**Neuron**

* Základní blok sítě
* Přijímá vstup, na který aplikuje váhy a přičítá bias
* Pomocí aktivační funkce generuje výstup

**Vstupní vrstva**

* První vrstva neuronové sítě, která přijímá surová data
* Bez výpočtů, pouze distribuuje data do dalších vrstev

**Skrytá vrstva**

* Všechny vrstvy mezi vstupní a výstupní vrstvou
* Provádí většinu výpočtu a zpracování

**Výstupní vrstva**

* Poslední vrstva sítě, poskytuje konečný výstup modelu
* Přijímá informace ze skrytých vrstev a generuje výstup

**Konvoluční vrstva**

* Používaná pro zpracování obrazu
* Aplikuje konvoluční operace na vstupní data, které pomáhají zachytit prostorovou hierarchii v obrazu
* Přes obraz se posouvají malé matice, které tyto operace aplikují
* Jako aktivační funkce se používá ReLu
* Používá se jako první (popř. prvních pár) vrstva při práci s obrazem

**Lineární (plně propojená) vrstva**

* Používá se jako pozdě vrstva
* Propojuje všechny neurony z přechozí vrstvy s každém neuronem této vrstvy
* Každé propojení má váhu a bias

**Váhy**

* Určují, jakou důležitost mají jednotlivé vstupy neuronu při výpočtu výstupu
* Každý vstup se násobí jeho váhou a součet těchto výsledků se používá při aplikování aktivační funkce

**Prahové hodnoty (bias)**

* Slouží k „posunutí“ vstupní hodnoty
* Hodnota, která je přičtena k váženému součtu vstupů před aplikování aktivační funkce

**Aktivační funkce**

* Rozhoduje, zdali se neuron aktivuje na základě jeho vstupu
* Využívá se nelineární transformace výstupu (např. ReLu nebo softmax)
* Aplikuje se váženém součtu vstupů a přidaní prahové hodnoty
* Určuje konečný výstup neuronu

**Ztrátová funkce (Loss, cost function)**

* Rozdíl mezi předpokládaným a skutečným výstupem,
* Určuje, jak dobře model funguje
* Slouží k optimalizaci (k aktualizaci vah)

**Optimalizační algoritmus**

* Upravuje váhy a prahové hodnoty sítě
* Např. gradientní sestup – upravuje na základě gradientu ztrátové funkce

**Zpětná propagace**

* Proces, kdy se chyby šíří zpět po sítí, aby byly aktualizovány váhy pomocí optimalizačního algoritmu