

---

# Veštačka inteligencija

---

Veštačka inteligencija je oblast računarstva koja je danas u velikoj ekspanziji. Skoro da nema oblasti ljudske delatnosti u kojoj veštačka inteligencija ne nalazi primene.

Veštačka inteligencija oslanja se na oblasti algoritmike, matematičke logike, numeričke matematike, matematičke analize, verovatnoće, statistike, itd.

Veštačka inteligencija kao informatička disciplina ustanovljena je na konferenciji *The Dartmouth Summer Research Conference on Artificial Intelligence* u Darmutu (Sjedinjene Američke Države), 1956. godine. Tom prilikom predloženo je, od strane Džona Makartija, i samo ime discipline, ne sasvim srećno, jer je to ime često izazivalo nedumice i podozrenje. Tokom godina koje su sledile, bilo je perioda optimizma i intenzivnog razvoja, ali i perioda pesimizma i opadanja interesovanja. Početkom dvadeset prvog veka javio se talas izuzetno uspešnih sistema zasnovanih na „dubokom učenju“ – sistema koji su uspešno vršili prepoznavanje lica na fotografijama, prevođenje prirodnih jezika, navigaciju vozila i drugo. Ovaj uzlet praćen je velikim probojima 2020-ih godina u oblastima obrade prirodnog jezika (poput sistema ChatGPT) i računarskog vida (poput sistema DALL-E). Sada, dakle, traje period izuzetnog razvoja veštačke inteligencije i on će sigurno još potrajati.

Ne postoji jedna, opšteprihvaćena definicija veštačke inteligencije, ali se pod njom obično podrazumeva sposobnost mašinskog usvajanja, pamćenja i obrade određenih znanja. Većina metoda veštačke inteligencije usmerena je na uske, konkretne oblasti primene. Ipak, u poslednje vreme javljaju se i podoblasti veštačke inteligencije koje imaju za cilj opšte rasuđivanje u stilu čoveka.

Pomenimo dve, verovatno ključne, teme veštačke inteligencije:

- rešavanje problema u kojima se javlja *kombinatorna eksplozija*, tj. u kojima je broj mogućnosti toliko veliki da se ne može sistematično, tj. iscrpno ispitati u razumnom vremenu. Jedna takva oblast je, na primer, igranje igara kao što je šah.
- *zaključivanje*, kada je cilj iz raspoloživih podataka kreirati neke nove uvide, nova znanja. Jedan vid zaključivanja je *deduktivno zaključivanje* – to je zaključivanje zasnovano na rigoroznom matematičkom rasuđivanju i ide od opšteg ka pojedinačnom. Primer takvog problema<sup>1</sup> je izračunavanje dijagonale nekog pravougaonika – Pitagorina teorema daje opštu vezu između stranica pravouglog trougla i ona se može primeniti za izračunavanje dijagonale bilo kog konkretnog pravougaonika. Drugi vid zaključivanja je *induktivno zaključivanje* – to je zaključivanje zasnovano na mnoštvu raspoloživih pojedinačnih podataka iz kojih se *generalizacijom* mogu kreirati neki novi uvide, nova znanja. Primer takvog problema je prepoznavanje vrste životinje na slici, a na osnovu hiljada slika za koje je već poznato šta prikazuju.

Mnoge metode veštačke inteligencije imaju sličnu opštu strukturu procesa rešavanja problema. Faze rešavanja obično su: modelovanje, tj. opisivanje zadatog problema u strogim, matematičkim terminima; rešavanje problema opisanog u matematičkim terminima; interpretiranje i analiza rešenja. Dublja priroda ovih faza razlikuje se između različitih podoblasti veštačke inteligencije.

## 1.1 Uska i opšta veštačka inteligencija

Dominantan pravac razvoja veštačke inteligencije dugo je bio razvoj sistema specijalizovanih za konkretne zadatke. To je pravac koji se naziva „uska veštačka inteligencija“. Alternativa je „veštačka opšta inteligencija“ (eng. *artificial general intelligence*, *AGI*). Njen cilj je razvoj meta-algoritama koji mogu da pokreću sistem

---

<sup>1</sup>Kada se govori o „problemu“, obično se misli na čitavu klasu srodnih zadataka. Pojedinačne zadatke koji pripadaju ovakvim klasama zovemo *instance problema* (ili *primerci problema*).

sposoban da uči, rasuđuje i rešava sve probleme koje može i čovek. Taj cilj neki istraživači pokušavaju da postignu matematičkim modelovanjem ljudskog mozga, iako mnogi smatraju da je to teško dostižno ili nemoguće. Istraživač i futurista Rej Kercvajl (Raymond Kurzweil, trenutno radi u kompaniji Google), u svojoj uticajnoj knjizi „Singularnost je blizu” iz 2005. godine, tvrdi da nije daleko trenutak kada će računari prevazići čoveka u svim intelektualnim aktivnostima i procenjuje da će računari već 2029. godine moći da prođu Tjuringov test (videti poglavlje 1.2), što će, dalje, oko 2045. godine dovesti do „sveopšteg preokreta u ljudskim mogućnostima“. Skorašnji prodori u ovoj oblasti učvrstili su mnoge u sličnim uverenjima.

## 1.2 Filozofski i etički aspekti veštačke inteligencije

Neke podoblasti veštačke inteligencije imaju za cilj oponašanje prirodne (ljudske ili životinjske) inteligencije. No, postavlja se pitanje šta je uopšte *inteligencija*. Možemo smatrati da inteligencija podrazumeva sledeće sposobnosti: sposobnost pamćenja, skladištenja znanja i mogućnost njegove obrade, sposobnost učenja – usvajanja novih znanja, sposobnost komunikacije sa drugim inteligentnim bićima ili mašinama, itd. Može se smatrati, dakle, da biće ili mašina imaju attribute inteligentnog, ako imaju navedena svojstva. Alen Tjuring (Alan Turing) formulisao je sledeći znameniti test: ako su u odvojene dve prostorije smeštene jedna ljudska osoba i neki uređaj i ako na identična pitanja pružaju odgovore na osnovu kojih se ne može pogoditi u kojoj sobi je čovek, a u kojoj uređaj, onda možemo smatrati da taj uređaj ima attribute veštačke inteligencije. Tjuring je 1947. godine, govoreći o računarima koji simuliraju čovekovo rasuđivanje, predložio i kreiranje mašina „programiranih da uče i kojima je dopušteno da čine greške” jer, kako kaže, „ako se od mašine očekuje da bude nepogrešiva, onda ona ne može biti inteligentna“. Ovakav pristup, koji dozvoljava greške, karakterističan je za mašinsko učenje. U vezi sa otvorenim pitanjem šta je uopšte ljudska inteligencija (ako želimo da je oponašaju računari), zanimljiv je i stav fon Nojmana: „ako mi preceizno opišete šta je to što ne može da uradi mašina, onda ću ja moći da napravim mašinu koja će raditi upravo to!“ Ovaj stav odnosi se na to da za svaku obradu podataka koju možemo da opišemo u terminima matematičkih izračunavanja, postoji računarski program koji može da je vrši.

Mnogi problemi veštačke inteligencije mogu se opisati u matematičkim terminima. Pitanje je za koje klase problema postoje opšti načini rešavanja (koje mogu da primene ljudi ili mašine). Rezultati Gedela i Tjuringa iz prve polovine dvadesetog veka pokazali su da postoje matematičke teorije (uključujući i jednostavne teorije kakva je aritmetika) koje su nepotpune i neodlučive. Na primer, postoje tvrđenja o prirodnim brojevima koja su tačna, ali se ne mogu dokazati iz aksioma aritmetike. Štaviše, skup aksioma aritmetike nije moguće proširiti tako da se to promeni. Dodatno, ne postoji algoritam koji može da dokaže svako aritmetičko tvrđenje koje jeste dokazivo. Ovi rezultati govore da za neke probleme ne mogu da postoje mašine koje mogu da reše svaku njihovu instancu, te da preostaje da se njima bave kao i ljudi: da koriste svojstva i zapažanja specifična za konkretne date instance. Drugim rečima, za instance ovakvih problema traganje za rešenjem neće uvek biti isto i neće garantovati uspeh.

Pored filozofskih pitanja o tome gde su granice ljudske i veštačke inteligencije, važna su i etička pitanja koja se odnose na mašine koje mogu da samostalno donose nekakve odluke. Još od ranih dana veštačke inteligencije postoji strah od „mašina koje misle“, a sa njihovim razvojem ta pitanja sve su češća a i ti strahovi jačaju: neki smatraju da ona donosi velike koristi, neki smatraju da od nje prete opasnosti, a neki veruje i u jedno i u drugo. Neke od najčešćih etičkih dilema odnose se na pitanja bezbednosti, cenzure, diskriminacije i privatnosti.

Jednostavn primer pitanja koje se tiče bezbednosti je vezan za autonomnu vožnju. Pitanje je kako treba definisati ponašanje vozila u slučaju da mora da ugrozi jednog od dva učesnika u saobraćaju. Još pre toga, pitanje je da li takve odluke prepustiti samom autonomnom sistemu, bez eksplicitne odluke ugrađene unapred. Ukoliko dođe do ugrožavanja bezbednosti učesnika u saobraćaju, povreda, ili štete, postavljaju se i mnoga dodatna, pravna pitanja, na primer – da li za štetu odgovara proizvođač autonomnog automobila ili čovek koji je u njemu sedeo. Mnoga ovakva pitanja će vrlo uskoro biti od praktičnog značaja, a odgovori na njih tek treba da se formulišu. Pored autonomne vožnje tu su i brojna pitanja vezana za vojne primene veštačke inteligencije.

Sistemi za preporučivanje sadržaja već godinama vrše izbor i filtriranje informacija na internetu i društvenim mrežama koje stižu do korisnika. Imajući u vidu da društvene mreže predstavljaju jedan od bitnih kanala informisanja i upoznavanja mišljenja drugih ljudi, ovakve primene bude podozrivost vezanu za mogućnosti cenzure od strane kompanija koje poseduju te društvene mreže i država koje na njih imaju uticaj. Sistemi za preporučivanje političkih ili društveno relevantnih sadržaja, čak i bez ikakve cenzure ili planskog usmeravanja od strane čoveka, mogu voditi korisnike do jednostranih ili polarizovanih stavova.

Brojni eksperimenti pokazali su da sistemi veštačke inteligencije mogu vršiti diskriminaciju pojedinaca na osnovu njihovog pola, boje kože, etničke pripadnosti i drugih faktora. Naime, sistemi zasnovani na učenju iz podataka, između ostalog uče i mnoštvo ljudskih predrasuda i diskriminativnog ponašanja koje se oslikava u tim podacima. Poznat je primer sistema COMPAS čija je uloga da sudijama u Sjedinjenim Američkim Državama pruža procene verovatnoće da će optuženi ponoviti krivično delo ukoliko se bude branio sa slobode. Ustanovljeno je da ovaj sistem precenjuje te verovatnoće u slučaju pripadnika crnačke zajednice.

Tehnike veštačke inteligencije mogu se koristiti i na načine koji mogu ugrožavati privatnost: na primer, za prepoznavanje i označavanje fotografija sa interneta.

### 1.3 Talasi veštačke inteligencije

Mnoge metode veštačke inteligencije zasnovane su na simboličkoj reprezentaciji: i problem i algoritam rešavanja opisani su eksplicitno, a osobine algoritma mogu se analizirati rigorozno, matematički. I opis problema i algoritam su vrlo specifični, prilagođeni jednom konkretnom zadatku i obično se teško uopštavaju. Ti eksplicitni opisi obično se daju u terminima teorije grafova ili matematičke logike. Za ove metode često se kaže da su GOFAI (od engleskog „Good Old-Fashioned Artificial Intelligence“ – „dobra stara veštačka inteligencija“) i čine takozvani „prvi talas“ veštačke inteligencije. One se najčešće oslanjaju na deduktivno zaključivanje.

Od 2010-ih godina, novi moćni računari omogućili su povratak neuronskih mreža (koje su razvijane još 1950-ih), kroz duboko učenje i ostvarili fantastične rezultate u mnogim poljima, kao što je računarski vid, automatsko prevođenje, automatsko upravljanje vozilima, igranje strateških igara, itd. Ovi rezultati zasnovani su na statistici i mašinskom učenju, tj. na induktivnom zaključivanju. U njima nema eksplicitnog opisivanja procesa rešavanja konkretnih primeraka problema, već se koriste meta-algoritmi (tzv. algoritmi učenja) kojima se, koristeći raspoložive podatke, kreiraju algoritmi za rešavanje konkretnih problema. Ovakvi sistemi obično nisu u stanju da ponude i neko objašnjenje za rešenja koje nude. Ipak, takozvani veliki jezički modeli (poput sistema ChatGPT) imaju i tu mogućnost i koriste je sa većom ili manjom uspešnošću. U razvoju i u primenama ovakvih sistema, uloga čoveka je glavna u fazi modelovanja problema i pripremi podataka za obučavanje. O ovim algoritmima i njihovim osobinama znatno je teže formalno rasuđivati nego o algoritmima koji se zasnivaju na eksplicitnom opisu problema, ali je moguće izvesti statističke ocene njihovog kvaliteta. Sistemi zasnovani na statistici i mašinskom učenju čine takozvani „drugi talas“ veštačke inteligencije.

Rasprostranjeno je uverenje da će u budućnosti dosadašnja dva pristupa morati da se integrišu, vodeći do takozvanog „trećeg talasa“ veštačke inteligencije i to se već dešava. Očekuje se da će u trećem talasu sistemi veštačke inteligencije samostalno kreirati modele koji će moći da objasne kako stvari funkcionišu.

### 1.4 Znanje i zaključivanje

Za pojam inteligencije suštinske su dve komponente: *znanje* i *zaključivanje*. Raspoloživo znanje nekada je toliko obimno da je potrebno koristiti specijalne tehnike *pretrage* kako bi se došlo do željene informacije (na primer, da li u skupu milion ljudi postoji niz poznanika koji povezuju zadate dve osobe).

Komponenta zaključivanja predstavlja takođe neku vrstu znanja – to je znanje (koje se naziva i meta-znanjem) o procesu izvođenja novog znanja iz raspoloživog znanja. Pod znanjem podrazumevamo i istinite, potvrđene činjenice, ali i hipoteze, nepotpune informacije i informacije date sa određenim verovatnoćama. Zaključivanje može biti deduktivno – zasnovano na rigoroznim opštim pravilima čijom primenom se dobijaju nove konkretne činjenice. Zaključivanje može biti i induktivno – u njemu se na osnovu mnoštva činjenica pokušava izvođenje opštih pravila. Postoje i druge forme i drugi okviri zaključivanja. Jednostavnim primerom ilustrovaćemo nekoliko različitih oblika zaključivanja. Razmotrimo odnos veze

$$(i) \quad \forall x(P(x) \Rightarrow Q(x))$$

i činjenica

$$(ii) \quad P(a)$$

$$(iii) \quad Q(a).$$

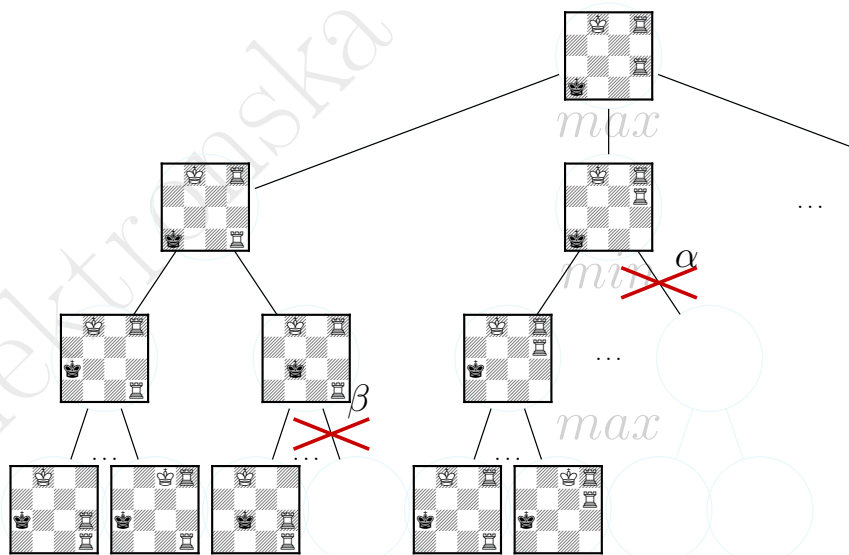
Izvođenje činjenice  $Q(a)$  na osnovu (i) i (ii) odgovara matematičkim principima deduktivnog zaključivanja. Izvođenje veze (i) na osnovu niza parova (ii) – (iii) za različite instance argumenta, odgovara nepotpunoj indukciji, nema matematičku egzaktnost, a u praktičnim primenama pouzdanost mu se povećava sa brojem instanci koje potvrđuju hipotezu. Izvođenje činjenice (ii) na osnovu (i) i (ii) zovemo *abdukcijom*. Ono takođe nije egzaktno, a opravdanje ima u odnosu uzroka i posledice. Na primer, ako za veliki broj osoba znamo da imaju povišenu temperaturu ako imaju grip i ako znamo da neka osoba ima povišenu temperaturu, moguće (ali ne nužno) objašnjenje je da ta osoba ima grip. Abduktivno rasuđivanje zaista se često primenjuje u medicinskim ekspertnim sistemima za utvrđivanje mogućeg uzroka na osnovu poznatih simptoma.

Izbor reprezentacije znanja jedan je od ključnih problema i on je u direktnoj vezi i sa prirodom određenog znanja, ali i sa prirodom mehanizama za zaključivanje. Mehanizmi za zaključivanje moraju biti prilagođeni reprezentaciji znanja i njegovoj prirodi, pa će u jednom slučaju biti zasnovani na klasičnoj logici, a u drugom na modalnoj logici, teoriji verovatnoće, fazi logici itd.

## 1.5 Pretraga

U mnogim praktičnim problemima broj postojećih mogućnosti je tako veliki da ne mogu biti sve ispitane sistematično u razumnom vremenu. Na primer, u sistemu za navigaciju, ukoliko je potrebno naći put između neke konkretne adrese u Lisabonu i neke konkretne adrese u Vladivostoku, broj ključnih tačaka koje se mogu razmatrati u traženju najboljeg puta mogao bi da bude prevelik za praktičnu primenu. U takvim situacijama, pretraga se ne vrši na sistematičan način, već se usmerava *heuristikama* — pravilima koja formalizuju smernice za rešavanje nekog problema. Heuristike ne garantuju uvek pronalaženje najboljeg rešenja, ali obično do rešenja dovode mnogo brže nego sistematična pretraga. Usmerena pretraga koristi se u mnogim oblastima računarstva — u pronalaženju najkraćih puteva, u rutiranju, u rešavanju optimizacionih problema, u računarskom igranju strateških igara kao što je šah.

**Igranje strateških igara.** U igri šah, u sredini partije prosečno ima oko 38 mogućih poteza. Ako ih u nekoj poziciji razmatramo sve - onda je to 38 mogućnosti. Ako razmatramo i sve moguće odgovore protivnika (tj. ako razmatramo dva polupoteza) - onda postoji  $38^2$  mogućnosti. Ako razmatramo svih mogućih deset polupoteza, onda ima  $38^{10}$  tj. više od  $6 \cdot 10^{15}$  mogućnosti, što se ne može obraditi u razumnom vremenu. Ukoliko bi se analizirale sve mogućnosti do samog kraja partije, onda bi taj broj bio još mnogo veći. Sistematična pretraga, dakle, u problemu kao što je ovaj nije primenljiva. Umesto toga, za igru šah najpre se definiše statička funkcija evaluacije koja omogućava nekakvu procenu pozicije iako se nije došlo do kraja partije. Ta funkcija omogućava da se pretraga vrši do neke fiksirane dubine, umesto samo do pozicija u kojima je partija završena. Sledeći ključni korak je onda primena algoritma koji koristi ocene pozicija na nekoj dubini za izbor poteza u tekućem potezu. Jedan od takvih algoritama (minimax sa alfa-beta odsecanjem) često može da odbaci veliki broj mogućnosti bez narušavanja kvaliteta izabranog poteza. Pojednostavljen prikaz ovakvog pristupa igranju igre šah (za pojednostavljenu igru na tabli  $4 \times 4$  umesto  $8 \times 8$ , i sa svega nekoliko figura) dat je na slici 1.1. Ovakav pristup (opisan ovde pojednostavljeno) korišćen je od strane računara DeepBlue koji je 1997. godine pobedio u šahu Garija Kasparova, tadašnjeg prvaka sveta. Današnji najbolji programi za igranje šaha mogu da za jednu poziciju analiziraju moguće nastavke do dubine od nekoliko desetine polupoteza i ocenjuju po nekoliko miliona pozicija u sekundi. Današnji najbolji programi kombinuju opisani pristup sa savremenim tehnikama mašinskog učenja čime postaju još moćniji.

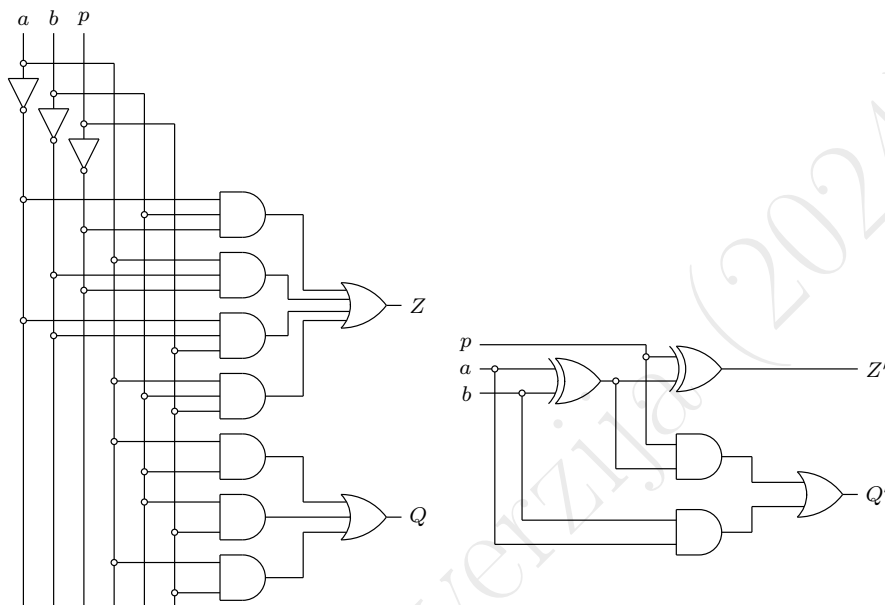


Slika 1.1: Pojednostavljeni prikaz usmeravanja pretrage u igranju igre šah

## 1.6 Automatsko deduktivno rasuđivanje

U mnogim primenama veštačke inteligencije nije kritično da odgovor koji se dobija od računara bude najbolji mogući i egzaktan. Na primer, nije kritično da li je put između dve tačke najbolji za nekog aktera u nekoj računarskoj igrici. Nije kritično ni da li je neki potez u šahu najbolji mogući, posebno jer je u mnogim situacijama skoro nemoguće utvrditi šta je najbolji mogući potez. U mnogim situacijama prihvatljivo je rešenje koje je „dovoljno dobro“. Takva, dovoljno dobra, a ne nužno tačna, egzaktna i najbolja moguća rešenja obično daju sistemi zasnovani na heuristički usmerenoj pretrazi, kao i sistemi zasnovani na mašinskom učenju i induktivnom

zaključivanju. Međutim, postoje mnoge oblasti primene gde je egzaktnost neophodna. Na primer, sistem metroa mora biti takav da je plan kretanja vozova takav da nikada ne dolazi do sudara. Slično, raspored časova za neku školu mora biti takav da nikad, na primer, jedan nastavnik ne treba da bude u dve učionice istovremeno. U takvim situacijama potrebno je koristiti sisteme koji se zasnivaju na deduktivnom rasuđivanju i koji, u principu, daju rešenja koja su uvek egzaktna. Naravno, to ima svoju cenu, te su takvi sistemi vremenski obično znatno neefikasniji nego sistemi zasnovani na heuristikama i mašinskom učenju. Sistemi zasnovani na automatskom rasuđivanju koriste se u mnogim oblastima računarstva, pa i u mnogim drugim. Na primer, koristeći tu tehnologiju moguće je proveravati ekvivalentnost logičkih kola, napraviti raspored časova za školu ili za fudbalsku ligu, pa čak i vršiti analizu rečenica na prirodnom jeziku.



Slika 1.2: Ilustracija dva ekvivalentna logička kola

**Ekvivalentnost logičkih kola.** Logička kola su u osnovi svakog savremenog računara. U procesoru računara hardverski su, u vidu logičkih kola, implementirane operacije poput sabiranja, množenja, poređenja i bitovskih operacija. Jednostavnost i efikasnost tih kola je ključna za funkcionisanje procesora, pa i čitavog računara. Nekad neko logičko kolo može biti zamenjeno drugim, jednostavnijim kolom, ali je neophodno najpre se uveriti da su ta dva kola ekvivalentna. Razmotrimo jedan jednostavan primer: neka jedno kolo odgovara logičkoj funkciji  $\neg(\neg a \vee \neg b)$  a drugo logičkoj funkciji  $a \wedge b$ . Da bi se pokazalo da su ova logička kola ekvivalentna potrebno je dokazati da je iskazna formula  $\neg(\neg a \vee \neg b) \Leftrightarrow a \wedge b$  tautologija, tj. da ni za koje vrednosti promenljivih  $a$  i  $b$  nije netačna. Slika 1.2 ilustruje nešto kompleksniji primer sa dva logička kola za sabiranje: ulazne vrednosti  $a$  i  $b$  su cifre brojeva koji se sabiraju,  $p$  je prethodni prenos, a na izlazu je formula koja opisuje cifru zbira  $Z$  i formula koja opisuje novi prenos  $Q$ . Da bi se dokazalo da su ova dva kola ekvivalentna potrebno je pokazati da su za sve moguće trojke vrednosti  $a$ ,  $b$  i  $p$  izlazi ova dva kola jednaki, što se ponovo može opisati nekom formulom za koju je potrebno utvrditi da je tautologija. U industrijskoj praksi, naravno, javljaju se i daleko kompleksniji zadaci ovog tipa.

U navedenim primerima, potrebno je imati softversku alatku koja može da proveri da li je data iskazna formula  $F$  tautologija. Takvu alatku zovemo *rešavačem*. U prikazanim primerima, postoji svega nekoliko iskaznih promenljivih koje figurišu u formuli, ali savremeni rešavači mogu za svega nekoliko sekundi da ispituju i neke formule koje imaju na stotine hiljada promenljivih.

Rešavači moraju da daju egzaktan odgovor na pitanje da li je neka formula tautologija ili zadovoljiva ili kontradikcija ili poreciva, ali zbog praktične upotrebljivosti ne mogu sistematično ispituju sve moguće vrednosti za sve promenljive koje se pojavljuju u formuli. Umesto toga, rešavači koriste mnoštvo algoritamskih tehnika koje ubrzavaju ispitivanje. Po tome, rad rešavača sličan je heurističkoj pretrazi, ali uz dodatak da su sve heuristike takve da ne ugrožavaju egzaktnost odgovora rešavača.

Pored alata za rasuđivanje u iskaznoj logici, postoje i alati za automatsko rasuđivanje u drugim, bogatijim logikama. Na primer, takvi alati mogu da rešavaju kompleksne matematičke probleme. Godine 1996. jedan takav alat po prvi put je dokazao teorem (iz algebre) koju pre toga ljudi nisu uspeli da dokažu.

**Automatsko dokazivanje u geometriji.** Jedno od najranijih polja primene automatskog rasuđivanja je geometrija – već sredinom pedesetih godina dvadesetog veka razvijeni su prvi geometrijski dokazivači a jedna od prvih dokazanih netrivialnih teorema tvrdila je da za proizvoljne međusobno različite tačke  $A, B, C, D$  takve da važi:  $\angle DB, BA = \angle DB, CB, BA \perp AD, CB \perp CD$ , mora da važi i  $AD \cong CD$ . Ako skup raspoloživih aksioma uključuje sledeće aksiome (zapisane na prirodnom jeziku):

**axp:** ako su kraci ugla normalni, onda je ugao prav

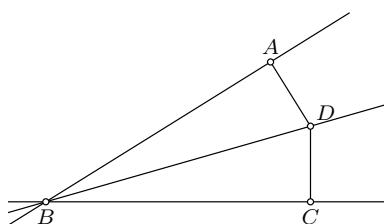
**axra:** dva prava ugla međusobno su podudarna

**axaaa:** ako su dva para uglova dva trougla podudarni, onda su podarni i treći uglovi

**axasa:** dva trougla sa jednim parom podudarnih stranica i podudarnim naleglim uglovima su podudarna

**axtc:** ako su dva trougla podudarna, onda su im parovi odgovarajućih stranica podudarni

savremeni dokazivači bi mogli da proizvedu ovakav dokaz:



1. uglovi  $\angle BA, AD$  i  $\angle CB, CD$  su pravi (axp)
2.  $\angle BA, AD = \angle CB, CD$  (axra)
3.  $\angle AD, DB = \angle CD, DB$  (axaaa)
4.  $\triangle DBA \cong \triangle DBC$  (axasa)
5.  $AD \cong CD$  (axtc)

Jedan pristup za dokazivanje navedene teoreme je *ulančavanje unapred*. U tom pristupu kreće se od raspoloživih znanja i izvode se nova. Na primer, kao u navedenom dokazu, iz činjenice  $BA \perp AD$ , na osnovu aksiome tj. pravila „ako su kraci ugla normalni, onda je ugao prav“, izvodi se zaključak da je ugao  $\angle BA, AD$  prav. Slično se izvode i drugi zaključci. Postoje i metode za automatsko dokazivanje zasnovane na *ulančavanju unazad*, u kojem se kreće od ciljnog tvrđenja i traže se traže činjenice koje mogu da impliciraju to tvrđenje i tako dalje, sve dok se ne dođe do zadatih činjenica. Postoje i metode za automatsko dokazivanje koje kombinuju ova dva pristupa, ali i metode koje su zasnovane na sasvim drugačijim principima. U dokazivanju netrivialnih tvrđenja, bez obzira na pristup koji se koristi, mogu se javiti milioni mogućnosti koje treba ispitati, te je za praktičnu upotrebljivost neohodno koristiti heuristike koje usmeravaju pretragu ka mogućnostima koje obačavaju više.

Današnji automatski dokazivači mogu da dokažu daleko složenije teoreme, iz raznih matematičkih disciplina, uključujući mnoge probleme sa međunarodnih matematičkih olimpijada. Automatsko dokazivanje matematičkih teorema nalazi primene ne samo u teorijskoj matematici, već je važno i u mnogim drugim oblastima, na primer u robotici, razvoju softvera, raspoređivanju, itd.

## 1.7 Mašinsko učenje i induktivno rasuđivanje

Mašinsko učenje, pre svega ono zasnovano na neuronskim mrežama, dalo je nov zalet oblasti veštačke inteligencije u prethodnih 10-15 godina. Pomoću njega napadnuti su neki problemi koji su ranije smatrani zabranom čoveka, a neki dugo otvoreni problemi su u potpunosti rešeni. Najčešće oblasti primene su u računarskom vidu, obradi prirodnih i programskih jezika, autonomnoj vožnji, igranju igara, bioinformatiki i šire. Mašinsko učenje omogućava razumevanje saobraćaja od strane autonomnih automobila na osnovu kamera i drugih senzora. Primera radi sa slike je moguće odrediti pozicije pešaka, automobila, semafora i saobraćajnih znakova. Nobelova nagrada za hemiju 2024. godine dodeljena je za rešavanje dugo otvorenog problema savijanja proteina – problem određivanja strukture u trodimenzionalnom prostoru kada je poznat niz aminokiselina koje čine taj protein. Veliki jezički modeli poput ChatGPT-a u stanju su da proizvode visoko kvalitetne prevode između različitih jezika (onih za koje su dostupne velike količine tekstova), popravljaju kvalitet napisanog teksta i da odgovaraju na pitanja i pružaju korisne informacije. Alati poput Kopilota, zasnovani na velikim jezičkim modelima u stanju su da pišu delove programskog koda u skladu sa zahtevom programera i da tako ubrzaju proces programiranja. OpetnAI o1, veliki jezički model opšte namene rangira se u prvih 12% na problemima takmičarskog programiranja platforme Codeforces. Rešavajući probleme sa Međunarodne matematičke olimpijade, jedan takav sistem specijalizovan za matematičke probleme osvojio je 28 poena – ekvivalent jake srebrne medalje, dok je raspon zlatne medalje počinjao od 29 poena (koje je ostvarilo 58 od 609 takmičara). U igranju strateških igara kao što su šah, go i drugi, neuronske mreže su odavno daleko nadmašile najbolje ljudske igrače.

### 1.7.1 Mašinsko učenje i generalizacija

Mašinsko učenje može se definisati na više načina. Mašinsko učenje je oblast veštačke inteligencije koja se bavi razumevanjem i formalizacijom induktivnog rasuđivanja, tj. generalizacije. Generalizacija znači formulisanje opštih zakonitosti na osnovu konačnog skupa opažanja. Primera radi, videviši veliki broj labudova od kojih su svi beli, indukcijom se može izvesti zaključak da su svi labudovi beli. Ovakvi zaključci očito mogu biti pogrešni. Teorija mašinskog učenja bavi se razumevanjem uslova pod kojima ovako doneseni zaključci mogu biti visoko pouzdani i metodama koje su u stanju da izvode takve zaključke. Iz perspektive primena, mašinsko učenje se može definisati kao oblast koja se bavi automatskim kreiranjem programa na osnovu datih ulaza i željenih izlaza. Primetimo da i ovo predstavlja vid generalizacije. Naime, ti automatski kreirani programi predstavljaju vid opisa zakonitosti koje važe između njihovih ulaza i izlaza. U nastavku će mašinsko učenje biti posmatrano iz ove praktične perspektive.

Programiranje nije lako. Da bi se uspešno napisao program, potrebno je: precizno opisati problem, detaljno ga razumeti, razložiti na potprobleme koji se lakše rešavaju pojedinačno, smisliti algoritme kojima se svaki od njih rešava i od njihovih rešenja proizvesti rešenje polaznog problema. Nijedan od tih koraka ne mora biti trivijalan, a neki problemi, iako rešivi, za čoveka predstavljaju nepremostiv izazov. Primera radi, neka je potrebno napisati program koji kao ulaz dobija matricu dimenzija  $N \times N$  koja sadrži brojeve od 0 do 255 i koja predstavlja sliku u nijansama sive, a na izlazu ispisuje DA ili NE u zavisnosti od toga da li se na slici nalazi ljudsko lice. Da bi se ovaj problem rešio, potrebno je prvo detaljno razumeti šta su karakteristike ljudskog lica. Lako je doći do nekih parcijalnih uvida i zaključiti da se lice sastoji od nosa, usta, očiju, ušiju i slično. To samo otvara dalja pitanja vezana za to šta je svaki od ovih pojmova i kakvi odnosi treba da važe među njima da bi oni činili lice. Međutim, suštinska težina ovog posla je u tome da je sve upotrebljene pojmove potrebno definisati u terminima piksela ulazne slike, a ne u bilo kakvim apstraktnijim pojmovima (kao što su zenice, usne, nozdrve, itd.) u kojima su ljudi navikli da razmišljaju i da se izražavaju. Ovo razmatranje sugerise da je pokušaj pisanja ovakvog programa osuđen na propast već u fazi razumevanja problema i osmišljavanja rešenja. Iako je teško precizno definisati šta je lice, pa samim tim i pristupiti pisanju prethodno pomenutog programa, nije teško sakupiti mnoštvo slika lica sa interneta. Štaviše, moguće je sakupiti skup primera slika koje sadrže lica i za koje odgovor tog programa treba da bude DA i skup primera slika koje ne sadrže lica i za koje odgovor programa treba da bude NE. Skup primera svakako ne čini preciznu specifikaciju problema, ali može se prihvatiti kao neka neprecizna implicitna specifikacija. Pitanje koje se postavlja u mašinskom učenju je da li program za prepoznavanje lica možemo automatski izvesti procesom *učenja* iz pomenutog skupa primera. Za takav program kažemo da je *naučen* u smislu da predstavlja ishod procesa učenja nekog sistema koji analizira dati skup primera. U velikom broju praktičnih problema moguće je naučiti takav program koji će često za date ulaze davati *dobre* izlaze. Šta u prethodnoj rečenici znači *često* i *dobro* nije precizno definisano, već se u praksi kvalitet izlaza ovakvih programa empirijski evaluira. Primera radi, procenat tačnih identifikacija lica može biti dobra mera kvaliteta ovakvog programa. U uobičajenom programiranju podrazumeva se zahtev da programi budu korektni, tj. da za date ulaze uvek daju dobre izlaze. U mašinskom učenju podrazumeva se da će biti grešaka, tj. da kreirani program za neke ulaze ne daje ispravne izlaze, ali se teži tome da greške budu retke i male. Kako se obično kaže, poželjno je da modeli *dobro generalizuju*. Naglasimo da za dobru generalizaciju nije dovoljno da naučeni program često daje dobre izlaze na podacima iz kojih je naučen. Od presudnog je značaja da to važi i na podacima iz kojih nije učeno. Pritom, greške su očekivane i na jednoj i na drugoj grupi podataka.

### 1.7.2 Podaci

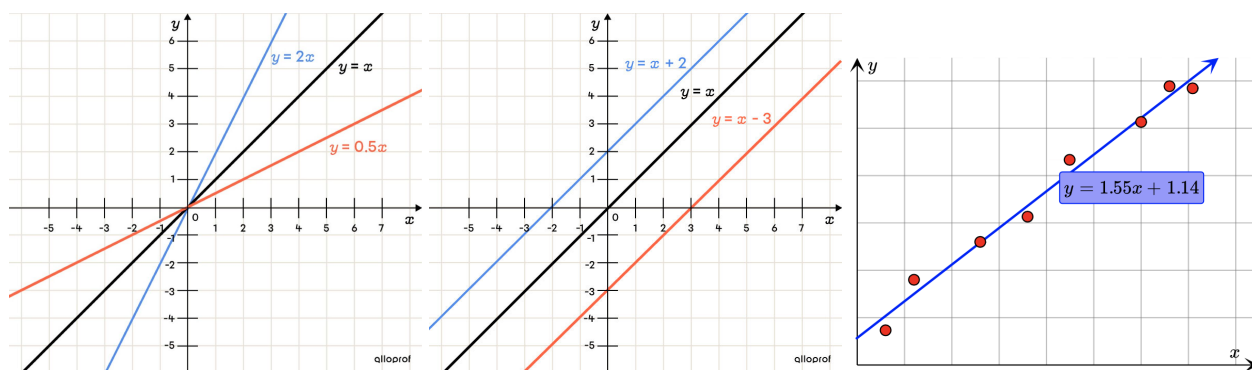
Proces učenja polazi od podataka. Podaci su opisani nekim numerčkim svojstvima. Primera radi, banka može odlučiti da svoje klijente opiše svojstvima kao što su ime, prezime, godine starosti, godišnja primanja, bračni status, itd. Istorija dnevnih cena nekih akcija na berzi može se opisati nizom vrednosti od kojih svaka odgovara nekom danu u posmatranom vremenskom intervalu i ima vredost cene tih akcija na dati dan. Slika se može opisati matricom trojki intenziteta crvene, zelene i plave. Na ovaj način, svi podaci od interesa mogu se opisati brojevima.

### 1.7.3 Modeli i njihovo obučavanje

Algoritam koji analizira podatke u potrazi za zakonitostima u njima naziva se *algoritmom učenja*. On kao izlaz daje nekakav matematički opis uočenih zakonitosti. Takvi opisi nazivaju se *modelima*. U razmatranom problemu prepoznavanja lica, program za prepoznavanje lica predstavlja upravo jedan model. Njegov kôd opisuje vezu između ulaza (slika) i izlaza (odgovora DA i NE). Programi koje pišemo na uobičajeni način obično su čitljivi i pišu se u nekom od standardnih programskih jezika, dok su modeli, tj. programi koji se automatski kreiraju mašinskim učenjem obično matematičke funkcije definisane velikim brojem parametara.

Nameću se mnoga pitanja u vezi sa tim matematičkim funkcijama. Kakve su to funkcije? I kako se pomenuti





Slika 1.3: Prikaz različitih modela oblika  $y = ax + b$  pri promeni parametara  $a$  (levo) i  $b$  (u sredini) i prikaz prave koja „dobro” odgovara nekim datim podacima (desno).

parametri izračunavaju iz podataka? Jednostavna vrsta parametrizovanih matematičkih funkcija su funkcije oblika  $y = ax + b$ , gde su  $y$  i  $x$  promenljive (recimo, telesna masa i visina osobe, tim redom), a  $a$  i  $b$  nepoznati parametri koje treba odrediti, tako da ova veza dobro opisuje što veći broj ljudi. Ilustracije takvih funkcija prikazane su na slici 1.3. Neka su za određeni broj osoba (recimo 20), izmerene telesna masa i visina. Da li je moguće naći vrednosti parametara  $a$  i  $b$  tako da model  $y = ax + b$  dobro odgovara podacima? Ukoliko je moguće, to znači da je moguće za datu visinu, pomoću ovog modela, predvideti telesnu masu. Ali šta uopšte znači da model dobro odgovara podacima? Prirodno je da im utoliko bolje odgovara, što su razlike između predviđenih i stvarnih vrednosti telesnih masa manje. Recimo, ako su dati parovi visina i masa  $\{(x_i, y_i) \mid i = 1, \dots, 20\}$ , želimo da vrednost ukupnog odstupanja

$$L(a, b) = \sum_{i=1}^{20} (y_i - (ax_i + b))^2$$

bude što manja. Ilustracija funkcije koja u ovom smislu dobro odgovara datim podacima data je na slici 1.3.

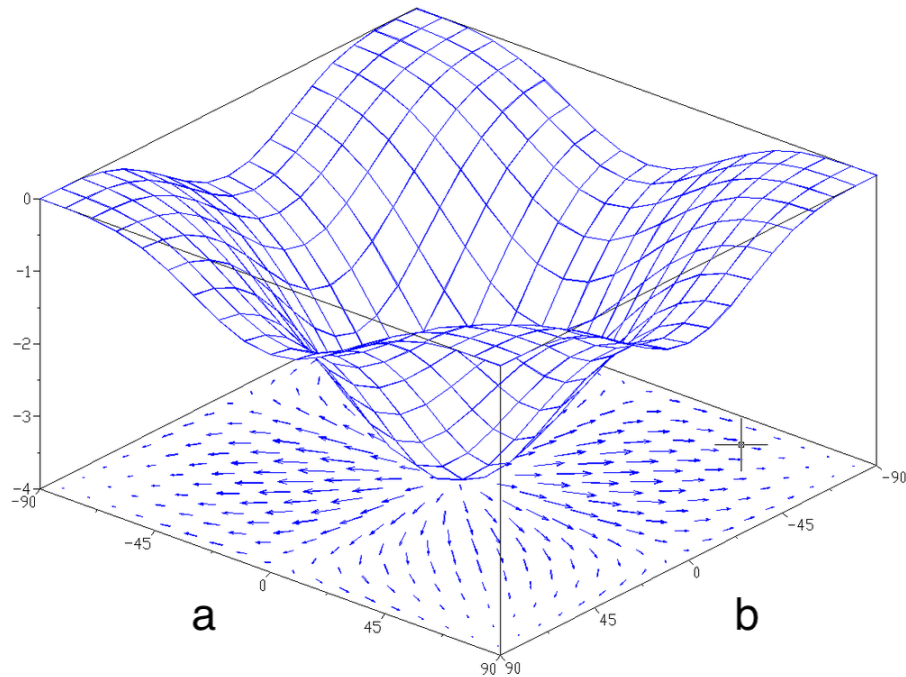
Očito vrednost  $L(a, b)$  zavisi od toga kako izabaremo parametre  $a$  i  $b$ . Vrednosti u kojima ova funkcija dostiže svoj minimum mogu se dobiti kretanjem nizbrdo pošavši od nekih proizvoljnih vrednosti. Ovaj postupak je moguće izvesti zahvaljujući poznavanju *gradijenta* funkcije u različitim tačkama. Gradijent je vektor koji u svakoj tački pokazuje pravac najbržeg rasta funkcije u okolini te tačke i postoje jednostavne formule za njegovo izračunavanje. Ovaj koncept ilustrovan je slikom 1.4. Ako pođemo od proizvoljne tačke  $(a, b)$  i krećemo se malim koracima u smeru suprotnom od gradijenta, nakon dovoljno koraka, naći ćemo se dovoljno blizu tačke u kojoj funkcija dostiže minimalnu vrednost. Funkcija  $L(a, b)$  koju posmatramo je jednostavna i ima samo jednu takvu tačku. U slučaju komplikovanijih funkcija, ovaj postupak ne mora uvek naći najbolje vrednosti za parametre funkcije, ali u praksi daje dobre rezultate.

#### 1.7.4 Neuronske mreže

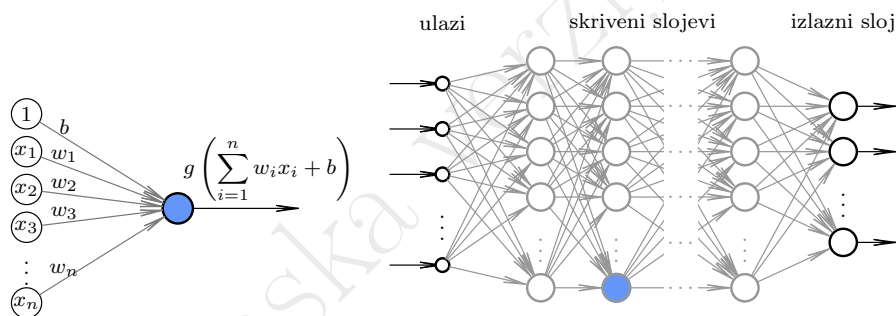
Modeli u vidu pravih kakve smo do sada razmatrali su jednostavni i korisni za rešavanje nekih relativno jednostavnih problema. Postavlja se pitanje postoje li neki komplikovaniji i moćniji modeli. Ideja za njihovu konstrukciju može se pozajmiti iz neurologije. Ljudski mozak sastoji se od relativno jednostavnih gradivnih elemenata – neurona. Neuroni sakupljaju ulazne signale od drugih neurona sa kojima su povezani i u zavisnosti od toga kakve su signale primili, daju nekakav izlazni signal drugim neuronima sa kojima su povezani. Analogno tome, moguće je organizovati izračunavanje u veštačkoj *neuronskoj mreži* prikazanoj na slici 1.5. Svaki od neurona (prikazanih krugovima) predstavlja jednu parametrizovanu funkciju. Neuronska mreža predstavlja složenu kompoziciju neurona, pa je i sama jedna parametrizovana funkcija (čiji su parametri zapravo parametri svih neurona skupa). Neuroni se mogu definisati na različite načine, ali se obično koristi linearna funkcija ulaza na koju se primenjuje neka jednostavna nelinearna transformacija. Kako se obučava jedna neuronska mreža? Isto kao i jednostavna prava iz prošlog primera! Ukoliko je dostupan skup parova ulaza i njima odgovarajućih izlaza, moguće je izračunati koliko izlazi mreže odstupaju od željenih izlaza za date ulaze. Ponovo je moguće smanjivati grešku koju mreža pravi varirajući vrednosti parametara krećući se nizbrdo, u smeru suprotnom od gradijenta.

Ovo objašnjenje neuronskih mreža dato je u vrlo grubim crtama. Postoje mnoge varijacije izloženog principa. Recimo, mreže koje predviđaju neprekidne vrednosti (poput boja na slici) i mreže koje predviđaju diskretne vrednosti (poput reči iz nekog rečnika) imaju nešto drugačije strukture. Načini na koji se mere odstupanja između ispravnih i predviđenih vrednosti takođe se razlikuju u različitim slučajevima. Čak se i cela konstrukcija





Slika 1.4: Prikaz gradijenta funkcije u različitim tačkama ( $a, b$ ).



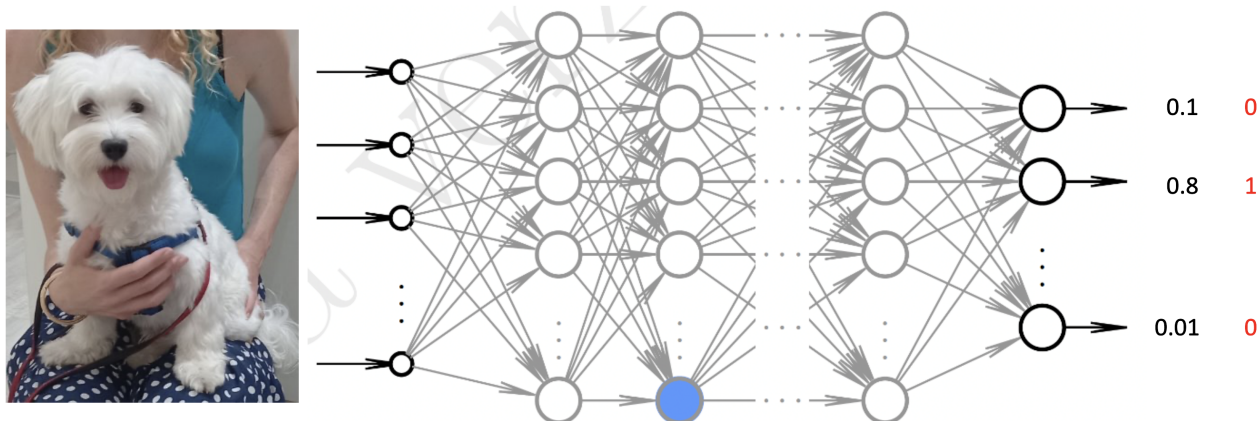
Slika 1.5: Struktura neurona i arhitektura potpuno povezane neuronske mreže.

mreže može značajno razlikovati u slučaju obrade slika, videa, zvuka, teksta ili tabelarnih podataka. Ono što sve te varijacije imaju zajedničko je da sve neuronske mreže predstavljaju kompoziciju velikog broja jednostavnih parametrizovanih funkcija koje se kombinuju u složenu celinu.

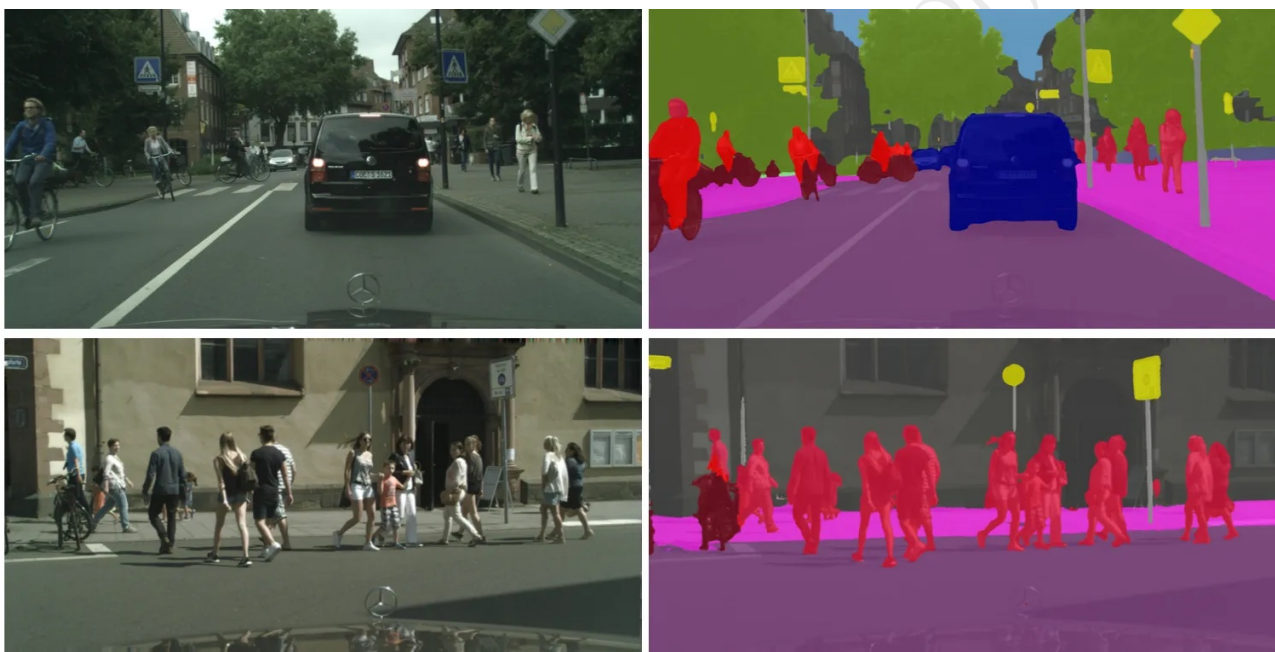
### 1.7.5 Neuronske mreže u praktičnim primenama

Razmotrimo nekoliko primera primene neuronskih mreža sa fokusom na tome kako se one treniraju u datim kontekstima. Navedene primene tiču se računarskog vida i obrade prirodnog jezika, ali uzbudljivih primena ima i u raznim drugim oblastima.

**Razvrstavanje slika.** Pretpostavimo da je potrebno prepoznati rasu psa na slici i da je dostupan veliki broj slika za koje je poznato koju rasu prikazuju. Slike se tipično predstavljaju kao matrice piksela od kojih svaki piksel ima pridružena tri broja — intenzitete crvene, zelene i plave boje. Svi ti brojevi, u nekom fiksiranom poretku, predstavljaju ulaze neuronske mreže. Mreža treba da ima onoliko izlaza koliko ima rasa pasa. Ideja je da svaki izlaz odgovara jednoj rasi i da veće vrednosti na izlazu sugerišu veću verovatnoću da slika prikazuje psa te rase. Ovakav dizajn mreže ilustrovan je slikom 1.6. Prirodno je očekivati od mreže da za ulaznu sliku, na izlazu koji odgovara tačnoj rasi, bude vrednost bliska jedinici, a da na izlazima koji odgovaraju pogrešnim rasama budu vrednosti bliske nuli. Ukupno odstupanje od ovih vrednosti za sve raspoložive podatke treba da budu što manje, što se postiže kao i pre — podešavanjem parametara metodom zasnovanom na kretanju nizbrdo. Broj neurona u mreži i njihov raspored obično se određuju eksperimentalno, tako da se na kraju dobiju što bolji



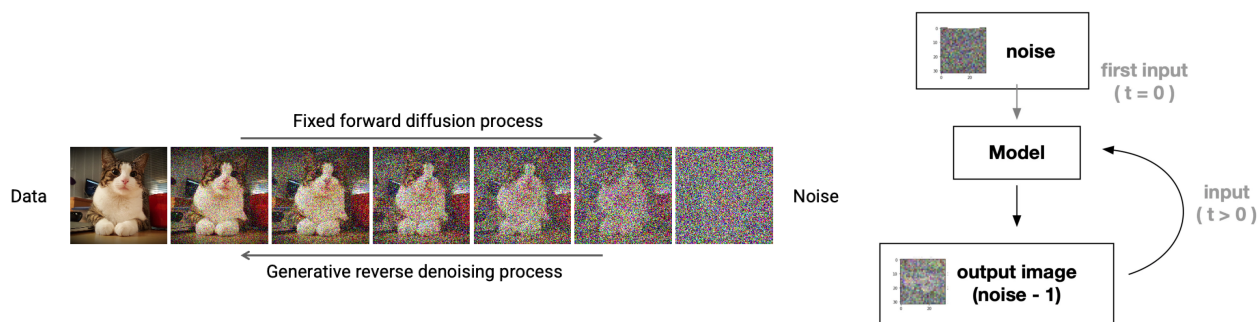
Slika 1.6: Prikaz mreže koja prepoznaje rasu psa. Slika se daje na ulazima mreže, a izlazi treba da odgovaraju stvarnoj rasi psa. Željeni izlazi su prikazani crvenom bojom.



Slika 1.7: Prikaz segmentacije slika iz saobraćaja na kojima se vide jasno označeni učesnici u saobraćaju i saobraćajni znakovi.

rezultati. To eksperimentisanje obično prati neke ustaljene prakse koje ovde nećemo objašnjavati.

**Semantička segmentacija slika.** Prethodni primer opisuje problem koji se smatra relativno jednostavnim — razvrstavanje slika u neke unapred definisane kategorije. Druga, izrazito korisna vrsta zadatka koju neuronske mreže mogu da vrše je razvrstavanje pojedinačnih piksela u te unapred definisane kategorije, čime se zapravo vrši segmentacija slike u smislene celine. Primera radi, neki pikseli na slici predstavljaju osobe, neki automobile, neki trotoar i tako dalje. Za neuronsku mrežu koja treba da upravlja autonomnim automobilom od velikog je značaja da razume u takvim terminima i da na osnovu toga razume, recimo, da se pešak nalazi ispred automobila. Prikaz ovakvog zadatka dat je slici 1.7. Neuronska mreža koja bi na ovaj način kategorisala piksele imala bi jednako organizovan ulaz kao i prethodna, ali bi izlaz bio značajno drugačije implementiran. Prethodna mreža bila je u stanju da pomoću  $N$  izlaza prepozna jednu od  $N$  rasa pasa. Mreža koju sada diskutujemo mora biti u stanju da na sličan način razvrsta svaki od  $m \times n$  piksela ulazne slike u neku od  $N$  kategorija, tako da mora imati  $m \times n \times N$  izlaza. Ukoliko je (najčešće zahvaljujući ljudskom trudu) dostupan skup podataka u kome je za veliki broj slika za svaki piksel poznato kojoj kategoriji pripada, moguće je obučiti neuronsku mrežu da vrši ovaj vid analize slike.



Slika 1.8: Niz slika dobijenih dodavanjem šuma na pravu sliku (levo) i shema modela koji uzima sliku sa više šuma i proizvodi sliku sa manje šuma (desno).

**Generisanje slika.** U problemu generisanja slika, neuronske mreže dostigle su izvanredne rezultate. Obično se mreži na ulazu pruži tekstualni opis slike koju je potrebno da generiše, kao i neki vektor pseudo-slučajnih brojeva. Na osnovu ta dva ulaza, mreža generiše izlaznu sliku. Za isti zadati opis, ali za različite izbore pseudo-slučajnih brojeva, mreža generiše različite slike. Kako ne bismo ulazili u detalje razumevanja teksta, pojednostavićemo problem tako što ćemo pretpostaviti da mreža ne uzima nikakav opis (recimo da je naučena samo da generiše slika pasa i da nije potrebno tražiti od nje više do toga), već da samo na osnovu pseudo-slučajnih brojeva na ulazu generiše sliku. Dobiti kvalitetnu sliku sigurno nije lako. Ali ono što jeste lako, to je pokvariti već postojeću dobru sliku. Recimo, na sledeći način. Pošavši od jedne slike, moguće je na nju kroz iteracije dodavati male količine šuma — pseudoslučajnih brojeva bliskih nuli, sve dok količina šuma u potpunosti ne uništi sadržaj slike i ne završimo sa slikom koja predstavlja samo neke pseudo-slučajne brojeve. Ovaj proces ilustrovan je na slici 1.8. Ovo je lako, ali nije mnogo korisno. Ipak, postavlja se pitanje, da li se ovaj proces može nekako obrnuti: možemo li, krećući se od neke slike popunjene pseudoslučajnim brojevima, doći do kvalitetne slike? Odgovor je pozitivan, i proces se može ugrubo razumeti na sledeći način. Niz slika  $I_1, I_2, \dots, I_n$  koji je generisan dodavanjem šuma može se iskoristiti za dobijanje parova ulaza i izlaza. To bi bili svi mogući parovi  $(I_i, I_{i-1})$  gde prva slika predstavlja ulaz (koji ima više šuma), a druga izlaz (koji ima manje šuma). Za sve te parove, mreža uči<sup>2</sup> da za datu sliku  $I_i$  generiše odgovarajuću sliku  $I_{i-1}$ , tj. da otklanja šum sa slike. Kada je mreža obučena, proces generisanja slike je iterativan i polazi od slike koja sadrži samo pseudo-slučajne brojeve. U svakoj iteraciji, naučena mreža se primenjuje na tekuću sliku dok se ne dostigne određeni broj iteracija. Taj broj je iskustveno određen tako da proces na kraju rezultuje verodostojnom slikom bez šuma.

**Veliki jezički modeli.** Trenutno najuzbudljiviju primenu neuronskih mreža predstavljaju veliki jezički modeli. Pojednostavljeno, jezički model je model koji za dati tekst (koji se često naziva kontekst) i svaku potencijalnu narednu reč, procenjuje njenu verovatnoću. Na taj način, jednim jezičkim modelom moguće je generisati verodostojne tekstove na nekom jeziku. Naravno, kao i u slučaju slika, i reči se predstavljaju brojevima. Veliki jezički modeli predstavljaju neuronske mreže obučene na ogromnim korpusima teksta sakupljenim sa čitavog interneta i iz drugih dostupnih izvora. Primetimo da u takvom učenju nije neophodan veliki ljudski trud u pripremi podataka — kada su tekstovi sakupljeni, svaki niz uzastopnih reči iz tog teksta predstavlja jedan kontekst za koji je poznata naredna reč (ako postoji). Iz ogromnog broja situacija, neuronska mreža naučiće ne samo da pogađa najverovatniju sledeću reč, već i verovatnoće mnogih drugih reči kao potencijalnih sledbenika. Naime, ukoliko se u tekstovima iz kojih mreža uči nakon konteksta „Marko igra” pet puta javlja reč fudbal, dva puta reč košarka i tri puta reč tenis i slično za kontekste „Jovana igra”, „Petar igra” i druge, mreža će, u nekom obliku, naučiti da razne osobe u približno 50% slučajeva igraju fudbal, u 20% košarku, a u 30% tenis, pa će tako proceniti verovatnoće i u slučaju konteksta „Janko igra” čak i ako Janko nije pomenut u tekstovima iz kojih je mreža učila. Kada je mreža naučena da ocenjuje verovatnoću naredne reči za dati kontekst, obično se dalje doobučava na posebno pripremljenim tekstovima koji predstavljaju parove pitanja i odgovora pošto je to očekivani vid interakcije sa čovekom. Ovakvi sistemi sposobni su da vode smislene konverzacije sa čovekom i da daju korisne informacije, da prevode tekst sa jednog jezika na drugi, da rešavaju matematičke probleme, da pišu delove programskog koda i slično. Neretko i greše, pa je u njihovoj upotrebi i dalje potreban nadzor čoveka. Ipak, dometi njihovog uspeha iznenadili su i mnoge profesionalce u oblasti veštačke inteligencije. Posebno iznenađenje predstavlja činjenica da je njihov uspeh postignut na principu predviđanja naredne reči, što je mnogima delovalo

<sup>2</sup>Primetimo da fraza „mreža uči” odudara od ranije korišćenih (i pravilnijih) fraza poput „program je naučen”, „model je naučen”, „model je naučen” i „mreža je obučena”. Te fraze impliciraju da su mreža ili model ishod nekog procesa učenja (koji sprovodi algoritam učenja). Fraza „mreža uči” sugerise samostalnost mreže u procesu učenja, što nije stvarno slučaj — proces učenja je pod potpunom kontrolom algoritma učenja. Ipak, zbog široke upotrebe ove fraze i lakoće izražavanja i mi ćemo koristiti i nju i slične jezičke konstrukcije.

kao previše grub i jednostavan princip za rešavanje sofisticiranih problema. Pokazalo se da je veliki jezički model učeći da korektno niže reč za rečju, zapravo naučio i sistem internih reprezentacija pročitano g teksta, a nad njima i mehanizme za rešavanje najrazličitijih problema. Ipak, kako bi veliki jezički model uspeo da da korektan izlaz nižući reč za rečju, zapravo je neophodno da nauči sistem internih reprezentacija pročitano g teksta, a nad njima mehanizme za rešavanje najrazličitijih problema. To je u nekom smislu prilično slično onome što čovek radi kad rešava probleme - čita njihovu specifikaciju, razume je u svom konceptualnom sistemu koji je oblikovan recimo školovanjem, primenjuje neki mehanizam rešavanja i na osnovu toga ispisuje rešenje. Naravno, dubina ovakvih analogija je otvoreno pitanje, jer u ovom trenutku ni funkcionisanje ljudskog mozga, a ni unutrašnje funkcionisanje velikih jezičkih modela nije dovoljno razjašnjeno.

Opisani primeri kriju kompleksnost procesa treniranja i evaluacije neuronskih mreža i daju samo obrise ideja na kojima su rešenja pomenutih metoda zasnovna. U praksi, kako bi se na ovakvim modelima zasnovali upotrebljivi proizvodi, potrebna je velika količina teorijskog razumevanja i praktičnog iskustva. Pored toga, potrebne su veštine razvoja softvera, razumevanje rada hardvera, operativnih sistema, izračunavanja u oblaku itd. Stoga se ovakvim problemima obično bave raznovrsni timovi inženjera, a nekad i istraživača, uz veliku pomoć ljudi koji pripremaju podatke za obučavanje. Pored raznovrsnog znanja, potreban je i specijalizovani hardver na kojem je moguće vršiti učenje velikih mreža na velikim količinama podataka (koji se nekad broje hiljadama, a nekad milijardama). Otuda je rešavanje praktičnih problema pomoću neuronskih mreža puno izazova, ali shodno tome i uzbuđenja.