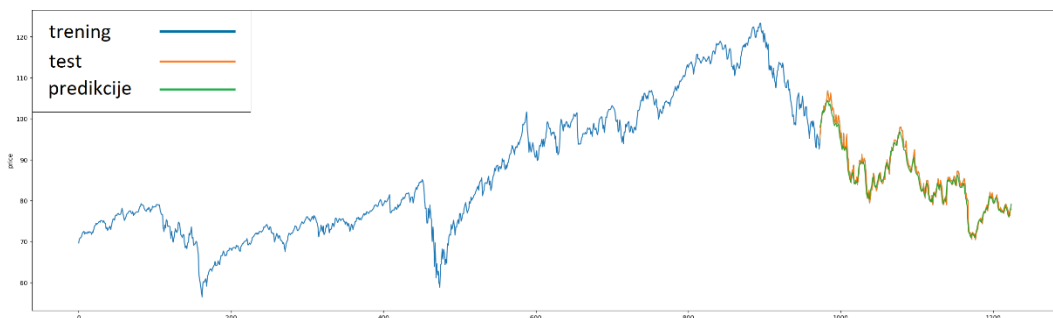
Prije same numeričke analize podataka, na slikama od 5.1 do 5.9, grafički su prikazane predikcije cijena za prethodno odabrane najbolje modele. Zbog velikog broja testnih primjera, grafički su prikazana ispitivanja modela na tri primjera.



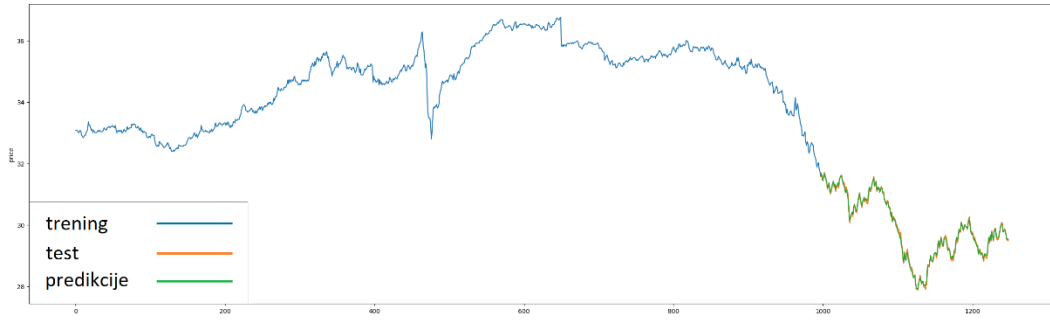
Slika 5.1: prvi testni primjer – model 2, 8 dana



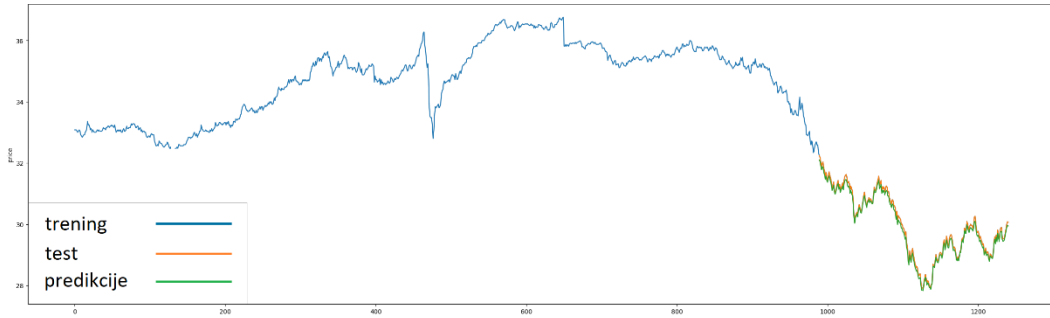
Slika 5.2: prvi testni primjer – model 4, 16 dana



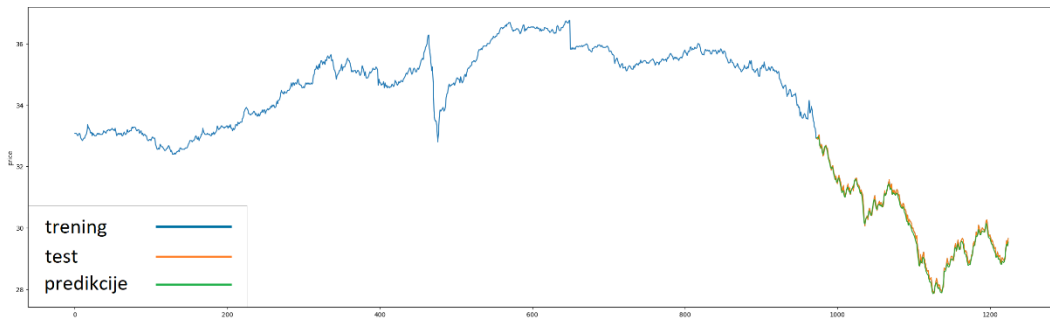
Slika 5.3: prvi testni primjer – model 1, 32 dana



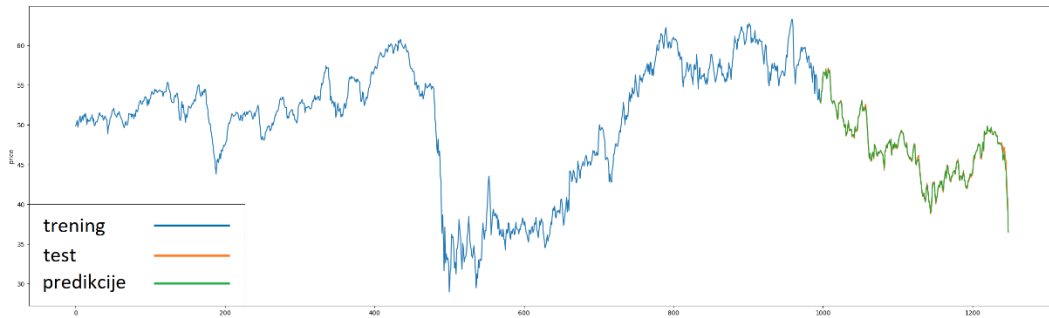
Slika 5.4: drugi testni primjer – model 2, 8 dana



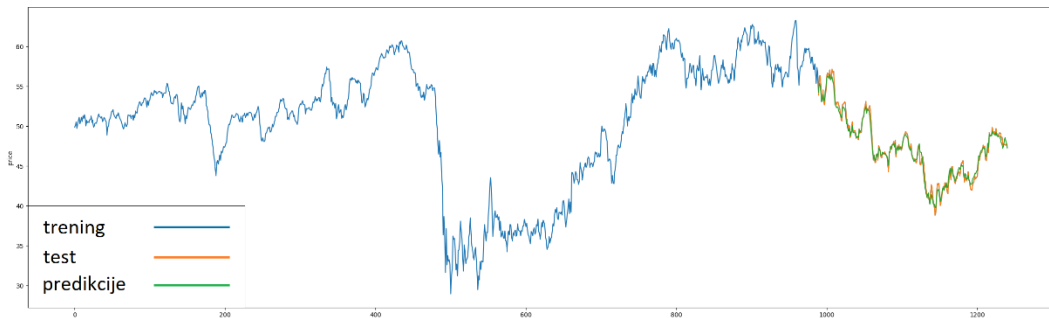
Slika 5.5: drugi testni primjer – model 4, 16 dana



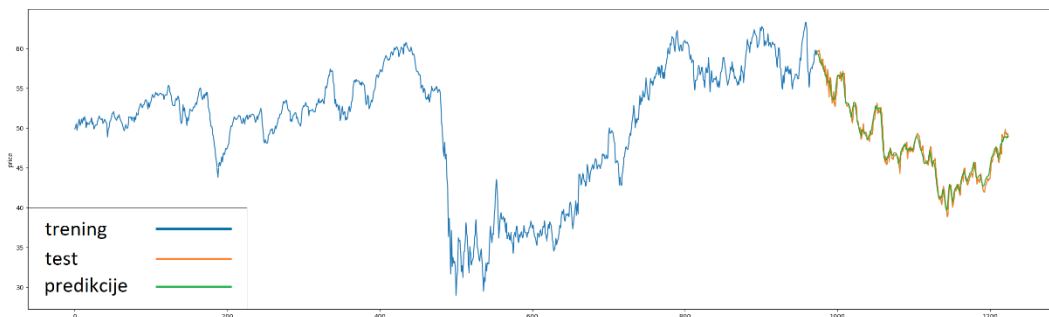
Slika 5.6: drugi testni primjer – model 1, 32 dana



Slika 5.7: treći testni primjer – model 2, 8 dana



Slika 5.8: treći testni primjer – model 4, 16 dana



Slika 5.9: treći testni primjer – model 1, 32 dana

Na slikama od 5.1 do 5.9, plavom bojom prikazani su stvarni podaci na kojima je model učen, narančastom bojom prikazani su stvarni podaci na kojima je model ispitan (ti podaci nisu poznati modelu, odnosno nije učen na njima) i zelena boja označava izlazne vrijednosti modela, odnosno predikcije cijena. Iako grafički prikaz može pomoći pri vizualizaciji rezultata, nemoguće je očitavanjem grafova s preciznošću reći koji model je najbolji. Kako bismo bili sigurni u odabir najboljeg modela, izračunat ćemo srednju kvadratnu pogrešku za testne primjere i rezultate prikazati numerički.

srednja kvadratna pogreška	duljina niza ulaznih podataka		
	8 dana	16 dana	32 dana
najbolji model	65.352	68.706	64.807

Tablica 5.2: prikaz iznosa greške modela na skupu podataka za validaciju

Iz rezultata u tablici 5.2 vidimo da sva tri modela daju slične rezultate te da nema velikih odstupanja među iznosima grešaka. Međutim, valja primijetiti kako najdulja i najkraća duljina ulaznog niza (32 i 8 dana) daju najslabije rezultate, s odstupanjem iznosa svega 0.545, dok predikcije koje koriste srednju duljinu niza (16 dana) imaju najveće odstupanje od stvarnih vrijednosti.

Na temelju vrijednosti srednjih kvadratnih pogrešaka najboljih modela (za različite duljine ulaznog niza) prikazanih u tablici 5.2, zaključujemo da je početni model (model 1) najbolji model te da su njegove performanse najbolje za duljinu ulaznog niza podataka jednaku trideset i dva. Spomenuti model bit će implementiran u skriptu.

5.2 Primjer korištenja skripte za predikciju cijena dionica

Cilj ovog poglavlja jest prikazati primjer korištenja skripte za predikciju cijena dionica, interakciju skripte s korisnikom i dobivene rezultate.

Pri pokretanju skripte, u konzoli se ispisuje poruka prikazana na slici 5.10.

datotekom naziva „AAPL (1).csv“, zbog čega će u oba slučaja skripta obrađivati prve skinute podatke, jer skripta pretražuje isključivo datoteku „AAPL.csv“. Brisanje preuzetih podataka nakon učitavanja istih u skriptu sprječava dupliciranje datoteka.

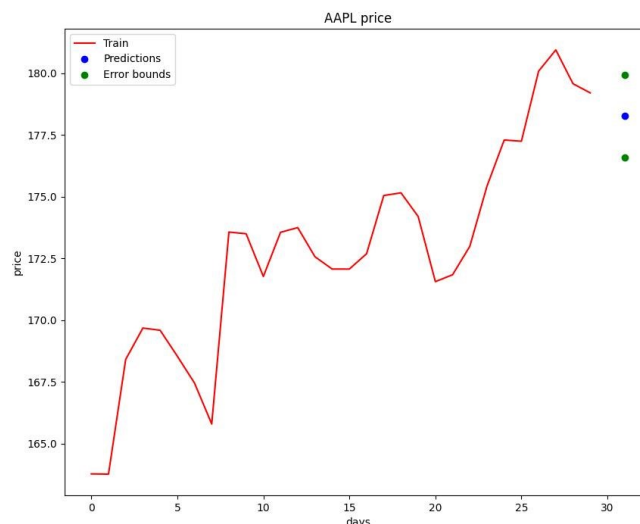
Završetkom učenja, model vraća četrnaest predikcija, od kojih se prvih trinaest koristi za računanje greške, a posljednja predikcija odnosi se na nadolazeći dan. Prvi dio prikazivanja rezultata jest ispis numeričke vrijednosti predikcije prikazan na slici 5.12.

```
Predicted closing value of AAPL stock for the next day: 178.26811±1.6618343591690063 USD  
  
Stock price has been predicted successfully!  
The prediction is stored in the 'AAPL_price_prediction.jpg' file.
```

Slika 5.12: ispis vrijednosti predikcije

Uz ispis numeričke vrijednosti predikcije, na slici 5.12 prikazana je poruka o spremanju datoteke „AAPL_price_prediction.jpg“. Navedena datoteka sadrži grafički prikaz predikcije cijene navedene dionice. Svi grafički prikazi spremaju se u mapu naziva „Images“. Ako u spomenutoj mapi već postoji datoteka imena „AAPL_price_prediction.jpg“, ona se prepisuje novom datotekom.

Grafički prikaz predikcije koji nadopunjuje numeričke vrijednosti sa slike 5.12 prikazan je na slici 5.13.



Slika 5.13: grafički prikaz vrijednosti predikcije

Prethodno je spomenuta mogućnost korisnikovog unosa nevažećeg (nepostojećeg) simbola dionice. U tom slučaju, skripta prepoznaje nevažeći unos i o njemu obavještava korisnika porukom prikazanom na slici 5.14.

```
Enter the valid ticker symbol of the stock you want to predict:  
>>> Predikcija Cijene Dionica  
The passed ticker symbol is invalid.  
Please try again.  
  
Enter the valid ticker symbol of the stock you want to predict:  
>>>
```

Slika 5.14: ispis poruke o nevažećem simbolu dionice

Uz ispis poruke, korisniku je omogućen ponovni unos simbola dionice, sve dok ne unese ispravni simbol. Po ispravnom unosu, skripta ispisuje dobivene rezultate i završava s radom.

Budući rad

Iako je u sklopu rada ispitano nekoliko različitih arhitektura modela neuronskih mreže i duljina nizova ulaznih podataka, za pronalaženje optimalnog modela potrebno je ispitati još velik broj modela s različitim arhitekturama, modela s ili bez aktivacijskih funkcija te modela s različitim brojevima neurona u slojevima.

Također, modeli različitih arhitektura reagirali su drugačije na iste duljine nizova ulaznih podataka. Model x može davati najbolje rezultate za ulazne nizove s osam podataka, dok model y može najbolje performanse imati s ulaznim nizovima koji sadrže dvanaest podataka.

Performanse modela nemoguće je predvidjeti ili unaprijed izračunati, zbog čega je za pronalazak najboljeg modela nužno provesti iscrpno ispitivanje svih modela na različitim duljinama ulaznih podataka.

Dodatan faktor kod performansi neuronskih mreža predstavlja veličina skupa podataka za učenje neuronskih mreža. Podatke je potrebno konstantno ažurirati kako bi model mogao pratiti aktualne trendove tržišta.

Postojeći model moguće je dograditi modelom koji proučava utjecaje vanjskih ili unutarnjih faktora poslovanja tvrtke. Jedno od rješenja može obuhvaćati pretragu novinskih članaka vezanih uz promatranu tvrtku te utjecaj ključnih riječi ili fraza kao što su „pad“, „bankrot“, „profit“ i sl. na vrijednost cijene dionice iste.

6. Zaključak

Cilj ovog rada bio je izraditi model neuronske mreže s funkcionalnošću predviđanja cijene dionica te implementacija istog u skriptu koja će korisnicima omogućiti lako korištenje modela, bez obzira na njihovo poznavanje tehničke domene neuronskih mreža. Skripta u potpunosti automatizira proces preuzimanja, dohvaćanja i obrade podataka, čime jedina zadaća korisnika ostaje unos simbola dionice tvrtke ili fonda čiju bi cijenu htio predvidjeti. Radi ostvarivanja maksimalne preciznosti uz mogućnost vizualizacije podataka, skripta rješenje ispisuje u obliku numeričke vrijednosti s očekivanom greškom temeljenom na odstupanjima prethodnih predikcija, ali i u grafičkom obliku koji pokazuje odnose cijena dionice u prethodnih trideset dana, vrijednosti predikcija te očekivane apsolutne greške.

Proces izrade modela neuronske mreže obuhvaćao je konstruiranje, učenje i ispitivanje različitih arhitektura neuronskih mreža koje koriste ćelije s dugoročnom memorijom (LSTM ćelije) sa zadaćom pronalaženja pravilnosti i pamćenja uzoraka u kretanju vrijednosti cijena dionica. Svaka arhitektura neuronske mreže trenirana je i ispitana s različitim duljinama niza ulaznih podataka. Svrha proučavanja performansi modela za različite duljine ulaznog niza podataka bila je pronalazak optimalne količine ulaznih podataka s kojima model postiže najbolje rezultate, točnije s kojom model ostvaruje najmanju grešku na testnom skupu podataka. Kriterij odabira najboljeg modela bila je najmanja ostvarena srednja kvadratna pogreška na ispitnom skupu primjera.

Krajnji cilj rada nije bio proizvesti financijskog stručnjaka kojem je moguće povjeriti sav kapital raspoloživ za ulaganje, već ispitati mogućnosti i rezultate koje povratne neuronske mreže mogu ostvariti na domeni analize tržišta dionica. Na kraju krajeva, kretanje vrijednosti dionica nepredvidiv je proces proporcionalan uspješnosti promatrane tvrtke na koju utječu brojni vanjski faktori, koji ne uključuju povijesne cijene dionica. Poslovanje tvrtke kroz posljednjih petnaest godina ne daje nikakve informacije o tome kako će ista poslovati u nadolazećoj godini. Usprkos tome, na temelju povijesnih kretanja vrijednosti moguće je uočiti pravilnosti kod reakcija investitora na pojedine intervale promjena cijena, što je sam temelj tehničke analize dionica.

Literatura

- [1] A. Hayes, »www.investopedia.com,« Investopedia, 21. travanj 2023.. [Mrežno]. Available: www.investopedia.com/terms/s/stock.asp. [Pokušaj pristupa 29. svibanj 2023.].
- [2] J. Salvucci, »www.thestreet.com,« TheStreet, 20. ožujak 2023.. [Mrežno]. Available: www.thestreet.com/dictionary/f/fractional-shares. [Pokušaj pristupa 29. svibanj 2023.].
- [3] A. Farley, »www.investopedia.com,« Investopedia, 13. travanj 2022.. [Mrežno]. Available: www.investopedia.com/articles/active-trading/022715/20-rules-followed-professional-traders.asp. [Pokušaj pristupa 29. svibanj 2023.].
- [4] »finance.yahoo.com,« Yahoo Finance, 30. svibanj 2023.. [Mrežno]. Available: finance.yahoo.com/quote/AAPL/chart. [Pokušaj pristupa 30. svibanj 2023.].
- [5] »www.ibm.com,« IBM, [Mrežno]. Available: <https://www.ibm.com/topics/machine-learning>. [Pokušaj pristupa 31. svibanj 2023.].
- [6] J. Chen, »www.investopedia.com,« Investopedia, 30. travanj 2023.. [Mrežno]. Available: <https://www.investopedia.com/terms/n/neuralnetwork.asp>. [Pokušaj pristupa 31. svibanj 2023.].
- [7] A. Bhardwaj, »iaviral.medium.com,« Medium, 6. srpanj 2021.. [Mrežno]. Available: iaviral.medium.com/what-are-neural-networks-an-introduction-to-machine-learning-algorithms-6b73383c9089. [Pokušaj pristupa 31. svibanj 2023.].
- [8] S. Dobilas, »towardsdatascience.com,« TowardsDataScience, 6. veljača 2022.. [Mrežno]. Available: towardsdatascience.com/lstm-recurrent-neural-networks-how-to-teach-a-network-to-remember-the-past-55e54c2ff22e. [Pokušaj pristupa 31. svibanj 2023.].
- [9] C. Olah, »colah.github.io,« Colah's blog, 27. kolovoz 2015.. [Mrežno]. Available: colah.github.io/posts/2015-08-Understanding-LSTMs/. [Pokušaj pristupa 31. svibanj 2023.].
- [10] Baeldung, »www.baeldung.com,« Baeldung, 16. ožujak 2023.. [Mrežno]. Available: www.baeldung.com/cs/training-validation-loss-deep-learning. [Pokušaj pristupa 1. lipanj 2023.].
- [11] jaintarun, »www.geeksforgeeks.org,« GeeksForGeeks, 4. travanj 2023.. [Mrežno]. Available: www.geeksforgeeks.org/stock-price-prediction-project.

using-tensorflow/. [Pokušaj pristupa 1. lipanj 2023.].

[12] »keras.io,« Keras, [Mrežno]. Available: keras.io/. [Pokušaj pristupa 1. lipanj 2023.].

[13] »finance.yahoo.com,« Yahoo, [Mrežno]. Available: finance.yahoo.com. [Pokušaj pristupa 2. lipanj 2023.].

[14] »finance.yahoo.com,« Yahoo, [Mrežno]. Available: finance.yahoo.com/quote/AAPL/history?p=AAPL. [Pokušaj pristupa 3. lipanj 2023.].

Procjena kretanja cijena dionica neuronskim mrežama

Sažetak

Rad obuhvaća izradu modela neuronske mreže koja koristi ćelije s dugoročnom memorijom (LSTM ćelije) za pronalaženje pravilnosti i uzoraka u kretanju cijena dionica. Dobiveni model implementiran je u skriptu koja, nakon što korisnik unese simbol dionice ili fonda čiju cijenu želi predvidjeti, preuzima podatke o aktualnim cijenama, obrađuje ih te predikcije prikazuje u numeričkom i grafičkom obliku. S ciljem pronalaska modela čije predikcije imaju najmanja odstupanja od stvarnih cijena, ispitane su različite arhitekture neuronskih mreža, kao i različite duljine ulaznih podataka. Dok je model učen i ispitan isključivo na povijesnim podacima, skripta model koristi za rad s tekućim vrijednostima.

Ključne riječi: Predikcija cijena dionica, LSTM, neuronske mreže

Predicting stock price movements with neural networks

Abstract

The paper includes the construction of a neural network model that uses cells with long short-term memory (LSTM cells) to find regularities and patterns in stock price movements. The obtained model is implemented into a script that, after the user enters the symbol of the stock or fund whose price he wants to predict, downloads data on current prices, processes it, and displays the predictions in numerical and graphical format. To find a model whose predictions have the smallest deviations from the actual values, different architectures of neural networks were tested, as well as different lengths of input data. While the model was trained and tested exclusively on historical data, the script uses the model to work with current values.

Keywords: Stock price prediction, LSTM, neural networks