Inference in Open-Domain Question-Answering

Mark Steedman *with* Nick McKenna, Tianyi Li, Javad Hosseini, and others

7th July 2022

The Motivation

- Most information out there is in the form of text or spoken natural language (NL), rather than structured data.

- Even in databases, there is often a free text "Comment" field for entering information that doesn't fit in the standard datas-tructure (e.g. `www.amazon.com; www.ebi.ac.uk/swissprot`).

- We would like to get these NL data into databases or knowledge-graphs using Natural Language Processing (NLP).

- We would also like to interrogate and update these data-structures in Natural Language.

The Problem of Open Domain NLP

- The Fundamental Problem of Open Domain NLP is that there are too many ways of asking and answering the same question:
- You want to know Who played against Manchester United? The text says:
  - Arsenal beat Manchester United.
  - Manchester United's defeat by Arsenal.
  - Arsenal obliterated Manchester United.
  - etc.
- So if you just build a knowledge graph based on relations found in text (a "Semantic Network"), you won't be able to interrogate it effectively.
- Conversely, if you build a knowledge graph using graph query language relations such as SPARQL/RDF triples, querying it in NL will need to be robust to the same variation in NL input.

The Problem of Open Domain NLP

- The problem arises from the lack of a usable NL semantics supporting common-sense inference, such as that $\langle team\rangle defeat\langle team\rangle$ entails $\langle team\rangle play\ against\langle team\rangle$

- Two solutions to the problem of Inference:

  1. Supervised fine-tuning of a pretrained LM using an NLI dataset;
  2. Unsupervised induction of an entailment graph from text, by machine reading, using the Distributional Inclusion assumption.

The Encoder-Decoder Architecture in NLP

- The basic encoder-decoder architecture with attention or context units, developed for neural machine translation offers a general method for sequence-to-sequence alignment.

- The method can be used to learn Semantic Parsers as direct transducers from NL to meaning representations such as KGQL. (Well, sort of: Wang *et al.*, 2021a,b; Li *et al.*, 2021.)

- Or as a Cloze model for QA, by Masking during training.

- Can LLM also be used to do text-inference or text-entailment (Schmitt and Schütze, 2021b,a)?

LLMs as Latent Entailment Graphs

- Evaluating supervised text inference is an open problem: NLI Datasets are:
  - Riddled with artefacts that ML can learn as a proxy;
  - Dominated by paraphrase and selection-bias; and
  - Fail to include false inverses of directional entailments.
- Our own investigations so far fail to support claims that supervised fine-tuning of LLMs on NLI datasets can to learn directional entailment (Li, 2022).
- LLMs appear to memorize the training data, and to organize the memory according to similarity of association (Schütze, this meeting).
- They excel at tasks where the memorized text actually contains something similar to the question (Zettlemoyer; Hovy, t.m.).
- The training text probably does not include explicit sequences like "X defeated Y, therefore/so/: X played against Y". (Entailments "go without saying").

An Unsupervised Approach to NLI

- Build an unsupervised natural language entailment graph from large amounts of multiply-authored text by machine-reading different articles about the same events grounded in the same named-entity tuples
- Learn from such observations that if one entity of type team *beat* another entity of that type in one document it's likely that the same two entities will *play against* each other in another.
- We have done this in English and Chinese: (Hosseini *et al.*, 2018, 2019, 2021; Li *et al.*, 2022).
- The method scales: (20M sentences $\Rightarrow$ >200M sentences). The Problem here is Zipfian Sparsity.
- Can we Smooth Entailment Graphs with non-directional LMs without compromising the directional precision of EG?

Some Statistics on Unsupervised KG/EG

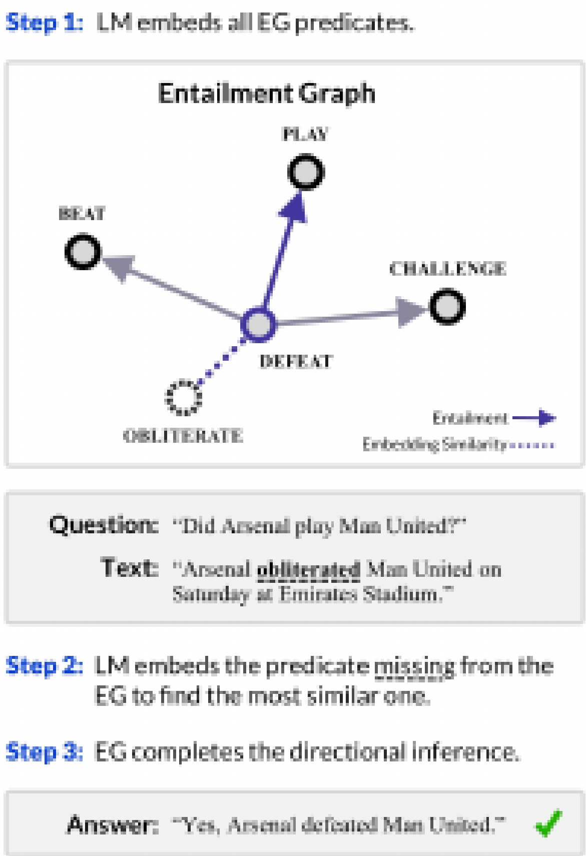
- Knowledge Graphs built on NewsSpike and NewsCrawl (Hosseini *et al.*, 2021)
  - Newspike is 0.5M multiply-sourced news articles over 2 months, 20M sentences; NewsCrawl is 5.4M articles sourced over 9 years, 256M sentences
  - NewsSpike KG has 326K typed relations, NewsCrawl, 1.05M
  - NewsSpike 29M relation triple tokens (before cutoff); NewsCrawl 729M.
  - NewsSpike 8.5M triple tokens (after cutoff); NewsCrawl 35m.
  - NewsSpike 3.9M triple types (after cutoff); NewsCrawl 13.4m
- We have built working typed global entailment graphs:
  - NewsSpike EG has 346 local typed subgraphs, NewsCrawl, 691
  - NewsSpike 23 subgraphs $>$1K nodes; NewsCrawl, 161
  - NewsSpike 7 subgraphs $>$10K nodes; NewsCrawl, 21

Smoothing Entailment Graphs with LMs

- If the P(remise)/Antecedent and/or H(ypothesis)/Consequent are missing from the EG through sparsity, EG loses.
- If we can find P' and/or H' that are in the graph, then:
  - if $P \models P'$ and/or $H' \models H$, and
  - $P' \models H'$ is in the graph, then by transitivity of entailment:
  - $P \models H$, else:
  - $P \not\models H$.
- The idea (McKenna *et al.*, 2022): Iff P and/or H are not in the graph, use Large LMs to find P' and/or H' that are,

This minimizes the non-directional influence of the LM.

Smoothing Entailment Graphs with LMs



Smoothing Entailment Graphs with LMs

- For P and/or H that is missing in the EG find the K nearest neighbour relations P' and/or H' that are in the EG, using contextualized embedding vectors.

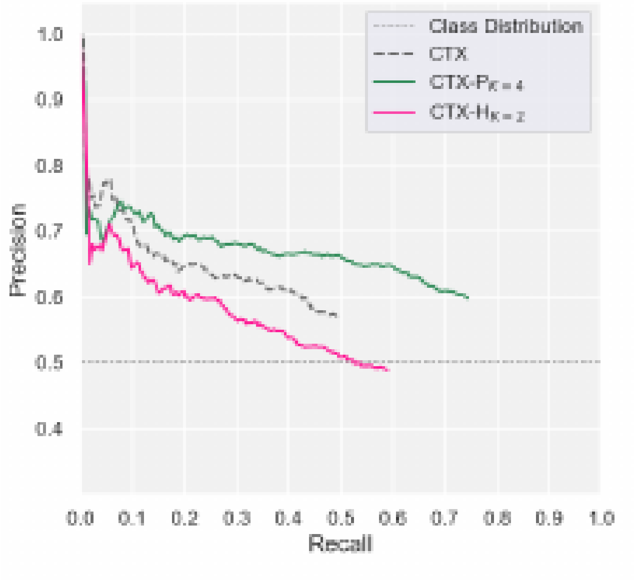
- Then try to establish $P/P' \models H/H'$.

- If $P/P' \models H'/H$, assume $P \models H$

- Note that there is no guarantee for LM-KNN P' and/or H' that $P \models P'$ and/or $H' \models H$.

Smoothing Entailment Graphs with LMs

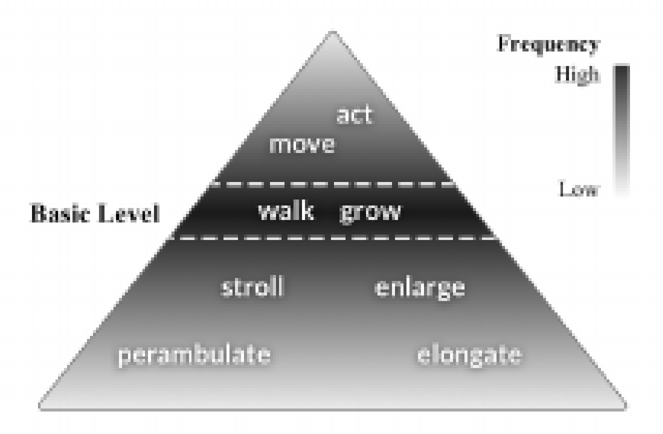
- Nevertheless, LM smoothing works for smoothing P, the antecedent:



- However, LM smoothing is deleterious for H, the consequent.
- Why is LM smoothing asymmetrical for P and H?

Why does LM smoothing work at all?

- There is a decrease in frequency with distance on either side of the basic level of "natural kinds" for terms on the hypernym-hyponym dimension of generality-specificity;
- There is also an increase in the number of terms with specificity:



This bias is well-known, as causing "translationese" in MT (Sennrich, t.m.).

Why is LM smoothing asymmetrical?

- This skewed distribution leads to a bias towards more frequent and more general predicates in generating nearest in-graph neighbours P' and/or H' for missing P and/or H using LMs.

- Since specifics are often hyponyms and related generics hypernyms, it is likely that $P \models P'$ obtained in this way.

- However, by the same reasoning, the nearest neighbours H' of H that are most likely to be in the EG are likely to be hypernyms of H, rather than hyponyms, so that it is less likely that $H' \models H$

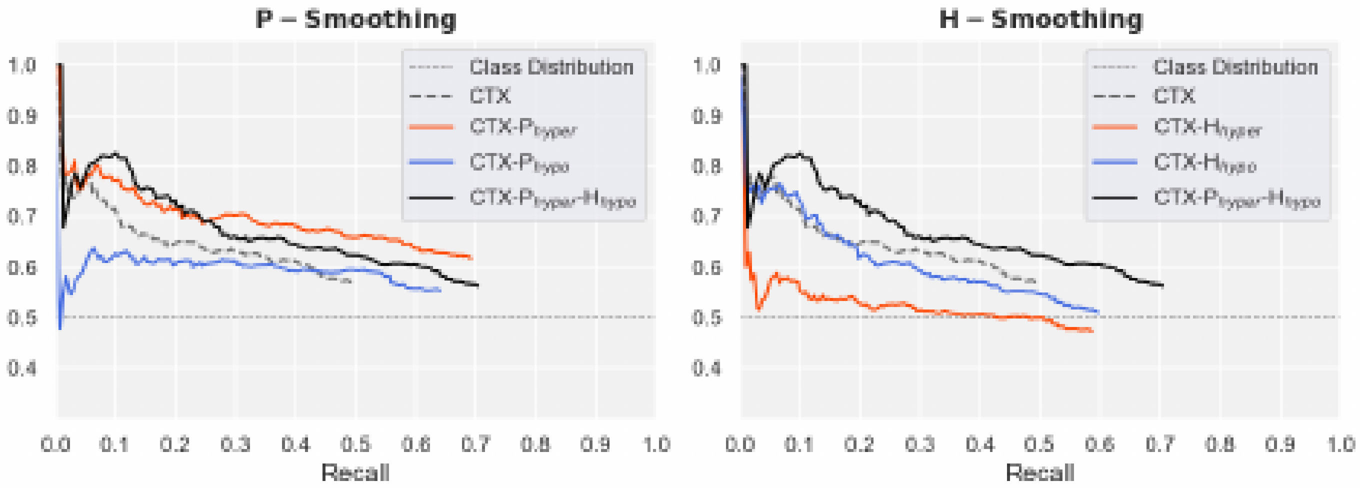
- Can we show that this is the explanation for the asymmetry?

Smoothing EGs with WordNet

- WordNet (and its mono- and multi-lingual generalization BabelNet) constitute largely distribution-neutral Hypo-Hypernym lattices.
- Use WordNet to optimally smooth P with guaranteed hypernyms and H with guaranteed hyponyms.
- McKenna *et al.* (2022) use WordNet *has hypernym* relation to identify hyper- and hypo-nyms P' and H' to smooth Hosseini *et al.* (2021) (CTX, our strongest EG).
- We test on the 2,930 question directional subset of our new ANT NLI dataset, constructed using WordNet antonyms as negative examples for comparison with supervised approaches (Bijl de Vroe *et al.*, 2022).
- The upward frequency bias still works asymmetrically for smoothing P and against smoothing H.

Smoothing EGs with WordNet

- Graphs respectively show effect of smoothing P and H with hypernyms and hyponyms against identical dashed baseline:
- They show the predicted opposite hyper/hypo effects for P and H, together with curves for predicted optimal joint P_{hyper} and H_{hypo} (identical black trace).



Smoothing EGs with WordNet

- There is the predicted hypernym facilitation in P_{hyper}.
- There is no significant hyponym facilitation for CTX in H_{hypo}.[1]
- Nevertheless, smoothing with $P_{hyper} + H_{hypo}$ significantly improves CTX over P_{hyper} alone (black trace.
- The additive effect seems to arise because, although present in EG, hyponym H' is even less frequent in text than absent H.
- It is therefore quite unlikely that EG-mining saw much evidence for $P \models H'$.
- However, P' is more frequent than P, so (given that both P' and H' are in the graph, it is a bit more likely that $P' \models H'$ is in the graph

[1] We do in fact see some H_{hypo} facilitation for our weaker EG Hosseini *et al.*, 2018.

Conclusion

- You can smooth EG with LMs on the premise/antecedent side.
- Can you also smooth the hypothesis/consequent side with WordNets?
- We are doing another experiment because our testset ANT is in part constructed using WordNet.[2]
- However, if so, the existence of BabelNet means we should be able to do so cross-linguistically, say for the Chinese Entailment graph of Li *et al.* (2021).
- Thanks be to BabelNet!

[2] Watch this space.

Thanks!

References

Hosseini, Javad, Chambers, Nathaniel, Reddy, Siva, Ricketts-Holt, Xavier, Cohen, Shay, Johnson, Mark, and Steedman, Mark, 2018. "Learning Typed Entailment Graphs with Global Soft Constraints." *Transactions of the Association for Computational Linguistics* 6:703–718.

Hosseini, Javad, Cohen, Shay, Johnson, Mark, and Steedman, Mark, 2019. "Duality of Link Prediction and Entailment Graph Induction." In *Proceedings of the 57th Annual Conference of the Association for Computational Linguistics (long papers)*. ACL, 4736–4746.

Hosseini, Javad, Cohen, Shay, Johnson, Mark, and Steedman, Mark, 2021. "Open-

Domain Contextual Link Prediction and its Complementarity with Entailment Graphs." In *Findings of the Association for Computational Lingustics: EMNLP*. 1137–1150.

Li, Tianyi, Li, Sujian, and Steedman, Mark, 2021. "Semi-Automatic Construction of Text-to-SQL Dataset for Domain Transfer." In *Proceedings of the 14th International Conference on Parsing Technology*. 38–49.

Li, Tianyi, Weber, Sabine, Hosseini, Javad, Guillou, Liane, and Steedman, Mark, 2022. "Cross-lingual Inference with a Chinese Entailment Graph." In *Findings of the Association for Computational Linguistics*. 1214–1233.

Li, Tianyi et al., 2022. "Language Models are Poor Learners of Directional Inference." In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. submitted.

McKenna, Nick *et al.*, 2022. "Smoothing Entailment Graphs with Language Models." *In Preparation* .

Schmitt, Martin and Schütze, Hinrich, 2021a. "Continuous Entailment Patterns for Lexical Inference in Context." In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 6952–6959.

Schmitt, Martin and Schütze, Hinrich, 2021b. "Language Models for Lexical Inference in Context." In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. 1267–1280.

Bijl de Vroe, Sander, Guillou, Liane, Johnson, Mark, and Steedman, Mark, 2022. "Temporality in General-Domain Entailment Graph Induction." In *submitted*.

Wang, Bailin, Lapata, Mirella, and Titov, Ivan, 2021a. "Learning from Executions for Semantic Parsing." In *NAACL*. 2747–2759.

Wang, Bailin, Lapata, Mirella, and Titov, Ivan, 2021b. "Meta-Learning for Domain Generalization in Semantic Parsing." In *NAACL*. 366–379.

Zhang, Congle and Weld, Daniel, 2013. "Harvesting Parallel News Streams to Generate Paraphrases of Event Relations." In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Seattle: ACL, 1776–1786.