NDMS I: Tekst som features i random forest

Flipped classroom øvelse til valgfaget "Nyere digitale metoder i samfundsvidenskaben I", 2021

I denne øvelse skal I lave en random forest model, der enten prædikterer, om en kommentar på reddit bliver "upvoted" eller prædikterer, om en kommentar bliver "downvoted".

Data er et udtræk af opslag og tilhørende kommentarer fra subreddit r/Denmark fra 1/7-2020 til 31/12-2020 med alle opslag, der indeholder ordet "Danmark" i titel eller opslagstekst ("r/Denmark on Denmark").

Direkte link til data her: https://raw.githubusercontent.com/CALDISS-AAU/course_ndms-I/master/datasets/reddit_rdenmark_q%3Ddanmark_01072020-31122020_tokendummies.zip

Kommentarerne er allerede tokenized (`comment_tokens`). Derudover er de 50 mest signifikante nøgleord i kommentarerne gjort til dummyvariable, som indikerer, hvorvidt ordet indgår i kommentaren eller ej (baseret på tokens). Disse variable har præfix "token_"

Med undtagelse af en outcome-variabel, så er data mere eller mindre klar til at blive fitted. Dog kan det give mening, at lave flere/nye tekstfeatures.

Øvelsen går ud på følgende

1. Dan en outcome-variabel baseret på variablen `comment_score`. I bestemmer selv, om jeres outcome skal være på, hvorvidt kommentaren er upvoted eller downvoted. Bemærk at kommentarer altid starter med en score på 1.
2. Fit en random forest model på jeres outcome variabel med brug af de eksisterende tekst features / token dummies (I kan enten udvælge specfikke features eller inkludere alle).
3. Hvis modellen ikke er god, forsøg da at lave noget hyperparametertuning - enten i et loop som vist i flipped classroom-videoerne eller manuelt.

Ekstra øvelse

1. Vi mistænker, at det primære problem med modellen er, at den er blevet tokenized før, at vi har lavet et resample og den derfor primært har medtaget ord fra ikke-upvoted-bunken. Prøv at danne nye tekst features, baseret på hyppigste termer i de kommentarer, hvor der er success i jeres outcome-variabel (husk at konvertere tokens-variablen sådan, at Python læser værdierne som lister - se under "Værd at vide om data").
2. Fit en ny random forest model med de nye tekst features - forsøg igen med at optimere parametrene.

Værd at vide om data

- Link til data: https://raw.githubusercontent.com/CALDISS-AAU/course_ndms-I/master/datasets/reddit_rdenmark_q%3Ddanmark_01072020-31122020_tokendummies.zip

- Alle variable, der vedrører opslag, har præfix "post_", mens alle variable, der vedrører kommentar, har præfix "comment_".
- I kan tjekke hvilke tekst features / token dummies, der er i data, med følgende kode:

  ```
  [column for column in df.columns if column.startswith('token_')]
  ```

  (hvor `df` referer til dataframe med data).
- Data er allerede tokenized. Tokens ligger som lister i variablen (`comment_tokens`). Pandas kan typisk ikke læse celler, bestående af lister, direkte ind som lister. For at sikre, at Python forstår at variablen består af lister, så skal I køre følgende kode:

  ```
  import ast
  df['comment_tokens'] = df['comment_tokens'].apply(ast.literal_eval)
  ```

  (hvor `df` referer til dataframe med data).
- Data er sammensat af to udtræk fra Pushshift Reddit API (https://github.com/pushshift/api): Et udtræk for opslag (https://github.com/pushshift/api#searching-submissions) og et udtræk af kommentarer for de opslag (https://github.com/pushshift/api#searching-comments). Rækkerne i data er kommentarer, hvilket betyder, at data om opslag gentages for hver kommentar, som vedrører det opslag.

Værd at vide om Reddit

Reddit (https://www.reddit.com/) er et forum-baseret socialt medie, der fungerer som en samling af "underfora". Underfora (kaldet "subreddits") kan oprettes af brugere om hvad som helst. Subreddit "r/Denmark" (https://www.reddit.com/r/Denmark/) er et subreddit rettet mod danskere med diskussion og opslag om dansk kultur, begivenheder, forhold, politik mm.. Opslag og kommentarer er typisk på dansk.

Brugere opretter opslag og kommentarer mere eller mindre anonymt. Hvert opslag kan kommenteres (i tråde) af andre brugere. Brugere kan tilkendegive, hvad de synes om et opslag eller en kommentar ved enten at stemme det op ("upvote") eller ned ("downvote"). Det antages, at en bruger altid stemmer sit eget opslag eller kommentar op, hvorfor et opslag eller kommentar altid starter med en score på 1.