



OpenReview 분석: 리뷰어들은 잘 될 연구의 떡잎을 알아볼까?

김기범

> Do Anything with python! 

소개

> Do Anything with python!



- 물리학과를 졸업하고 뇌파, 반도체, 배터리 연구를 하다 미디어 플랫폼 얼룩소에서 자연어 연구를 하고 있음.
- 세상 물정을 알 수 있는 데이터에 관심이 많음.



- OpenReview라는 플랫폼을 통해 논문 리뷰 과정을 다른 사람들도 확인할 수 있게 됨.
- 이 발표에서는 논문을 처음 접하는 리뷰어들의 평가와 향후 논문의 인용수의 상관관계를 분석한 결과를 소개하고자 함.
- OpenReview와 google scholar에서 논문에 대한 리뷰어들의 평가와 인용수를 스크래핑 하는 방법, 논문의 인용수와 리뷰어들의 상관관계 분석, 그리고 그 분석결과로 도출된 새로운 가설들을 검증하는 내용을 다룸.

오픈리뷰란?

> **Do Anything with python!**



- Massachusetts Amherst 대학의 Andrew McCallum 교수가 처음 제안.
- 논문의 리뷰 과정을 투명하게 공개하는 플랫폼.
- ICLR 등의 다양한 컨퍼런스에서 사용 중.



오픈리뷰란?

> Do Anything with python!



OpenReview.net [Login](#)

Open Peer Review. Open Publishing. Open Access. Open Discussion. Open Recommendations. Open Directory. Open API. Open Source.

Active Venues

- NeurIPS 2022 Conference
- TMLR
- ACL ARR 2022
- ACMMM 2022 Track Reproducibility
- NAACL 2022 Track Demo
- ACL 2022 Workshop Insights
- KU Spring2022 COMP547
- CoLLAs 2022 Conference
- ACL ARR 2022 March
- TU 2022 TUML

Open for Submissions

- EMNLP 2022 Workshop SereTOD
⌚ Due 15 Jul 2022, 09:00 Korean Standard Time
- NeurIPS 2022 Workshop Queer In AI
⌚ Due 16 Jul 2022, 08:59 Korean Standard Time
- ECCV 2022 Workshop VIPriors
⌚ Due 16 Jul 2022, 08:59 Korean Standard Time
- ACMMM 2022 Workshop NarSUM
⌚ Due 16 Jul 2022, 20:59 Korean Standard Time
- ACL ARR 2022 July
⌚ Due 16 Jul 2022, 20:59 Korean Standard Time
- ACMMM 2022 Workshop MMSports
⌚ Due 17 Jul 2022, 08:59 Korean Standard Time
- EMNLP 2022 Workshop LOUHI
⌚ Due 17 Jul 2022, 08:59 Korean Standard Time

“OpenReview는 과학 커뮤니케이션의 개방성을 촉진하는 것을 목표로 합니다.”



- *Tran, et al. "An open review of openreview: A critical analysis of the machine learning conference review process." arXiv preprint arXiv:2010.05137 (2020).*
 - 젠더 불균형: 여성 저자들은 더 낮은 리뷰 스코어, 더 낮은 게재 비율을 가짐.
 - 탑스쿨에서 나온 논문들이 리뷰 마감 전 arXiv에 논문을 공개한 경우, 평균적으로 rating이 0.67점 높았다.
- *Bharadhwaj, et al. "De-anonymization of authors through arXiv submissions during double-blind review." arXiv preprint arXiv:2007.00177 (2020).*
 - 논문 리뷰기간 중 아카이브에 미리 올려놓은 ICLR 논문들을 분석.
 - 리뷰어가 자신의 리뷰에 대한 확신이 낮은 경우, 이미 Arxiv에 공개된 유명한 저자의 논문에 더 높은 점수를 줌.
- 리뷰 과정이 공개되는 플랫폼이 생겨 리뷰 생태계에 대한 연구를 할 수 있게 되었다.



- 한 논문이 다른 논문들에 인용된 횟수를 뜻함.
- 인용수 자체가 논문의 질을 그대로 대변하는 것은 아님. 하지만 논문의 영향력을 평가할 수 있는 가장 객관적인 지표.

Deep residual learning for image recognition

[K He](#), [X Zhang](#), [S Ren](#), [J Sun](#) - ... and pattern **recognition**, 2016 - [openaccess.thecvf.com](#)

... **Deeper** neural networks are more difficult to train. We present a **residual learning** framework to ease the training of networks that are substantially **deeper** than those used previously. ...

☆ 저장 77 인용 12578회 인용 관련 학술자료 전체 72개의 버전 >>



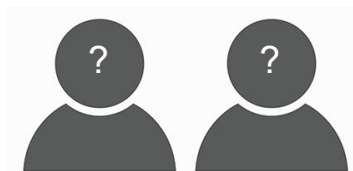
리뷰 점수 vs 인용수

> Do Anything with python!



리뷰 점수

(ICLR의 경우)익명의 전문가들이 투고된 논문을 심사해 리뷰 점수를 매김.
리뷰 점수를 기반으로 저널/컨퍼런스에 논문을 게재할지 결정.



인용수

논문이 출판된 이후 다른 논문에 인용된 횟수.
논문의 파급력, 영향력.

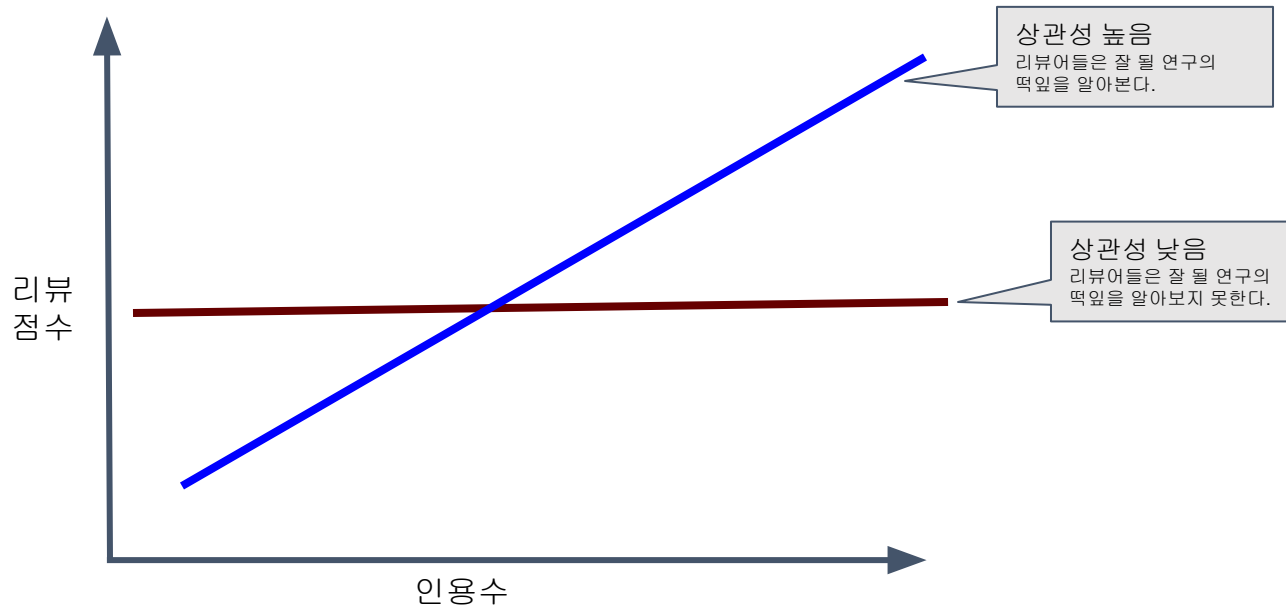


VS



리뷰 점수 vs 인용수

> Do Anything with python!



데이터 수집: OpenReview

> Do Anything with python!



- Openreview에서 리뷰의 진행은 댓글 포럼 형태.
- 각 리뷰어들이 댓글을 통해 리뷰를 남기면, 저자들이 이에 답하는 형식을 취함.

Recurrent Batch Normalization

Tim Coolijmans, Nicolas Ballas, César Laurent, Çağlar Gülçehre, Aaron Courville

05 Nov 2016 (modified: 22 Jul 2022) ICLR 2017 Poster Readers: Everyone Show BibTeX Show Revisions

TLDR: Make batch normalization work in recurrent neural networks

Abstract: We propose a reparameterisation of LSTM that brings the benefits of batch normalization to recurrent neural networks. Whereas previous works only apply batch normalization to the input-to-hidden transformation of RNNs, we demonstrate that it is both possible and beneficial to batch-normalize the hidden-to-hidden transition, thereby reducing internal covariate shift between time steps.

We evaluate our proposal on various sequential problems such as sequence classification, language modeling and question answering. Our empirical results show that our batch-normalized LSTM consistently leads to faster convergence and improved generalization.

Keywords: Deep learning, Optimization

Conflicts: umontréal.ca, google.com

Reply Type: Author: Visible To: Hidden From:

16 Replies

ICLR committee final decision

ICLR 2017 PCS

07 Feb 2017 ICLR 2017 conference acceptance Readers: Everyone

Comment: The reviewers believe this paper is of significant interest to the ICLR community, as it demonstrates how to get the popular batch normalization method to work in the recurrent setting. The fact that it has already been cited a variety of times also speaks to its interest within the community. The extensive experiments are convincing that the method works. One common criticism is that the authors don't address enough the added computational cost of the method in the text or empirically. Plots showing loss as a function of wall-clock time instead of training iteration would be more informative to readers deciding whether to use batch norm.

Pros:

- Gets batch normalization to work on recurrent networks (which had been elusive to many)
- The experiments are thorough and demonstrate the method reduces training time (as a function of training iterations)
- The paper is well written and accessible

Cons:

- The contribution is relatively incremental (several tweaks to an existing method)
- The major disadvantage to the approach is the added computational cost, but this is conspicuously not addressed.

Decision: Accept (Poster)

Batch normalisation brought to LSTM

ICLR 2017 conference AnonReviewer1

20 Dec 2016 (modified: 20 Jan 2017) ICLR 2017 conference official review Readers: Everyone

Review: The paper shows that BN, which does not work out of the box for RNNs, can be used with LSTM when the operator is applied to the hidden-to-hidden and the input-to-hidden contribution separately. Experiments are conducted to show that it leads to improved generalisation error and faster convergence.

The paper is well written and the idea well presented.

- i) The data sets and consequently the statistical assumptions used are limited (e.g. no continuous data, only autoregressive generative modelling).
- ii) The hyper parameters are nearly constant over the experiments. It is ruled out that they have not been picked in favor of one of the methods. E.g. just judging from the text, a different learning rate could have lead to equally fast convergence for vanilla LSTM.

Concluding, the experiments are flawed and do not sufficiently support the claim. An exhaustive search of the hyper parameter space could rule that out.

Rating: 7: Good paper, accept

Confidence: 4: The reviewer is confident but not absolutely certain that the evaluation is correct

Answer to the review

Nicolas Ballas

24 Dec 2016 ICLR 2017 conference public comment Readers: Everyone

Comment: Thank you for your feedback,

- 1) We evaluate our BN-LSTM proposal on 5 different datasets, two sequential classification tasks, two language modelling tasks and one large-scale question answering task. The different datasets also cover a variety of inputs, including characters (PTB and Text8), words (CMN question answering), and pixels (MNIST and pMNIST). In addition, all the proposed tasks have been extensively used in the RNN literature and can be considered as standard benchmarks for such models.

We think that the proposed set of tasks is fairly representative of the usual problems on which RNN model are applied.

데이터 수집: OpenReview

> Do Anything with python!



- openreview.net은 데이터 수집, 리뷰 등을 할 수 있는 python library openreview-py를 제공.
- openreview-py로 ICLR18~20 Blind review 페이지 노트 객체 얻기.
- Invitation 링크는 컨퍼런스마다 다름.

```
1 pip install openreview-py
```

```
1 base_url = 'https://api.openreview.net'
2 c = openreview.Client(baseUrl=base_url)
3 invitation = 'ICLR.cc/2018/Conference/-/Blind_Submission'
4
5 blind_notes = [note for note in openreview.tools.iterget_notes(c,
6                                                                invitation=invitation,
7                                                                details='original')]
8 forum_list = set([h.forum for h in blind_notes])
9
10 all_data = []
11 for forum in tqdm(forum_list):
12     forum_comments = c.get_notes(forum=forum)
13     all_data.append(forum_comments)
```





- Openreview에서 리뷰의 진행은 댓글
포럼 형태.
- 각 리뷰어들이 댓글을 통해 리뷰를
남기면, 저자들이 이에 답하는 형식을
취함.
- openreview-py로 얻어진 리뷰 객체는
각각의 댓글 데이터를 담고 있음.
- 이중 전담 리뷰어들의 댓글은
“confidence”, “rating” 속성을 가짐.

```
{'cdate': 1511987146689,  
  'content': {'confidence': '5: The reviewer is absolutely certain that the '  
              'evaluation is correct and very familiar with the '  
              'relevant literature',  
              'rating': '4: Ok but not good enough - rejection',  
              'review': 'This paper presents a modification of the objective '  
                        'used to train generative networks with an MMD '  
                        'adversary (i.e. as in Dziugaite et al or Li et al '  
                        '2015), where importance weighting is used to evaluate '  
                        'the MMD against a target distribution which differs '  
                        'from the data distribution. The goal is that this '  
                        'could be used to correct for known bias in the '  
                        'training data - the example considered here is for '  
                        'class imbalance for known, fixed classes.\n'
```



데이터 수집: OpenReview

> Do Anything with python!



- Google Scholar을 통해 각 논문의 인용수를 얻을 수 있음.
- scholarly 라이브러리를 사용하여 구글 scholar 검색 쿼리 시행.
- 하지만 몇백 번 이후의 검색은 구글이 막아버림.
- 이를 해결하기 위해 scraper api의 프록시를 사용.
- scraper api는 회원가입 후 5000번의 무료 크레딧이 주어짐.

```
from scholarly import scholarly, ProxyGenerator

pg = ProxyGenerator()
API_KEY = apikey
success = pg.ScraperAPI(API_KEY)
scholarly.use_proxy(pg)

citations_raw = {}

for title in tqdm(title_list):
    search_reslut_iter = scholarly.search_pubs(title)
    data = next(search_reslut_iter)
    citations_raw[title] = data

for x in citations_raw.keys():
    del citations_raw[x]["source"]

with open(saving_path, 'w') as f:
    json.dump(citations_raw, f)
```

인용수와 리뷰 점수 비교하기

> Do Anything with python!



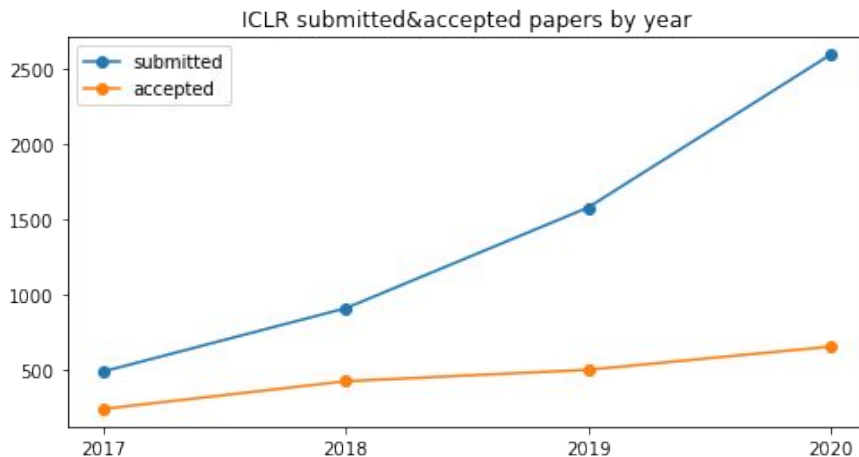
- 앞에서 소개한 데이터 수집 및 파싱 과정을 통해 ICLR2017~20 데이터를 획득.
- 분석 시점에서 ICLR2021, ICLR2022 데이터도 얻을 수 있지만, 2022년 현재 유의미한 수의 인용이 존재하지 않을 것이라 가정하여 제외하였음.
- reject된 논문도 인용되지만, 한 번 reject된 논문이 수정되어 다른 저널/컨퍼런스에 다시 제출되는 경우가 많아 reject된 논문의 인용수는 정확하지 못하다고 판단해 제외.

수집한 데이터 살펴보기

> Do Anything with python!



- 매년 ICLR에 제출된 논문의 수는 가파르게 증가하고 있음.

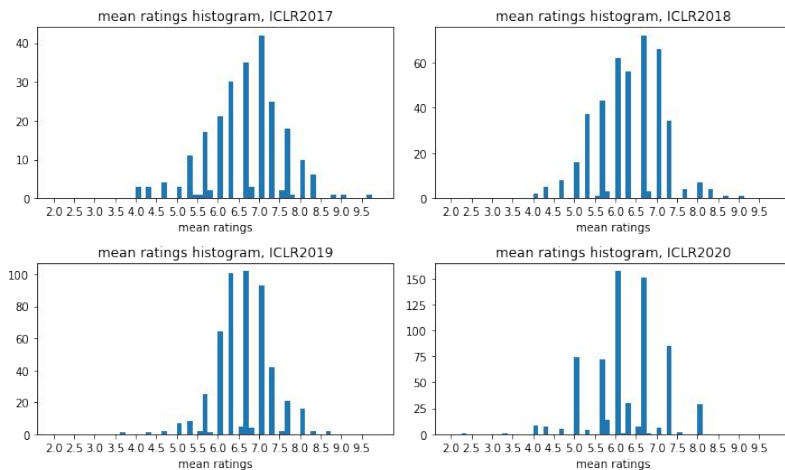
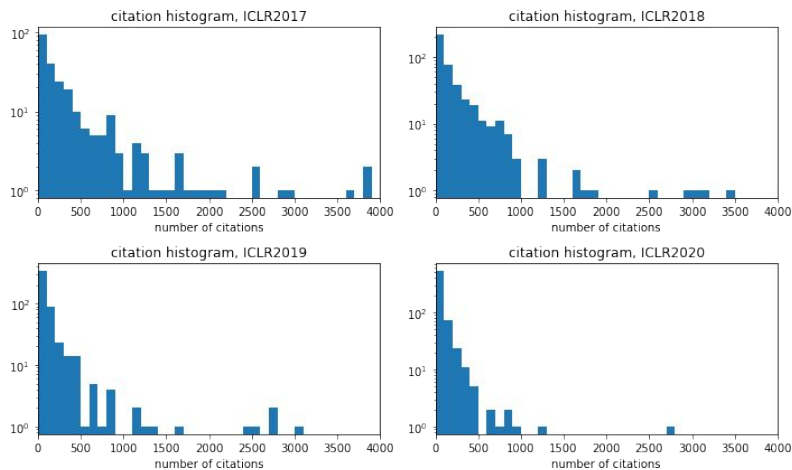


수집한 데이터 살펴보기

> Do Anything with python!



- 전체 제출된 논문의 인용수 히스토그램을 그려보니 꼬리가 두꺼운 분포를 가짐.
- 논문들의 평균 리뷰 스코어는 노멀 분포에 가까움.

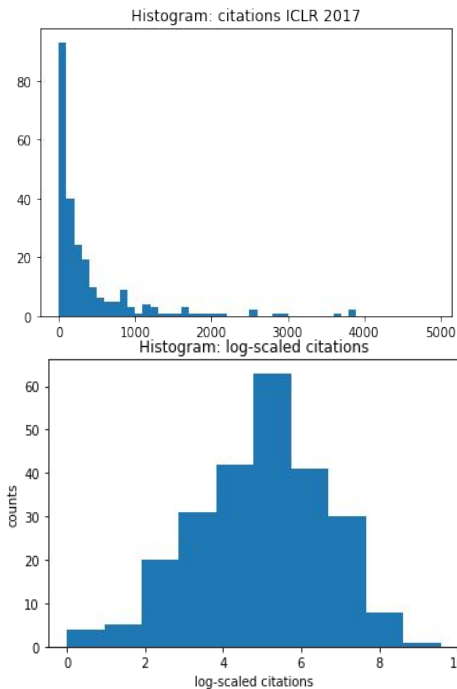


인용수와 리뷰 점수 비교하기

> Do Anything with python!



- 논문은 많이 인용될수록 더 많이 인용되는 성질을 가짐.
- 인용수의 로그값을 상관성 분석에 사용하는 것이 더 적합할 것이라 가정.
- 인용수에 로그를 취하면 정규분포와 유사한 분포가 됨.

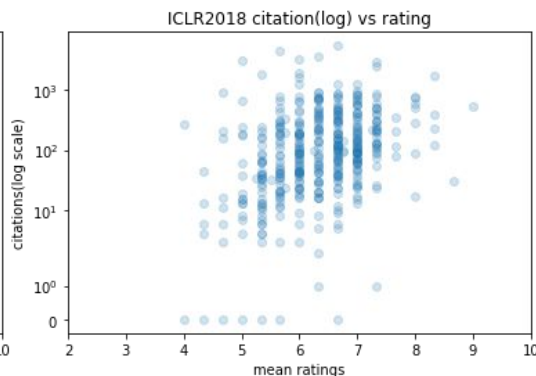
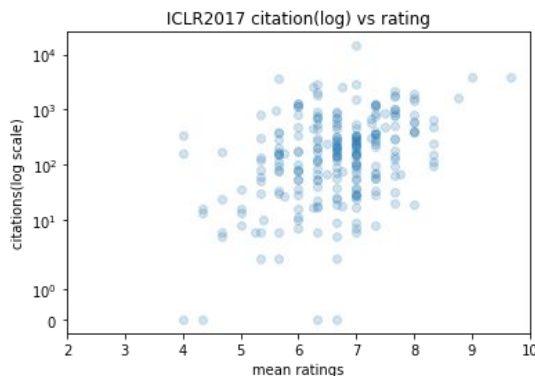


인용수와 리뷰 점수 비교하기

> Do Anything with python!



- 인용수의 log값과 리뷰어들의 리뷰 점수 사이의 관계를 pearson correlation coefficient로 나타냄.
- 해를 거듭할수록 인용수와 리뷰 점수의 상관성이 떨어지는 결과가 도출됨.
- 즉 갈수록 리뷰어들은 더 영향력이 높은(높을) 논문을 판별하는데 어려움을 겪고 있음.



Year	correlation coefficient
2017	0.401
2018	0.374
2019	0.194
2020	0.127



- **추정1.** 해가 거듭할수록 페이퍼 수와 세부 분야가 많아져 리뷰어들이 많이 필요했고, 세부 주제를 잘 모르는 리뷰어들이 리뷰에 참여해야 했기에 평균적으로 리뷰어의 질적 수준이 하락함.
 - In ICLR2020, 47% of the reviewers have not published in the related areas
<https://twitter.com/cHHillee/status/1191823707100131329>
- **추정2.** 해가 거듭할수록 페이퍼 수가 많아졌고, 그에 반해 리뷰어들의 수가 충분히 증가하지 않아 리뷰어당 더 많은 리뷰를 해야 하기에 시간 부족으로 질적 수준이 하락.
- **추정3.** 2019, 2020년 ICLR 논문들은 학계에서 아직 제대로 평가가 되지 않았다.
Citation 분포가 명확해지려면 더 많은 시간이 필요함.
 - 2018년에 얻은 ICLR2017 논문 인용수를 통해 간접적으로 검증해보자.

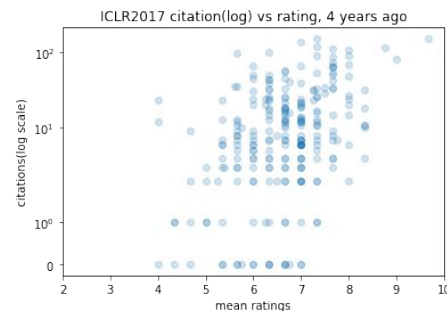
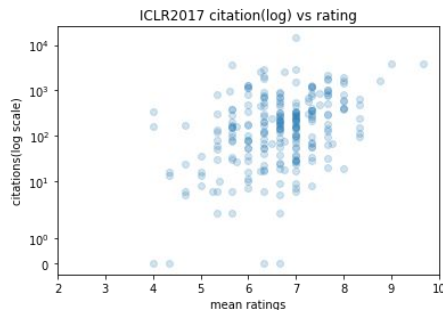
가설3의 간접 검증

> Do Anything with python!



- 2022년과 2018년에 얻은 인용수로 계산한 ICLR2017 인용수-리뷰점수 correlation coefficient는 차이가 거의 없었음.

연도	Correlation Coefficient
2018	0.401
2022	0.411



Github & Alookso

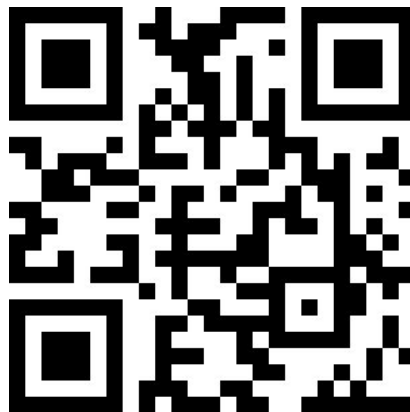
> Do Anything with python!



- 수집&전처리&분석 코드&데이터를 깃허브에 기록.



https://github.com/isingmodel/openreview_ratings_vs_citations



얼룩소에도 놀러오세요~