Szymon M., 14.05.2023

Przetwarzanie Języków Naturalnych

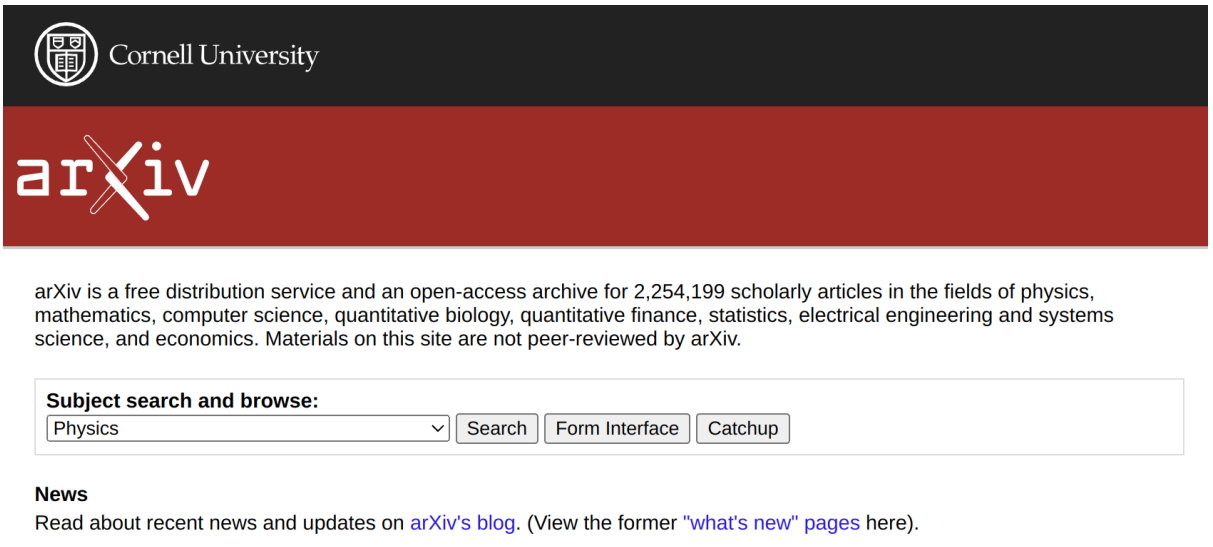
projekt pod tytułem

Generacja tytułów publikacji naukowych

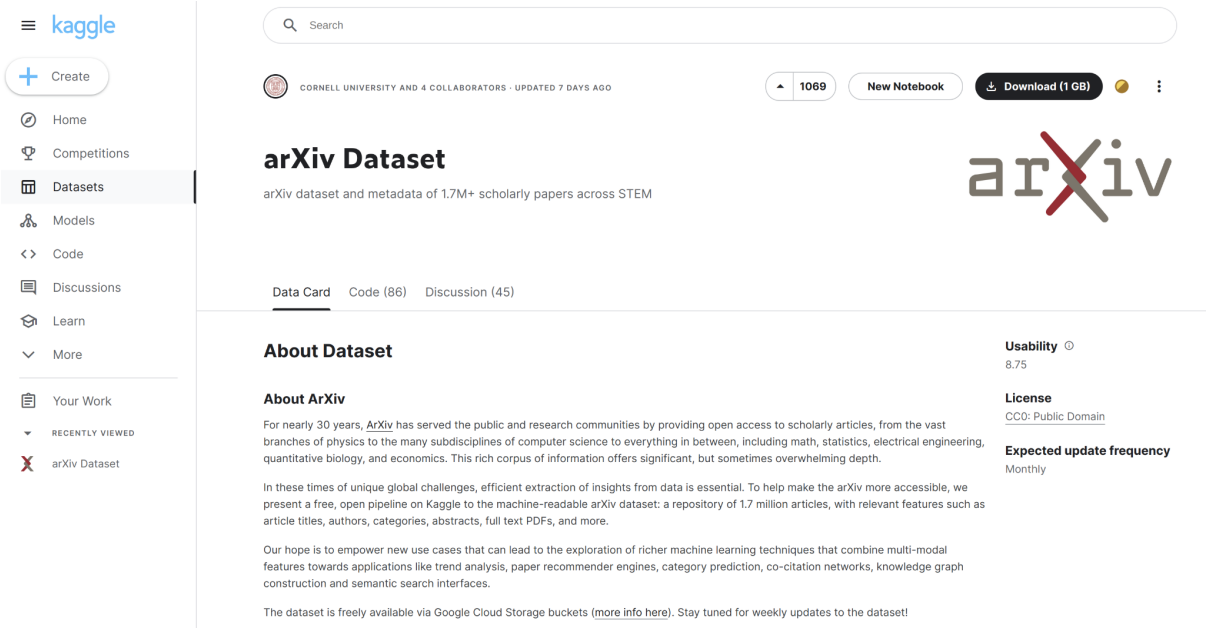
W ostatnich czasach w środowisku naukowym pojawia się coraz więcej prac naukowych tematycznie związanych z uczeniem maszynowym, sieciami neuronowymi i szeroko pojętą sztuczną inteligencją. Możliwość przewidywania tempa rozwoju badań jest bardzo istotną umiejętnością. Z tego powodu wykorzystać można dostępne obecnie narzędzia do przewidywania tego jakie będą prace naukowe które powstaną w przyszłości, a w rozważanym przypadku tytuły tych prac.

W celu przybliżenia procesu jaki należy przeprowadzić, by przewidzieć tytuły prac naukowych posłużę się przykładem programu dostępnego pod linkiem https://github.com/csinva/gpt-paper-title-generator którego autorami są Chandan Singh oraz Jamie Simon. Skonstruowany przez nich program miał za zadanie, wykorzystując ogólnodostępne źródła prac naukowych oraz narzędzia takie jak modele generatywne aby, wygenerować tytuły prac naukowych jakie mogą pojawić się w przyszłości.

Pierwszym etapem jest pozyskanie danych w postaci tytułów artykułów naukowych. Bazą danych która została wykorzystana jest witryna arXiv na której znajduję się około 2 250 000 artykułów naukowych z takich dziedzin jak fizyka, matematyka, informatyka i inne pokrewne nauki.



Dane potrzebne do analizy zostały pobrane z witryna kaggle która udostępnia darmowe datasety które można wykorzystać we własnych programach. W tym przypadku pobraliśmy dataset składający się z artykułów dostępnych na arXiv



Dane te pobrać można bezpośrednio ze strony lub wykorzystać aplikacje konsolową keggle

```
! pip install kaggle
! kaggle datasets download Cornell-University/arxiv
! unzip arxiv.zip
✓ 1m 52.2s

Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: kaggle in /home/szymon/.local/lib/python3.8/site-packages (1.5.13)
Requirement already satisfied: certifi in /usr/lib/python3/dist-packages (from kaggle) (2019.11.28)
Requirement already satisfied: python-dateutil in /home/szymon/.local/lib/python3.8/site-packages (from kaggle) (2.8.2)
Requirement already satisfied: urllib3 in /usr/lib/python3/dist-packages (from kaggle) (1.25.8)
Requirement already satisfied: python-slugify in /home/szymon/.local/lib/python3.8/site-packages (from kaggle) (8.0.1)
Requirement already satisfied: six≥1.10 in /usr/lib/python3/dist-packages (from kaggle) (1.14.0)
Requirement already satisfied: tqdm in /home/szymon/.local/lib/python3.8/site-packages (from kaggle) (4.65.0)
Requirement already satisfied: requests in /home/szymon/.local/lib/python3.8/site-packages (from kaggle) (2.28.2)
Requirement already satisfied: text-unidecode≥1.3 in /home/szymon/.local/lib/python3.8/site-packages (from python-slugify→kaggle) (1.3)
Requirement already satisfied: idna<4,≥2.5 in /usr/lib/python3/dist-packages (from requests→kaggle) (2.8)
Requirement already satisfied: charset-normalizer<4,≥2 in /home/szymon/.local/lib/python3.8/site-packages (from requests→kaggle) (3.1.0)

[notice] A new release of pip is available: 23.0.1 → 23.1.2
[notice] To update, run: python3 -m pip install --upgrade pip
Downloading arxiv.zip to /home/szymon/PJN/projekt
100%|████████████████████████████████| 1.15G/1.15G [01:14<00:00, 13.6MB/s]
100%|████████████████████████████████| 1.15G/1.15G [01:15<00:00, 16.4MB/s]
Archive:  arxiv.zip
  inflating: arxiv-metadata-oai-snapshot.json
```

Archiwum waży około 1.2 GB, natomiast po rozpakowaniu rozmiar wynosi około 3.5GB zawierając ponad 2 200 000 rekordów w postaci JSON.

```
❯ ls -lh
total 4.7G
-rw-rw-r-- 1 szymon szymon  52K May 13 14:14 1.ipynb
-rw-rw-r-- 1 szymon szymon 3.5G May  6 23:52 arxiv-metadata-oai-snapshot.json
-rw-rw-r-- 1 szymon szymon 1.2G May 13 14:37 arxiv.zip
~/PJN/projekt ❯
```

```python
import json

attributes = ["authors", "title", 'update_date','categories' ]

papers = []
count = 0
with open('arxiv-metadata-oai-snapshot.json') as f:
    for line in f:
        count += 1
        if count % 100000 == 0:
            print(f'on paper {count}')
        json_object = json.loads(line)
        paper = {}
        for attr in attributes:
            paper[attr] = json_object[attr]
        papers.append(paper)

import pandas as pd
df = pd.DataFrame(papers)

df
```

✓ 32.8s

Pobrane dane w postaci JSON zostały przetworzone na ramkę danych biblioteki pandas w celu dalszej obróbki

	authors	title	update_date	categories
0	C. Bal\'azs, E. L. Berger, P. M. Nadolsky, C.-...	Calculation of prompt diphoton production cros...	2008-11-26	hep-ph
1	Ileana Streinu and Louis Theran	Sparsity-certifying Graph Decompositions	2008-12-13	math.CO cs.CG
2	Hongjun Pan	The evolution of the Earth-Moon system based o...	2008-01-13	physics.gen-ph
3	David Callan	A determinant of Stirling cycle numbers counts...	2007-05-23	math.CO
4	Wael Abu-Shammala and Alberto Torchinsky	From dyadic Λ_{α} to $\Lambda_{\a...	2013-10-15	math.CA math.FA
...	...	...	...	...
2250218	R. Prozorov, M. Konczykowski, B. Schmidt, Y. Y...	On the origin of the irreversibility line in t...	2009-10-30	supr-con cond-mat.supr-con
2250219	Durga P. Choudhury, Balam A. Willemsen, John S...	Nonlinear Response of HTSC Thin Film Microwave...	2016-11-18	supr-con cond-mat.supr-con
2250220	Balam A. Willemsen, J. S. Derov and S.Sridhar ...	Critical State Flux Penetration and Linear Mic...	2009-10-30	supr-con cond-mat.supr-con
2250221	Yasumasa Hasegawa (Himeji Institute of Technol...	Density of States and NMR Relaxation Rate in A...	2009-10-30	supr-con cond-mat.supr-con
2250222	Naoki Enomoto, Masanori Ichioka and Kazushige ...	Ginzburg Landau theory for d-wave pairing and ...	2009-10-30	supr-con cond-mat.supr-con

2250223 rows × 4 columns

Jak widzimy, ramka danych zawiera około 2 500 000 rekordów prac naukowych ze wszystkich dziedzin dostępnych na platformie arXiv

```python
ai_papers = [paper for paper in papers if ("cs.AI" in paper["categories"] or "stat.ML" in paper["categories"] or "cs.LG" in paper["categories"])]
df = pd.DataFrame(ai_papers)

import re
regex = re.compile(f" and |,")

df["authors"] = [[str(a).strip() for a in regex.split(author) if len(str(a).strip()) > 3] for author in df["authors"]]
df = df.sort_values(by="update_date", ascending=False)

df
```

✓ 3.2s

	authors	title	update_date	categories
168839	[Shuting Shen, Xi Chen, Ethan X. Fang, Junwei Lu]	Combinatorial Inference on the Optimal Assortm...	2023-05-05	stat.ML cs.LG
179649	[Ming-Kun Xie, Jia-Hao Xiao, Gang Niu, Masashi...	Class-Distribution-Aware Pseudo Labeling for S...	2023-05-05	cs.LG
169390	[Alon Albalak, Colin Raffel, William Yang Wang]	Improving Few-Shot Generalization by Exploring...	2023-05-05	cs.LG cs.CL
179618	[UngJin Na, Moonhee Choi, HangJin Jo]	Critical heat flux diagnosis using conditional...	2023-05-05	physics.flu-dyn cs.LG
171498	[Soham Rohit Chitnis, Sidong Liu, Tirtharaj Da...	Domain-Specific Pre-training Improves Confiden...	2023-05-05	cs.CV cs.AI cs.LG
...	...	...	...	...
180173	[Petro M. Gopych]	A Neural Network Assembly Memory Model Based o...	2007-05-23	cs.AI cs.IR cs.NE q-bio.NC q-bio.QM
180172	[Peter D. Turney, Michael L. Littman, Jeffrey ...	Combining Independent Modules to Solve Multipl...	2007-05-23	cs.CL cs.IR cs.LG
180171	[Peter D. Turney (National Research Council of...	Measuring Praise and Criticism: Inference of S...	2007-05-23	cs.CL cs.IR cs.LG
180170	[Wolfgang Mayer, Markus Stumptner]	Model-Based Debugging using Multiple Abstract ...	2007-05-23	cs.SE cs.AI
1	[T. Kosel, I. Grabec]	Intelligent location of simultaneously active ...	2007-05-23	cs.NE cs.AI

181083 rows × 4 columns

Następnie dane te zostały odfiltrowane tak, aby pozostały jedynie rekordy których tematyką jest sztuczna inteligencja czy uczenie maszynowe. Po filtracji zostało około 180 tyś. rekordów.

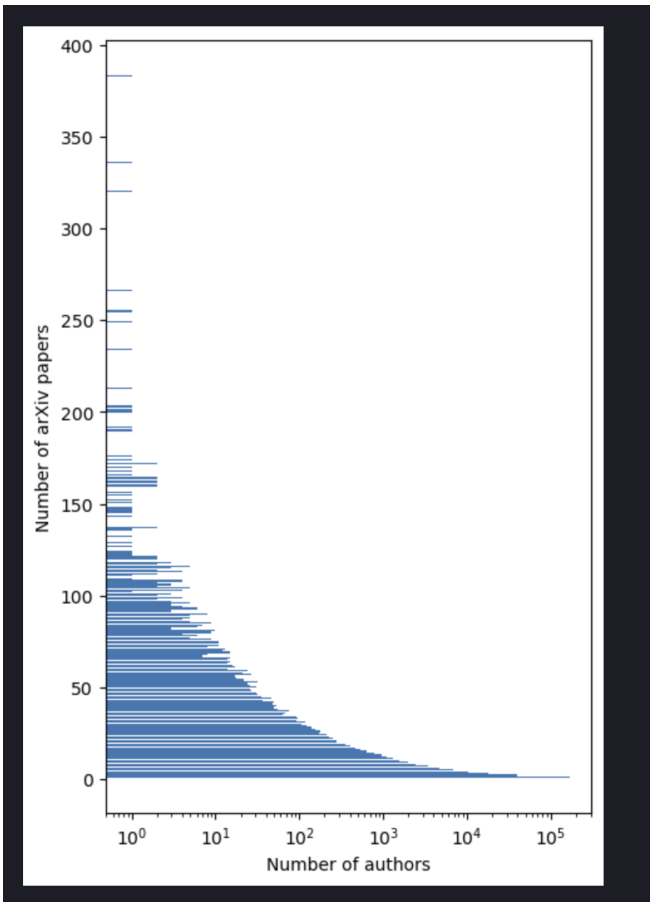
```
authors_with_papers = {}

for paper in df.values:
    authors = paper[0]
    title = paper[1]
    for author in authors:
        if author in authors_with_papers:
            authors_with_papers[author].append(title)
        else:
            authors_with_papers[author] = [title]

output = pd.DataFrame([{"author": author, "papers": authors_with_papers[author], "paper_count": len(authors_with_papers[author])} for author in

output.sort_values(by="paper_count", ascending=False)
```

✓ 2.9s

	author	papers	paper_count
10152	Yoshua Bengio	[Hyena Hierarchy: Towards Larger Convolutional...	383
2130	Yang Liu	[Performative Prediction with Bandit Feedback:...	336
3403	Sergey Levine	[Offline RL for Natural Language Generation wi...	320
4749	Bo Li	[Model Explainability in Physiological and Hea...	266
15088	Michael I. Jordan	[Online Learning in Stackelberg Games with an ...	255
...	...	...	...
128610	Igor Khokhlov	[Attacks, Defenses, And Tools: A Framework To ...	1
128611	Mehdi Mirakhorli	[Attacks, Defenses, And Tools: A Framework To ...	1
128616	Songqing Yue	[Imbalanced Malware Images Classification: a C...	1
128619	Zhengren Wang	[Listing Maximal k-Plexes in Large Real-World ...	1
267544	Markus Stumptner	[Model-Based Debugging using Multiple Abstract...	1

267545 rows × 3 columns

Po zliczeniu ilości prac według konkretnych pojedynczych autorów otrzymano następujące wyniki



Jak widać na wykresie, stosunek ilości prac ilości autorów którzy napisali tę pracę ma charakter wykładniczy.

Po przygotowaniu danych, zostały one wprowadzone do modelu GPT w formie pięciu kolejnych prac konkretnego autora w oczekiwaniu na to, że model przewidzi jakie tematy mogą mieć kolejne pięć prac tego autora.

1 Author-specific paper titles (prompting gpt3)

To generate author-specific titles, we take the five most recent titles from each author with atleast 3 arXiv AI papers (cs.ML, cs.LG, stat.ML). We then format the papers using the following template and query for a new title using GPT-3:

```
Here is a list of related machine-learning papers:

> [title 1]
> [title 2]
...
> [title 5]
> ____
```

```python
for i, author in enumerate(tqdm(authors)):
    if not author in authors_save:
        query = prompt + '\n> '.join(authors_dict_titles[author][-papers_in_context:]) + '\n>'
        completion = openai.Completion.create(
            engine="text-davinci-002", prompt=query,
            n=gens_per_author, stop='>'
        )
        authors_save[author] = [completion.choices[i].text for i in range(len(completion.choices))]
    if i % 200 == 0:
        pkl.dump(authors_save, open(f'gen_titles/authors_save_{i}.pkl', 'wb'))
pkl.dump(authors_save, open(f'gen_titles/authors_save_full.pkl', 'wb'))      Chandan Singh, 7 months
```

Wykorzystany został modeł GPT-3 od OpenAI, a do komunikacji z modelem posłużyło oficjalne Pythonowe API które pozwala na prostą komunikację z modelem. Konkretnie został wykorzystany model “text-davinci-002”, w trybie uzupełniania wprowadzonego tekstu.

Wprowadzając pięć następujących tytułów

```
> Hierarchical Shrinkage: improving the accuracy and interpretability of tree-based methods
> Fast Interpretable Greedy-Tree Sums (FIGS)
> Adaptive wavelet distillation from neural networks through interpretations
> Emb-GAM: an Interpretable and Efficient Predictor using Pre-trained Language Models
> Explaining Patterns in Data with Language Models via Interpretable Autoprompting
> ____
```

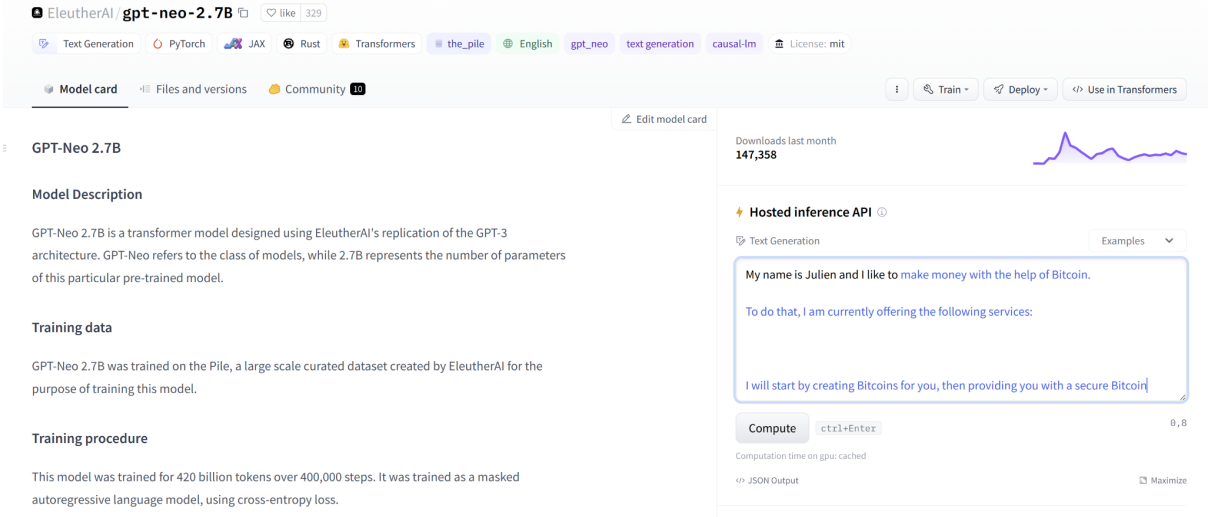
Otrzymano następujący wynik

1. Towards Interpretable Natural Language Processing: A Survey
2. A Unified Framework for Interpretable Machine Learning
3. Compositional Attention Networks for Machine Reasoning
4. Achieving Open Vocabulary Neural Machine Translation
5. A Deep Understanding of Neural Networks through Deep Visualization

Zauważono jednak, że otrzymane wyniki nie mają bezpośredniego związku z podawanymi tytułami, a jedynie z popularnymi w ostatnim czasie tematami prac naukowych.

Z tego powodu postanowiono douczyć model konkretnymi przykładami tematów prac, które pojawiały się po okresie uczenia się modelu.

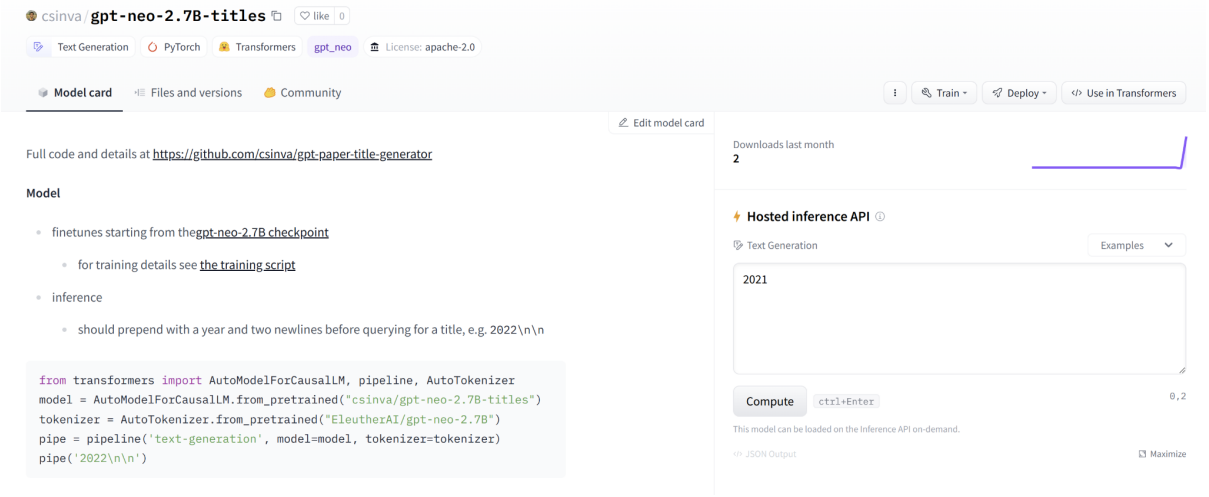
Jako model bazowy wykorzystano model gpt-neo z 2.7 mld parametrów.



Został on douczony danymi spreparowanymi w następujący sposób. Jak widzimy model potrzebuje parametru prompt który oznacza tekst wpisywany przez użytkownika w momencie kiedy chce uzyskać autouzupełnianie, natomiast completion oznacza odpowiedź którą chcielibyśmy uzyskać


```
{"prompt":"2020\n\n","completion":" Collective Learning by Ensembles of Altruistic Diversifying Neural Networks\n"}
{"prompt":"2021\n\n","completion":" Zeroth-order non-convex learning via hierarchical dual averaging\n"}
{"prompt":"2010\n\n","completion":" Computationally Efficient Modulation Level Classification Based on Probability Distribution Distance Functions\n"}
{"prompt":"2020\n\n","completion":" Thermal Prediction for Efficient Energy Management of Clouds using Machine Learning\n"}
{"prompt":"2021\n\n","completion":" Label-GCN: An Effective Method for Adding Label Propagation to Graph Convolutional Networks\n"}
{"prompt":"2019\n\n","completion":" Stock Forecasting using M-Band Wavelet-Based SVR and RNN-LSTMs Models\n"}
{"prompt":"2020\n\n","completion":" A Centralised Soft Actor Critic Deep Reinforcement Learning Approach to District Demand Side Management through CityL
{"prompt":"2017\n\n","completion":" On the Design of LQR Kernels for Efficient Controller Learning\n"}
{"prompt":"2020\n\n","completion":" Factual Error Correction for Abstractive Summarization Models\n"}
{"prompt":"2021\n\n","completion":" Machine Learning Regression based Single Event Transient Modeling Method for Circuit-Level Simulation\n"}
{"prompt":"2021\n\n","completion":" MedAug: Contrastive learning leveraging patient metadata improves representations for chest X-ray interpretation\n"}
{"prompt":"2019\n\n","completion":" Estudo comparativo de meta-heur\\'isticas para problemas de colora\\c{c}\\~oes de grafos\n"}
{"prompt":"2019\n\n","completion":" BOBBY2: Buffer Based Robust High-Speed Object Tracking\n"}
{"prompt":"2020\n\n","completion":" Ansor : Generating High-Performance Tensor Programs for Deep Learning\n"}
{"prompt":"2018\n\n","completion":" Outlier Detection using Generative Models with Theoretical Performance Guarantees\n"}
{"prompt":"2019\n\n","completion":" A context-aware knowledge acquisition for planning applications using ontologies\n"}
{"prompt":"2021\n\n","completion":" TabularNet: A Neural Network Architecture for Understanding Semantic Structures of Tabular Data\n"}
{"prompt":"2017\n\n","completion":" Efficient Spatio-Temporal Gaussian Regression via Kalman Filtering\n"}
{"prompt":"2018\n\n","completion":" Ladder Networks for Semi-Supervised Hyperspectral Image Classification\n"}
{"prompt":"2020\n\n","completion":" A New Similarity Space Tailored for Supervised Deep Metric Learning\n"}
{"prompt":"2020\n\n","completion":" Frequency Estimation in Data Streams: Learning the Optimal Hashing Scheme\n"}
{"prompt":"2013\n\n","completion":" The Role of Tuning Uncertain Inference Systems\n"}
{"prompt":"2019\n\n","completion":" Task-Driven Common Representation Learning via Bridge Neural Network\n"}
{"prompt":"2017\n\n","completion":" Navigating Occluded Intersections with Autonomous Vehicles using Deep Reinforcement Learning\n"}
{"prompt":"2019\n\n","completion":" Evolving Neural Networks in Reinforcement Learning by means of UMDAc\n"}
{"prompt":"2020\n\n","completion":" XAI-P-T: A Brief Review of Explainable Artificial Intelligence from Practice to Theory\n"}
{"prompt":"2020\n\n","completion":" A pipeline for fair comparison of graph neural networks in node classification tasks\n"}
{"prompt":"2020\n\n","completion":" GENNI: Visualising the Geometry of Equivalences for Neural Network Identifiability\n"}
{"prompt":"2021\n\n","completion":" TabGNN: Multiplex Graph Neural Network for Tabular Data Prediction\n"}
{"prompt":"2017\n\n","completion":" A unified method for super-resolution recovery and real exponential-sum separation\n"}
{"prompt":"2020\n\n","completion":" BinaryDuo: Reducing Gradient Mismatch in Binary Activation Network by Coupling Binary Activations\n"}
{"prompt":"2021\n\n","completion":" A Non-asymptotic Approach to Best-Arm Identification for Gaussian Bandits\n"}
{"prompt":"2020\n\n","completion":" Minimax optimal goodness-of-fit testing for densities and multinomials under a local differential privacy constraint\
{"prompt":"2020\n\n","completion":" Leveraging Siamese Networks for One-Shot Intrusion Detection Model\n"}
{"prompt":"2021\n\n","completion":" Shaped Policy Search for Evolutionary Strategies using Waypoints\n"}
{"prompt":"2021\n\n","completion":" Speaker-Conditioned Hierarchical Modeling for Automated Speech Scoring\n"}
{"prompt":"2014\n\n","completion":" Sparsistency and agnostic inference in sparse PCA\n"}
{"prompt":"2017\n\n","completion":" A minimax and asymptotically optimal algorithm for stochastic bandits\n"}
{"prompt":"2012\n\n","completion":" Justice blocks and predictability of US Supreme Court votes\n"}
{"prompt":"2019\n\n","completion":" Learning to Synthesize: Robust Phase Retrieval at Low Photon counts\n"}
{"prompt":"2022\n\n","completion":" FedQAS: Privacy-aware machine reading comprehension with federated learning\n"}
{"prompt":"2021\n\n","completion":" Multidimensional Assignment Problem for multipartite entity resolution\n"}
{"prompt":"2022\n\n","completion":" Automated machine learning for secure key rate in discrete-modulated continuous-variable quantum key distribution\n"}
{"prompt":"2020\n\n","completion":" When deep learning models on GPU can be accelerated by taking advantage of unstructured sparsity\n"}
{"prompt":"2011\n\n","completion":" On the Universality of Online Mirror Descent\n"}
{"prompt":"2020\n\n","completion":" Understanding Crowd Behaviors in a Social Event by Passive WiFi Sensing and Data Mining\n"}
{"prompt":"2018\n\n","completion":" A Generation Method of Immunological Memory in Clonal Selection Algorithm by using Restricted Boltzmann Machines\n"}
```


Uzyskany model został umieszczony w sieci i jest ogólnie dostępny



Następnie model poproszono o wygenerowanie tytułów prac naukowych dla poszczególnych lat. Wyniki prezentują się następująco:

2022

- Diverse Datasets for Learning to Rank via Knowledge Graph Embedding
- Machine learning-driven method for high-throughput single-cell analysis of differentiation and lineage commitment
- On the Sample Complexity of Differentially Private Learning
- Data-Dependent Weight Normalization for Improved Image Resolution
- Adaptive Densely Connected Networks for the Generation and Visualization of Object Deformations
- Exploring the Implicit Bias in Transfer Learning using Imitation Learning

2023 (These samples tend to just be similar to 2021/2022 where the majority of the training data lies)

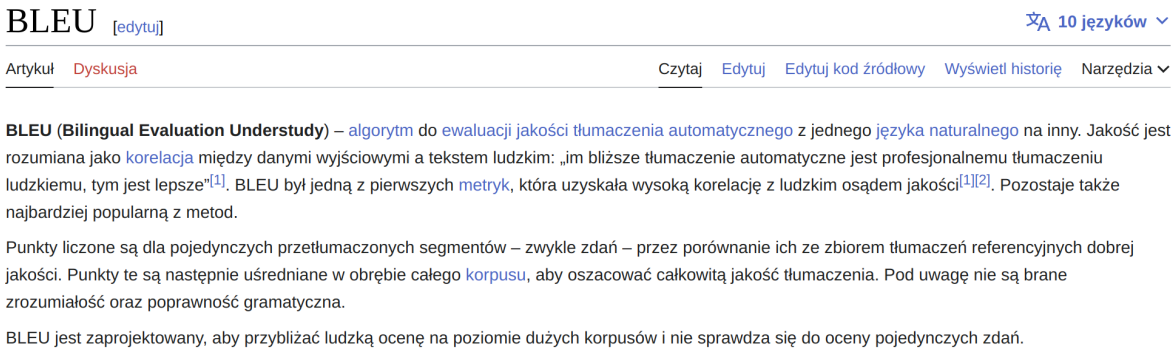
- An Interpretable Dynamic Network for Spatiotemporal Pattern Prediction in High-Dimensional Time Series Data
- Multimodal Deep Learning for Automated Cancer Histopathology Analysis
- Reinforced Learning for Robust and Accurate Object Detection
- A Machine Learning Approach to High Sensitivity Data Processing
- Adversarial Robustness for Graph Neural Networks in Network Intrusion Detection
- Reinforcement Learning via Exploration and Rehearsal for Learning from Demonstrations

2010 (Seems to properly generate older titles)

- Learning in a Dynamic, Clustered and Homogeneous Heterogeneous Markov Decision Process
- An Empirical Analysis of the Regularization of the Gaussian Process Regression
- A Scalable Clustering Algorithm under Heterogeneous Data
- A Unified Representation for Probabilistic Time Series Forecasting
- A Hybrid Approach to Automatic Alignment and Localization of Digital Object Platforms
- Bayesian nonparametric modeling of random fields

Wygenerowane tytuły z pewnością przypominają rzeczywiste tytułu prac naukowych.

Postanowiono sprawdzić dokładność wygenerowanych tytułów i porównać je z rzeczywistymi tytułami powstałymi w tym okresie. Do tego celu postanowiono wykorzystać algorytm BLEU który powstał do porównywania automatycznego tłumaczenia tekstu z tekstem przetłumaczonym przez profesjonalnego tłumacza.



BLEU [edytuj]

10 języków

Artykuł Dyskusja Czytaj Edytuj Edytuj kod źródłowy Wyświetl historię Narzędzia

BLEU (**Bilingual Evaluation Understudy**) – algorytm do ewaluacji jakości tłumaczenia automatycznego z jednego języka naturalnego na inny. Jakość jest rozumiana jako korelacja między danymi wyjściowymi a tekstem ludzkim: „im bliższe tłumaczenie automatyczne jest profesjonalnemu tłumaczeniu ludzkiemu, tym jest lepsze"[1]. BLEU był jedną z pierwszych metryk, która uzyskała wysoką korelację z ludzkim osądem jakości[1][2]. Pozostaje także najbardziej popularną z metod.

Punkty liczone są dla pojedynczych przetłumaczonych segmentów – zwykle zdań – przez porównanie ich ze zbiorem tłumaczeń referencyjnych dobrej jakości. Punkty te są następnie uśredniane w obrębie całego korpusu, aby oszacować całkowitą jakość tłumaczenia. Pod uwagę nie są brane zrozumiałość oraz poprawność gramatyczna.

BLEU jest zaprojektowany, aby przybliżać ludzką ocenę na poziomie dużych korpusów i nie sprawdza się do oceny pojedynczych zdań.

Ewaluując wygenerowane tytuły postarano się znaleźć tytuły im odpowiadające. Oto pięć pierwszych wyników:

A	B
Understanding the effect of data augmentation in generative adversarial networks	Understanding the effect of data augmentation in self-supervised anomaly detection
Adversarial attacks on graph neural networks	Sparse vicious attacks on graph neural networks
Differentiable reinforcement learning for continuous control	Normality-guided distributional reinforcement learning for continuous control
Multilevel representation learning for time series forecasting	Out-of-distribution representation learning for time series classification
Unsupervised feature learning for medical image segmentation	Distributed contrastive learning for medical image segmentation

Wygenerowane tytuły są praktycznie nie do odróżnienia od tych które stanowią nazwy rzeczywistych prac naukowych. Autorzy zwracają jednak uwagę na fakt, że wygenerowane treści są bardziej ogólne od odpowiadającym im rzeczywistym tytułom.

Przedstawiona w tym raporcie analiza pokazuję potencjał drzemiący w językowych modelach generatywnych w kontekście przewidywania tudzież generowania treść na podstawie danych historycznych, co może być przydatne przy próbach ustalenia kierunku rozwoju różnych technologii czy gałęzi nauki. Już teraz istnieją narzędzia które połączone ze sobą dają zaskakująco świetne rezultaty.