The Paper Review Process

and How to Critique a Paper

CS290b

Harvard John A. Paulson
School of Engineering
and Applied Sciences

Today:

- Overview of the paper review process
- How to be a “good” reviewer (and how to be a “bad” reviewer)
- Simulating the program committee decision-making process
  - You will be acting as “Area Chairs”
  - The teaching staff will be the senior area chairs/program committee chairs

What is peer review?

- **[Wikipedia]** Scholarly peer review or academic peer review is the process of having a draft version of a researcher's methods and findings reviewed (usually anonymously) by experts (or "peers") in the same field.

- **[NIH]** A peer-reviewed publication contains original articles that have been written by researchers and evaluated for technical and scientific quality and correctness by other experts in the same field.

Why do we need peer review?

- Why not just post papers online (e.g., Arxiv) and move on with our lives?

Why do we need peer review?

- Why not just post papers online (e.g., Arxiv) and move on with our lives?
- **Science is done in writing.**
- Results need to be reproducible and understandable by your peers.
- You (and your co-authors) are too close to your work.
- Peer-review can improve papers, help identify key contributions, confirm (or reject) claims of novelty, and position a paper within the literature.

A bit of context…

- Peer-review is a relatively new process in scholarly research.
  - First peer-reviewed publications was *Medical Essays and Observations* by the Royal Society of Edinburgh in 1731

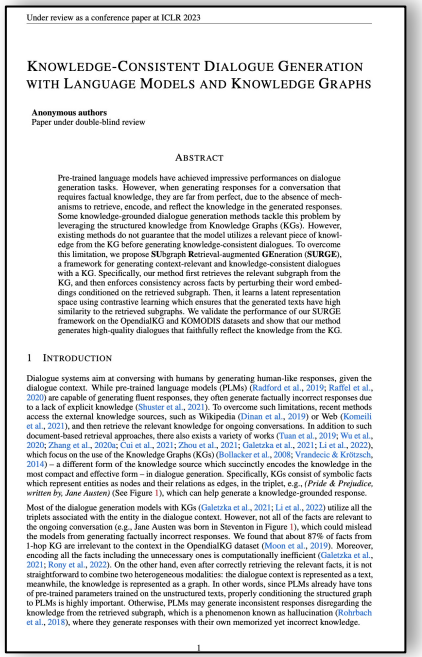
- Nature – one of the premier scientific journals – only made peer review mandatory in 1973
  - Decisions were made by editors before that
  - The journal was founded in 1869!

- Peer-review is now wide-spread.
  - Conferences and journals that are not peer-reviewed have less impact.

Typical venues where you find peer review in CS

- Workshops and small conferences (<100 submissions, often co-located with other conferences)
- Medium to Large Conferences (100s of submissions, e.g. KDD, COLT)
- Very Large Conferences (1000s of submission, e.g. ICML, NeurIPS)
- Journals (JMLR, ACM Transactions, IEEE Transactions)

Overview of the conference review process



Under review as a conference paper at ICLR 2023

KNOWLEDGE-CONSISTENT DIALOGUE GENERATION WITH LANGUAGE MODELS AND KNOWLEDGE GRAPHS

Anonymous authors
Paper under double-blind review

ABSTRACT

Pre-trained language models have achieved impressive performances on dialogue generation tasks. However, when generating responses for a conversation that requires factual knowledge, they are far from perfect, due to the absence of mechanisms to retrieve, encode, and reflect the knowledge in the generated responses. Some knowledge-grounded dialogue generation methods tackle this problem by leveraging the structured knowledge from Knowledge Graphs (KGs). However, existing methods do not guarantee that the model utilizes a relevant piece of knowledge from the KG before generating knowledge-consistent dialogues. To overcome this limitation, we propose **SU**bgraph **R**etrieval-augmented **GE**neration (**SURGE**), a framework for generating context-relevant and knowledge-consistent dialogues with a KG. Specifically, our method first retrieves the relevant subgraph from the KG, and then enforces consistency across facts by perturbing their word embeddings conditioned on the retrieved subgraph. Then, it learns a latent representation space using contrastive learning which ensures that the generated texts have high similarity to the retrieved subgraphs. We validate the performance of our SURGE framework on the OpendialKG and KOMODIS datasets and show that our method generates high-quality dialogues that faithfully reflect the knowledge from the KG.

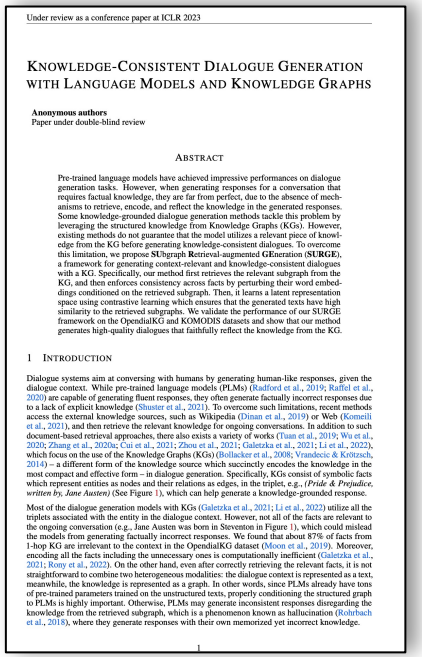
1 INTRODUCTION

Dialogue systems aim at conversing with humans by generating human-like responses, given the dialogue context. While pre-trained language models (PLMs) (Radford et al., 2019; Raffel et al., 2020) are capable of generating fluent responses, they often generate factually incorrect responses due to a lack of explicit knowledge (Shuster et al., 2021). To overcome such limitations, recent methods access the external knowledge sources, such as Wikipedia (Dinan et al., 2019) or Web (Komeili et al., 2021), and then retrieve the relevant knowledge for ongoing conversations. In addition to such document-based retrieval approaches, there also exists a variety of works (Tuan et al., 2019; Wu et al., 2020; Zhang et al., 2020a; Cui et al., 2021; Zhou et al., 2021; Galetzka et al., 2021; Li et al., 2022), which focus on the use of the Knowledge Graphs (KGs) (Bollacker et al., 2008; Vrandecic & Krötzsch, 2014) – a different form of the knowledge source which succinctly encodes the knowledge in the most compact and effective form – in dialogue generation. Specifically, KGs consist of symbolic facts which represent entities as nodes and their relations as edges, in the triplet, e.g., (*Pride & Prejudice, written by, Jane Austen*) (See Figure 1), which can help generate a knowledge-grounded response.

Most of the dialogue generation models with KGs (Galetzka et al., 2021; Li et al., 2022) utilize all the triplets associated with the entity in the dialogue context. However, not all of the facts are relevant to the ongoing conversation (e.g., Jane Austen was born in Steventon in Figure 1), which could mislead the models from generating factually incorrect responses. We found that about 87% of facts from 1-hop KG are irrelevant to the context in the OpendialKG dataset (Moon et al., 2019). Moreover, encoding all the facts including the unnecessary ones is computationally inefficient (Galetzka et al., 2021; Rony et al., 2022). On the other hand, even after correctly retrieving the relevant facts, it is not straightforward to combine two heterogeneous modalities: the dialogue context is represented as a text, meanwhile, the knowledge is represented as a graph. In other words, since PLMs already have tons of pre-trained parameters trained on the unstructured texts, properly conditioning the structured graph to PLMs is highly important. Otherwise, PLMs may generate inconsistent responses disregarding the knowledge from the retrieved subgraph, which is a phenomenon known as hallucination (Rohrbach et al., 2018), where they generate responses with their own memorized yet incorrect knowledge.

1

Overview of the conference review process



Under review as a conference paper at ICLR 2023

KNOWLEDGE-CONSISTENT DIALOGUE GENERATION WITH LANGUAGE MODELS AND KNOWLEDGE GRAPHS

Anonymous authors
Paper under double-blind review

ABSTRACT

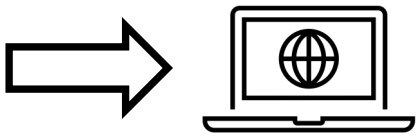
Pre-trained language models have achieved impressive performances on dialogue generation tasks. However, when generating responses for a conversation that requires factual knowledge, they are far from perfect, due to the absence of mechanisms to retrieve, encode, and reflect the knowledge in the generated responses. Some knowledge-grounded dialogue generation methods tackle this problem by leveraging the structured knowledge from Knowledge Graphs (KGs). However, existing methods do not guarantee that the model utilizes a relevant piece of knowledge from the KG before generating knowledge-consistent dialogues. To overcome this limitation, we propose **SU**bgraph **R**etrieval-augmented **GE**neration (**SURGE**), a framework for generating context-relevant and knowledge-consistent dialogues with a KG. Specifically, our method first retrieves the relevant subgraph from the KG, and then enforces consistency across facts by perturbing their word embeddings conditioned on the retrieved subgraph. Then, it learns a latent representation space using contrastive learning which ensures that the generated texts have high similarity to the retrieved subgraphs. We validate the performance of our SURGE framework on the OpendialKG and KOMODIS datasets and show that our method generates high-quality dialogues that faithfully reflect the knowledge from the KG.

1 INTRODUCTION

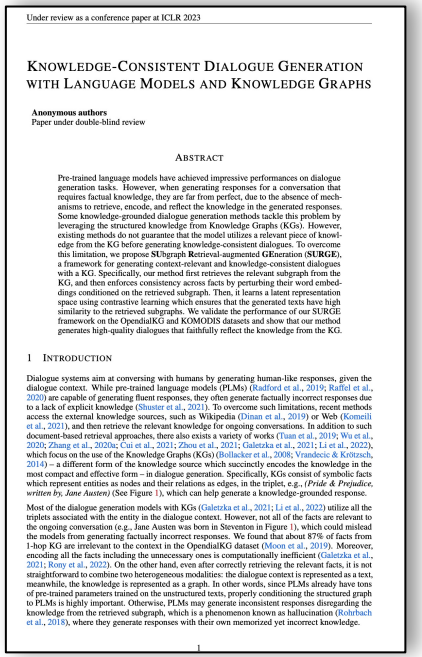
Dialogue systems aim at conversing with humans by generating human-like responses, given the dialogue context. While pre-trained language models (PLMs) (Radford et al., 2019; Raffel et al., 2020) are capable of generating fluent responses, they often generate factually incorrect responses due to a lack of explicit knowledge (Shuster et al., 2021). To overcome such limitations, recent methods access the external knowledge sources, such as Wikipedia (Dinan et al., 2019) or Web (Komeili et al., 2021), and then retrieve the relevant knowledge for ongoing conversations. In addition to such document-based retrieval approaches, there also exists a variety of works (Tuan et al., 2019; Wu et al., 2020; Zhang et al., 2020a; Cui et al., 2021; Zhou et al., 2021; Galetzka et al., 2021; Li et al., 2022), which focus on the use of the Knowledge Graphs (KGs) (Bollacker et al., 2008; Vrandecic & Krötzsch, 2014) – a different form of the knowledge source which succinctly encodes the knowledge in the most compact and effective form – in dialogue generation. Specifically, KGs consist of symbolic facts which represent entities as nodes and their relations as edges, in the triplet, e.g., *(Pride & Prejudice, written by, Jane Austen)* (See Figure 1), which can help generate a knowledge-grounded response.

Most of the dialogue generation models with KGs (Galetzka et al., 2021; Li et al., 2022) utilize all the triplets associated with the entity in the dialogue context. However, not all of the facts are relevant to the ongoing conversation (e.g., Jane Austen was born in Steventon in Figure 1), which could mislead the models from generating factually incorrect responses. We found that about 87% of facts from 1-hop KG are irrelevant to the context in the OpendialKG dataset (Moon et al., 2019). Moreover, encoding all the facts including the unnecessary ones is computationally inefficient (Galetzka et al., 2021; Rony et al., 2022). On the other hand, even after correctly retrieving the relevant facts, it is not straightforward to combine two heterogeneous modalities: the dialogue context is represented as a text, meanwhile, the knowledge is represented as a graph. In other words, since PLMs already have tons of pre-trained parameters trained on the unstructured texts, properly conditioning the structured graph to PLMs is highly important. Otherwise, PLMs may generate inconsistent responses disregarding the knowledge from the retrieved subgraph, which is a phenomenon known as hallucination (Rohrbach et al., 2018), where they generate responses with their own memorized yet incorrect knowledge.

1



Overview of the conference review process



Under review as a conference paper at ICLR 2023

KNOWLEDGE-CONSISTENT DIALOGUE GENERATION WITH LANGUAGE MODELS AND KNOWLEDGE GRAPHS

Anonymous authors
Paper under double-blind review

ABSTRACT

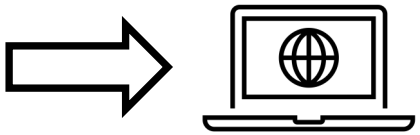
Pre-trained language models have achieved impressive performances on dialogue generation tasks. However, when generating responses for a conversation that requires factual knowledge, they are far from perfect, due to the absence of mechanisms to retrieve, encode, and reflect the knowledge in the generated responses. Some knowledge-grounded dialogue generation methods tackle this problem by leveraging the structured knowledge from Knowledge Graphs (KGs). However, existing methods do not guarantee that the model utilizes a relevant piece of knowledge from the KG before generating knowledge-consistent dialogues. To overcome this limitation, we propose **SU**bgraph **R**etrieval-augmented **GE**neration (**SURGE**), a framework for generating context-relevant and knowledge-consistent dialogues with a KG. Specifically, our method first retrieves the relevant subgraph from the KG, and then enforces consistency across facts by perturbing their word embeddings conditioned on the retrieved subgraph. Then, it learns a latent representation space using contrastive learning which ensures that the generated texts have high similarity to the retrieved subgraphs. We validate the performance of our SURGE framework on the OpendialKG and KOMODIS datasets and show that our method generates high-quality dialogues that faithfully reflect the knowledge from the KG.

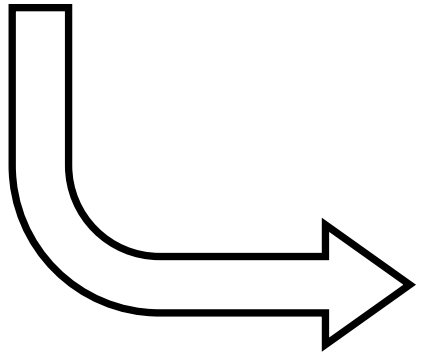
1 INTRODUCTION

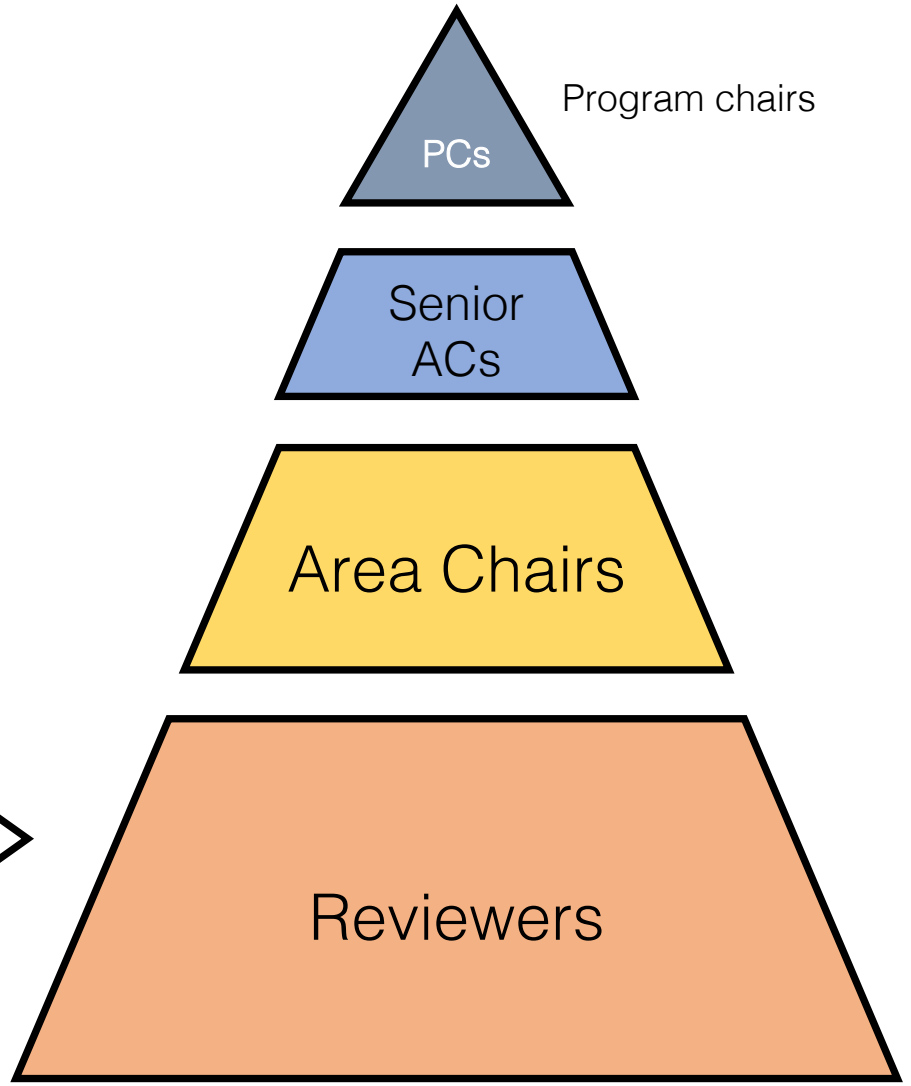
Dialogue systems aim at conversing with humans by generating human-like responses, given the dialogue context. While pre-trained language models (PLMs) (Radford et al., 2019; Raffel et al., 2020) are capable of generating fluent responses, they often generate factually incorrect responses due to a lack of explicit knowledge (Shuster et al., 2021). To overcome such limitations, recent methods access the external knowledge sources, such as Wikipedia (Dinan et al., 2019) or Web (Komeili et al., 2021), and then retrieve the relevant knowledge for ongoing conversations. In addition to such document-based retrieval approaches, there also exists a variety of works (Tuan et al., 2019; Wu et al., 2020; Zhang et al., 2020a; Cui et al., 2021; Zhou et al., 2021; Galetzka et al., 2021; Li et al., 2022), which focus on the use of the Knowledge Graphs (KGs) (Bollacker et al., 2008; Vrandecic & Krötzsch, 2014) – a different form of the knowledge source which succinctly encodes the knowledge in the most compact and effective form – in dialogue generation. Specifically, KGs consist of symbolic facts which represent entities as nodes and their relations as edges, in the triplet, e.g., *(Pride & Prejudice, written by, Jane Austen)* (See Figure 1), which can help generate a knowledge-grounded response.

Most of the dialogue generation models with KGs (Galetzka et al., 2021; Li et al., 2022) utilize all the triplets associated with the entity in the dialogue context. However, not all of the facts are relevant to the ongoing conversation (e.g., Jane Austen was born in Steventon in Figure 1), which could mislead the models from generating factually incorrect responses. We found that about 87% of facts from 1-hop KG are irrelevant to the context in the OpendialKG dataset (Moon et al., 2019). Moreover, encoding all the facts including the unnecessary ones is computationally inefficient (Galetzka et al., 2021; Rony et al., 2022). On the other hand, even after correctly retrieving the relevant facts, it is not straightforward to combine two heterogeneous modalities: the dialogue context is represented as a text, meanwhile, the knowledge is represented as a graph. In other words, since PLMs already have tons of pre-trained parameters trained on the unstructured texts, properly conditioning the structured graph to PLMs is highly important. Otherwise, PLMs may generate inconsistent responses disregarding the knowledge from the retrieved subgraph, which is a phenomenon known as hallucination (Rohrbach et al., 2018), where they generate responses with their own memorized yet incorrect knowledge.

1







Nomenclature may change. This is from ML

Reviewers

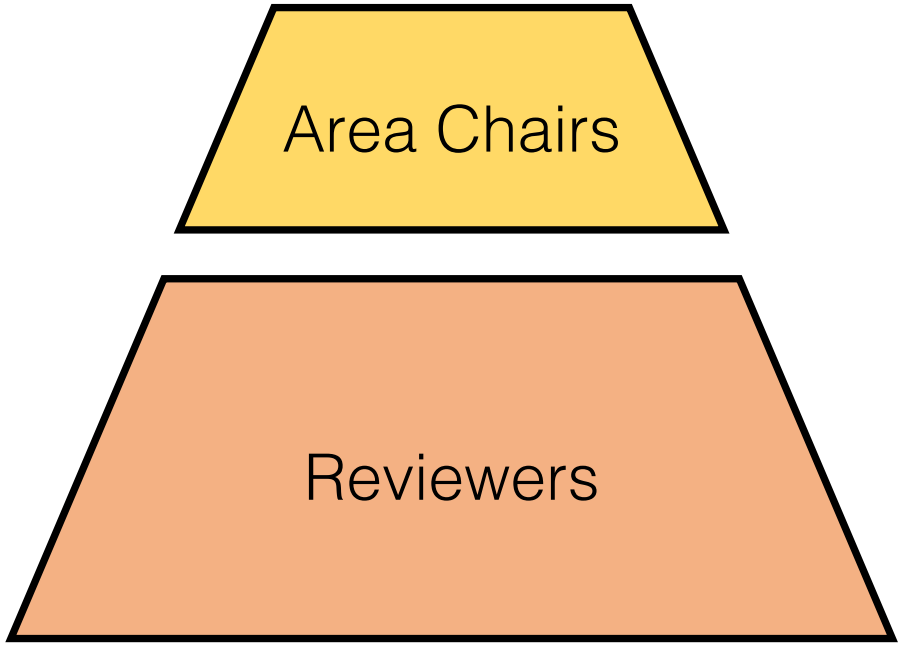
Reviewers

- Usually 3-4 per submission (sometimes more, rarely less)
- Their job is to evaluate in terms of the following intertwined aspects:
  - **Technical correctness:** are the experiments/claims/theorems/algorithms correct?
  - **Significance:** do the contributions of the paper merit publication?
  - **Novelty:** does the paper introduce new contributions compared to existing literature?
  - **Clarity:** is the paper clearly written? Are the contributions explained?
  - **Reproducibility:** can the authors' claims be reproduced?

Reviewers

- The review process is usually categorized in terms of who the reviewer can identify:
  - **Blind**: Authors do not know reviewer identities. Common in journals
  - **Double blind:** Blind + reviewers don't know the identity of the authors, but may know who are the other reviewers [reduces bias based on authors' reputation]
  - **Triple blind:** Double blind + reviewers do not know the identity of other reviewers [reduces bias based on reviewers' reputation]

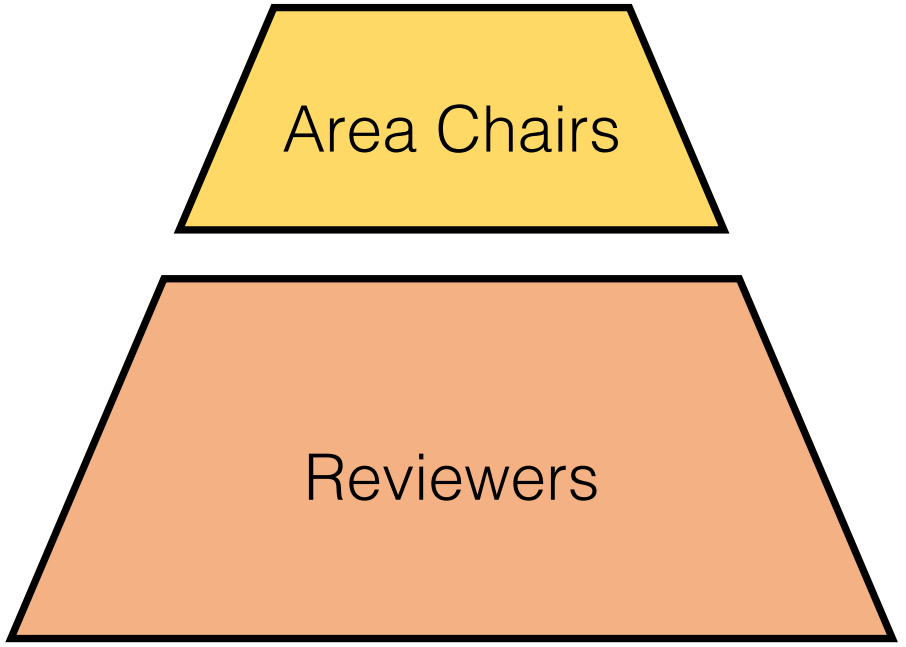
Area Chairs



Nomenclature may change. This is from ML

Area Chairs

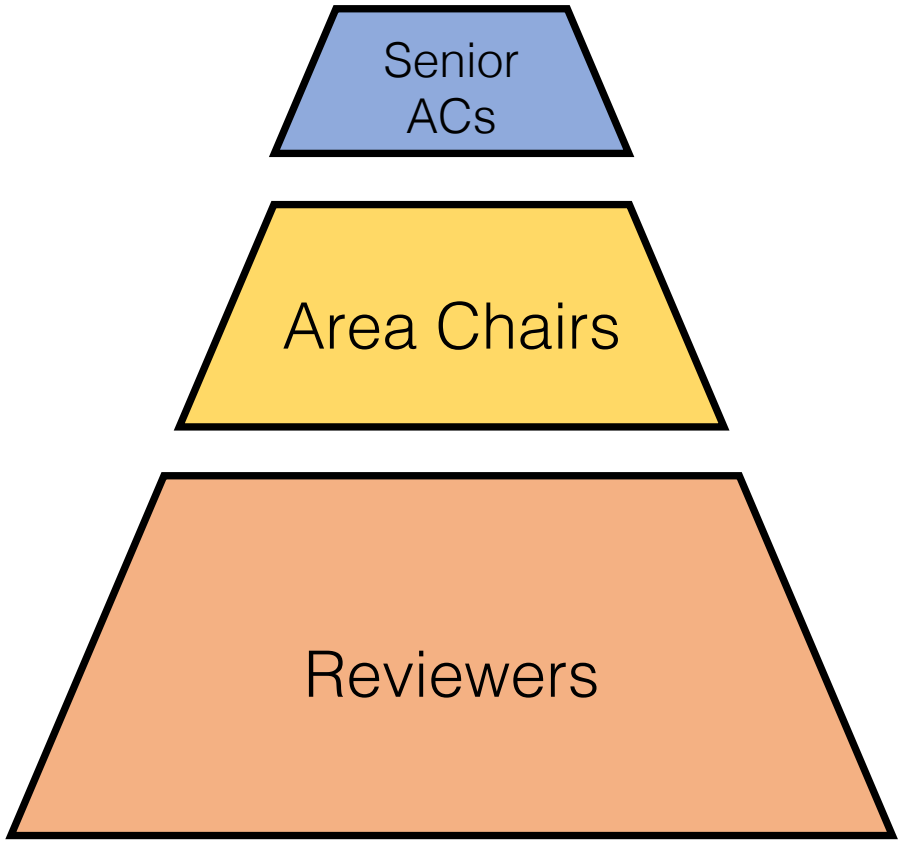
- Usually handles 10-15 papers.
- Assigns reviewers and requests additional reviews if needed.
- Leads discussion between reviewers and ensures that reviewers are responsive and on time.
- Writes a meta-review summarizing discussion and provides a recommendation.



Nomenclature may change. This is from ML

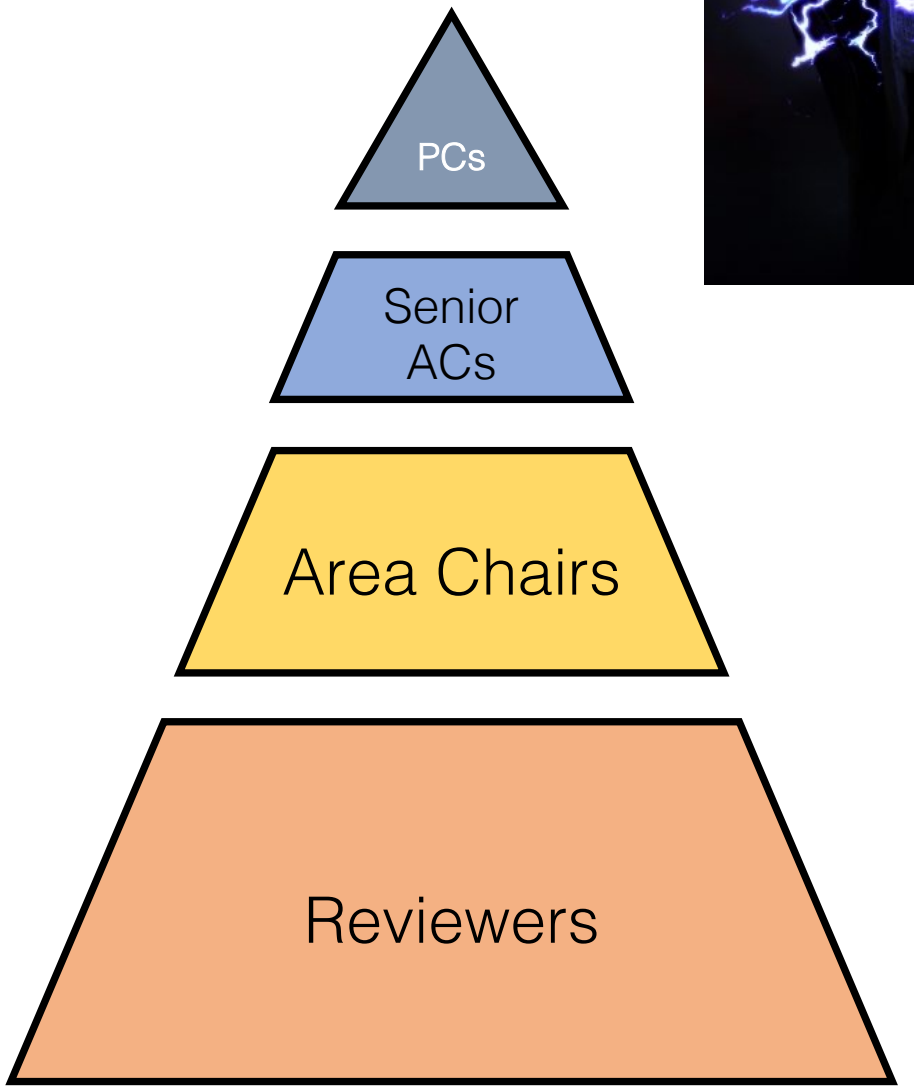
Senior Area Chairs

- In large conferences, a senior area chair is often responsible for ~100 papers.
- Handles unresponsive ACs, difficult cases (e.g., plagiarism), and award nominations.
- In some cases, may overturn AC decisions.



Program Chairs

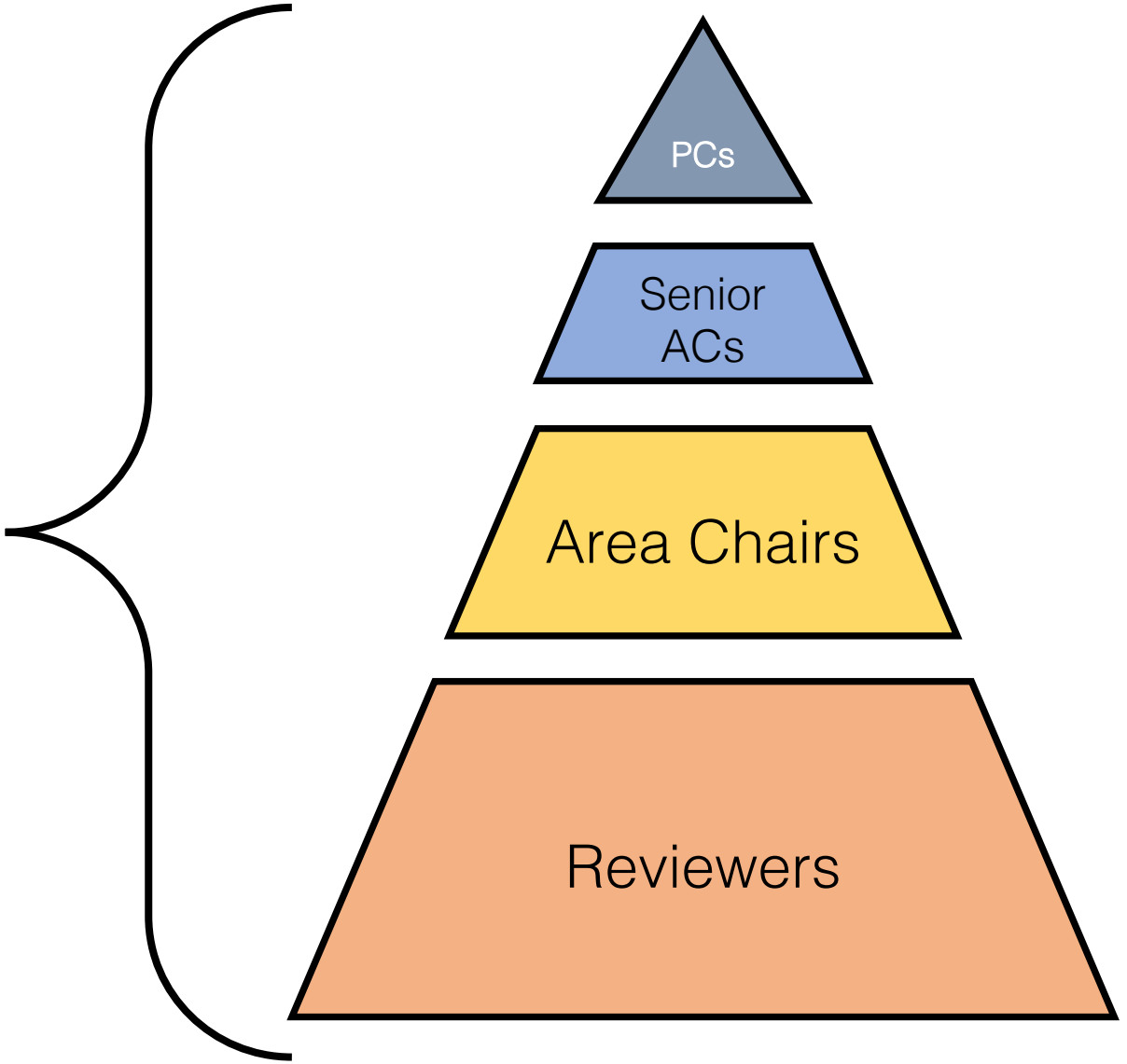
- The PCs usually form and approve the SAC committee
- Organize meeting with SAC to finalize list of accepted papers and select award-winning papers.
- Deal with issues such as reviewer misconduct, dual-submission, etc.
- At conference, the PCs change from year to year.



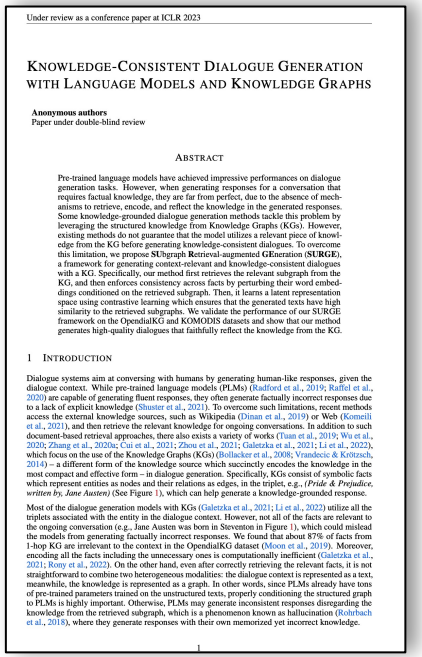
This hierarchy appears in some form across publication venues that are peer-reviewed.

Journals have **editors** instead of area chairs. Editors often serve a pre-determined term.

Editors have a similar role to ACs, but often can be suggested by submitting authors.



Interaction between authors and reviewers



Under review as a conference paper at ICLR 2023

KNOWLEDGE-CONSISTENT DIALOGUE GENERATION WITH LANGUAGE MODELS AND KNOWLEDGE GRAPHS

Anonymous authors
Paper under double-blind review

ABSTRACT

Pre-trained language models have achieved impressive performances on dialogue generation tasks. However, when generating responses for a conversation that requires factual knowledge, they are far from perfect, due to the absence of mechanisms to retrieve, encode, and reflect the knowledge in the generated responses. Some knowledge-grounded dialogue generation methods tackle this problem by leveraging the structured knowledge from Knowledge Graphs (KGs). However, existing methods do not guarantee that the model utilizes a relevant piece of knowledge from the KG before generating knowledge-consistent dialogues. To overcome this limitation, we propose **SU**bgraph **R**etrieval-augmented **GE**neration (**SURGE**), a framework for generating context-relevant and knowledge-consistent dialogues with a KG. Specifically, our method first retrieves the relevant subgraph from the KG, and then enforces consistency across facts by perturbing their word embeddings conditioned on the retrieved subgraph. Then, it learns a latent representation space using contrastive learning which ensures that the generated texts have high similarity to the retrieved subgraphs. We validate the performance of our SURGE framework on the OpendialKG and KOMODIS datasets and show that our method generates high-quality dialogues that faithfully reflect the knowledge from the KG.

1 INTRODUCTION

Dialogue systems aim at conversing with humans by generating human-like responses, given the dialogue context. While pre-trained language models (PLMs) (Radford et al., 2019; Raffel et al., 2020) are capable of generating fluent responses, they often generate factually incorrect responses due to a lack of explicit knowledge (Shuster et al., 2021). To overcome such limitations, recent methods access the external knowledge sources, such as Wikipedia (Dinan et al., 2019) or Web (Komeili et al., 2021), and then retrieve the relevant knowledge for ongoing conversations. In addition to such document-based retrieval approaches, there also exists a variety of works (Tuan et al., 2019; Wu et al., 2020; Zhang et al., 2020a; Cui et al., 2021; Zhou et al., 2021; Galetzka et al., 2021; Li et al., 2022), which focus on the use of the Knowledge Graphs (KGs) (Bollacker et al., 2008; Vrandecic & Krötzsch, 2014) – a different form of the knowledge source which succinctly encodes the knowledge in the most compact and effective form – in dialogue generation. Specifically, KGs consist of symbolic facts which represent entities as nodes and their relations as edges, in the triplet, e.g., *(Pride & Prejudice, written by, Jane Austen)* (See Figure 1), which can help generate a knowledge-grounded response.

Most of the dialogue generation models with KGs (Galetzka et al., 2021; Li et al., 2022) utilize all the triplets associated with the entity in the dialogue context. However, not all of the facts are relevant to the ongoing conversation (e.g., Jane Austen was born in Steventon in Figure 1), which could mislead the models From generating factually incorrect responses. We found that about 87% of facts from 1-hop KG are irrelevant to the context in the OpendialKG dataset (Moon et al., 2019). Moreover, encoding all the facts including the unnecessary ones is computationally inefficient (Galetzka et al., 2021; Rony et al., 2022). On the other hand, even after correctly retrieving the relevant facts, it is not straightforward to combine two heterogeneous modalities: the dialogue context is represented as a text, meanwhile, the knowledge is represented as a graph. In other words, since PLMs already have tons of pre-trained parameters trained on the unstructured texts, properly conditioning the structured graph to PLMs is highly important. Otherwise, PLMs may generate inconsistent responses disregarding the knowledge from the retrieved subgraph, which is a phenomenon known as hallucination (Rohrbach et al., 2018), where they generate responses with their own memorized yet incorrect knowledge.

1

Reviewers

- In many conferences, authors have a chance to respond to reviewers.
- This is done in a short “rebuttal phase”

- In journals, papers can be marked as “major” or “minor” revision. Authors respond to reviews and resubmit an updated manuscript.

The rebuttal/discussion phase

- Rebuttal phases are a chance to ask **clarifying questions** to the authors.
- As a reviewer, ask questions that pave a path forward for the paper.
- Avoid asking questions that are unrealistic or would require another round of peer-review.

The rebuttal/discussion phase

- Rebuttal phases are a chance to ask **clarifying questions** to the authors.
- As a reviewer, ask questions that pave a path forward for the paper.
- Avoid asking questions that are unrealistic or would require another round of peer-review.

- Good questions for reviewers to ask in a rebuttal:
  - How does your result compare with [reference]?
  - Can you clarify why the step from Eq. (40) to (41)?
  - Can you explain the parameters used to generate Figure 3?

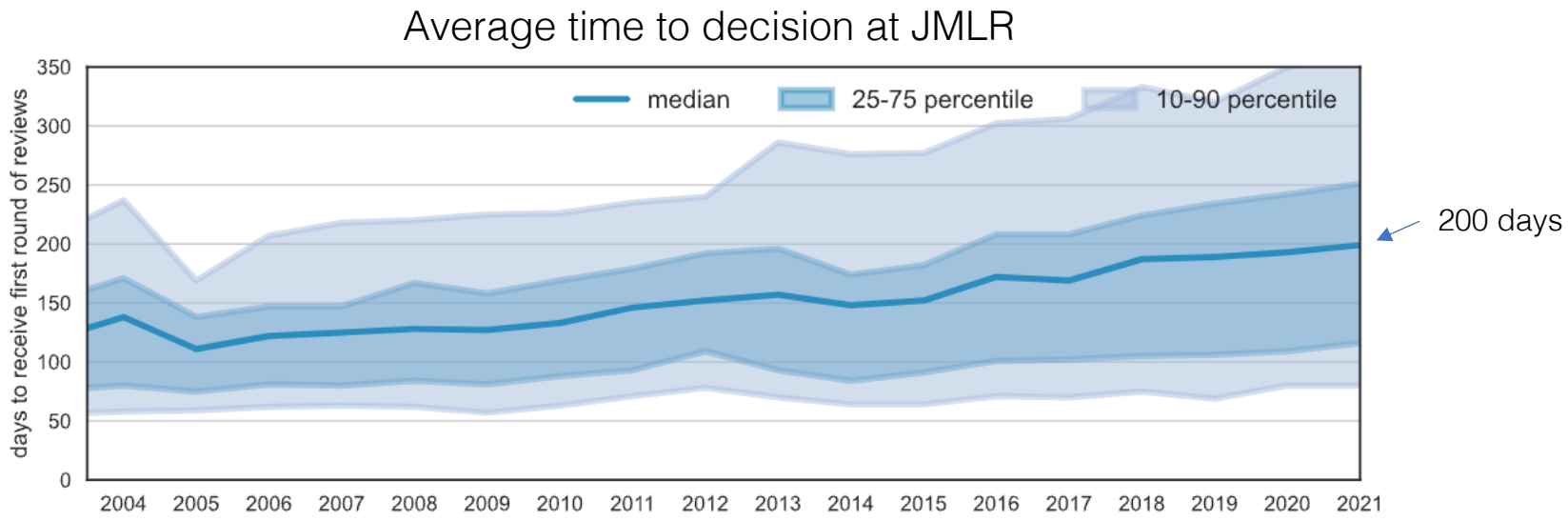
The rebuttal/discussion phase

- Rebuttal phases are a chance to ask **clarifying questions** to the authors.
- As a reviewer, ask questions that pave a path forward for the paper.
- Avoid asking questions that are unrealistic or would require another round of peer-review.

- Good questions for reviewers to ask in a rebuttal:
  - How does your result compare with [reference]?
  - Can you clarify why the step from Eq. (40) to (41)?
  - Can you explain the parameters used to generate Figure 3?

- Bad questions:
  - Can you run this on 10 more datasets?
  - Your paper is about X, but why did you not talk about Y?
  - Can you generalize Theorems 1-4 to a much more complicated setting?
  - [Blank] (i.e., asking no questions at all).

Reviewing takes time!



Conferences have a fixed timeframe (usually ~2-4 months)

The review process is noisy

The review process is noisy

- NeurIPS' 21 consistency experiment
  - 882 papers were each given to two separate groups of reviewers, ACs, and SACs.
  - Authors were asked to answer each review separately.

The review process is noisy

- NeurIPS' 21 consistency experiment
  - 882 papers were each given to two separate groups of reviewers, ACs, and SACs.
  - Authors were asked to answer each review separately.

Original \ Copy	Oral	Spotlight	Poster	Reject	Withdrawn
Oral	0	0	4	0	0
Spotlight	0	3	9	13	0
Poster	2	7	74	94	0
Reject	0	13	83	462	0
Withdrawn	0	0	0	0	118

The review process is noisy

- NeurIPS' 21 consistency experiment:
  - 882 papers were each given to two separate groups of reviewers, ACs, and SACs.
  - Authors were asked to answer each review separately.

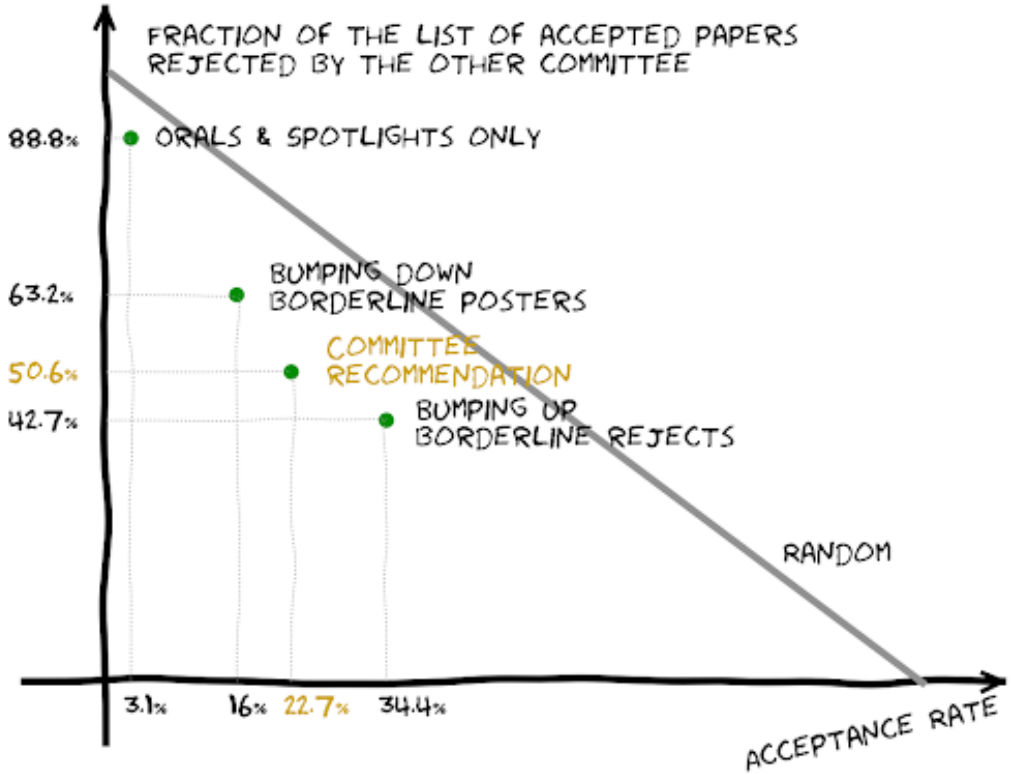
Only 3 agreement in Oral/spotlight papers

Original \ Copy	Oral	Spotlight	Poster	Reject	Withdrawn
Oral	0	0	4	0	0
Spotlight	0	3	9	13	0
Poster	2	7	74	94	0
Reject	0	13	83	462	0
Withdrawn	0	0	0	0	118

The review process is noisy

NEURAL INFORMATION PROCESSING SYSTEMS

- Fraction of accepted papers that would have changed if we reran the review process
- Equivalent to the probability that a randomly chosen paper would have been rejected if re-reviewed



The review process is noisy

- Fraction of accepted papers that would have changed if we reran the review process
- Equivale ve been reje

Takeaway:
Do not take a review personally!

Next: Being a Good Reviewer

First rule of reviewing:

First rule of reviewing:

Never accept a review you don't have time to do.

First rule of reviewing:

Never accept a review you don't have time to do.

- **You can say no!**
- Only accept papers you have time for.
- **Do not** feel obligated to do a review that you know will be rushed.

Second rule of reviewing:

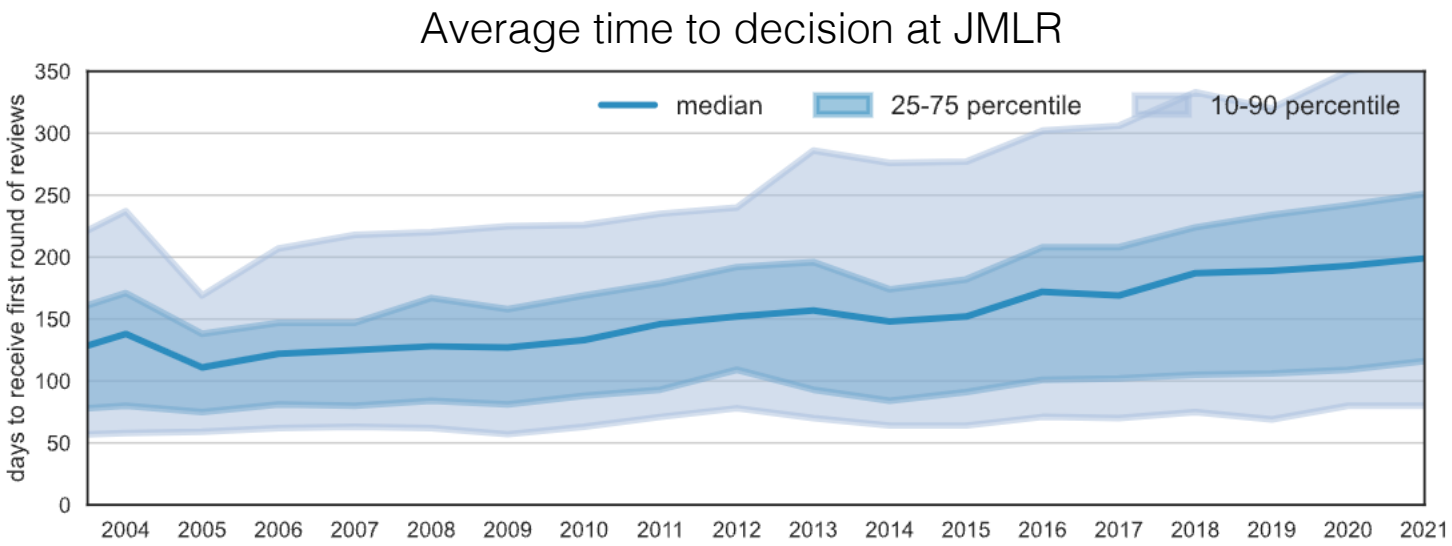
Second rule of reviewing:

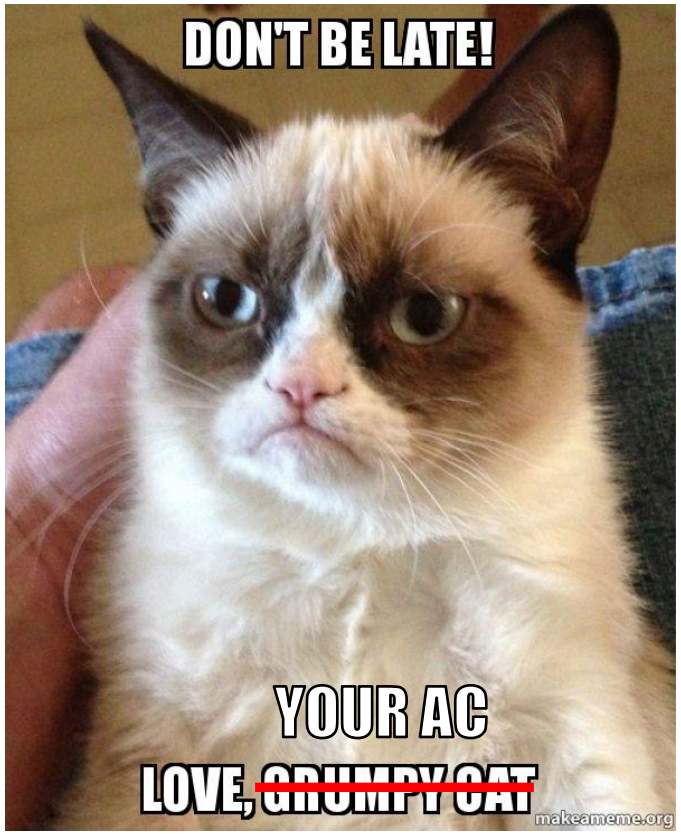
Never, ever accept a review you don't have time to do.

Third rule of reviewing:

Third rule of reviewing:

If you accept a review, turn in your review on time.





If you follow these 3 rules, you will already be in the top-25% of reviewers:

1. **Never** accept a review you don't have time to do.
2. **Never, ever** accept a review you don't have time to do.
3. **Don't be late.**

Now for the rest…

Tasks of a reviewer

- Help authors improve their paper
  - Review as you would like your papers to be reviewed (be firm but polite)
  - Give **actionable** feedback
  - If you think a paper should be rejected, **clearly explain why**
- Help the AC or Editor understand if the paper should be accepted/rejected.
  - Explain your rationale using facts and arguments based on the paper **(avoid subjective opinions)**
  - You can usually send comments directly to the AC/Editor that will not be seen by authors

Is this a useful review?

Summary:
The authors proposed a spectral tensor decomposition of a DNN into independent, parallel and computationally cheaper subnets that process corresponding spectrally decomposed data, resulting in speedup in training; ensembling of these subnets yielded reasonable generalization performance.

Strengths And Weaknesses:
[+] Novel idea

[-] Inadequate supporting experiments

Questions:
- Though a recurrent spectral tensor layer was derived, it was not demonstrated with experiments.
- The example DNNs experimented are not representative or practically relevant.
- Do spectral tensor transformers exist?
- It seems decomposition of data with orthogonal bases is not unique. Nor is the number of components and the grouping thereof. How does one choose optimal or reasonable decomposition?
- With a certain data decomposition, task-relevant information in different components naturally vary. Corresponding subnet experts might need to have different capacities to realize maximal efficiency? Or how does one equalize component importance, or do spectral pruning?
- Fig. 2, why was the FC training numerically unstable, with a non-monotonic training loss curve?

Limitations:
See above.

Ethics Flag: No

Soundness: 2 fair

Presentation: 1 poor

Contribution: 2 fair

Rating: 4: Borderline reject: Technically solid paper where reasons to reject, e.g., limited evaluation, outweigh reasons to accept, e.g., good evaluation. Please use sparingly.

Confidence: 4: You are confident in your assessment, but not absolutely certain. It is unlikely, but not impossible, that you did not understand some parts of the submission or that you are unfamiliar with some pieces of related work.

Code Of Conduct: Yes

And this one?

Summary:
This paper studies the tensor completion problem for tensors with non-negative entries. The paper proposes a non-negative analog of the nuclear norm, denoted $\|\cdot\|_+$ whose 1-ball is defined to be a convex combination of rank-1 $0,1$-tensors. The paper proposes to solve the problem, dual to $\|\cdot\|_+$-minimization to recover unknown non-negative tensor from a subset of observed entries. This approach is a natural analog of the nuclear norm minimization for general tensors. The paper proves that this approach w.h.p. can recover the unknown tensor from an essentially statistically optimal number of unknown samples. The authors prove that this minimization problem is NP-hard. At the same time, they show that the BCG algorithm can solve the problem with a linear number of calls to a linear separation oracle. Moreover, the paper proposes a heuristic algorithm for the oracle and studies its performance on synthetic data.

Strengths And Weaknesses:
Significance: Tensors with non-negative entries frequently appear in practice, hence improved algorithms for this setup are likely to have further applications. This paper proposes a natural idea for the problem, which does not seem to be explored in the prior work. The experimental results presented by the authors show that their heuristic algorithm has better performance on tensors with low non-negative rank compared to some state-of-the-art approaches.

Clarity: The technical proofs seem to be correct and the paper is reasonably well-written. However, there is a number of vague statements, typos, or claims that are potentially overstated. See questions below. Additionally, it will be very helpful for the reader if details of BCG algorithm are included at least in the appendix.

Finally, the paper uses notation that is non-standard in the literature, which makes it a bit harder to read. For instance, r is typically reserved for the rank of the tensor, while this paper uses k for the rank and r_i for the dimensions of the tensor.

Questions:
1. Can you please clarify what results in the papers cited on lines 33-34 achieve information-theoretic bounds? Are there specific numbered theorems? As far as I know, for $n \times n \times n$ tensors of rank r, Yuan and Zhang'16 prove recovery from r^{1/2}n^{3/2} samples, which is not the information-theoretic rate ($\widetilde{O}(rn)$). I was not able to locate the corresponding information-theoretic rate results in the other two cited papers. I would like to kindly suggest to include exact statements of the prior work that achieves information-theoretic bounds in the appendix, as they are quite rare and are directly related to the main contribution of this paper.
2. Cor 4.3 does not seem to achieve the information-theoretic bound unless the rank $k = O(1)$. Do I understand correctly that if $e = 0$, Cor 4.3 proves recovery from essentially $k^4 \cdot \rho$ entries (as opposed to $k \cdot \rho$)? If that's correct, I believe the authors should either explicitly say that $k = O(1)$, or they should be more explicit in the description of their results.
3. The abstract and the text of the paper contain the claim: "We prove that our algorithm converges in a linear (in numerical tolerance) number of oracle steps, while achieving the information-theoretic rate." However, the nature of the oracle does not seem to be mentioned until the end of the paper. In particular, it is not mentioned that this oracle solves an NP-hard problem. I believe those are important details, which if mentioned early give a better understanding of the results of the paper.
4. Are there any additional simple assumptions that guarantee that the oracle can be implemented in polynomial time? This will give specific assumptions under which your algorithm for (8) works in polynomial time. Note that for standard nuclear norm (NN) minimization there are some regimes in which problem can be solved in polynomial time, even though the general problem is NP-hard. For example, using SOS, NN minimization is known for tensors with orthogonal components by Potechin-Steurer'17 and for low-rank tensors with random components by Kivva-Potechin'20.
5. Prop. 3.1 and Cor 3.3 are a bit hard to understand, and I needed to look into the paper of Lecue et al to understand the claim. For example, in Prop 3.1, what is y? This variable seems to have meaning only when comes in pair with x, i.e., in pair $x\langle i\rangle, y\langle i\rangle$. It will also simplify reading if the definition of $\mathbb{E}$ used in L197 is included. I believe it just stands for average over all entries of the tensor, but there are other ways to interpret it. What is the difference between ψ and $\widehat{\psi}$ in Prop. 4.1 and Cor 4.3?
6. Did you try to run your algorithm on any real-world dataset? How does it compare to state-of-the-art approaches on such datasets? How does your

Not your job as a Reviewer:

- Evaluate the authors' intelligence or if they are "hardcore."
- Change the tone/notation/presentation of the paper to a form that you would prefer.
- Evaluate the authors' command of the English language.
- Increase citations to your own work.
- Impose your personal bias on of a given field research sub-field.

Writing a good review

- Start with a brief summary of the paper and its (claimed) main contributions

- Why a summary?
  - Helps the authors understand what **you** identified as important in the paper.
  - Useful for the AC/SAC who are handling dozens of papers.
  - Provide context for the remainder of the review.

Writing a good review

- After the summary, describe the **strengths** and **weaknesses** of the paper.
  - Explain **why** a given point is either a strength or a weakness.
  - Motivate claims with **specific examples** from paper.
  - Provide arguments that are **refutable!**
  - Give **constructive feedback**.
- Ask **concrete questions** that would sway you and are realistic.
  - Don’t give busy work to the authors if those results won’t change your mind.
- At conferences, there are usually pre-defined review fields.

Writing a good review

- Do we gain new knowledge that we can build in future research?

Writing a good review

- Do we gain new knowledge that we can build in future research?

- Is this paper within the scope of this venue?

Writing a good review

- Do we gain new knowledge that we can build in future research?
- Is this paper within the scope of this venue?
- Is the paper understandable?
  - Is a paper difficult to understand due to ambiguity in writing or unnecessary complexity?
  - Some papers are hard. But they should not be **unnecessarily** hard.
  - You are an expert in your field. If you can't understand a paper, the problem may not be you.
  - Let the AC know if you are **not** an expert, or if you did not have time to check details.

Writing a good review

- Do we gain new knowledge that we can build in future research?
- Is this paper within the scope of this venue?
- Is the paper understandable?
  - Is a paper difficult to understand due to ambiguity in writing or unnecessary complexity?
  - Some papers are hard. But they should not be **unnecessarily** hard.
  - You are an expert in your field. If you can’t understand a paper, the problem may not be you.
  - Let the AC know if you are **not** an expert, or if you did not have time to check details.
- Does the paper give enough information to **independently** reproduce the results?
  - This is more than ”attaching code”!

Writing a bad review

- Stating that correct/novel results are too simple or trivial.
  - A paper doesn’t have to be difficult!

Writing a bad review

- Stating that correct/novel results are too simple or trivial.
  - A paper doesn't have to be difficult!

"This is folklore in the field, though I couldn't find a reference for a proof of this result."

- Reviewer 2 crushing my young grad-student soul.

Writing a bad review

- Stating that correct/novel results are too simple or trivial.
  - A paper doesn't have to be difficult!

"This is folklore in the field, though I couldn't find a reference for a proof of this result."

- Reviewer 2 crushing my young grad-student soul.

- Basing a review on personal opinion.
  - For example, "Information theory is a semi-dead field."

Writing a bad review

- Stating that correct/novel results are too simple or trivial.
  - A paper doesn't have to be difficult!

"This is folklore in the field, though I couldn't find a reference for a proof of this result."

- Reviewer 2 crushing my young grad-student soul.

- Basing a review on personal opinion.
  - For example, "Information theory is a semi-dead field."

- Asking for unnecessary extensive experimentation (e.g., "try it out on 5 more datasets" or "apply this to LLMs).
  - Ask yourself: can you articulate why the current results are not convincing?

Writing a bad review

- Stating that correct/novel results are too simple or trivial.
  - A paper doesn't have to be difficult!

"This is folklore in the field, though I couldn't find a reference for a proof of this result."

- Reviewer 2 crushing my young grad-student soul.

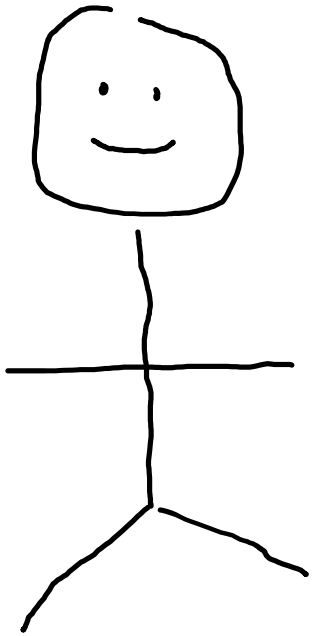
- Basing a review on personal opinion.
  - For example, "Information theory is a semi-dead field."

- Asking for unnecessary extensive experimentation (e.g., "try it out on 5 more datasets" or "apply this to LLMs).
  - Ask yourself: can you articulate why the current results are not convincing?

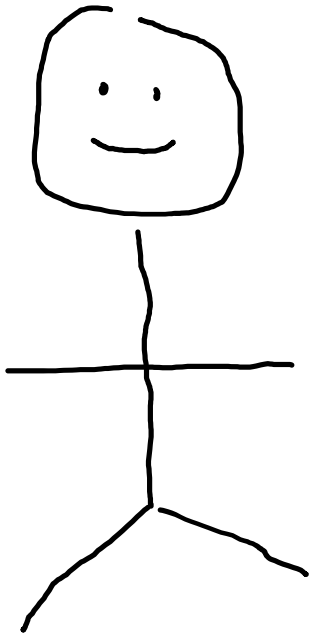
- Accepting a paper because the author is well-established.
  - Hey, I follow this person on Twitter! They have the hottest takes on xyz.

The paper killer

This is Reviewer 2.



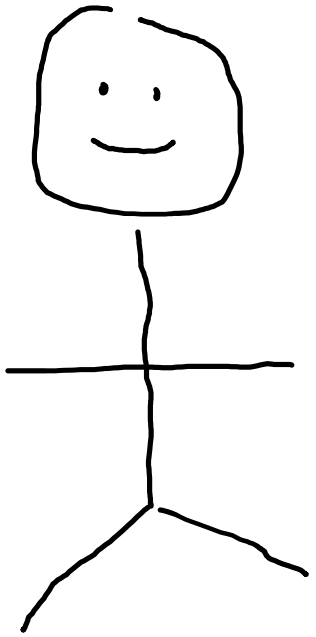
The paper killer



This is Reviewer 2.

Reviewer 2 had a few of their papers rejected in the past.

The paper killer



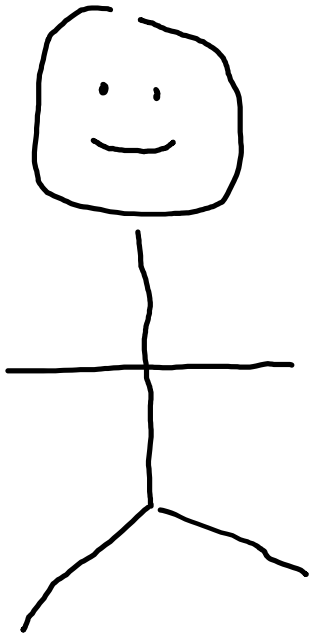
This is Reviewer 2.

Reviewer 2 had a few of their papers rejected in the past.

Now, Reviewer 2 is an angry and bitter scholar who exacts revenge on their peers through critical anonymous rejections.

It’s been three years since Reviewer 2 accepted a paper.

The paper killer



This is Reviewer 2.

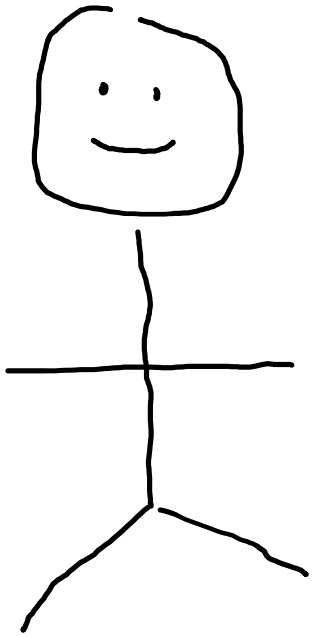
Reviewer 2 had a few of their papers rejected in the past.

Now, Reviewer 2 is an angry and bitter scholar who exacts revenge on their peers through critical anonymous rejections.

It’s been three years since Reviewer 2 accepted a paper.

Reviewer 2 loves to gatekeep.

The paper killer



This is Reviewer 2.

Reviewer 2 had a few of their papers rejected in the past.

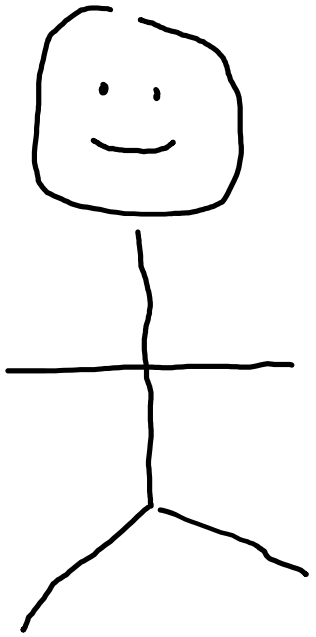
Now, Reviewer 2 is an angry and bitter scholar who exacts revenge on their peers through critical anonymous rejections.

It’s been three years since Reviewer 2 accepted a paper.

Reviewer 2 loves to gatekeep.

Reviewer 2 hates puppies and kittens.

The paper killer



This is Reviewer 2.

Reviewer 2 had a few of their papers rejected in the past.

Now, Reviewer 2 is an angry and bitter scholar who exacts revenge on their peers through critical anonymous rejections.

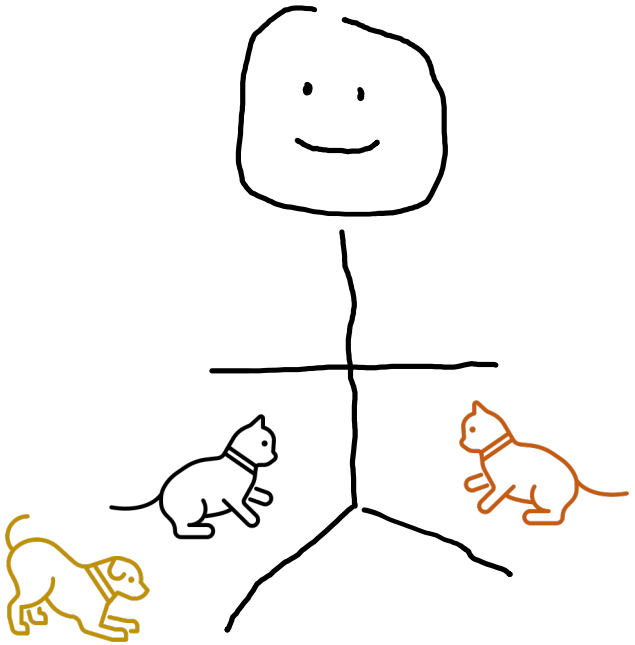
It’s been three years since Reviewer 2 accepted a paper.

Reviewer 2 loves to gatekeep.

Reviewer 2 hates puppies and kittens.

Don’t be like Reviewer 2.

The paper killer



- “Paper killers” are made, not born that way.
- If you start rejecting papers out of a sense of gatekeeping, you are only making matters worse.
- You want your field to grow and thrive!
- Be scholarly, objective, thorough, and polite.
- Be confident but know your limits. Be open to change your mind.

The AC Role

- Main task: “review the reviews” and produce a single recommendation.
- Write a meta-review that summarizes reviews, discussion, and activity during rebuttal. The meta-review explains the recommendation.
- Run after reviewers who are late (remember the first 3 rules!)
- Check quality of reviews and, if necessary, request more reviews.
- Lead discussion among reviewers and ensure they react to rebuttals.
- If necessary, read the paper yourself.

The AC Role

- A meta-review and decision is not just an average of review scores!

- The AC can disagree with the reviewers.
  - Should start a discussion with reviewers first.
  - Explain why you disagree in your meta-review.

- Do not abuse your power.

The AC Role

- A meta-review and decision is not just an average of review scores!

- The AC can disagree with the reviewers.
  - Should start a discussion with reviewers first.
  - Explain why you disagree in your meta-review.

- Do not abuse your power.

Up next: Write a meta-review in groups!