Laboration Maskininlärning

1 Rekommendationssystem

Rekommendationssystem är en idag avgörande funktion för sociala medier och distribution av musik, film, spel med mera. De två största metoderna som används är content filtering och collaborative filtering. Använd internet och resurserna i kursrepot för att läsa om dessa metoder. Denna laboration antar främst att det rör sig om collaborative filtering, men det är på inget sätt fel att använda collaborative filtering– teknikerna är snarlika och det handlar främst om hur man designar sina features.

1.1 Förberedelser

Ladda ned ml-latest.zip under sektionen "recommended for education and development" från movielens. Undersök datamängden.

Filerna genome_* är utdatan från ett annat maskininlärningsprojekt och kan ignoreras för denna uppgift.

Filen `link.csv` innehåller korsreferenser mellan olika filmsajter och kräver mer avancerade tekniker för att dra nytta av. Om du bygger en dash applikation kan det vara värt att inkludera länkarna.

Undersök `movies.csv` filen och särskilt kolumnen genres. Det finns strax över 85000 filmer och ca 20 genrer.

Undersök `ratings.csv` och fundera över vad distributionen av värden innebär. Borde någon sorts skalering användas? Det finns över 30 miljoner rader i denna fil och den tar ca 1 GB i minnet. Beroende på hur mycket datakraft du har tillgänlig kan det bli väldigt krävande att processa hela filen. Fundera på om du kan göra något statistikt urval eller kanske kan du dra värden slumpmässigt ur denna fil under ett träningsskede (om du använder en metod som kräver träning).

Undersök `tags.csv` och fundera på om du kan använda den datan på något sätt. Går det att använda någon teknik för naturligt språk?

Filmer och användare har unika identifierare i datan och korsrefereras mellan filerna. Fundera på hur du vill bygga en gemensam datamängd– vilken data vill du använda?

1.2 Utförande

I denna laboration skall du själv välja ut en featuremängd och metod för att rekommendera filmer. Du skall sedan implementera ett enklare system som rekommenderar fem filmer givet en inmatad film.

De två mest rättframa sätten att använda datan är antingen att försöka förutsäga ett betyg för filmerna eller att identifiera likheter mellan filmer baserat på genrer. I det första fallet räcker det att göra en värderegression på ”rating” och slumpmässigt dra träningsmängder ur datan. I det andra fallet räcker en one-hot encoding på genrer och en KNN-Transform med cosinus-likhet som metrik. Den första tekniken fungerar bäst som content-filtering metod. Genom att träna på mängder av liknande användare kan förutsägelserna bli bra. Resultaten av att bara använda genre som likhetsmått är dock en besvikelse.

I exemplet från lektion kodades likhet mellan betyg genom att skapa en kolumn per användare med värdet för filmens betyg. I detta exempel innebär det en 86500×330000 matris, vilket tar cirka 26 GB minne, som är en begränsande faktor på många hemdatorer. Genom att filtera dessa mängder, i likhet med vad som gjordes i bok-exemplet så kan denna metod ge bättre rekommendationer.

Mycket bättre resultat kan uppnås genom att kombinera metadatan i `tags.csv` och genrerna i `movies.csv`. Genom att behandla genrerna som fler nyckelord i tags-datan, och använda till exempel TF-IDF vektorisering, kan likhet mellan filmer fångas på ett lite djupare plan.

Tillslut kan förstås flera tekniker kombineras, tillexempel genom att först hitta en större mängd lika filmer och sedan förutsäga ratings på dem för att hitta de bästa fem. Eller KMeans klustring kring filmerna som rekommenderas för att öka diversiteten bland rekommenderade filmer.

Det är upp till dig att välja ambitionsnivå. För det högre betyget skall åtminstone någon förbättrad teknik användas och dokumentationen skall vara utförlig och av god kvalité.

1.2.1 Inlämning

Denna laboration skall lämnas in som kod i första hand. Skriv koden enligt något av följande:

- filen kan köras som ett icke-interaktivt skript (dvs filmen anges på kommandoraden)
- progammet körs som en dash-applikation (eller motsvarande)
- programmet har en funktion/klass som kan importeras i en .ipynb fil, som då inte skall innehålla någon annan kod än den som exporteras från python-filen. Inga listningar eller onödiga figurer här.

Du behöver dokumentera din kod och även skriva en utförlig README.md eller .ipynb Jupyter notebook som förklarar vilka metoder du använder, vilka begränsningar som gäller och vilka val du gjort.