Raport

Dopasowywanie wersów 对联 (duìlián) (chińskich dwuwierszy)

Cel projektu

Celem projektu było stworzenie modelu dopasowującego do pierwszego wersu pewnego dwuwiersza jego drugi wers.

Dane treningowe modelu: pary pierwszy i drugi wiersz.

Wejście modelu: pierwszy wers pewnego dwuwierszu, zbiór drugich wersów dwuwierszy

Wyjście modelu: pasujący do pierwszego wersu drugi wers ze zbioru.

Dane

Dane pochodzą ze zbioru „chinese couplets” z platformy kaggle. Zbiór zawiera 740 tys. dwuwierszy.

Dla modeli 1a i 1b wybrano losowo 1% dwuwierszy ze zbioru, które podzielono na zbiór uczący (6 tys. par wersów) i zbiór testowy (1,5 tys. par wersów).

Dla modelu 2 wybrano losowo 5% dwuwierszy ze zbioru, które podzielono na zbiór uczący (30 tys. par wersów) i zbiór testowy (7,5 tys. par wersów).

Wersy duilian są podporządkowane zasadom określającym m.in. metrum, liczbę znaków i ich ton. Zasady te sprawiają, że wers pierwszy koresponduje z drugim.

Modele

Model 1 (biorący pod uwagę znaczenie):

- Każdy wers jest reprezentowany przez tensor złożony z liczbowych reprezentacji:
  - znaczenia zdania (reprezentowane zanurzeniem BEE) (512 liczb),
  - początkowych głosek znaków w notacji pinyin (pinyin initials) (ograniczane/dopełniane do 35 liczb),
  - końcowych głosek znaków w notacji pinyin (pinyin finals) (ograniczane/dopełniane do 35 liczb),
  - tonów znaków (ograniczane/dopełniane do 35 znaków).
- Dla każdego wersu pierwszego ze zbioru treningowego do uczenia modelu użyto 3 wejść:
  - pary reprezentacji wersu pierwszego i odpowiadającego mu wersu drugiego z etykietą 1,
  - 2 par reprezentacji wersu pierwszego i losowego nieodpowiadającego mu wersu drugiego z etykietą 0.

Model 2 (nie biorący pod uwagę znaczenia):

- Każdy wers jest reprezentowany przez tensor złożony z liczbowych reprezentacji:
  - początkowych głosek znaków w notacji pinyin (pinyin initials) (ograniczane/dopełniane do 35 liczb),
  - końcowych głosek znaków w notacji pinyin (pinyin finals) (ograniczane/dopełniane do 35 liczb),
  - tonów znaków (ograniczane/dopełniane do 35 znaków),

    - liczby znaków wersu (1 liczba),
    - początkowej i końcowej głoski ostatniego znaku wersu i jego tonu (3 liczby).
- Dla każdego wersu pierwszego ze zbioru treningowego do uczenia modelu użyto 5 wejść:
    - pary reprezentacji wersu pierwszego i odpowiadającego mu wersu drugiego z etykietą 1,
    - 4 par reprezentacji wersu pierwszego i losowego nieodpowiadającego mu wersu drugiego z etykietą 0.

Przetestowano modele:

- 1a – model 1 używający MLPClassifier sklearn.
    - Do danego wersu pierwszego dopasowuje wszystkie wersy drugie, dla których MLPClassifier przewiduje etykietę 1 dla reprezentacji pary wersów.
- 1b – model 1 używający MLPRegressor sklearn.
    - Do danego wersu pierwszego dopasowuje wszystkie wersy drugie, dla których MLPRegressor przewiduje wartość większą od 0,9.
- 2 – model 2 używający MLPRegressor sklearn.
    - Do danego wersu pierwszego dopasowuje wszystkie wersy drugie, dla których MLPRegressor przewiduje wartość większą od 0,9.

Ewaluacja

Zmierzono dokładność na danych testowych dobieranych tak jak dane do uczenia (jedna poprawna i 2/5 niepoprawne pary).

Dla 74 wersów zastosowano przystosowaną metrykę precyzji, sprawdzającą czy oczekiwany wers jest wśród proponowanych przez model spośród możliwych 74 drugich wersów i oceniającą wyżej modele proponujące mniej wersów.

Model	Dokładność	Precyzja
1a	0.840268	0.000594
1b	0.750783	0.005119
2	0.808725	0.008174

Wnioski

Znaczenie wersu nie pomaga w dopasowywaniu. Model 2 niebiorący pod uwagę znaczenia zdania, ale uczący się na 5 zamiast 2 przykładach osiąga lepsze wyniki.

Model 1a używający MLPClassifier nieco lepiej dopasowuje wersy spośród małego wyboru drugich wersów. Model 1b używający MLPRegressor znacznie lepiej dopasowuje wersy spośród dużego wyboru drugich wersów.

Prawdopodobnie najważniejsza jest równa liczba znaków w obu wersach. Ważne są wymowa i tony, być może szczególnie ostatniego znaku.