

Storyline - Advanced Neural Networks

Monday, January 27, 2025 3:34 PM

Секој елемент од моделот има своја тежина која го контролира неговото учење за да се подобрат неговите предвидувања.

- **Backpropagation** е техника која е се користи за оптимизација на тежините преку минимизирање на грешката на излезот (ја подобрува прецизноста и ја намалува грешката за време на процесот на обука)
- 1) **Forward pass=>** се пресметува грешката како разлика помеѓу излезот и целта
- 2) **Backward pass=>** грешката се враќа назад низ мрежата така што се пресметуваат градиенти на тежина и bias за секој слој
- Откако ќе се пресметаат градиентите во овој чекор тие се користат за ажурирање на тежините преку алгоритми како ГРАДИЕНТЕН СПУСТ

Ажурирање на тежини:

---> Како да се најдат најдобрите тежини ? На секоја тренинг сесија извршуваме градиентен спуст: го прегледуваме излезот на моделот, ја пресметуваме грешката и ја ажурираме тежината за истата да се намали

- Понекогаш ова се прави за секоја слика/пример поединечно (**ОБУКА ПО ПРИМЕР** - како што е Stochastic Gradient Descent)
- Или се прави со групирање на **сите примери во mini batches** - за големи податоци и + е помалку подложно на предрасуди од издвоени податоци

Кога моделите стануваат премногу сложени за да ги запаметат сите детали тие најчесто стануваат преобучени за тренинг податоците но не и нови (overfitting) треба да се примени регуларизација - казнување

- **Dropout** - техника каде случајно исклучуваме некои неврони од процесот на обука на секоја итерација. Така се спречува мрежата да се потпре само на неколку неврони и станува поопшта.
- **Early stopping** - техника каде ја запирате обуката кога гледаме дека почнува да учи премногу специфични детали и моделот не станува подобар/поефикасен

Споредба на Dropout и Early Stopping:

- Dropout го трга фокусот од претерано зависни неврони и воведува елемент на случајност во мрежата
- Early Stopping го спречува понатамошното учење и претставува превентативна мерка

Невронската мрежа е изградена врз основа на слоеви кои работат заедно

- **Целосно поврзан слој:** секој неврон е **поврзан со сите** неврони од претходниот
 - Претераното поврзување може да предизвика преоптоварување на системот
- **Слој со конвулација (CNN):** се фокусира на помали детали наместо да гледа се наеднаш.
 - Секој неврон е поврзан само со локален регион од претходниот слој
 - Пример, ако се работи за голема слика наместо да се обработи целата наеднаш конвулацијата ќе се фокусира на мал дел и ќе ги анализира деталите во тој мал регион.
 - Значително се намалува бројот на параметри и ја прави мрежата поефикасна

За да се намалат димензиите и да се фокусира на најзначајните карактеристики и да ги подобри истите:

Концепт на Pooling --> кога има голем број на детали и треба да се сумираат сите важни нешта, така овој процес извлекува најважни информации од секој регион

- Maxpooling --> ќе се избере најголемиот број од сите броеви во даден регион и така останува најважната вредност

- Моделот не се оптеретува со премногу детали туку ги задржува клучните карактеристики кои му помагаат да прави точни одлуки

Автоенкодери

- Ако слика е претставена како голем блок податоци треба да ја компресираме или намалиме па затоа служат автоенкодерите.
 - Енкодер - ги зема сите информации и ги компресира во помал формат
 - Декодер - ги враќа компресираните податоци назад во поголеми
- Ова помага во намалување на сложеноста и намалување на шумот
- **Длабоки автоенкодери** -> се додаваат **повеќе скриени слоеви** што овозможува моделот да долови и **повисоко апстрактни карактеристики** од податоците
- Автоенкодерите се користат за намалување на димензионалноста така што учат компактни репрезентации на податоците во скриениот слој.
 - Пример слика со 10 000 карактеристики може да биде претставена од скриениот слој како 100 карактеристики

Заклучок:

- 1) чистење на податоци
- 2) Откривање на аномалии
- 3) Компресија на податоци
- 4) Зачувување на структурата

Меморија

- Секој неврон во обична мрежа е без меморија односно се сеќава само на тековната информација, но кога извршуваме задача како распознавање на текст или покомплексна анализа тогаш треба да се сетиме на претходните информации.
 - За тоа служат повторните невронски мрежи **RNN кои имаат моќ да запамтат информации од претходните чекори (погодно за обработка на секвенциски податоци како текст, аудио)**

Обучувањето на RNN е преку BPTT (Backpropagation Through Time)

- Се минимизира вкупниот трошок при ажурирање на тежините из сите чекори
- Накратко, кај RNN излезот од секој чекор не зависи само од моменталниот влез туку и од претходните излези, но имаат ограничувања за долготрајни зависности.
- **Исчезнувачки градиенти проблем**
- **Примена: анализирање кратки текстови и распознавање звуци**
- Понекогаш и овие агенти можат да заборават или да не задржат информација па затоа доаѓаат **LSTM (Long Short-Term Memory)** мрежите кои имаат капацитет да задржат информацији долго време а оние кои не се толку важни да ги избришат.
 - Möglichkeit за избор кои информации ќе бидат задржани а кои заборавени што е клучно за задачи како разбирање на текст каде се следат долготрајни зависности.
 - **Се чуваат, запишуваат, читаат и бришат**

Карактеристики на LSTM:

- LSTM невроните имаат **4 влезови и 1 излез** што им овозможува поефикасно управување со информациите, има меморија со геирани клетки
 - Input Gate (кои информации ќе влезат)
 - Forget Gate (кои информации ќе се избришат)
 - Output Gate (генерира излез на клетката)

Обучувањето на LSTM е преку BPTT и Gradient Descent

- **Го решава проблемот со исчезнувачки градиенти**
- **Примена: превод на јазици, генерирање текст**
- **Двоначочни RNN =>** обработка на информации од минатото и иднината, пример при анализа на реченици се земат контекстот пред и по зборот.

Паралела:

- Pooling помага да се извличат најважните информации, автоенкодерите го компресираат значењето за го направат поефикасно и меморијата ја овозможува способноста на мрежата да се секава на претходни информации.