Računarski fakultet

DIPLOMSKI RAD

Slika je vredna 16x16 reči: Vision Transformeri

Vanja Kovinić
RN 42/2020

Mentor:
dr Nemanja Ilić

Komisija:
dr Nemanja Ilić
dr Nevena Marić

Beograd, septembar 2024.

Sadržaj

Apstrakt

Ovaj diplomski rad istražuje **Vision Transformere** (**ViT**), nov pristup u oblasti računarskog vida koji koristi arhitekturu transformera prvobitno razvijenu za obradu prirodnog jezika. Prvi deo rada pruža detaljan pregled arhitekture transformera, uključujući ključne komponente kao što su ***self-attention*** **mehanizam** i **poziciono enkodovanje**, i diskutuje njihove svrhe i funkcionalnosti. Nakon toga, fokus se prebacuje na Vision Transformere, objašnjavajući kako se slike transformišu u **tokene** i obrađuju kroz **enkoder transformera** kako bi se primenili na rešavanje vizuelnih zadataka.

Rad zatim ulazi u praktične aspekte implementacije Vision Transformera, uključujući izbor i podešavanje **hiperparametara** za poboljšanje performansi. Izvršeno je i poređenje sa referentnim implementacijama, i predložen pristup za poboljšanje performansi. Prikazani su različiti eksperimenti, zajedno sa diskusijom njihovih rezultata, pružajući uvid u efikasnost i izazove povezane sa Vision Transformerima.

Na kraju, rad naglašava značaj Vision Transformera u oblasti računarskog vida, prikazujući njihov potencijal i ograničenja, kao i njihove praktične primene.

1 Uvod

"Pre otprilike 540 miliona godina, Zemlja je bila obavijena tamom. Ovo nije bilo zbog nedostatka svetlosti, već zato što organizmi još uvek nisu razvili sposobnost da vide. Iako je sunčeva svetlost mogla da prodre u okeane do dubine od 1.000 metara i hidrotermalni izvori na dnu mora isijavali svetlost u kojoj je život cvetao, nijedno oko nije se moglo naći u tim drevnim okeanima, nijedna retina, rožnjača ili sočivo. Sva svetlost i život nikada nisu viđeni. Koncept gledanja nije ni postojao tada i ova sposobnost nije ostvarena sve dok nije stvorena.

Iz nama nepoznatih razloga, trilobiti su se pojavili kao prva bića sposobna da spoznaju svetlost. Oni su prvi prepoznali da postoji nešto izvan njih samih, svet okružen višestrukim jedinkama. Rađenje vida se smatra da je pokrenulo kambrijsku eksploziju, period u kojem se veliki broj vrsta životinja pojavljuje u fosilnom zapisu. Vid je započeo kao pasivno iskustvo, jednostavno propuštanje svetlosti, ali je ubrzo postao aktivniji. Nervni sistem je počeo da evoluira, vid je prešao u uvid, gledanje je postalo razumevanje, a razumevanje je dovelo do akcije, a sve to je dovelo do nastanka inteligencije.

Danas nismo više zadovoljni vizuelnom spoznajom koju nam je priroda dala. Radoznalost nas je navela da stvorimo mašine koje mogu da "vide" kao mi, pa čak i inteligentnije." - Li Fei-Fei [3]

1.1 Istorija i motivacija

1.1.1 Rani Razvoj u Računarskom Vidu

Koreni računarskog vida potiču iz ranih pokušaja da se razume i interpretira vizuelni podatak korišćenjem matematičkih modela i računara. U početku, istraživanja u oblasti računarskog vida fokusirala su se na jednostavne zadatke kao što su detekcija ivica, prepoznavanje objekata i osnovna obrada slika. Rane metode su se u velikoj meri oslanjale na ručnu izradu karakteristika (engl. ***features***) slike i algoritme dizajnirane da imitiraju osnovne aspekte ljudskog vida.

1.1.2 Uspon Dubokog Učenja

Značajan preokret u računarskom vidu dogodio se sa pojavom **dubokog učenja**. **Konvolucione Neuronske Mreže** (**CNNs**), koje su predstavili Yann LeCun i drugi [9] krajem 1980-ih i početkom 1990-ih, revolucionisale su ovu oblast uvođenjem automatskog ekstraktovanja karakteristika kroz slojeve koji se uče (engl. ***learnable features***). **CNN**-ovi su pokazali izuzetne performanse u različitim zadacima klasifikacije slika, omogućavajući računarima da nauče složene reprezentacije vizuelnih podataka. Ovo otkriće je kasnije propaćeno uspehom modela kao što su **AlexNet** [7], **VGGNet** [12] i **ResNet** [4], koji su postavili nove standarde u izazovima prepoznavanja slika.

1.1.3 Ograničenja CNN-ova

Uprkos svom uspehu, **CNN**-ovi imaju inherentna ograničenja koja su motivisala potragu za novim pristupima. Jedan od značajnih nedostataka je njihova poteškoća u povezivanju udaljenih zavisnosti i globalnog konteksta unutar slike. **CNN**-ovi obično obrađuju slike kroz seriju lokalizovanih konvolucionih operacija, što može ograničiti njihovu sposobnost da razumeju odnose između udaljenih elemenata na slici.

1.1.4 Motivacija za uvođenje Vision Transformera

Pojava **Vision Transformera** (**ViT**) predstavlja odgovor na ova ograničenja. Inspirisani uspehom modela transformera u obradi prirodnog jezika (**NLP**), istraživači su pokušali da primene iste principe u računarskom vidu. Transformeri koriste ***self-attention*** mehanizam za povezivanje globalnih zavisnosti, što ih čini pogodnim za zadatke koji zahtevaju razumevanje složenih odnosa unutar vizuelnih podataka.

Vision Transformeri rešavaju nekoliko izazova sa kojima se suočavaju **CNN**-ovi. Pretvaranjem slika u sekvence parčića (engl. ***image patches***) i primenom ***self-attention*** mehanizma preuzetim iz **transformera**, **ViT**-ovima mogu efikasnije modelovati globalni kontekst. Ovaj pristup omogućava **ViT**-ovima da postignu vrhunske performanse na različitim testovima klasifikacije slika i pokazuje njihov potencijal da unaprede oblast računarskog vida.

1.2 Konvolucione Neuronske Mreže (CNN)

1.2.1 Istorija CNN-ova

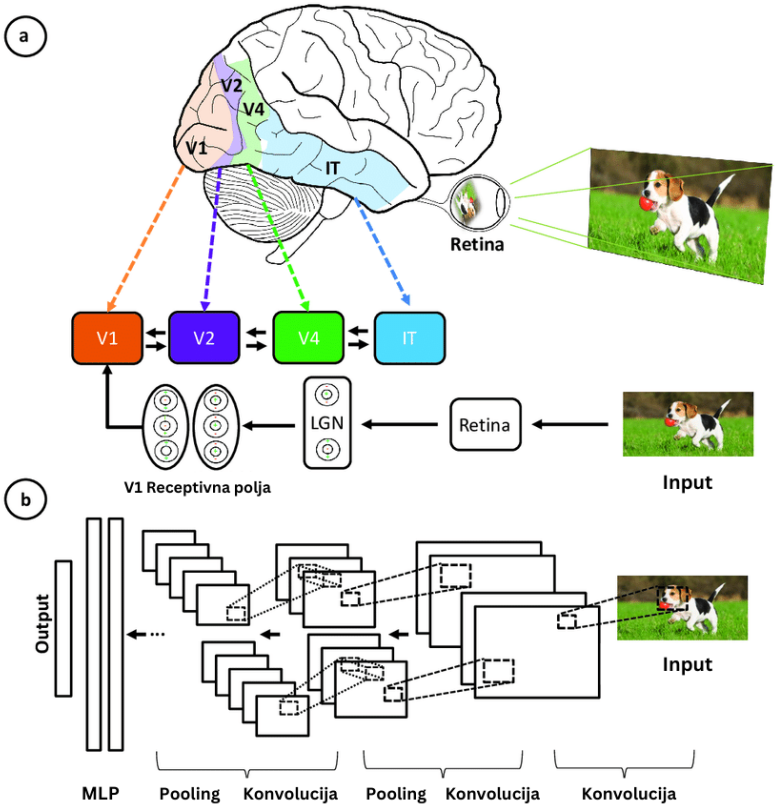
Konvolucione neuronske mreže su razvijene i prvi put korišćene oko 1980-ih. Tokom tog perioda, primarna primena **CNN**-ova bila je prepoznavanje rukom pisanih cifara, što je našlo praktičnu primenu u poštanskom sektoru za čitanje poštanskih i PIN kodova. Rani modeli **CNN**-ova, kao što je **LeNet** [8] koji je razvio Yann LeCun, pokazali su potencijal CNN-ova za zadatke prepoznavanja cifara.

Međutim, šira primena **CNN**-ova bila je ograničena značajnim izazovima. Duboki modeli učenja, uključujući **CNN**-ove, zahtevaju ogromne količine podataka za obuku i značajne računarske resurse, koji u to vreme nisu bili lako dostupni. Pored toga, ***backpropagation*** algoritam, koji je neophodan za obuku neuronskih mreža, bio je računarski skup. Ova ograničenja su ograničila upotrebu **CNN**-ova uglavnom na poštanski sektor, i tehnologija nije uspela da stekne širu primenu u oblasti mašinskog učenja.

Oživljavanje **CNN**-ova došlo je 2012. godine, kada su Alex Krizhevsky, zajedno sa Ilyom Sutskeverom i Geoffreyjem Hintonom, prepoznali potencijal dubokog učenja sa višeslojnim neuronskim mrežama. Ovo oživljavanje je pokrenuto nekoliko ključnih faktora: dostupnost velikih skupova podataka, kao što je ***ImageNet*** [1] skup sa milionima označenih slika, i značajna unapređenja u računarskim resursima, posebno **GPU**-ovima (engl. ***graphics processing unit***). Ovi razvojni događaji omogućili su istraživačima da prevaziđu prethodna ograničenja i u potpunosti iskoriste mogućnosti konvolucionih neuronskih mreža.

1.2.2 Ljudski vizuelni sistem kao inspiracija

Arhitektura konvolucionih neuronskih mreža je analogna načinu na koji su neuroni u ljudskom mozgu povezani i inspirisana je organizacijom **vizuelnog korteksa**. Pojedinačni neuroni reaguju na stimuluse samo u ograničenom delu vizuelnog polja poznatom kao **receptivno polje** (engl. ***receptive field***). Ova polja se preklapaju kako bi se pokrilo čitavo vizuelno područje.



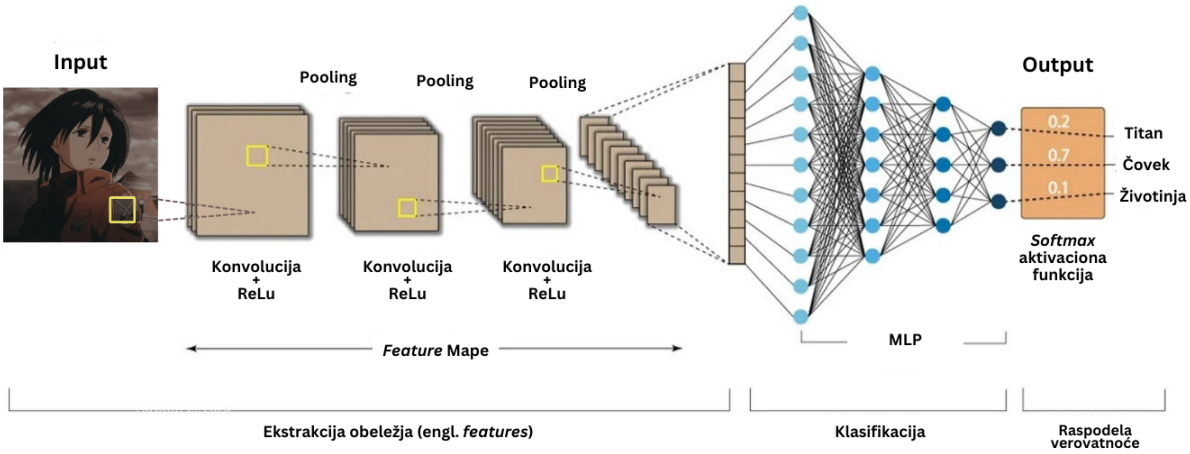
Slika 1: Paralela između organizacije vizuelnog korteksa i **CNN** arhitekture

Glavne sličnosti [6] između organizacije vizuelnog korteksa i arhitekture **CNN**-ova su:

- **Hijerarhijska arhitektura**: I **CNN** i vizuelni korteks imaju hijerarhijsku strukturu, gde se jednostavni oblici izvlače u početnim slojevima, a složeniji oblici se grade u dubljim slojevima. Ovo omogućava kompleksniju reprezentaciju vizuelnog inputa.
- **Lokalna povezanost**: Neuroni u vizuelnom korteksu su povezani samo sa lokalnim regionom inputa, a ne sa celim vizuelnim poljem. Slično tome, neuroni u sloju **CNN**-a su povezani samo sa lokalnim regionom inputa putem **konvolucione operacije**. Ova lokalna povezanost omogućava efikasnost.
- **Translaciona invarijantnost**: Neuroni vizuelnog korteksa mogu detektovati karakteristike bez obzira na njihovu lokaciju u vizuelnom polju. ***Pooling*** slojevi u **CNN**-u pružaju određeni stepen **translacione invarijantnosti**.
- **Višestruke *feature* mape**: Na svakoj fazi vizuelne obrade, izvlače se različite ***feature*** mape. **CNN**-ovi ovo imitiraju putem višestrukih jezgara (engl. ***kernels***) koji detektuju različite karakteristike u svakom konvolucionom sloju.
- **Nelinearnost**: Neuroni u vizuelnom korteksu pokazuju osobine nelinearnosti. **CNN**-ovi postižu nelinearnost putem **aktivacionih funkcija**, koje se primenjuju nakon svake konvolucije.

1.2.3 Arhitektura CNN-ova

Arhitektura **CNN**-a biće prikazana na primeru klasifikacionog problema[1], gde je cilj modela da klasifikuje slike u jednu od P klasa, kao što je prikazano na slici 2.



Slika 2: Primer arhitekture **CNN**-a za klasifikaciju slika

Sastavni deo **CNN**-a su:

- **Konvolucioni slojevi**
- ***Pooling* slojevi**
- **Višeslojni perceptron (engl. *Multi-Layer Perceptron* - MLP)**

[1]Gradivni elementi CNN-a su identični, bez obzira na to da li je u pitanju problem regresije ili druge prirode; jedina razlika leži u MLP sloju

Konvolucioni slojevi

1. Konvoluciona operacija

U **konvolucionom sloju**, osnovna operacija je **konvolucija** ulazne slike sa **kernelom** (poznatim i kao **filter**). Cilj ove operacije je detektovanje karakteristika kao što su ivice, teksture ili šare u ulaznim podacima.

Neka je ulazna slika predstavljena kao 2D matrica I sa dimenzijama $H \times W$, gde je H visina, a W širina slike (pogledati sliku 3). Neka je K 2D kernel sa dimenzijama $f \times f$, gde je f veličina kernela.

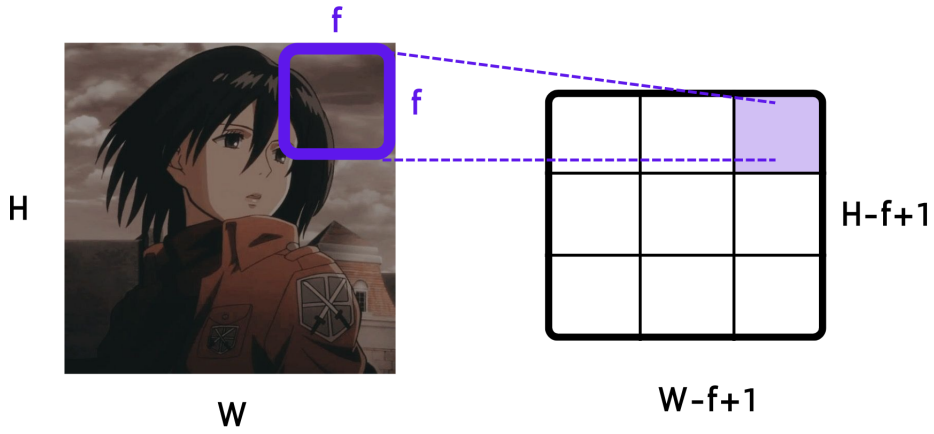
Konvolucija operacija se može matematički izraziti kao:

$$S(i,j) = \sum_{m=0}^{f-1} \sum_{n=0}^{f-1} I(i+m, j+n) \cdot K(m,n)$$

gde $S(i,j)$ predstavlja vrednost izlazne ***feature*** mape na poziciji (i,j). Ovde, (i,j) označava poziciju na ulaznoj slici gde se primenjuje kernel.

2. Izlaz konvolucione operacije

Rezultat primene kernela na ulaznu sliku je *feature* mapa F sa dimenzijama $(H - f + 1) \times (W - f + 1)$, pod pretpostavkom da se ne koristi *padding*. Veličina *feature* mape je smanjena u odnosu na ulaznu sliku zbog klizne operacije kernela.



Slika 3: Dimenzije ulazne slike i *feature* mape u jednom konvolucionom sloju

3. Popunjavanje (engl. Padding)

Popunjavanje se koristi za kontrolu prostornih dimenzija izlazne *feature* mape. Popunjavanje dodaje dodatne piksele oko ivica ulazne slike. Neka je p veličina popunjavanja. Popunjena ulazna slika I' ima dimenzije $(H + 2p) \times (W + 2p)$.

Konvolucija operacija sa popunjavanjem može se izraziti kao:

$$S(i,j) = \sum_{m=0}^{f-1} \sum n = 0^{f-1} I'(i+m, j+n) \cdot K(m,n)$$

4. Korak (engl. Stride)

Korak određuje koliko piksela se filter pomera pri svakom koraku tokom konvolucije. Neka je s vrednost koraka. Korak utiče na dimenzije izlazne karakteristične mape. Sa korakom s, izlazna karakteristična mapa F ima dimenzije:

$$H_{\text{out}} = \frac{H - f + 2p}{s} + 1$$

$$W_{\text{out}} = \frac{W - f + 2p}{s} + 1$$

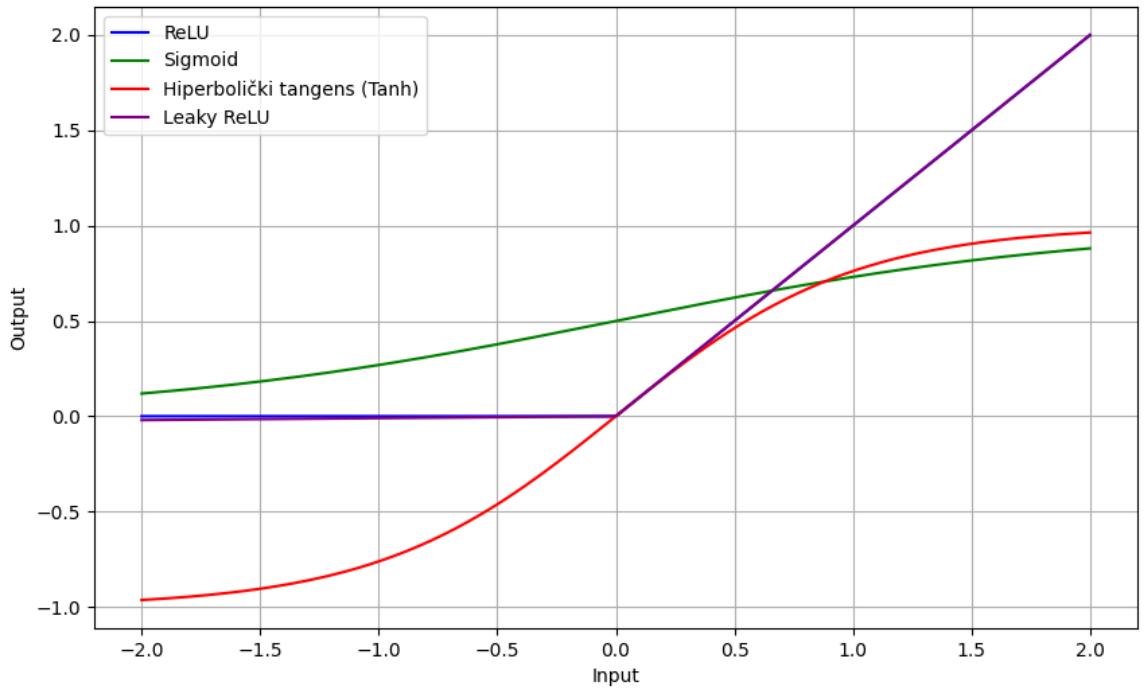
gde su H_{out} i W_{out} visina i širina izlazne *feature* mape, respektivno.

5. Aktivaciona Funkcija

Nakon konvolucione operacije, aktivaciona funkcija se primenjuje element po element da bi se uvela nelinearnost u model. Najčešća aktivaciona funkcija koja se koristi je **ReLU** (*Rectified Linear Unit*), definisana kao:

$$\text{ReLU}(x) = \max(0, x)$$

gde je x ulaz u aktivacionu funkciju. Neke od najčešćih aktivacionih funkcija se mogu videti na slici 4.



Slika 4: Grafik najčešće korišćenih aktivacionih funkcija

Pooling slojevi

Pooling slojevi su još jedna ključna komponenta **CNN**-ova koja vrši operaciju uzorkovanja po određenoj strategiji kako bi se smanjile prostorne dimenzije ulazne *feature* mape, čime se smanjuje računarska složenost i sprečava prekomerno prilagođavanje (engl. *overfitting*). Najčešće korišćene operacije pooling-a su ***max pooling*** i ***average pooling***.

Pooling operacija

Pooling operacije se primenjuju na svaku *feature* mapu nezavisno. Ulazna *feature* mapa F ima dimenzije $H \times W$, gde je H visina, a W širina. *Pooling* operacija pomera prozor veličine $f \times f$ preko *feature* mape , sa korakom s.

Max Pooling

Kod *max pooling*-a, izlazna vrednost za svaki prozor je maksimalna vrednost unutar tog prozora. Matematički, za *feature* mapu F i *pooling* prozor veličine $f \times f$ sa korakom s, operacija *max pooling*-a se može izraziti kao:

$$M(i,j) = \max_{0 \leq m < f, 0 \leq n < f} F(s \cdot i + m, s \cdot j + n)$$

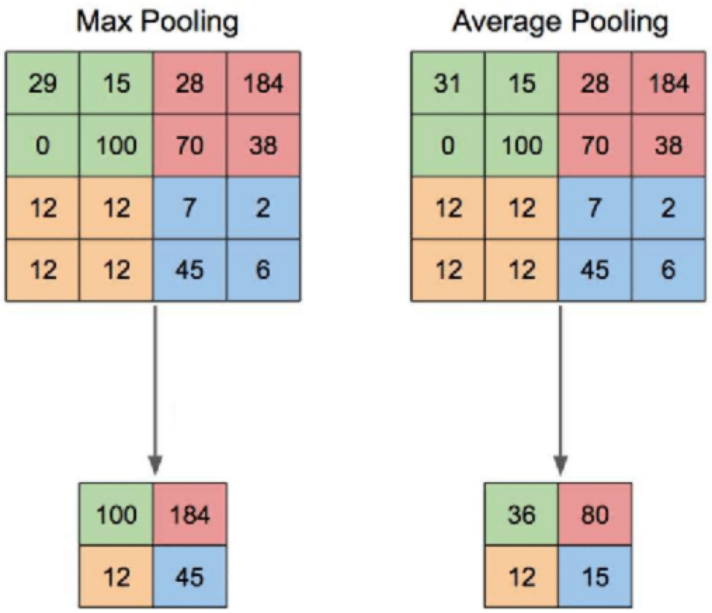
gde $M(i,j)$ predstavlja vrednost izlazne *feature* mape na poziciji (i,j).

Average Pooling

Kod *average pooling*-a, izlazna vrednost za svaki prozor je prosečna vrednost unutar tog prozora. Matematički, za *feature* mapu F i *pooling* prozor veličine $f \times f$ sa korakom s, operacija *average pooling*-a se može izraziti kao:

$$A(i,j) = \frac{1}{f^2} \sum_{m=0}^{f-1} \sum_{n=0}^{f-1} F(s \cdot i + m, s \cdot j + n)$$

gde $A(i, j)$ predstavlja vrednost izlazne mape karakteristika na poziciji (i, j).



Slika 5: Primer izvođenja *max pooling* i *average pooling* operacije

Izlaz Pooling-a

Dimenzije izlazne *feature* mape zavise od veličine prozora za pooling f i koraka s. Data ulazna *feature* mapa F sa dimenzijama $H \times W$, izlazna *feature* mapa O (bilo M za *max pooling* ili A za *average pooling*) ima dimenzije:

$$H_{\text{out}} = \frac{H - f}{s} + 1$$
$$W_{\text{out}} = \frac{W - f}{s} + 1$$

gde su H_{out} i W_{out} visina i širina izlazne *feature* mape, respektivno.

Višeslojni Perceptron (MLP)

Perceptron

Perceptron prima više ulaznih signala, primenjuje težine na njih, sabira ih i prosleđuje rezultat kroz aktivacionu funkciju kako bi proizveo izlaz (*slika 6*).

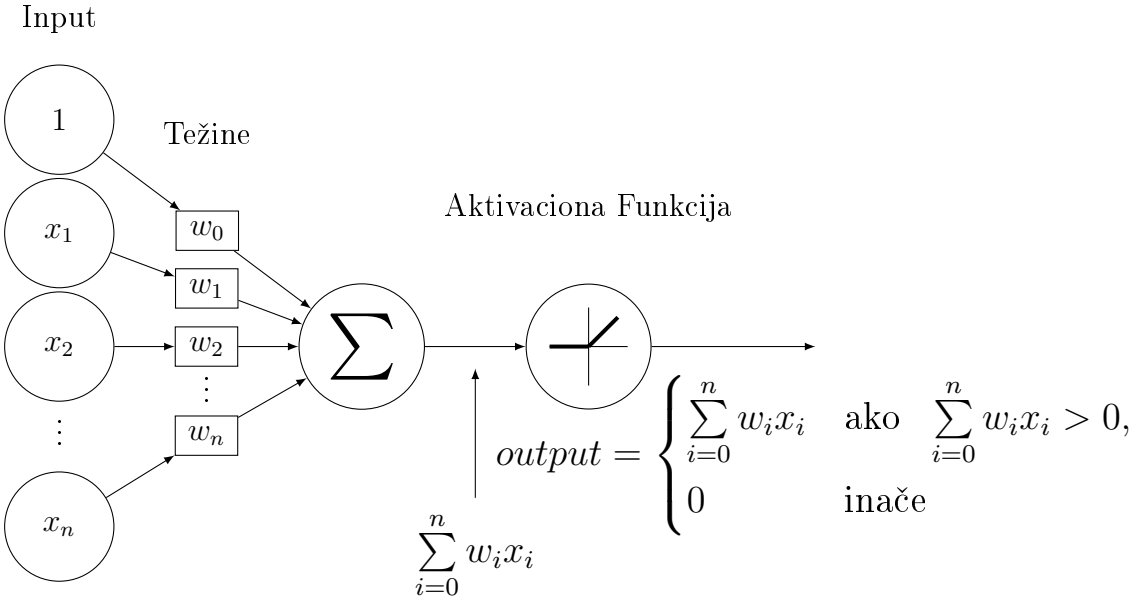
Za dat ulazni vektor $\mathbf{x} = [x_1, x_2, \ldots, x_n]$ i odgovarajuće težine $\mathbf{w} = [w_1, w_2, \ldots, w_n]$, perceptron računa ponderisanu sumu na sledeći način:

$$z = \sum_{i=1}^{n} w_i x_i + b$$

gde je b (*bias*) slobodan član.

Izlaz y se zatim dobija primenom aktivacione funkcije $f(z)$. U našem primeru smo za aktivacionu funkciju koristili **ReLU** funkciju, koja se definiše na sledeći način:

$$y = \begin{cases} z & \text{ako } z > 0 \\ 0 & \text{inače} \end{cases}$$



Slika 6: Grafički prikaz perceptrona

Višeslojni Perceptron (MLP)

MLP je potpuno povezana veštačka neuronska mreža koja se sastoji od više slojeva perceptrona, obično uključujući ulazni sloj, jedan ili više skrivenih slojeva i izlazni sloj.

Za **MLP** sa L slojeva, ulaz u mrežu je $\mathbf{x} \in \mathbb{R}^n$. Izlaz svakog sloja l se računa kao:

$$\mathbf{h}^{(l)} = f(\mathbf{W}^{(l)}\mathbf{h}^{(l-1)} + \mathbf{b}^{(l)})$$

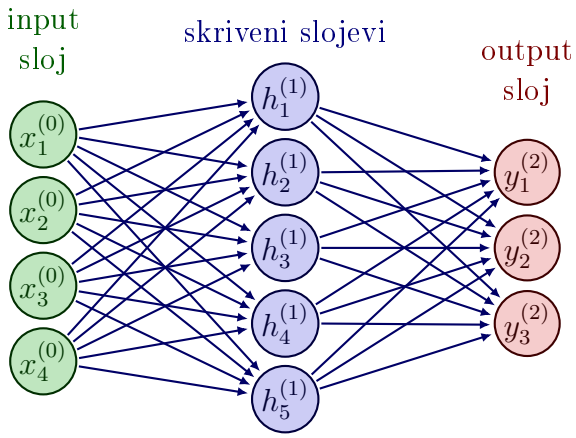
gde:

- $\mathbf{h}^{(0)} = \mathbf{x}$ je ulazni vektor,
- $\mathbf{W}^{(l)}$ i $\mathbf{b}^{(l)}$ su matrica težina i vektor pristrasnosti za sloj l,
- f je aktivaciona funkcija (obično *ReLU*, *sigmoid* ili *tanh*).

Finalni sloj često koristi *softmax* funkciju za klasifikacione zadatke, koja pretvara izlaz u distribuciju verovatnoće za date klase:

$$\text{softmax}(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{P} e^{z_j}}$$

gde je $\mathbf{z}$ ulaz u softmax funkciju, a P je broj klasa.



Slika 7: Grafički prikaz **MLP**-a sa 4 ulazna, 5 skrivenih i 3 izlazna čvora

2 Transformeri

2.1 Istorija i razvoj

Model **Transformera**, predstavljen u seminalnom radu *"Attention is All You Need"* [13] od Vaswani-ja i saradnika 2017. godine, označio je značajan napredak u oblasti obrade prirodnog jezika (**NLP**). Pre toga, modeli kao što su *Recurrent Neural Networks* (**RNNs**) [11] i *Long Short-Term Memory Networks* (**LSTMs**) [5] bili su dominantne arhitekture za *sequence-to-sequence* zadatke. Međutim, ovi modeli su imali ograničenja, posebno sa udaljenim zavisnostima i paralelizacijom.

Transformeri su revolucionisali **NLP** uvođenjem nove arhitekture koja se u potpunosti zasniva na ***self-attention*** mehanizmu, eliminišući potrebu za rekurentnim slojevima. Ova inovacija je omogućila efikasnije treniranje i sposobnost da se bolje modeluju veze između udaljenih reči u sekvenci. Uvođenje transformera dovelo je do dramatičnog poboljšanja performansi na različitim **NLP** zadacima, kao što su mašinsko prevođenje, sažimanje teksta i odgovaranje na pitanja.

Njihov uspeh brzo je postao evidentan razvojem moćnih modela koji su ih koristili kao osnovu. Jedna od prvih značajnih primena bila je u mašinskom prevođenju, gde je transformer nadmašio prethodne najbolje modele na referentnim skupovima podataka. Ovaj uspeh je dodatno pojačan stvaranjem modela kao što su **BERT** *(Bidirectional Encoder Representations from Transformers)* [2] i **GPT** *(Generative Pre-trained Transformer)* [10], koji su postavili nove standarde za različite **NLP** zadatke.

Reference

[1] Jia Deng et al. "ImageNet: A large-scale hierarchical image database". In: (2009), pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[2] Jacob Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: (2019). arXiv: 1810.04805 [cs.CL]. URL: https://arxiv.org/abs/1810.04805.

[3] Li Fei-Fei. *With spatial intelligence, AI will understand the real world.* URL: https://www.ted.com/talks/fei_fei_li_with_spatial_intelligence_ai_will_understand_the_real_world.

[4] Kaiming He et al. "Deep residual learning for image recognition". In: (2016), pp. 770–778.

[5] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-term Memory". In: *Neural computation* 9 (Dec. 1997), pp. 1735–80. DOI: 10.1162/neco.1997.9.8.1735.

[6] Zoumana Keita. *An Introduction to Convolutional Neural Networks (CNNs).* URL: https://www.datacamp.com/tutorial/introduction-to-convolutional-neural-networks-cnns.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet classification with deep convolutional neural networks". In: NIPS'12 (2012), pp. 1097–1105.

[8] Y. Lecun et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324. DOI: 10.1109/5.726791.

[9] Yan LeCun. "Backpropagation applied to handwritten zip code recognition". In: *Neural Computation* 1 (1989), pp. 541–551. DOI: 10.1162/neco.1989.1.4.541.

[10] Alec Radford et al. "Language Models are Unsupervised Multitask Learners". In: (2019).

[11] David E. Rumelhart and James L. McClelland. *Learning Internal Representations by Error Propagation.* 1987, pp. 318–362.

[12] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: (2015).

[13] Ashish Vaswani et al. “Attention Is All You Need”. In: (2023). arXiv: 1706.03762 [cs.CL]. URL: https://arxiv.org/abs/1706.03762.