Filmfortjeneste

Gruppe 18, 15.11.2023

Hvordan skal en filmprodusent kunne vite om filmen de nå vurderer å produsere vil gi en real fortjeneste eller ikke? Er det enkelte egenskaper ved en film som gjør det mer eller mindre sannsynlig at den slår an? Eller en kombinasjon av egenskaper?

Mål

Målet med prosjektet har vært å lage en maskinlæringsmodell som vurdere utifra noen parametere hvor mye den aktuelle filmen kommer til å tjene. I dag finnes det ingen gode tydelige retningslinjer man kan gå etter for å forutse om en film kommer til å gjøre det bra eller dårlig.

Vi ønsket derfor å lage en modell som kunne hjelpe med å undersøke om det er noen egenskaper ved en film som gjør det mer eller mindre sannsynlig at den gjør det bra. En vellykket modell ville ha revolusjonert filmbransjen, da det er mye penger i film og mange er i bransjen for å tjene penger. Hvis man med sikkerhet kunne sagt at enkelte egenskaper vil garantert sikre god inntekt eller garantert fått en film til og floppe ville dette vært verdt en god del penger for en del produsenter.

For at denne modellen skal regnes som noe vellykket må den klare å produsere en treffsikkerhet på over 90% for at vi skulle klart og selge resultatene videre til de mulige aktørene. Hvis modellen klarer å være så treffsikker at produsentene selv søker oss ut for å få en vurdering på sine mulige filmer vil dette videre skape en etterspørsel som vi kunne ha satt en høy prislapp på. Dette er bare mulig om modellen klarer å bli så treffsikker at dem som får en vurdering opplever at dette stemmer ofte nok til at de kan stole på vurderingen modellen gir.

Egenskaper

Da vi skulle velge egenskaper for å trene modellen så vi først på hvilke egenskaper som ikke ville gi modellen mye å jobbe med. Dette er blant annet filmtittel, diverse linker og om filmen tilhører en samling. Vi regnet disse egenskapene for å være mindre verdifulle for resultatet og ville skapt veldig mange kolonner.

Vi valgte å bruke egenskapene: Budsjett, originalt språk, popularitet, utgivelsesdato og lengden på filmen. Disse egenskapene var dem vi tenkte det var størst mulighet for å kunne se en sammenheng i, og som ikke skapte for stort datasett.

Dette er data som kommer fra virkelige filmer fra tidligere år, hentet fra siden “the movie database”. Testsettet består av 4398 filmer. Dataen består av tall og setninger, noe som krevde at vi måtte jobbe gjennom datasettet for å skape et testsett som modellen kunne jobbe med. Etter hvert som flere filmer blir lansert vil all informasjonen vi trenger for å fortsette å trene modellen vår bli offentlig tilgjengelig data, inkludert hvor mye filmene tjener.

Datasettet krevde en del rensing før vi kunne trene modellen på det. Flere egenskaper måtte fjernes, gjøres om til tall og alt måtte skaleres slik at det ikke ble voldsomme hopp mellom de forskjellige numeriske verdiene på egenskapene våre. Vi valgte å beholde egenskapen originalt språk, der har vi gitt alle språkene en numerisk verdi i stedet for en strengrepresentasjon. For utgivelsesdato valgte vi å konvertere dato og tid til millisekunder.

Vi har i noe grad brukt ChartGPT i prosjeket, men kun til å generere kode som konver datoene (release_date) til millisekunder. I tillegg ble ChatGPT brukt får å genere koden som plotter «finne feature importenace» verdiene. Vi brukte ChatGPT får å spare tid i kodeprosessen.

MODELL

Vi har valgt å sjekke ut modellene: Linear regression, Random forest, Decision tree og Bayesian ridge. For å finne den beste modellen har vi valgt cross validation med neg_mean_squared_error, neg_mean_absolute_error og r2. Vi har valgt å bruke linear regresion som baseline-modell og har med det fått

```
Linear reg:
neg_mean_squared_error: 8867734745149617.0
neg_mean_absolute_error: 52890028.34942323
r2: -0.6011309216497362
Random forest:
neg_mean_squared_error: 9482458837792690.0
neg_mean_absolute_error: 48520314.22667322
r2: -0.5648697707059933
Decision tree:
neg_mean_squared_error: 1.5963705847710854e+16
neg_mean_absolute_error: 61834238.233641066
r2: -0.2640278626616787
Bayesian ridge:
neg_mean_squared_error: 1.0367698754362612e+16
neg_mean_absolute_error: 57109315.14075634
r2: -0.5314729041058569
```

Av de modellene vi har sett på er det Random Forest som kommer best ut, som vi kan se på figuren over. Vi velger derfor å velge denne modellen.

For å undersøke feil-vurderinger har vi tenkt å visualisere vurderinger mot korrekt verdi slik at det er enklere å se hvor modellen sliter og se om det er noe felles for disse datapunktene. Vi har også tenkt å se på egenskap-viktighet (feature importance) og se om vi har noen egenskaper som gjør det vanskeligere å gi en god vurdering enn og gjøre det lettere. Utifra dette samt optimalisering av hyperparameterene har vi prøvd og optimalisere modellens vurdering.

For å optimalisere modellen har vi forsøkt å jobbe med hyperparameterene til modellen. Dette ble gjort i to omganger, først ble det brukt random search for å først se om det var noen parameter som påvirket resultatet mer enn andre. Derifra brukte vi resultatet for å justere hyperpameterene i en GridSearch. Dette ga bedre resultater enn modellen gjorde

før hyperparameterjustering og vi valgte derfor å bruke denne versjonen av modellen til vurderingene våre.

Resultat

Modellen ga ikke de resultatene vi ønsket og vil i denne omgang ikke settes i drift. Vi mener vi ikke har fått nok data inn i modellen og at egenskapene den vurderer utifra ikke gir et godt nok helhetlig bilde på hva som skal til for at en film tjener mye penger.

Hvis den hadde fungert som vi hadde ønsket hadde vi solgt det som en tjeneste rettet mot filmprodusenter, hvor de kunne betale oss en viss sum og få en vurdering på om filmen vil tjene penger eller ikke. Vi hadde fortsatt å trene modellen på nye data etter hvert som det kom, og generelt holdt et godt øye med treffsikkerheten.

REFERANSER

https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html
https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74

https://github.com/HVL-ML/DAT158/blob/main/notebooks/DAT158-1.5-Regression.ipynb

https://chat.openai.com