Generowanie tytułów na podstawie abstraktu

Tomasz Torbus
Przetwarzanie Języka Naturalnego

Abstrakt

Wykonanie projektu mającego na celu generowanie tytułów na podstawie abstraktów z prac naukowych. Projekt został zaimplementowany w języku Python, wykorzystując wstępnie wyuczony model T5 do generowania tytułów.

Użyto zbioru prac naukowych z arXiv. Ten zbiór danych zapewnia jakość jak i różnorodność tematyczną. Ogromny zbiór danych wysokiej jakości jest idealny do fine-tuningu modelu T5 który doskonale radzi sobie z wyciąganiem wniosków.

Przeprowadzone testy wykazały, że projekt oparty na modelu T5 osiąga rezultaty zbliżone do oryginalnych tytułów. Model potrafił wygenerować sensowne i zwięzłe tytuły, które dobrze odzwierciedlały treść abstraktu.

Tytuł wygenerowany na podstawie powyższego abstraktu:
Using the T5 model to generate titles from scientific papers

Wstęp

Wykonanie projektu opierało się na zbiorze prac naukowych z arXiv [1]. ArXiv jest to elektroniczne archiwum prac naukowych które gromadzi prace już od ponad 30 lat. Prace zbierane w wspomnianym archiwum pochodzą z szerokiego zakresu dziedzin: fizyki, astronomii, matematyki, informatyki, statystyki i biologii itp. Zbiór ten nadaje się idealnie do fine-tuningu modelu T5 (o którym później) ponieważ dzięki niemu mamy zbiór do wielu, wysokiej jakości tekstów. Przygotowany zbiór na kaggle jest gotowym zestawem do machine learningu.

Podstawą działania projektu jest model T5 (Text-To-Text Transfer Transformer) [2]. Największą zaletą tego modelu jest to, że na podstawie tekstowego wejścia danych generuje on dane wyjściowe również w formie tekstu. Jest to inna metodologia która wykorzystywana była wykorzystywana u nas na zajęciach.

Model T5 został stworzony i jest rozwijany przez Google. Firma ta dąży do “token-free future” co niesie za sobą wiele zalet. Podczas gdy większość wstępnie wyuczonych modeli językowych operuje na sekwencjach tokenów (odpowiadających słowom lub ich “paczkami”) rozwiązanie “token-free” operuje bezpośrednio na tekście (bity lub litery). Potrafią procesować tekst “out of the box”, są bardziej odporne na błędy w tekście, minimalizuje techniczną wiedzę która jest potrzebna by operować modelem. [3]

Model ten został wyuczony na zbiorze danych C4 [5] który jest ogromnym zbiorem “całego” internetu. Jego domyślna konfiguracja, czyli angielski z wyczyszczonym tekstem

(przekleństwa, nic nie wnoszące teksty, spam, powtórzenia) ma 806 GB tekstu. Natomiast wersja angielska “noclean” to aż 6.2 TiB tekstu. Natomiast najbardziej masywna wersja wielojęzyczna ma aż 38 TiB tekstu.

Rozwinięcie

Projekt wymaga niewielkiej ilości przygotowań środowiska. Głównym czynnikiem, który ułatwił pracę nad tym projektem jest biblioteka simpleT5 [4], który upraszcza jeszcze bardziej proces korzystania z tego modelu. Zawiera dostęp do wstępnie wytrenowanych modeli T5, fine-tuning modelu można zawrzeć w 3 linijkach kodu.

Kod z GitHuba projektu [4]:

```
# import
from simplet5 import SimpleT5

# instantiate
model = SimpleT5()

# load (supports t5, mt5, byT5 models)
model.from_pretrained("t5","t5-base")

# train
model.train(train_df=train_df, # pandas dataframe with 2 columns:
source_text & target_text
            eval_df=eval_df, # pandas dataframe with 2 columns:
source_text & target_text
            source_max_token_len = 512,
            target_max_token_len = 128,
            batch_size = 8,
            max_epochs = 5,
            use_gpu = True,
            outputdir = "outputs",
            early_stopping_patience_epochs = 0,
            precision = 32
            )
```

Dane z kaggle [1] są w formacie .json gdzie każda linia pliku odpowiada jednej pracy z archiwum. Struktura danych podana na stronie kaggle [1]:

- `id`: ArXiv ID (can be used to access the paper, see below)
- `submitter`: Who submitted the paper
- `authors`: Authors of the paper
- `title`: Title of the paper
- `comments`: Additional info, such as number of pages and figures
- `journal-ref`: Information about the journal the paper was published in

- `doi`: [https://www.doi.org](Digital Object Identifier)
- `abstract`: The abstract of the paper
- `categories`: Categories / tags in the ArXiv system
- `versions`: A version history

W zbiorze nie ma samej treści prac naukowych, ale znajduje się tam abstrakt dzięki któremu uzyskujemy bardzo dobre spojrzenie na to, o czym jest praca. W projekcie korzystam jedynie z: title, abstract, year.

Rok pracy jest jedynie na potrzeby ograniczenia wielkości zestawu danych który zostanie wyuczony.

Kluczowe elementy:

Po uzyskaniu tytułów prac i ich abstraktów z lat 2013-2023. Uzyskujemy 39 361 wyników.

```
df = pd.DataFrame({
      'target_text': titles,
      'source_text': abstracts
})
```

	target_text	source_text
0	Entanglement in a Jaynes-Cummings Model with T...	We investigate the conditions of entanglemen...
1	Banach-like metrics and metrics of compact sets	We present and study a family of metrics on ...
2	On the Cohomological Derivation of Yang-Mills ...	We present a brief review of the cohomologic...
3	Geometric Computational Electrodynamics with V...	In this paper, we develop a structure-preser...
4	A presentation for the mapping class group of ...	Finite presentations for the mapping class g...

Dane został podzielone w proporcji 80/20 gdzie 80% to są dane treningowe a 20% to treści ewaluacyjne.

```
model = SimpleT5()
model.from_pretrained(model_type="t5", model_name="t5-base")

model.train(train_df=train_df,
```

```
            eval_df=test_df,
            source_max_token_len=128,
            target_max_token_len=50,
            batch_size=8,
            max_epochs=3,
            use_gpu=True)
```

Model został wgrany na Huggingface w celu łatwej interakcji bez potrzeby pobierania modelu. Jest to platforma umożliwiająca tworzenie API, hostowania modeli itp.

https://huggingface.co/spaces/Skydem/PJN_predict_titles

Podsumowanie

Projekt w bardzo dobry sposób potrafi przewidzieć tytuł, dodatkowo łatwość fine-tuningu modelu daje ogrom możliwości w doborze zbiorów i przyszłych możliwości.

Bibliografia

[1] https://www.kaggle.com/datasets/Cornell-University/arxiv?datasetId=612177
[2] https://ai.googleblog.com/2020/02/exploring-transfer-learning-with-t5.html
[3] https://arxiv.org/abs/2105.13626
[4] https://github.com/Shivanandroy/simpleT5
[5] https://www.tensorflow.org/datasets/catalog/c4?hl=pl