Notes from ACL 2020

Zining Zhu

University of Toronto

@zhuzining

Table of Contents

Category: Generation, Discourse & Pragmatics

☐ Designing precise and robust dialogue response evaluators

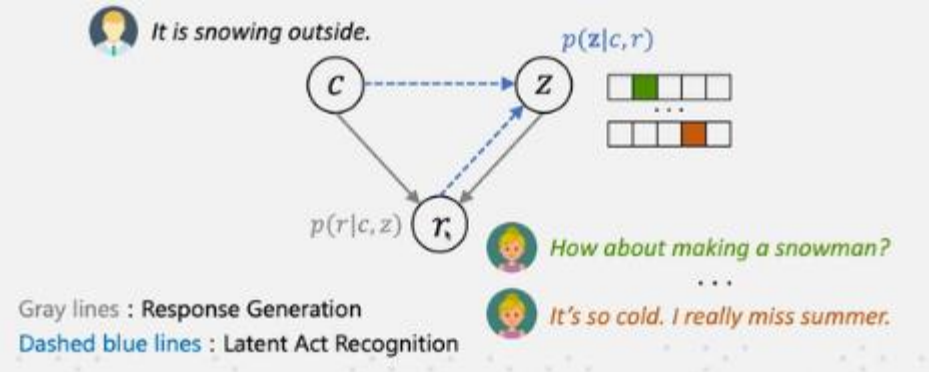
https://www.aclweb.org/anthology/2020.acl-main.4/

- Propose a reference-free, semi-supervised (improve upon ADEM, training on small amount of annotated data), RoBERTa-based evaluator.

☐ PLATO: pre-trained dialogue generation model with discrete latent variable

https://www.aclweb.org/anthology/2020.acl-main.9/

- Set up a discrete latent variable

- 

☐ A complete shift-reduce Chinese discourse parser with robust dynamic oracle https://www.aclweb.org/anthology/2020.acl-main.13/

- Four stages:
  - EDU segmentation
  - Tree construction
  - Relation recognition
  - Center labeling
  - Add two steps: binary tree conversion, and beam search
- Evaluation against previous works

☐ Fact-based text editing https://www.aclweb.org/anthology/2020.acl-main.17/

- Novel problem: fact-based text editing. Delete the unsupported facts and insert the missing facts.
- Method: LSTM Sequential tagger. Predict action from {Keep, Drop, Gen}.
- Create new datasets (WebEdit, RotoEdit) for this task.

☐ Reverse engineering configurations of neural text generation models https://www.aclweb.org/anthology/2020.acl-main.25/

- Motivation: generation models leave out artefacts in generated texts.
  - RQ1: Do some modeling choices leave behind more artefacts than others?
  - RQ2: Can we distinguish between text generation models based on the text generated alone?
  - RQ3: Which model configurations leave behind the most detectable artifacts?
- Experimental setup:
  - Multi-class classification problem.
  - CNN/DailyMail dataset as a starting point. Use Grover (Zellers et al) to generate.

- Results:
  - Artifacts are present. All models do much better than random chance.
  - Model architecture doesn't matter too much.
  - Sampling methods are easily distinguishable.
  - Even condition length can be predicted (but is hard)

☐ Unsupervised paraphrasing by simulated annealing https://www.aclweb.org/anthology/2020.acl-main.28/

- Task: paraphrase generation
- Previous methods:
  - Supervised learning with S2S: paraphrasing datasets are small. Domain specific datasets restrict generalization
  - Unsupervised learning with VAE: might have topic drifts.
  - Unsupervised, sampling-based method: e.g., CGMH. Sample from {Insertion, replacement, deletion}. Probabilistic sampling procedures are less "controllable".
- Proposed approach: UPSA
  - Randomly choose an editing operation and a position; edit the sentence; accept or reject; decrease the temperature.
  - Objective function: semantic preservation, expression diversity, language fluency.
- Future work:
  - Improve the measurement of paraphrase
  - Encourage more syntactically different edits

☐ Opportunistic decoding with timely correction for simultaneous translation https://www.aclweb.org/anthology/2020.acl-main.42/

- Simultaneous translation task
- Method:
  - Decode fixed number of extra words ("revision window") at each step to reduce latency
  - These extra words can be corrected in the future when more source words are revealed.
  - Revision-aware average lagging (RAL)
  - Average Lagging is not sensitive to the revisions of committed words. RAL only starts to calculate the latency for the word once it agrees with the final results, and doesn't change anymore.
- Results

  - Substantial reduction in latency
  - Up to +3.1 increase in BLEU, with revision rate under 8%.

☐ E2E neural pipeline for goal-oriented dialogue systems using GPT-2

https://www.aclweb.org/anthology/2020.acl-main.54.pdf

- Goal: DSTC8: multi-domain task-completion task, developing and e2e multi-domain dialogue system.
- Input representation: delexicalization (replace dialogue-dependent words e.g., phone, name, postcodes) with generating slot tokens as [DOMAIN_SLOTNAME]
- Evaluation:
  - Automatic evaluation: success rate
  - Human evaluation: success rate, language understanding score, response appropriateness score.
  - Also: MultiWOZ benchmarks: dialogue state tracking and dialogue context-to-text generation
  - Interpretable results. Easy integration with external systems. Natural human-level interaction.

☐ Learning to tag OOV tokens by integrating contextual representation and background knowledge

https://www.aclweb.org/anthology/2020.acl-main.58.pdf

- Task: context-aware slot filling models
- Approach: main components.
  - Pretrained BERT encoder: get the context-aware representation
  - Knowledge integration layer: get the knowledge-aware representation. Use Wordnet as the first-level candidate set. Use hyponyms of synsets as the second-level candidate set. Apply multi-level graph attention.
  - BiLSTM matching layer: match the two types of representations
  - CRF layer: to model the relationship between tags
- Datasets: ATIS, Snips. Evaluate with F1.
- Baseline models: RNN, BERT, slot-filling models, ablation study models.

☐ USR: An unsupervised and reference free evaluation metric for dialog generation

https://www.aclweb.org/anthology/2020.acl-main.64.pdf

- Why is evaluating dialogue hard?
  - One-to-many nature of dialog: can have many valid responses.

    - Dialogue quality is multi-faceted. A response is not just good or bad. There should be multiple qualities (relevance, interesting-ness, fluency, etc.)
    - There's not "one size fits all" definition of good dialog. E.g., chit-chat: interesting > relevant.
- Motivation
    - Evaluation is important & hard
    - The only perfect evaluation mechanism is human evaluation, but we still need an automatic metric (e.g., during development).
- Overview
    - Trained response generation models on Topical-Chat and PersonaChat
    - Evaluate with the USR metric.
- How to evaluate evaluation metrics?
    - Generate responses (different models + human generated + original in the dataset). Score them using human labels on different aspects.
    - 6 qualities: understandable (0-1), natural, maintains context, interesting (1-3), uses knowledge (0-1), overall quality (1-5).
- Propose solution
    - Model-based metric. Multiple models for different qualities. Self-supervised training to approximate quality.
    - Fine-tune RoBERTa on self-supervised tasks approximating qualities we want to capture:
        - Understandable / natural -> MLM
        - Interesting / engaging -> dialog retrieval
        - Uses knowledge -> fact to response selection
    - Put the metrics together, and regress them, to get the USR metric.

☐ Improved NLG via loss truncation https://www.aclweb.org/anthology/2020.acl-main.66/

- Neural NLG can generate unfaithful texts.
- Related work:
    - Better decoders (top-k, top-p)
    - Better models: copying, neural checklists
    - Unfaithfulness penalties: unlikelihood training, GANs
    - This work: focus on generic losses

- Observation: large datasets contain unfaithful examples. Hypothesis: unfaithful data leads to unfaithful models.
- Observation: Noisy data dominates high loss. Many high loss examples may cause unwanted distortions.
- Observation: Log loss is not robust to noise: toy example of estimating Gaussian mixture.
- Observation: distinguishability (TV) can work under noise, gives guarantees on generation performance, but is not optimizable. (Donoho '98)
- Want to learn under noise, but differentiable. Propose loss truncation.
    - Key idea: drop high log loss examples.
- Truncated loss upper bounds distinguishability (total variation) -> low truncated loss implies low distinguishability!
- Another technique: Rejection sampling for high-quality sequences.
- Evaluation:
    - Task: Gigaword summarization
    - Baselines: top-k, top-p, full sampling, beam search, GAN
- Loss truncation outperforms on distinguishability
    - Metric: HUSE-Q (quality) and HUSE-D (diversity)
    - (Hashimoto et al., 2019)
- Loss truncation outperforms on factuality.
    - Factuality of generated titles (Novikova et al., 2017)

☐ Rigid formats controlled text generation https://www.aclweb.org/anthology/2020.acl-main.68/

- Rigid formats generation include: lyrics, 宋词 SongCi, sonnet, etc.
    - Constraints in formats include num words, num sentences, rhyming rules, etc.
    - Example: old music staff, new lyrics (constraints: format, rhyming, sentence integrity)
- Methods
    - Define a subtask: polishing.
    - Framework: SongNet.
    - Embed the format and rhyme symbols, and concatenate with other embeddings.
    - Beam-search algorithm and truncated top-k sampling.
- Evaluation
    - Sentence integrity: PPL of the constraints
    - Metrics: PPL, diversity, format, rhyme, integrity

    - Human evaluation: relevance, fluency, style

☐ Syn-QG: Syntactic and shallow semantic rules for question generation

https://www.aclweb.org/anthology/2020.acl-main.69/

- Rule-based vs neural-based question generation
- Syn-QG is a rule-based framework generating questions by identify potential short answers in
    - The nodes of crucial dependency relations
    - The modifying arguments of each predicate in the form of semantic roles
    - Named entities and other generic entities
    - The states of VerbNet's thematic roles in the form of semantic predicates
    - PropBank roleset natural language descriptions
- +custom rules + handling negations using implicatives, + back translation.

☐ Pronunciation-attentive contextualized pun recognition

https://www.aclweb.org/anthology/2020.acl-main.75/

- What is a pun? E.g., "I'd tell you a chemistry joke but I know I wouldn't get a reaction."
    - Both local and global contexts are consistent with the pun word "reaction"
    - This kinds of puns are homographic puns.
    - Another type: heterographic puns. "The boating store had its best sail (sale) ever." Similar pronunciation connects two words.
- Task: pun detection and location
    - Previous work: word sense disambiguation methods or using external knowledge base. They can't tackle heterographic puns. Leveraging static word embedding techniques can't model puns well.
- Proposed method: pronunciation-attentive Contextualized Pun Recognition.
    - Encode both phonemes and the word contexts.
    - Classify at each location whether the word is at the key word of the pun (localization task), or whether the sentence is a pun (detection task).
    - Datasets: SemEval 2017 shared task 7, and Pun of the Day (PTD). Two largest publicly available pun datasets.
    - Attentions visualization.
    - Release implementations and pre-trained phoneme embeddings to github.

☐ Learning to identify follow-up questions in conversational QA

https://www.aclweb.org/anthology/2020.acl-main.90/

- Determine whether a question is part of an ongoing conversation.
- LIF Dataset:
  - Derived from the QuAC dataset (Choi et al., 2018)
  - Valid instances from the "should ask" follow-up questions.
- Methods
  - Creating the invalid instances
  - Challenges of the dataset
  - Propose model (Three-way attentive pooling network), outperform several strong baseline models.

☐ Keyphrase generation for scientific document retrieval

https://www.aclweb.org/anthology/2020.acl-main.105/

- Task: retrieving relevant papers, by generating keyphrases.
- Method: two seq2seq-based models
- Extrinsic evaluation framework:
  - Contrasting the retrieval effectiveness
  - NTCIR-2 collection.
- Discussion
  - First study of the usefulness of keyphrase generation for scientific document retrieval.
  - Keyphrases produced by SOTA models are consistently helpful for document retrieval
  - New extrinsic evaluation framework for keyphrase generation.

☐ Generating counter narratives against online hate speech

https://www.aclweb.org/anthology/2020.acl-main.110.pdf

- Standard approach: content moderation (e.g., deletion or suspension)
- Novel approach: direct invention in the discussion with textual responses (counter narratives)
- Proposed solution: an "author (GPT-2) - reviewer (human or machine)" architecture.
- Datasets:
  - Crawling
  - CROWD (Crowdsourcing)
  - Nichesourcing
- Metrics:
  - Novelty
  - Diversity

    - Time (operator experiments; post-editing and writing)

☐ The TechQA Dataset https://www.aclweb.org/anthology/2020.acl-main.117/

- Motivation: lack of industry dataset. Existing datasets are not in specific domains.
- Data source: IBM external forums.
- Questions asked by IT personnel (e.g., system administrators). Answers are by SMEs.
- The answers contain links from technotes.
- Dataset released:
    - Training: 450 answerable questions + non non-answerable as per 2 annotators
    - Dev: 160 answerable, 150 non-answerable, as per 2 annotators
    - Test: ~500 questions, with roughly the same answerable / non-answerable statistics as the dev set.
    - Technotes: 800k technotes, e.g., for training language models.
- Each training / dev sample consists of
    - Question title, text
    - 50 technotes
    - The id of the technote with the answer, if the question is answerable
    - The start-end character offsets, if the question is answerable.
- Metrics:
    - Main metric: F1
    - Ancillary metrics: best_F1, HA_F1@1, HA_F1@5

☐ You impress me: dialogue generation via mutual persona perception https://www.aclweb.org/anthology/2020.acl-main.131/

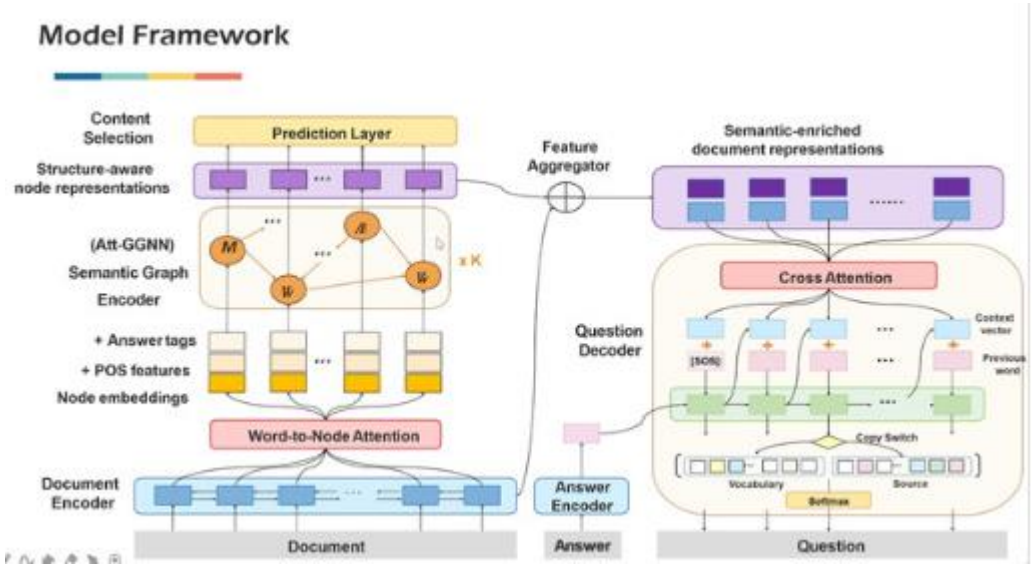
- Personalized dialogue generation problem: two interlocutors meet for the first time, and have a conversation to get to know each other.
    - This requires modeling their (configurable and persistent) personalities
    - The understanding between interlocutors (i.e., i=persona perception) is essential for a high-quality conversation
- Methods
    - Transmitter: GPT-2. Generate the response given dialogue history
    - Receiver: Responsible for persona perception. Learn the proximity between sentences and persona by a contrastive learning paradigm.

- Transmitter is fine-tuned during a self-play procedure (Lewis et al., 2017) with persona perception as a kind of reward.

- Experiments:
  - Dataset: Persona-Chat (Original & Revised)
  - Visualization of the relevance scores between a sampled dialogue and its corresponding revised persona

☐ Semantic graphs for generating deep questions https://www.aclweb.org/anthology/2020.acl-main.135/

- Background: question generation.
- Motivation: use semantic graphs to generate deep questions.
  - S2S is not suitable for deep questions generation (DQG).
  - S2S directly learns the mapping from unstructured document to question.
- Propose a model to incorporate structured semantic graph to assist question generation



- Experiments:
  - Dataset: HotpotQA
  - Baselines:
    - S2S+Attn, NQG++, ASs2s, S2sa-at-mp-gsa, CGC-QG
  - Evaluation metrics: BLEU1-4, METEOR, ROGUE-L

☐ Parallel sentence mining by constrained decoding https://www.aclweb.org/anthology/2020.acl-main.152/

- Task: parallel sentence mining. Sentence alignment using NMT from large corpora.

- Related work:
  - Pairwise comparison of web document is intractable
  - Document alignment + sentence alignment
  - SOTA: Multilingual embeddings + modified cosine on neighbors
- NMT: Given a source, NMT can force-decode any target, and produce a translation score.
  - Force-decode all target against all sentences -> Can't afford O(n^2) search
  - Trie-constrained beam search
    - Build a trie (prefix tree) over all target sentences
    - Constrain NMT decoding to follow the trie (*) This reduces the search space.
    - The most "parallel" target found for each source.
    - Then remove bad pairs by e.g., thresholding on cross entropy scores.
- Experiment: building and using comparable corpora (BUCC) shared task (Zweigenbaum et al., 2018)
- Advantages: End-to-end. High precision. Convert NMT to a mining tool easily.

☐ Learning to update natural language comments based on code changes https://www.aclweb.org/anthology/2020.acl-main.168/

- Task: generate natural language comments based on changes in programming codes.
- Learn to edit old comments into new ones
- Model:
  - Learn representation for C_old
  - Learn representation for code changes
  - Predict NL (natural languages) edits
  - Apply NL edits to existing comment
  - Rerank + produce updated comments
- Dataset:
  - Mine simultaneous updates to {comment, method} pairs from consecutive commits of open-source Java projects on GitHub
- Evaluation metrics: xMatch, Generation {METEOR, BLEU-4}, Editing {SARI, GLEU} + human

☐ Politeness transfer: a tag and generate approach https://www.aclweb.org/anthology/2020.acl-main.169/

- Task: convert non-polite sentences to polite ones while preserving the meaning
- Challenges:

    - Politeness is culturally diverse
    - Politeness is subtle. E.g., indirect, greet, 1st person plural (e.g., "Let's go and remove it" better than "Go and remove it").
    - Data paucity.
- Focus: politeness in north American English speakers + a formal setting. Focus on converting request / action-directives to polite requests.
- Proposed methodology:
    - Transfer desiderata: Successful transfer into target style. Retain content words (non-attribute markers)
    - Tag and generate pipeline
- Create dataset:
    - Step 1: remove attribute markers
    - Step 2: generate tags
    - Step 3: Use attribute markers of the style target to generate artificial parallel data.
    - Dataset at https://github.com/tag-and-generate

☐ Learning an unreferenced metric for online dialogue evaluation
https://www.aclweb.org/anthology/2020.acl-main.220.pdf

- Background: dialogue evaluation examples:
    - ADEM (Referenced), RUBER
- Propose MaUde: un-referenced, single-turn.
- Issues with a referenced metric: lack of generalization. Trained models can't be reused. Also, reference issue -- true labels are not available when using.
- "un-referenced"
- Hypothesis: dialog consists of a particular temporal structure. (More in dialog, but less clustering in chit-chat)
- Can leverage the structure to learn a dialog transition function (DTF)
- Noise Contrastive Estimation Training. (Semantic / Syntactic NCE).

☐ Simple and effective retrieve-edit-rerank text generation
https://www.aclweb.org/anthology/2020.acl-main.228/

- Retrieve-edit module
    - Generate text using retrieved examples from training set
- Post-generation ranking

    - Retrieve N example, generate a candidate output with each
    - Then rank these candidates
- Experiments:
    - 2 MT tasks
    - Gigaword Summarization task

☐ Towards holistic and automatic evaluation of open-domain dialogue generation https://www.aclweb.org/anthology/2020.acl-main.333

- Human evaluation of open-domain dialogue is a natural choice, but is costly and inefficient.
- Automatic metrics:
    - Heuristic-based: BLEU, METEOR, ROUGE -> not suitable for high conditional entropy task evaluation
    - Distributed-representation-based: RUBER, BERT+RUBER
- Holistic evaluation
    - Context coherence and response fluency
    - Response diversity
    - Logical self-consistency
- Proposed metrics: (see paper for equations)
    - Context coherence score.
    - Response fluency score.
    - Response diversity score (n gram entropy for the set u_{t+1}^(k))
    - Logical self-consistency: score of NLI classifier taking two consecutive responses from the same speaker
- Evaluate these metrics by the correlation between human ratings (response fluency score) and between the scores computed on augmented dataset and baseline dataset.

☐ Learning implicit text generation via feature matching https://www.aclweb.org/anthology/2020.acl-main.354/

- Most GANs use REINFORCE or Gumbel softmax and pretrain the generators
- This work builds on generative feature matching networks (GFMN). This avoids instabilities of adversarial learning
- SeqGFMN: GFMN for sequential discrete data.
    - SeqGFMN uses a feature extractor (FEs) instead of a discriminator
    - Feature-matching loss + reconstruction loss + classification loss + back-translation loss

- Results:
  - Dataset: MSCOCO captions + EMNLP2017 WMT news
  - Evaluation: BLEU, Self-BLEU, Fréchet InferSent Distance
  - There is no need of RL or Gumbel Softmax. The proposed loss is effective for unsupervised text generation, e.g., style transfer.

☐ Dscorer: a fast evaluation metric for discourse representation structure parsing

https://www.aclweb.org/anthology/2020.acl-main.416/

- Background: the Discourse Representation Structure
- Related: Counter (Rik et al., 2018) conversion then score.
  - Problem 1: searching optimal variable mapping is NP-complete.
  - Problem 2: only considers local clauses without taking larger window sizes into account.
- Propose Dscorer.
  - First induct a graph
  - Then perform n-gram extraction
  - Then score based on the extracted n-gram
- Evaluation
  - Data: Parallel Meaning Bank, Groningen Meaning Bank - sent / doc.
  - Metrics: seconds.

☐ Tangled up in BLEU: re-evaluating the evaluation of automatic MT evaluation metrics

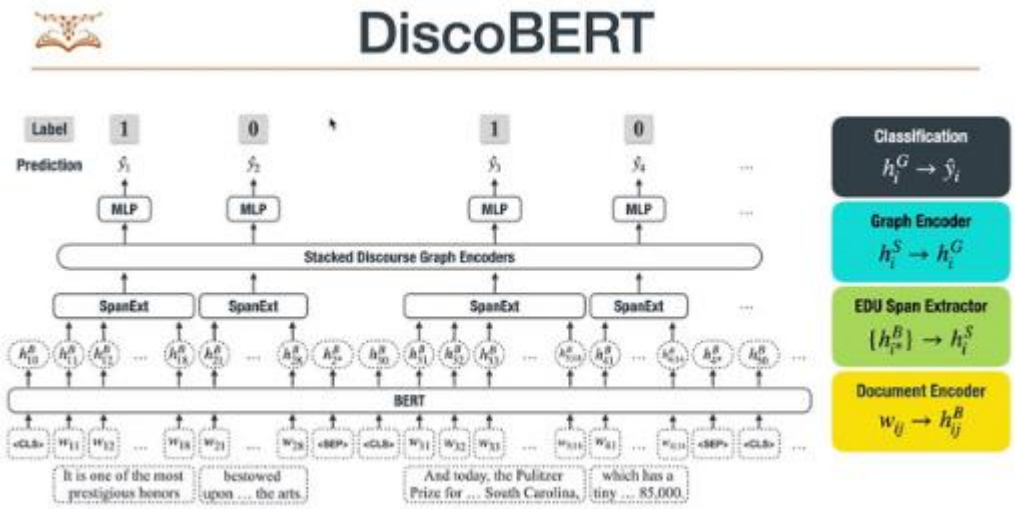
https://www.aclweb.org/anthology/2020.acl-main.448/

- Best paper runner-up
- Pearson correlation e.g., 0.9 with human evaluation doesn't tell us a lot of information!
  - Problem 1: outliers
  - Problem 2: Heteroskedasticity
- Q1: How much do outliers influence the correlation of metrics with human scores?
- Q2: How much is metric reliability influenced by MT system quality?
- Q3: How reliable are metrics when comparing two systems?
- Recommendations:
  - When evaluating metrics, should also report results without outliers.
  - During MT system development, stop using BLEU, and instead use chrF, YiSi-1 or ESIM.
  - Always support your final conclusions with human evaluation.
  - Always visualize your data.

☐ Discourse-aware neural extractive text summarization

https://www.aclweb.org/anthology/2020.acl-main.451/

- Challenge: discourse understanding. Build structural connections to capture local or global context
- Propose model: DiscoBERT
  - EDUs from RST as the minimal selection units
  - Discourse relations as graphs



  - Document encoder: basically Liu& Lapata 2019. <CLS>sent1<SEP><CLS>sent2<SEP>. Init with bert-base-uncased, fine-tuning on summarization datasets.
  - EDU representation extractor: self-attentive span extractor. Random init, fine-tune on summarization datasets.
  - Stacked discourse graph encoder: basically GCN (Kipf & Welling, 2016). Node is EDU embedding; edge: pre-defined discourse relations and coref mentions. Output node: new EDU embedding after graph propagation. Random init; fine-tune on summarization datasets
  - Decision: binary classification. MLP+sigmoid. Rank all of the EDUs. Pick up the top k EDUs and their dependencies.
- Building the discourse graph.
  - First acquire RST discourse tree graph (constituency). Then construct a converted RST discourse graph (dependency)
  - Coreference mention graph.
- Experiment:

    - Datasets: CNN/DM, NYT
    - Metrics: ROUGE against benchmarks, assessment of grammaticality (Grammarly + human evaluation, manual inspection of error analysis)
    - DiscoBERT outperform BERT and other baselines.

☐ Discourse as a function of event: profiling discourse structure in news articles around the main event https://www.aclweb.org/anthology/2020.acl-main.478/

- Background: discourse profiling.
- Introduce a content structures in Newspaper articles.
- Grounded on the news content schemata proposed by Van Dijk
    - M1, M2, C1, C2, D1 - D4
- The NewsDiscourse Corpus
    - 802 news articles covering 4 domains.

☐ Implicit discourse relation classification: we need to talk about evaluation https://www.aclweb.org/anthology/2020.acl-main.480/

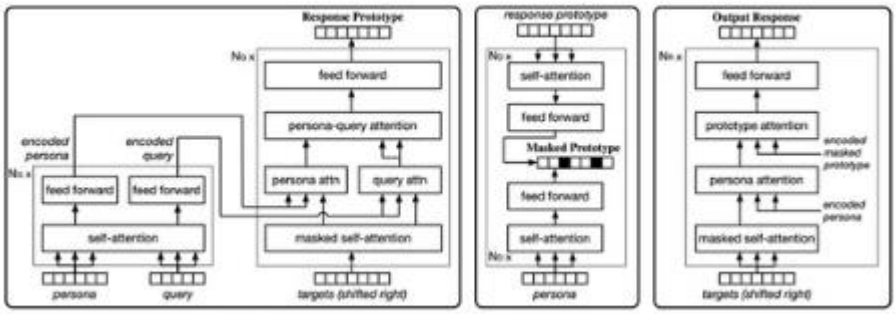
- Implicit discourse relations: those not signaled by overt discourse markers.
- PDTB 2 vs PDTB 3
- PDTB 2 evaluation problems:
    - Inconsistences in choice of split.
    - Choice of labels. Most evaluate on L2 (layer 2) labels.
    - Variation across runs.
- Preprocesssing codes available for PDTB 2 and 3.
- More about cross-validation.

☐ Generate, delete, and rewrite: a three-stage framework for improving persona consistency of dialogue generation https://www.aclweb.org/anthology/2020.acl-main.516/

- Background: persona-based dialogues
- Task definition: Given some persona texts and an input message, generate a model corresponding to the persona of the person.
- Challenge: the change of one persona-related word may not significantly affect the overall loss, but it could turn a good response into a totally inconsistent one.
- Ideas:
    - In the traditional single decoding stage, the models focus on generating fluent responses.

- If we have a fluent template, we can focus on more the persona-related words in the response.
- Converting into two mappings.

Overall framework

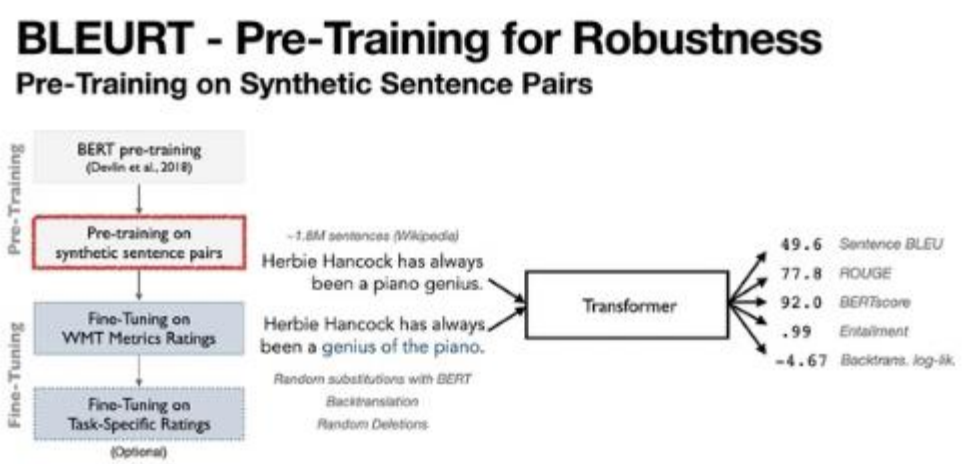


(1) Generate (2) Delete (3) Rewrite

- Experiments:
  - Dataset: PersonaChat Dataset (Zhang et al., ACL '18), and Dialogue NLI Dataset (Welleck et al., ACL '19)
  - Compared methods: S2SA, Per-S2SA, Generative Profile Memory Network, DeepCopy, Per-CVAE, Transformer.
  - Evaluation metrics:
    - Automatic: perplexity, Distinct1/2, Ent_diin, Ent_bert
    - Human: fluency, relevance, informativeness, consistency, pair-wise comparison (5 professional annotators from a third-party company)
  - Ablation study

☐ BLEURT: learning robust metrics for text generation https://www.aclweb.org/anthology/2020.acl-main.704/

- Background: NLG evaluation
- Current metrics (based on ML):
  - Hybrid metrics: BERTscore, YiSi, Sentence Mover's similarity (more robust)
  - E2E models: BEER, RUSE, ESIM (more flexible)
  - Can we be both robust and flexible?
- BLEURT: pre-training for robustness

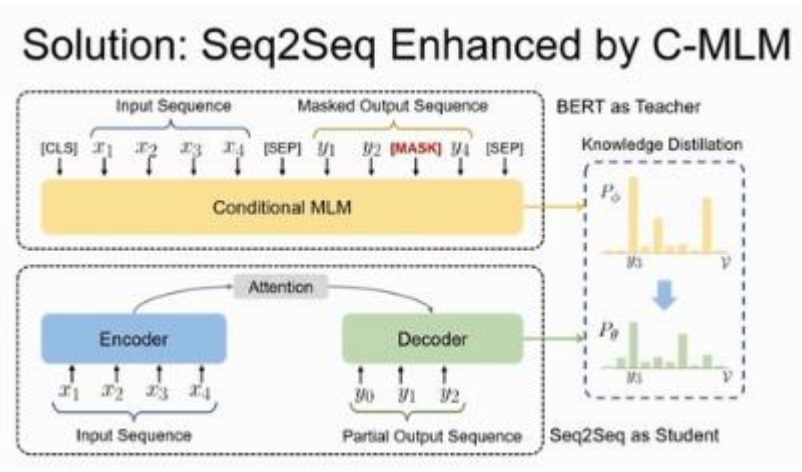


- How accurate is BLEURT?
  - Agreement with human ratings on WMT Metrics Shared Task '19 (segment level)
  - Impact of pre-training and model size
  - How robust is BLEURT? Extrapolation experiments on WMT Metrics '17
  - Whether BLEURT Transfer to other tasks: agreement with human ratings on WebNLG. Consider the semantics, fluency, and grammar.

☐ Distilling knowledge learned in BERT for text generation

https://www.aclweb.org/anthology/2020.acl-main.705/

- BERT is dominating NLU; can we use it to improve text generation?
- Proposal: conditional MLM for S2S generation
- How to make it sequential? Use knowledge distillation. BERT as a teacher.



☐ Stimulating creativity with Funlines: a case study of humor generation in headlines

https://funlines.co/humor/

- Funline tool: a competitive game to collect data for a human creativity task.
- Make AMT tasks more funny.

Category: Interpretable AI

☐ Probing linguistic systematicity https://www.aclweb.org/anthology/2020.acl-main.177/

- Background: linguistic systematicity: individual words will mean the same thing when put in new contexts.
- Natural languages are not always systematic.
  - E.g., Idioms. "She's the cat's pyjamas". Compound nouns, sarcasms, presuppositions, implicatures.
  - Less systematic words tend to be open class (nouns, verbs, etc,).
  - Systematicity bias: stronger for closed-class words.
- Propose a test suite:
  - E.g., The "jabberwocks" test.
  - Suite of systematicity tests include jabberwocky items that systematic learners can reasonably be expected to correctly classify.
  - Based on the NLI task.
- Identical Open-class words test
  - Pairs of sentences with the same subject and verb.
  - E.g., "All A run. Some A don't run."
- Consistency Test
  - Correctly classified test items, when transposed should also be correctly classified.
- Known word perturbation test
  - Correctly classified pairs with novel words are edited with a familiar word.
  - Should also be correctly classified.
- Block structure: expose learner to closed-class items with a variety of open-class items.

☐ "None of the above": measure uncertainty in dialog response retrieval https://www.aclweb.org/anthology/2020.acl-main.182/

- Retrieval-based dialog agents. Model input: context + proposed response r. Output: score.
- How to tell if the "correct response" is not in the candidate set?

- Goal: classify the absence of ground truth while maintaining models' regular response retrieval performance.
- Method: Learn the relationship among the candidates as a set instead of looking at pointwise matching.
- Dataset: Ubuntu Dialog Corpus
- Baseline models: Dual LSTM Encoder (one for context c, one for response r.)
- Evaluation: NOTA metrics.

☐ "Can you put it all together": evaluating conversational agents' ability to blend skills https://www.aclweb.org/anthology/2020.acl-main.183.pdf

- Goal: train a model multi-task on multiple single-purpose conversation datasets.
- Collect new dataset blending all previous purposes
  - BlendedSkillTalk: ParlAI
- Show the proposed model does well on the collected tasks.

☐ ExpBERT: representation engineering with natural language explanations https://www.aclweb.org/anthology/2020.acl-main.190/

- Background: language as a tool for communicating inductive biases
- Strategy for using language explanations:
  - Interpret language explanations in the context of the input text to produce features
  - Use these features along with the input representation to train classifiers.
- Experiments:
  - Spouse dataset (Hancock et al., 2018) Given an input paragraph + 2 names, predict if they are married.
  - TACRED (Zhang et al., 2018) Given an input paragraph + 2 entities, predict the relation between them.

☐ Stolen probability: a structural weakness of NLMs https://www.aclweb.org/anthology/2020.acl-main.198/

- The dot-product softmax is a structural weakness of the Transformer LMs
- Stolen Probability Effect -- words embedded interior to the convex hull are probability impoverished by words on the convex hull.
  - The convex hull are those to the embeddings of a vocabulary.
  - Proof: See Appendix A of paper.
- Empirical analysis: use a detection algorithm.

    - Identify whether the points are inside the convex hull. Previous algs like Quickhull became prohibitively slow at ~10 dimensions, but word embeddings have hundreds of dimensions.
    - Propose an alg based on the intuition: if a vertex p is not interior (i.e., a vertex) to the vocabulary, then there exists a vector h such that: <h, x_i-p> < 0 for all vertices x_i excluding p.
- Experiment shows clear separation between interior vs. Non-interior words.

☐ Contextual embeddings: when are they worth it? https://www.aclweb.org/anthology/2020.acl-main.236/

- Motivation: contextualized embeddings work incredibly well, but are extremely expensive.
- When are contextual embeddings worth their cost?
- Impacts of training data volume
    - Performance vs. Task training data volume on 15 tasks.
    - Cost tradeoff: Random / GloVe has lower computational cost, but requires more training data.
- Impact of linguistic properties
    - Context helps for complex, ambiguous, and unseen language.
    - Complex structure: how interdependent are different words in a sentence?
    - Ambiguous word sense: are words likely to appear with multiple labels during training?
    - Prevalence of unseen words: how likely is encountering a word never seen during training?
    - GLUE Diagnostic task.
    - Contextual embeddings give larger gains on difficult language.

☐ Mitigating gender bias amplification in distribution by posterior regularization https://www.aclweb.org/anthology/2020.acl-main.264/

- Predictions contain gender bias. "bias amplification"
    - (Zhao et al., '17) bias in top predictions.
- "Posterior regularization" for mitigation:
    - First define the constraints and the feasible set Q. The posterior bias should be smilar to the bias in the training set.
    - Then minimize the KL divergence between Q and p_\theta
    - Do MAP inference based on the regularized distribution

- PR removes almost all of the bias amplification effects.
- The reason for bias amplification is left as an open question.

☐ On exposure bias, hallucination and domain shift in Neural Machine Translation
https://www.aclweb.org/anthology/2020.acl-main.326/

- What is hallucination?
  - Fluent translations but completely unrelated to input.
  - Hallucination is especially common under domain shift.
- MLE training vs inference history mismatch: exposure bias.
- Conjecture: exposure bias contribute to hallucination. Try to verify this, and try to address the exposure bias problem.
- Method: Minimum Risk Trianing (MRT): See Section 2 in paper for details.
- Experiments
  - Datasets: OPUS (German to English), German to Romansh (Allegra / Convivenza)
  - Metrics: Annotators identify when hallucination occurs.
  - Uncertainty analysis. Aim: to acquire better understanding, visualize how MRT find-tuning affects model bias towards hallucinations. Plot the per-token-probability of the model to a random sampled sentence (distractor) and the ground-truth reference.

☐ BERTRAM: Improved word embeddings have big impact on contextualized model performance
https://www.aclweb.org/anthology/2020.acl-main.368/

- Motivation: BERT fails with rare words. (e.g., "a kumquat is a ___", "an arghul is an ___").
- Propose solution: BERTRAM
  - Want to use an embedding connecting the e.g., (almost)bigrams of "unicycle" with "un ##ic ##y ##cle"
  - Single Context case: aggregate the embeddings of subwords of a rare concept. Prepend to the front of sentence, followed by a column, then plug into the model.
  - Multiple context: use attentive mimicking (Pinter et al., 2017) as a training objective. BERTRAM = BERT for Attentive Mimicking
- Dataset rarification: modify existing datasets so that rare words are guaranteed to be important
  - Train a baseline model M for the task. Select only examples which M classifies correctly. Find words for which M changes its prediction when they are replaced with [MASK] tokens. Replace the identified words with rare synonyms.

- Evaluation
    - MNLI and AG's News, DBPedia (all rarified). BERTRAM outperform BERT.

☐ Perturbed masking: parameter-free probing for analyzing and interpreting BERT
https://www.aclweb.org/anthology/2020.acl-main.383/

- The probe confounder problem: shall we credit the good accuracy to the representation or the probe?
    - Previous approaches to fix it: control tasks (Hewitt and Liang, 2019) and MDL (Voita et al., 2020)
- Propose approach: unsupervised probing with perturbed masking.
    - Perturb the inputs (replacing with [MASK]). Compute impact matrix from the LM embedding.
    - Then try to induce a syntactic parse from the impact matrix, using e.g., graph-based dependency parsing (after all, a task-specific one) algorithm.
- Applications: dependency / discourse probe, or Chinese word segmentation tasks.

☐ Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness?
https://www.aclweb.org/anthology/2020.acl-main.386/

- Core questions of interpretation:
    - What counts as an explanation of a model's decision?
    - How to evaluate the quality of an explanation?
- This work tries to make sense of where things stand w.r.t faithful explanations
- What makes an interpretation useful?
    - Plausability, readability, faithfulness
    - Focus on faithfulness in this paper
- Guidelines: pitfalls to avoid when evaluating models for faithfulness
    - Many evaluations conflate evaluating faithfulness and evaluating plausibility.
    - Be explicit .
    - Model decision process != human decision process. Human cannot judge if an interpretation is faithful.
    - Don't trust untested claims of "inherent interpretability" of models.
- Survey: three assumptions underlying current literature on faithful explanations
    - Two models make the same predictions <=> they use the same reasoning process.
    - On similar inputs / decisions, interpretations should be similar

    - The linearity assumption: Heat-maps can be faithful under certain circumstances.
- Opinion: is faithful interpretation doomed to fail? And what should we do about it?
    - Assumptions make it easy to show by counter-example that an interpretation is not faithful.
    - Position: this fatalistic view is unproductive.
- Possible way forward?
    - Domain restrictions. We care about natural input spaces and specific tasks.
    - Targeted interpretations. Interpret work only on specific examples.

☐ Towards transparent and explainable attention models

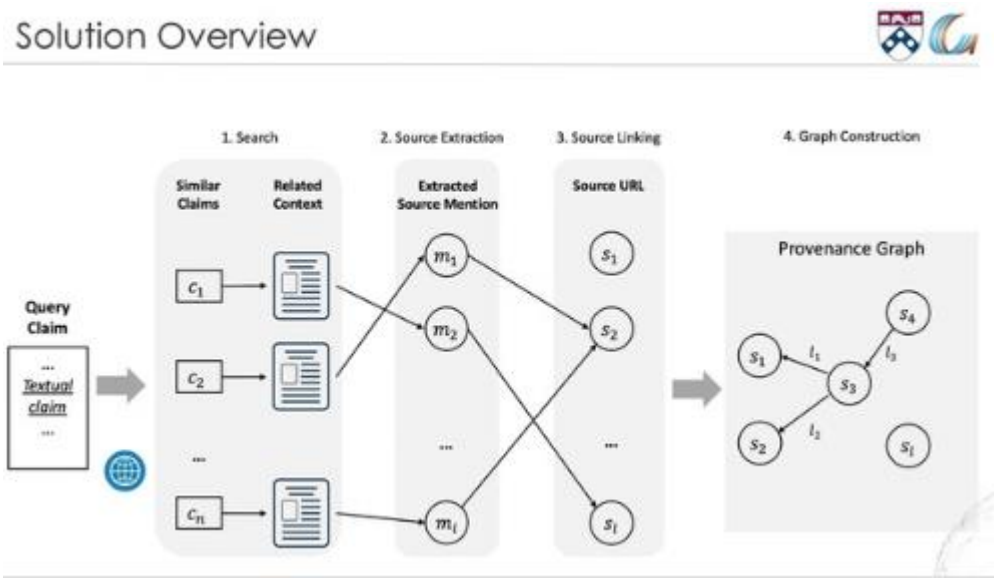
https://www.aclweb.org/anthology/2020.acl-main.387/

- Motivation: explain the attentions. Attention distributions do not provide a faithful explanation for the model's predictions.
- Wiegreffe and Pinter (2019): there is still a possibility that attention distributions may provide a plausible explanation which can be understood by a human even it is not faithful to how the model works.
- Do attention distributions provide a faithful explanation?
    - Case 1: high similarity in input representations -> sentiment classification might not change much
    - Case 2: low similarity in input representations -> results might change a lot. They are faithful here.
- Looking closely at an LSTM based model.
    - Quantify the similarity between these vectors? Conicity(.) = avg of ATM, where ATM is cosine similarity between hidden representation h and hs' means.
    - LSTM have high conicities. Therefore not faithful explanations.
- Observation: Heavy attentions are assigned to punctuations. -> Doubtful that they will provide reasonable explanations.
    - Possible reason: might capture a summary instead of the individual positions.
- Main goal: design models with lower conicity in their attentions.
    - Method 1: Orthogonalization. Modify the loss term.
    - Method 2: Diversity Driven Training. Add the conicity to loss term.
- Evaluations: accuracy & conicity.

☐ "Who said it, and why?" Provenance for natural language claims

https://www.aclweb.org/anthology/2020.acl-main.406/

- Motivation: people can generate and publish content very easily.
  - Call for a longer-term, holistic, and systematic approach to navigate information influence and pollution.
  - We need Provenance!
- Provenance:
  - Use case example. How do we know whether a claim is true? Linking to the evidence text. Question: is this a reliable source? Are the authors themselves influenced by someone else?
  - Provenance encodes claims and (1) relationship between the claim and its source, (2) reference between sources and changes between their corresponding claims, (3) composition of claims.
  - This is important for (beyond claim verification) (1) describing the context an history of the claim. (2) studying "who" influences the claim / author, and (3) understanding the relationship between documents with claims.
- Formalizing claim provenance: a labeled directed acyclic graph



☐ ERASER: A benchmark to evaluate rationalized NLP models

https://www.aclweb.org/anthology/2020.acl-main.408/

- Benchmark for evaluating interpretability (re: Rationales)
- Evaluating Rationales And Simple English Reasoning (ERASER)

- Tasks:
  - Classification: Sentiment, claim verification (FEVER, MultiRC), ...
  - QA: BoolQ, CoS-E
  - All tasks can be phrased as either
    - Doc+Query classifications
    - Q+A vs Q+A ranking
  - All rationales are extractive.
- Metrics
  - Plausibility: do the rationales match human intuition?
  - Faithfulness: does rationale reflect information used to make a prediction?
    - Comprehensiveness (Yu et al., 2019) are all features needed to make a decision present?
    - Sufficiency: Are we able to make a decision from *only* the predicted rationales?
    - Unincluded metrics: "tokens to flip" (Serrano and smith, 2019). "leave-one-out" (Jain and Wallace, 2019)
- Benchmark models
  - BERT to BERT
  - Lei et al., Extractor + Classifier

☐ Learning to faithfully rationalize by construction https://www.aclweb.org/anthology/2020.acl-main.409/

- What is a rationale?
  - Snippet of text "explaining" model's decision
- Discrete rationalization paradigm (Lei, Barzilay, Jaakkola, EMNLP 2016)
  - This is faithful by design.
  - Easy to train if human rationales are available. (but we don't usually have.)
- Propose FRESH (faithful rationale extraction from saliency thresholding)
  - Threshold saliency scores from existing post-hoc explanations
- Compare to previous works:
  - Lei et al 2016: extractor and classifier.
  - Bastings et al., ACL 2019: generator and a classifier
  - (for human evaluation) Also compare to the ground-truth
- Evaluation metrics:

    - Performance comparison on SST, AGNews, Ev. Inf, Movies, MultiIRC
    - Rationale plausibility (AMT user study): sufficiency and coherence

☐ Information-theoretic probing for linguistic structure https://www.aclweb.org/anthology/2020.acl-main.420/

- Propose control function and measure the information gain
- What do control functions mean?
- Argue that BERT can know nothing -- contextualized representations of a sentence contain exactly the same amount of information about syntax as does the sentence itself.

☐ Similarity analysis of contextual word representation models https://www.aclweb.org/anthology/2020.acl-main.422/

- Q: How do the design parameters of contextualized LMs affect their representations?
- Approach: similarity analysis
- Related work:
    - Probing / diagnostic classifiers. Need annotations + sensitive to choice of features.
    - Similarity analysis to investigate learning dynamics (Raghu et al., 2017) etc.
- Methodology
    - Generate representation at every layer of every model, given a set of models and corpus.
    - Compute various similarity measures: nerosim, reprsim
    - Neuron-level similarity: high if there are pairs of similar neurons. (Bau et al., 2019)
    - Representation-level similarity: CKA, SVCCA, PWCCA. High if models share a similar subspace.
    - Models: ELMo variants, GPT, BERT, XLNet
- Observations
    - Similar representations, different neurons
    - Models in the same family are similar
- Localization measure. One concept captured in one neuron (Hinton, 1984). Measure: normalized difference of Aggregate(neuronsim) -Aggregate(reprsim).
    - If high neuronsim: unlikely that both models learned the same distributed representations.
    - Localization appears to increase with layer number.
- Attention-based similarity. JSD then aggregate.

☐ How does BERT's attention change when you fine-tune? An analysis methodology and a case study in negation scope https://www.aclweb.org/anthology/2020.acl-main.429/

- Probing the attention: Clark et al (2019) found in the pretrained BERT attention encoding many syntactic relationships in very intuitive ways (dependent -> head)
- Question: How to verify whether a specific encoding of certain knowledge is relevant to the downstream task?
- Method:
    - finetune the LM to downstream task vs control task.
    - If the knowledge is really relevant in downstream task, then the downstream task representation is detectable; and the control task is undetectable.
    - Example: a case study of negation scope.

☐ Interpreting pretrained contextualized representations via reductions to static embeddings https://www.aclweb.org/anthology/2020.acl-main.431/

- Interpreting pretrained contextualized representations; new interpretability techniques.
- Propose context-agnostic models.
- To decontextualize: Average the contexts.
- Experiment 1: word similarity / relatedness
    - Clarification on where lexical semantics is best encoded
    - Evidence that representations are over-contextualized (potentially related to anisotropy)
    - High quality word embedding can be easily extracted
- Experiment 2: social bias
    - Groups: gender, race, religion
    - Normative considerations: representational harms, stereotypes pertaining to adjectives, professions may precipitate allocative harms.
    - Use several estimators for social bias

☐ Learning to deceive with attention-based explanations https://www.aclweb.org/anthology/2020.acl-main.432/

- Attention as explanation
- Setup
    - Use tasks that we know certain words a-priori to be useful for prediction
    - Measure attention mass on these tokens

    - Examine if the models can be manipulated to reduce attention mass on these impermissible tokens.
- Method
    - Apply loss term on the impermissible tokens
    - Found attention is easy to manipulate with negligible drop in accuracy
    - Models find interesting alternative workarounds.
- Evaluation with human study
    - Q1: do you think that this prediction was influenced by the gender of the individual?
    - Q2: Do you believe that highlighted tokens capture the model's prediction?

☐ Spying on your neighbors: fine-grained probing of contextual embeddings for information about surrounding words https://www.aclweb.org/anthology/2020.acl-main.434/

- How much context is encoded in a contextual word embedding? What kinds of information are obtained from surrounding words?
- Example task 1: probe the word class of each word
- Task 2: Probe the tense of words in the sentence.
- See the paper for more details on tasks set-up.

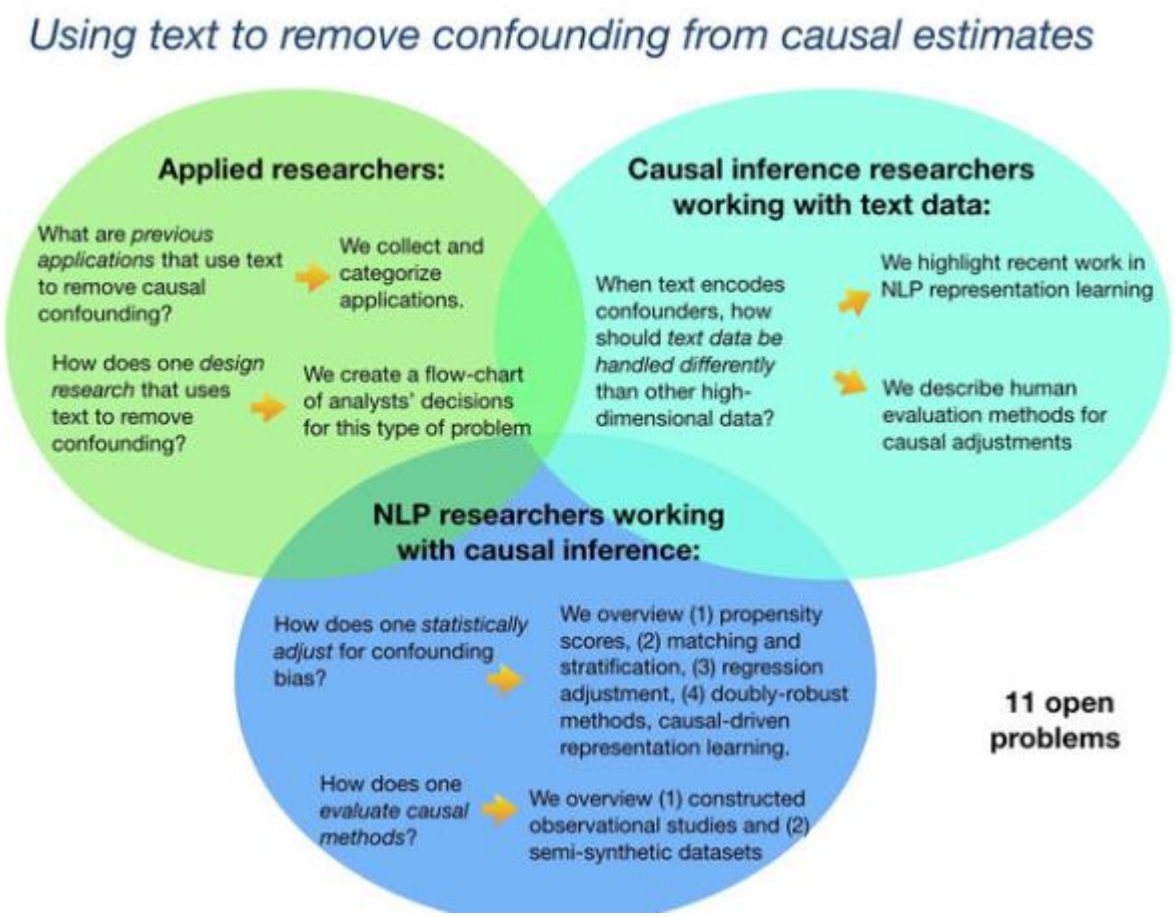
☐ Beyond accuracy: behavioral testing of NLP models with checklist https://www.aclweb.org/anthology/2020.acl-main.442.pdf

- Best paper award
- How do I check if my model works?
- Propose: test NLP models, like we test software, following a Checklist.
- The Checklist test suite:
    - What to test: Linguistic capabilities (rows)
        - Temporal, negation, coreference, SRL, logic, POS, taxonomy, robustness, NER, …
    - How to test: 4 conditions (columns)
        - Min functionality test. Simple small datasets.
        - Perturbation tests.
        - Invariance tests.
        - Directional Expectation tests.
    - Fill in this matrix
- Writing tests at scale: tooling
    - Templating + RoBERTa

    - Lexicons
    - Perturbation library
    - Visualizations, etc.
- Test some SOTA models.
    - E.g., sentiment analysis: commercial models + research models
    - A lot of bugs in high-performing systems.
- Case study: MSFT's sentiment analysis (already stress-tested)
    - With Checklist: found many new bugs.
- User study: testing BERT on QQP
- https://github.com/marcotor/checklist

☐ Text and causal inference: a review of using text to remove confounding from causal estimates https://www.aclweb.org/anthology/2020.acl-main.474/

- Review: Applications and methods for which text encodes confounders.
-



☐ Finding universal grammatical relations in multilingual BERT

https://www.aclweb.org/anthology/2020.acl-main.493.pdf

- Background: Unsupervised LMs learn linguistic structures. Multilingual models show strong cross-lingual performance.
- Q: Does Multilingual-BERT learn a cross-lingual representation of syntactic structure?
- Cross-lingual probing experiments give affirmative answer to this question.
- Method: Following (Hewitt and Manning, 2019)
- Evaluation metrics: UUAS (Unlabeled undirected attachment score), DSpr. (Distance Spearman Correlation)

☐ Obtaining faithful interpretations from compositional neural networks

https://www.aclweb.org/anthology/2020.acl-main.495.pdf

- Background: compositional reasoning
  - Neural module networks: Andreas et al., 2016
  - Module execution is not faithful.
- Propose:
  - Ways to improve module-wise faithfulness
    - Visual-NMN: count module mediates backprop
    - Decontextualized word vectors improve faithfulness
    - Supervising module output (pretrain on the GQA dataset) improves faithfulness
  - Systematic evaluation of intermediate module execution
- Evaluation
  - How do we evaluate faithfulness?
  - Collect intermediate outputs for 536 programs. Compute precision, recall, F1.
  - DROP dataset: Cross entropy between gold and predicted span probabilities.
  - Dataset on github.com/allenai/faithful-nmn

☐ Neural-DINF: a neural network based framework for measuring document influence

https://www.aclweb.org/anthology/2020.acl-main.534/

- Goal: measure the influence of articles without looking at citations.
- Method:
  - Step 1: generate static word embeddings in each time slice separately
  - Step 2: Adversarial training + refinement procedure to align these embeddings to the same vector space

    - Step 3: Present a new metric to calculate the influence of a document without citations.
    - Influence evaluation score: computed with the semantic shift, term frequency C_{d,w}^t and the document frequency C_w^t.
- Evaluation: outperform DIM on ACL anthology

☐ A tale of a probe and a parser https://www.aclweb.org/anthology/2020.acl-main.659/

- Background: syntactic probe
- Question: does a parser make a good probe?
- Experiments:
    - On UUAS: Parser wins
    - On DSpr: probe wins (the probe designed for DSpr is more attuned to it)
- Choice of metric is very important for probing.

☐ Do transformers need deep long-range memory? https://www.aclweb.org/anthology/2020.acl-main.672/

- Background: language modeling, Transformer XL.
- Scientific Q: does the transformer's superior performance arise from having long-range memory representations at every layer?
- Engineering Q: The memory matrix is huge; can we save space and computation by having fewer long-range memories?
- Approach: intervention experiment
    - Replace long-range memories with short-range memories for a subset of layers
    - Consider different arrangements of long-range memories vs short-range memories.
- Results:
    - Obtain comparable performance with 4 long-range memories vs 24
    - Placing long-range memories in first layers works poorly
    - Placing long-range memories interleaved or in last layers works very well.

Category: NLP+Society

☐ Simple, interpretable and stable method for detecting words with usage change across corpora https://www.aclweb.org/anthology/2020.acl-main.51/

- Task: the word usage-change task

- Existing approaches
  - Identifying words that are used differently (1) over time, or (2) by different populations.
  - Hamilton et al.: semantic shift detections (not stable across runs)
- Proposed methods:
  - Intuition: Words whose usage changed are likely to be interchangeable with different sets of words, and thus have different neighbors in the two embedding spaces.
  - Proposal: nearest neighbors (in the shared vocabulary space) as a proxy for meaning.
  - Requires very basic filtering (stop words and frequency cut-off)
- Evaluation:
  - Stability: the overlapping proportion of words across time. Alignment-based methods are highly sensitive to names.
  - Interpretable: can just plot the nearest neighbors around each word at each time.

☐ iSarcasm: a dataset of intended sarcasm https://www.aclweb.org/anthology/2020.acl-main.118/

- Background: sarcasm
  - Sarcasm is a form of verbal irony (Wilson, 2006). Literal meaning != intended meaning.
  - Why study sarcasm? Very frequently present. Disruptive of sentiment / emotion analysis systems.
- Formulation as a classification problem.
- Previous labelling methods do not account for the contextual nature of sarcasm, leading to noisy labels.
  - Manual labeling or Distant supervision both use a proxy for labeling -> false positives / negatives.
  - Propose to remove the proxies. Directly asking the authors to label their texts.
- Methodology: crowdsource sarcastic tweets.
  - Survey + explain + rephrase.
  - Quality control: automatic + linguistic expert (filtering out spurious samples; not alter labels).
- Observations:
  - A lot of perceived but not intended, and intended but not perceived.
  - Previous models may adapt to labelling noise (and still perform better than benchmarks on iSarcasm).
  - Need novel models with sociocultural considerations.

    - Sarcasm detection is not just NLP. It is computational social science.

☐ Is your classifier actually biased? Measuring fairness under uncertainty with Bernstein bounds https://www.aclweb.org/anthology/2020.acl-main.262/

- Background: classification bias.
    - Methods in (Hardt et al., 2016): demographic parity, equal opportunity, equalized odds.
    - Datasets are not annotated with protected attributes.
    - Previous work: create a small dataset annotated with a protected attribute and use them to estimate the bias.
- Q: How can we quantify our uncertainty about the bias estimate?
- Method: propose Bernstein bound unfairness (BBU).
- More uncertainty <-> larger t <-> less tight bound.
- Takeaways: it is possible to claim the existence of classification bias - with some level of confidence - without knowing the exact magnitude. Datasets currently used to estimate bias in NLP are too small to conclusively identify bias, except in the most egregious cases.

☐ It takes two to lie: one to lie, and one to listen https://www.aclweb.org/anthology/2020.acl-main.353/

- Present the dataset involving playing the Diplomacy game.
- Available at Convokit.cornell.edu
- Very impressive presentation video.

☐ He said "who's gonna take care of your children when you are at ACL?" https://www.aclweb.org/anthology/2020.acl-main.373/

- Social media networks:
    - Freedom of speech, but also source of hate speech
    - The tweets reporting sexist messages should not be classified as sexist.
- Previous work: sexism detection is casted as a binary classification task.
- A novel characterization of sexist content-force relation.
- A new French dataset for social media text annotated for sexism detection.
- A set of experiments to detect sexist content.
    - Binary classification
    - Three classes (reporting content vs non-reporting vs non-sexist)
    - Cascade classifier (first sexist content, then reporting)

☐ When do word embeddings accurately reflect surveys on our beliefs about people? https://www.aclweb.org/anthology/2020.acl-main.405/

- How can we use text to better understand beliefs about people?
- Questions:
  - What is a "belief about people"
    - A perception of a given identity on a given dimension
  - Why should we care about measuring them?
    - Because they hold people accountable for the labels they apply to others
  - How should we measure them?
  - How good are existing approaches at doing so?
- Background:
  - Identity
  - Beliefs about identities exist along a series of sociocultural dimensions.
- Methods
  - E.g., dimensions: color, gender, evaluation (good - bad), potency (weak - strong)
  - Semantic difference scale: let people drag the bar along a scale. Asking people is expensive; historical data is sparse, and people are bad at remembering.
  - Evaluating beliefs using embeddings.
  - What dimension you are evaluating is much more important than how you are measuring it.
  - Belief-level analysis

☐ S2ORC: The Semantic Scholar Open Research Corpus https://www.aclweb.org/anthology/2020.acl-main.447/

- 81M+ papers, 380M+ citation links, including 8M+ papers with structured full text.
- Academic disciplines in S2ORC: a lot.
- Behind the scene:
  - Identify copies from Semantic Scholar search engine.
  - Normalize their metadata to a single store representation. E.g., resolving conflicts.
  - Find open-access PDFs.
  - Parse PDF to S2ORC JSON
  - Resolve citation links
- Example: CORD-19 dataset.

☐ How does NLP benefit legal system: a summary of legal AI https://www.aclweb.org/anthology/2020.acl-main.466/

- What is Legal AI?
- Challenges of the tasks of Legal IA?
- How to use NLP technologies to benefit the research of legal AI?
    - Three representative tasks: legal judgment prediction, similar case matching, legal question answering
- What should be the role of legal AI in the justice system?

☐ Predictive biases in NLP models: a conceptual framework and overview https://www.aclweb.org/anthology/2020.acl-main.468/

- Predictive models in NLP are biased.
- Goal: provide a conceptual framework and math definitions for organizing work on biased predictive models in NLP.
- Use the target population. Check if there are disparity of outcome / errors.
- Bias, as outcome and error disparities, can result from many origins:
    - The embedding model
    - The feature sample
    - The fitting process
    - The outcome sample

☐ Detecting perceived emotions in hurricane disasters https://www.aclweb.org/anthology/2020.acl-main.471/

- Propose HurricaneEmo dataset
    - 15k English hurricane disaster-related tweets
    - Temporal: spans multiple hurricanes in the 2017 Atlantic hurricane season
    - Labeled: annotated for fine-grained emotions derived from the Plutchik Wheel of Emotions

- 

☐ Language technology is power: a critical survey of "bias" in NLP
https://www.aclweb.org/anthology/2020.acl-main.485/

- Recent vital work demonstrate that NLP systems exhibit "bias". Many works struggle to define "bias".
- Calls us to be precise about what we mean by "bias".
- Survey 146 papers on "bias" in NLP, focus on text.
- Afterpath: recommendations for going forward
  - Analyze language and social hierarchies together. Ask: how are social hierarchies, language ideologies, and NLP systems co-produced?
  - Articulate conceptualizations of "bias". Provide explicit statements of why system behaviors that are described as "bias" are harmful, in what ways, and to whom. Make sure motivations and quantitative techniques are well-matched, for example.
  - Examine language use in practice.

☐ Social bias frames: reasoning about social and power implications of languages
https://www.aclweb.org/anthology/2020.acl-main.486/

- Two ways harmful social biases are expressed
  - Overt -- easy to be detected by APIs
  - Subtle -- hard to be detected.
- Machine should do proactive reasoning about social biases.
- Binary hate speech detection is not enough. (e.g., annotations subject to variability)
- Related efforts:
  - Denotational / connotational frames

    - Commonsense inference.
    - Toxicity detection. More structured.
- Corpus: Social Bias Inference Corpus.
- Model: linearize the conceptual frame, on Transformer. Check paper for details.

☐ State and fate of linguistic diversity and inclusion in the NLP world

https://www.aclweb.org/anthology/2020.acl-main.560/

- Example: Dutch vs Somali. Both have similar number of speakers, but Dutch has much more corpus resources and better translation systems than Somali.
- Questions:
    - How many resources are available across the world's languages? Do they correlate with the number of speakers?
    - Which typological features have NLP systems been exposed to? Which features have been underrepresented?
    - How inclusive has ACL been in conducting and publishing research for different languages?
    - Does resource availability influence the research questions and publication venue?
    - What role does an individual researcher or community have in bridging the resource division?
- Website: https://microsoft.github.io/linguisticdiversity

☐ Sentence meta-embeddings for unsupervised semantic textual similarity

https://www.aclweb.org/anthology/2020.acl-main.628.pdf

- Task: STS (semantic textual similarity) predict similarity score of a sentence pair.
    - Supervised STS: SOTA is T5. Sentence-pair encoder with cross-sentence attention
    - Supervised "siamese". Finetuned sentence encoder + similarity measure.
    - Unsupervised: pretrained sentence encoder + similarity measure. Efficient for retrieval. No labeled STS data required. Deal with unsupervised setting in this paper.
- Recap: word meta-embedding
- Recap: unsupervised STS methods
    - Diversity of architectures
    - Diversity of pretraining data / tasks
    - Why not use the word-embedding methods and apply to sentence embeddings?
- Methods:

    - Concatenation; averaging (as baselines)
    - M1: SVD, adapted from Yin and Schutze (2016)
    - M2: GCCA adapted from Rastogi et al., 2015
    - M3: Autoencoder with reconstruction loss. Adapted from Bollegala and Bao (2018)
- Experiment:
    - ParaNMT, SentenceBERT, Universal Sentence Encoder
    - Tasks: STS12 - STS16 from Agirre et al., (2016). STS benchmark test set Cer et al., 2017. BWC (Chelba et al., 2014)
    - Combining sentence embedding is a good idea. GCCA is better.

☐ Automated evaluation of writing - 50 years and counting

https://www.aclweb.org/anthology/2020.acl-main.697/

- Task: automated evaluation of writing (AWE)
- In 1966, Ellis Page provided a proof-of-concept demonstration of AWE and outlined some challenges.
- The most visible current use is arguably AWE for standardized testing, leading to new challenges.
- AWE is becoming universally available. What value will users find in it?
    - Supporting decisions about the user
    - Augmentation the user's skill for best written product
    - Helping the user improve writing skill
- Future: can design the tools and their evaluations to focus on specific type of use -- by considering the user's goals and by engaging partners from other disciplines.

☐ On forgetting to cite older papers: an analysis of the ACL Anthology

https://www.aclweb.org/anthology/2020.acl-main.699/

- Example: the citation age.
- Q: Can we identify the change of citation ages over time?
- Data: ACL Anthology dataset. 2010 - 2019
- Pipeline: PDFs -> Extract text -> ParsCit reference parser
- Observations:
    - A decrease in avg age of citations
    - But more publications = more papers to cite
    - We cite older papers at a steady rate.

    - There is less variety in citations of older papers

☐ Gender gap in NLP research: disparities in authorship and citations

https://www.aclweb.org/anthology/2020.acl-main.702/

- Gender gaps
- This work examines gender gaps in NLP research
    - Disparities in authorship
    - Disparities in citations
- Raises concerning gaps between male and female authors: number of papers, citations, etc. There is no easy answer to this question.

Category: Language with linguistic theory + cognitive psychology + semantics

☐ A three-parameter rank-frequency relation in natural languages

https://www.aclweb.org/anthology/2020.acl-main.44/

- An extension of Zipf's law
    - Zipf's law is a general tendency, but the linearity is not so perfect.
- Propose the three-parameter rank-frequency relation
- See paper for equations.
- What do the parameters mean?
    - Alpha: an axis of analysis-synthesis on syntax.
        - Smaller: syntactic role more afforded by affixes within content words e.g., Uralic and Slavic with heavy declensions
        - Larger; a group of frequently used function words to afford syntactic role. E.g., Germanic and Romance with abundant articles and prepositions.
    - Beta + gamma_norm: analysis-synthesis on morphology
        - Smaller: many rare words by derivation or compounding e.g., compounds in Germanic
        - Larger: fewer rare words, using multi-word expressions for rare concepts. E.g., multi-word expressions in French and English
    - More evidence.
        - Chinese: small alpha and huge beta. Nearly all words are composed by multiple characters.

        - Tokenized Japanese: smaller alpha for larger granularity. Less functional suffixes segmented.
        - On smaller data: stable alpha and larger beta+gamma_norm. alpha related to function words; beta+gamma_norm related to vocabulary.

☐ Modeling code-switch languages using bilingual parallel corpus
https://www.aclweb.org/anthology/2020.acl-main.80.pdf

- Code switch happens when the speaker mixes lexicons from different languages.
- Previous theories:
    - Matrix language frame theory (a dominant / matrix language) and inserted (embedded) language
    - Equivalence constraint theory: imposes an additional constraint that the CS points will only happen at the boundaries shared by both languages.
- Proposed methods:
    - Monolingual objective + establish cross-lingual word level correspondence.
    - Dataset: LDC2015S04 South East Asia Mandarin-English (SEAME)

☐ Explicit memory tracker with coarse-to-fine reasoning for conversational machine reading
https://www.aclweb.org/anthology/2020.acl-main.88/

- Setting: conversational machine reading e.g., CoQA. The text to read contains a recipe to derive the answer, instead of containing the literal answer.
- Task: ShARC: Shaping answers with rules through conversation.
- Proposed solution:
    - Explicit memory tracker
    - Coarse-to-fine reasoning
- New SOTA results on ShARC benchmark.

☐ Injecting numerical reasoning skills into LMs https://www.aclweb.org/anthology/2020.acl-main.89/

- In LM, some skills (e.g., numerical reasoning) are missing when training MLM
- Numerical reasoning requires:
    - Locating the relevant entities & numeric strings
    - Map to floats
    - Then perform arithmetics
- Approaches
    - Previous work: Symbolic approach: using a non-differentiable external calculator

    - Text-to-text approach. End-to-end differentiable. E.g., GenBERT. (For now: GenBERT is the Transformer with some structural modifications. This doesn't work well; reduces too much accuracy compared to symbolic systems. Will add techniques and new training data below.)
- What to do?
    - Data: Synthesize text and numerical data. ND (numerical) + TD (textual data).
    - Technique: digit tokenization (DT). Wordpiece tokenization. Then further tokenize into individual digits.
    - Technique: random shift (RS). Randomly right-shift the position Ids of tokens. This reduces overfitting to short instances.
    - Then do MLM training.
- Experiments: performance on different datasets, with ablation studies.

☐ Moving down the long tail of WSD with gloss informed bi-encoders

https://www.aclweb.org/anthology/2020.acl-main.95.pdf

- Motivation: WSD datasets suffer from data sparsity. Has a long tail.
- Previous:
    - lexical overlap between context and gloss (Lesk, 1986) Gloss are lexical information e.g., sense definitions.
    - Recent work: neural models integrate glosses (Luo et al., 2018a,b)
    - Pretrained models for WSD: simple probe classifiers have been shown to perform well.
    - Also: GlossBERT
- Propose: gloss informed bi-encoder
    - Two encoders independently encode the context and gloss, aligning the target word embedding to the correct sense embedding.
    - Both encoders initialized with BERT and fine-tuned end-to-end.
    - Bi-encoder is more efficient than a cross-encoder.

☐ Learning and evaluating emotion lexicons for 91 languages

https://www.aclweb.org/anthology/2020.acl-main.112.pdf

- Collection method: translation + prediction
- Evaluation: strong results across most datasets and variables
- Even on par with many monolingual results.
- MEmoLon Dataset: https://github.com/JULIELab/MEmoLon

☐ Dialogue coherence assessment without explicit dialogue act labels

https://www.aclweb.org/anthology/2020.acl-main.133/

- Task: Dialogue coherence assessment
  - Composed of 2 aspects: Entities and dialogue acts.
- SOTA model for dialogue coherence: e.g., Cervone et al., 2018
- Propose DiCoh model:
  - Use dialogue act prediction (DAP) as an auxiliary task for training coherence model in a multi-task learning scenario
  - Synthetically define perturbation methods to destroy the coherence of dialogues:
    - Utterance Ordering, Utterance Insertion, Utterance Replacement, Even Utterance Ordering (only shuffle within each speaker)
- Datasets: DailyDialogue, SwitchBoard
- Compare against several baseline models.

☐ A Systematic Assessment of Syntactic Generalization (SG) in Neural Language Models

https://www.aclweb.org/anthology/2020.acl-main.158.pdf

- Traditional evaluation of LMs: perplexity
  - As perplexity improves, can we expect models to become more human-like?
- Good evaluate metrics should correlate more to human-like generalization.
- Proposed evaluation scheme: (1) train on grammatical sentences, and (2) test on two variants: (2a) Unseen, ungrammatical, and (2b) Unseen, grammatical. (3) Show that model prefers 2b > 2a
- 34 English test suites
  - E.g., Subject-verb agreement
  - 2x2 design, success defined by conjunction of inequalities
  - Controlled experiments.
- Results:
  - Dissociation between test set ppl and syntactic generalization
  - Architecture has larger effect on SG score than data size
  - Purely sequence-based LSTM underperforms other models

☐ Overestimation of syntactic representation in neural LMs

https://www.aclweb.org/anthology/2020.acl-main.160

- Question: can NNs induce hierarchical syntax without supervision?

- Previous work: mixed results on a range of neural formalisms
  - "Does a given system behave as though it has syntactic representations?"
  - Prasad et al., 2019: template-based syntactic priming.
- This work: reproduced positive results from the paper with two non-syntactic baselines.
- Illustrates a fundamental problem with tasks introduced by (Prasad et al., 2019)

☐ Automatic detection of generated text is easiest when humans are fooled

https://www.aclweb.org/anthology/2020.acl-main.164/

- Review: neural LM and decoding strategies
  - e.g., beam search of beam size k
  - Top-k sampling
  - Nucleus sampling
- Tradeoff of decoding strategy:
  - Humans are fooled (k->1) vs automatic systems are fooled (k-> vocab size)
- Investigate the impact of priming the LM with text using either no priming (nowordcond) or a single word of priming (1 wordcond)
  - Total of 6 corpora of generated text, each containing 250k training examples
  - Build ~250k examples of machine-generated text and 250k examples of human-produced text.
  - Finetune a BERT to decide generated or human. Compare against different decoding strategies.
- Conclusions
  - Even SOTA LMs are not good enough at modeling language for us
  - Humans notice bad word choices; they don't tend to notice when text is a little less interesting or diverse.
  - Automatic discriminators are quick to pick up on an over-representation of common words.

☐ A tale of two perplexities: sensitivity of NLMs to lexical retrieval deficits in AD

https://www.aclweb.org/anthology/2020.acl-main.176/

- Motivation: Alzheimer's disease detection
- A tale of two perplexities
  - LM perplexity within subjects: decreases over time;
  - LM perplexity across subjects: higher with dementia

    - N-gram LM perplexity: perplexed by dementia / controls (Wankerl et al., 2017)
    - Difference in neural LM perplexity
- DemBank problems...? Propose two methods:
    - Interrogation by perturbation.
    - Interrogation by interpolation.
- Experimental setup:
    - DemBank LOOCV. RWTHLM (LSTM-150), then GluoNLP (standard-LSTM-lm-200)
    - By perturbation: more frequent word: AD model less perplexity. Less frequent words: control models more perplexed.
    - By regression: control model PPL increases with mean lexical frequency. Dementia model PPL decreases.
    - By interpolation
    - Got 0.93 to 0.94 in LOOCV accuracy for classifying AD.

☐ Recollection vs imagination: exploring human memory and cognition via NLMs

https://www.aclweb.org/anthology/2020.acl-main.178/

- NLP tools as new lens on cognitive processes?
    - How does writing change when the event is imagined vs recalled?
    - What effect do time elapsed and narrativization of the event have on writing?
- Dataset: Hippocorpus. 6854 stories of 15-25 sentences.
    - Recalled stories (N=2279) + Imagined stories (N=2756) + retold stories (N=1319)
- NLP measures to analyze narratives
    - How does the narrative flow? Delta_l: Story linearity
    - What type of events is the story about? Episodic vs semantic knowledge
    - Lexicon-based measures (psychometrics from LIWC)
    - See paper for details.
- Results
    - Imaged vs recalled stories: density plot shows imaged stories have higher linearity.
    - Narrativization of memories over time: retold stories flow more fluently.
    - Frequency of recalling the memory: stories became more linear, less hierarchical over time.
    - Narrativization lead stories to be more similar to imagined stories.

☐ Speakers enhance contextually confusable words https://www.aclweb.org/anthology/2020.acl-main.180/

- Background: Speech production and efficient communication.
  - Contextually predictable words, syllables, speech sounds tend to be phonetically shortened.
  - Speakers seem to prefer shorter variants of the same word when the underlying word is more contextually predictable.
  - Less clear: are speaker choices shaped by pressures for effective (robust to noise) communication?
- This work: evaluate whether more contextually confusable words are associated with longer durations in natural speech.
- Listener model:
  - Use a generative model to sample an intended word
  - Use a noise channel to sample an listened word
  - Listener computes a posterior distribution over intended words given the perceived word
- Word perceptibility: measures the prob that the listener will accurately recover it, given that it was intended.
- Word confusability: the log reciprocal of its perceptibility
- Statistical analysis controlling variables, including:
  - Neighborhood size
  - Log weighted neighborhood density
  - Unigram confusability
- Contextual confusability remains significant after controlling for unigram confusability. (for Switchboard, Buckeye)
- Conclusion:
  - Speakers increase the duration of more confusable words
  - Confusability is distinct from neighborhood density
  - Effects of confusability are context-sensitive.

☐ What determines the order of adjectives in English? Comparing efficiency-based theories using dependency treebanks https://www.aclweb.org/anthology/2020.acl-main.181/

- Adj ordering preferences are not arbitrary.

- Pre-nominal languages vs post-nominal languages
  - Big brown bag vs bag brown big.
  - They have reversed adj ordering preferences.
- Four hypotheses:
  - One hypothesis: "brown" has more subjectivity. "big" less, so it's placed further from the noun.
  - H2: Integration cost theory. Ordering by integration cost gradually reduces the entropy of possible noun referents.
  - H3: Information gain increases from "big" to "bag" increases?
  - Hypothesis 4: PMI increases.
- Experiment to test these hypotheses:
  - Theory are in the form {A1, A2, N} -> rank condition.
  - Dependent measure: corpus frequencies of (A1A2N) vs (A2A1N).
  - For subjectivity: use AMT. For the H2-H4 information theoretic hypotheses: use AN pairs. (SyntaxNet-Parsed Common Crawl, and EWT data).
- How to test all this?
  - PMI, IC, IG are estimated from joint frequencies of wordforms, but counts are sparse.
  - Also estimate based on semantic clusters. -> IG gets better, PMI gets worse.
  - Get cluster? Apply k-means to GloVe.
- Why these shifts?
  - Probably something semantic captured by cluster-based IG.
  - Probably something collocational is captured by wordform PMI.

☐ Enabling LMs to fill in the blanks https://www.aclweb.org/anthology/2020.acl-main.225.pdf

- Motivation: filling in the blanks should consider both previous and subsequent contexts. Useful for writing emails to tune the tones. Also useful for connecting ideas.
- Text infilling task: with arbitrary number of blanks.
- Previous works:
  - GPT-*: Can't consider future text
  - BERT: Must know exact number of tokens
  - SA (Zhu et al., 2019) can’ t leverage the pretrained models
- Propose ILM
  - Usage: Finetune existing LMs.

Manufacture infilling examples.

Set-up training data for in-filling.

Fine-tune LM on infilling examples.

- Data: Stories (Mostafazadeh et al., 2016), Abstracts, Lyrics
- Metric: Score, perplexity. E.g., Human evaluation: Turing test.

☐ Representations of syntax [MASK] useful: effects of constituency and dependency structure in recursive LSTMs https://www.aclweb.org/anthology/2020.acl-main.303/

- Q: Does natural language contain enough signal for sequential networks to robustly learn syntactic structures?

  If not, how do we provide stronger inductive biases toward syntactic structure?

  Which types of syntactic representations should we encourage models to learn?
- Two representations:

  Dependency representation

  Constituency representation
- Models

  BiLSTM

  Constituency LSTM

  Dependency LSTM

  Head-lexicalized LSTM
- Experimental task: English subject-verb agreement

  Non-sequential syntactic dependency

  Data: Linzen et al., 2016 dataset.

☐ Analysing lexical semantic change with contextualized word representations https://www.aclweb.org/anthology/2020.acl-main.365/

- What type of word representations best capture semantic change?
- Background:

  One vector for each word form or word sense?

  Word usage (occurrence)?
- Methods: Unsupervised approach to lexical semantic change.

  Step 1: Look for a target word. Extract a vector for each occurrence.

  Step 2: Cluster representations into "user types".

Step 3: Bring in the temporal dimension. How many occurrences belong to a user type? "Usage distribution".

Step 4: Obtain a "shift score" -- a number that quantifies the amount of change from time interval t to t'. JSD or Entropy Difference, or Average Pairwise Distance.

- Setup:

  Corpus: COHA

  Target words with semantic shift scores: GEMS (Gulordava & Baroni, 2011)

  LM: BERT-base

  Also present new dataset: Diachronic Usage Pair Similarity (DUPs)

- Analysis

  What do usage types capture? (Synchronic)

  - Literal vs metaphorical
  - Polysemy and homonymy
  - Syntactic functionality
  - Entity names

  What kinds of lexical change are detected? (Diachronic)

  - Broadening; Narrowing; Shift; New syntactic role (e.g., "download").

☐ Do NLMs show preferences for syntactic formalisms? https://www.aclweb.org/anthology/2020.acl-main.375/

- Motivation: syntax in neural LMs

  RQ1: does the syntax captured by LMs align more with a function-head or content-head dependency analysis?

  RQ2: Are patterns consistent across different languages?

- Background: Hewitt and Manning (2019)
- UD vs SUD annotation trees
- Method:

  Input: ELMo and BERT

  Probe with UD / SUD. SUD focus more on the function head. UD are more on the content heads.

  Evaluate the UAS.

  Differences with tree heights.

  Consistency between languages.

☐ Compositionality and generalization in emergent languages

https://www.aclweb.org/anthology/2020.acl-main.407/

- Why compositionality?

  Efficient strategy where we need to only encode finite set of words and systematic rules to encode infinite meanings.

  Enables generation to novel meanings.

- Reconstruction Game

  Receiver and a sender GRU

  Unlike previous studies (Kottur et al., 2017), we show that NN agents develop a productive language when trained on rich environment.

- How to measure compositionality?

  Topographic similarity: whether the cosine distance between two meanings correlates with the edit distance between the messages expressing them. (Brighton & Kirby, 2006)

  Positional disentanglement: measures whether symbols in specific positions of the message tend to univocally refer to the values of a specific attribute. (See paper for equation)

  Bag-of-symbol disentanglement: captures the intuition of a permutation-invariant language, where only symbol counts are informative.

- Results

  Compositionality is not necessary for NN agents' generalization

  To generalize without compositionality, agents rely on more symbols than needed to succeed in the game.

- Again, why compositionality?

  Train new Receivers with the emergent languages.

  New agents, with different architectures, show better performances when trained on compositional emergent languages.

☐ SenseBERT: Driving some sense into BERT https://www.aclweb.org/anthology/2020.acl-main.423/

- Pretraining method on word-sense information.
- BERT: predict a masked word
- Our method: predict the sense of a masked word.
- Word in Context task: improves upon SOTA.
- Model and objective:

Model: Transformer encoder. Input & output both word and sense

Data: WordNet: a lexical database. Use its supersense.

☐ On the spontaneous emergence of discrete and compositional signals

https://www.aclweb.org/anthology/2020.acl-main.433/

- Short version: This paper show that agents trained to communicate about a discrete world spontaneously evolve discrete words.
- More detailed version:

  Use an autoencoder to learn language. A sender (encoder) and receiver (decoder).

  Show discreteness emerges spontaneously in production and perception. The messages are not compositional, but they are categorical.
- The game setting: extremity games.

  Sender wants to guess a property. Receiver returns the answer.

  The receiver sees a possibly different context c', and pick the action.

  Note that the messages (latent spaces) are *continuous*.
- Results:

  They succeed in learning communications (with strict setting having better results)

  Measure discreteness of encoding: use DBSCAN cluster algorithms.

  To measure discreteness of perception: randomly sample messages from the clusters.

  Feed to the receiver.

  To measure compositionality: (1) vector analogies and (2) composition network

  Categorical perception.
- Take-aways

  Discreteness emerges naturally, in production and perception

  No evidence of compositional structure

  Hints of categorical perception.

☐ (Re)constructing meaning in NLP https://www.aclweb.org/anthology/2020.acl-main.462/

- Key question: what's missing from NLP?
- We argue: construal

  Well-studies in psycholinguistics

  Provide a review of what it means.
- Example: who is Nora?
- Multiple views of meaning

Database view: just the facts

Conceptulization view: meaning includes how a scene is construed.

Linguistic choices systematically manipulate dimensions of construal.

- Do these differences matter?

  Easy to spot where construal fails in modern NLP systems.

  Construal can be a source of ambiguity. Context matters.

- Meanings are relativized to content, context, and construal.
- Dimensions of construal: an incomplete taxonomy

  Prominence (salience, profiling): language can highlight different aspects of the same scene. E.g., active / passive, transitive / intransitive.

  Resolution (granularity / specificity). Entities and events can be described in many granularities

  Metaphor (structure mapping). Language allows one domain to be understood in the other. Metaphors can influence reasoning and judgment.

  Perspective (vantage point). The same scene can be described from multiple perspectives.

  Configuration. Language can affect how internal / structural properties of entities and events are viewed (boundedness, homogeneity, discrete / continuous, plexity)

- Example: how construal of meaning surfaces in an application.
- When / How does construal matter for NLP? Some possible research directions:

  Flexibility in construal -> variation

  Construal dimension checklist: interrogate datasets, representations, and systems for different construal dimensions. Diversity of inputs / outputs. Sensitivity of systems, etc.

  Construal annotation of texts.

- Construal invites the connection with cognitive science.

☐ Climbing towards NLU: on meaning, form, and understanding in the age of data

https://www.aclweb.org/anthology/2020.acl-main.463/

- Best theme paper
- Human-analogous NLU is a grand challenge of AI
- While large neural LMs are undoubtedly useful, they are not nearly-there solutions to this grand challenge.
- Any systems trained only on linguistic form cannot in principle learn meaning.

- Genuine progress in our field depends on maintaining clarity around big picture notions such as meaning and understanding in task design and reporting of experimental results.
- Working definitions

  Form

  Meaning

  Understanding: given an expression e in a context, recover the communicative intent I
- How do babies learn language?
- Thought experiment: Java
- Thought experiment: meaning from form alone.

☐ How can we accelerate progress towards human-like linguistic generalization?

https://www.aclweb.org/anthology/2020.acl-main.465/

- Best theme paper runner-up
- Evaluation in NLP

  Ideally: want human-like generalization in language systems

  But currently, systems are evaluated with "the pretraining-agnostic identically distributed (PAID)" paradigm
- Current paradigm:

  Based on machine learning. Pretrain then fine-tune.

  PAID: pretraining-agnostic

  Don't reward the generalization-efficient algorithms.
- Conclusions: how can we measure progress towards robust generalization from less data?

  Standardized, moderately sized pretraining corpora

  Expert-created evaluation sets that are inaccessible during fine-tuning

  Should reward few-shot learning

☐ Exploiting syntactic structure for better LM: a syntactic distance approach

https://www.aclweb.org/anthology/2020.acl-main.591/

- Motivation: why structure-aware LM?

  Grammar induction with neural LMs: an unusual replication, EMNLP '18

  LSTMs can learn syntax-sensitive dependencies well, but modeling structure makes them better, ACL '18

  A systematic assessment of syntactic generalization in neural LMs, ACL '20
- Bring syntax into inductive bias when designing structures.

- Method

  Baseline: Ordered Neuron LSTM. Use master input / master forget gates to replace the vanilla input / forget gates.

  How to inject structure? Split-head approach and learning-to-rank loss.

- Results:

  LM perplexity: PTB-Concat, CTB-Sepsent

  Results on syntax & ablation: WSJ

- Conclusion:

  Improved LM perplexity + better induced syntax + extensibility

☐ Predicting declension class from form and meaning https://www.aclweb.org/anthology/2020.acl-main.597/

- Declension class: some nouns fall into this class based on their morphological properties. E.g., number and case.
- Human need to remember the classes of these nouns when learning languages.
- Task: predict which class does each noun fall into.
- How to predict?

  Form and meaning both share a significant amount of information with declension class

  For both languages, form shares more information with class than meaning does

  Some of this information is redundant (given the tripartite MI results)

  In paper: estimating conditional MI, by-class results.

☐ A two-stage masked LM method for term set expansion https://www.aclweb.org/anthology/2020.acl-main.610/

- The TSE task

  Small seed set -> entire semantic class.

  Polysemous words should be disambiguated

  The class words can contain multi-word nouns.

- Previous:

  Pattern-based methods, distributional-based methods.

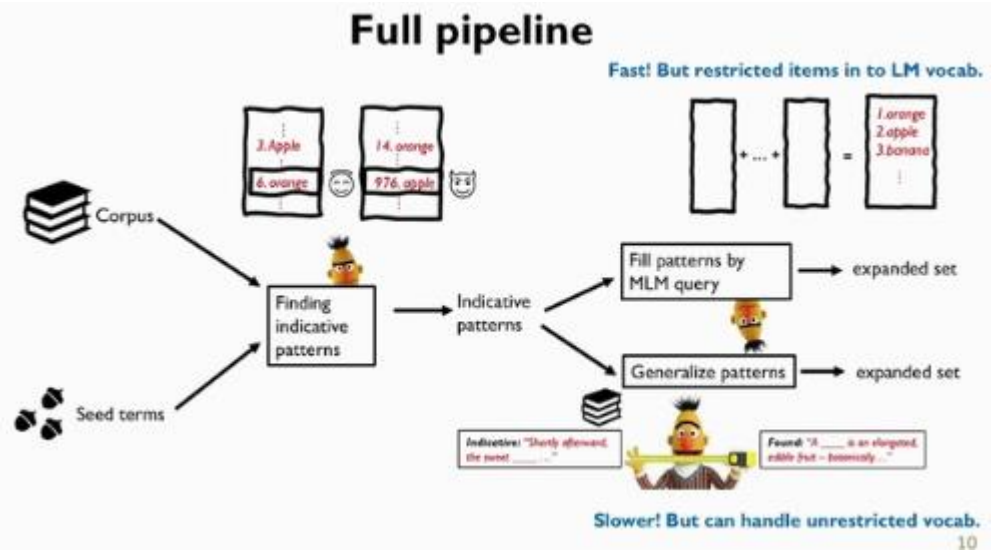
  This work finds a middle ground on this spectrum

- Our method: masked LM pattern-based

  Indicative patterns: patterns in which all class terms have a high probability to appear in.

Query masked LM for terms that are likely to fill the patterns.

Another approach: search for terms that appear in patterns similar to the indicative patterns.



☐ What are the goals of distributional semantics? https://www.aclweb.org/anthology/2020.acl-main.663/

- Goals of semantics: top-down vs bottom-up

  Koller, 2016: bottom-up theories are intrinsically unfalsifiable.

  Top-down goal of semantics: characterize the meanings of all utterances in a language.

- Grounding:

  Goal: connect language to the world.

  Possible approaches: combine distributional and grounded models.

  Map between distributional and grounded models.

  Joint learn from distribution and grounded data. (most promising)

- Concepts and Referents. Goal: know how to evaluate truth, and generalize to new situations.

  Need to distinguish "concept" and "referent". Use in one space or two spaces? Concepts as regions / classifiers are more promising.

- Vagueness. Goal: capture uncertainty about truth.
- Polysemy. Goal: capture multiple senses of words. One representation per sense? Can learn a single, flexible representation.
- Hyponymy. Goal: capture hyponymy relation between words. E.g., asymmetry in coherence; specially defined space; distributional inclusion hypothesis. Regions and distributions.

Hyperonym: more general term

Hyponym: more specific term

- Compositionality. Goal: derive meanings from their parts. Vector composition? E.g., addition and multiplication are surprisingly effective. Neural models.
- Logic. Goal: support truth and entailment. Possible approaches: hybrid model, specially defined space, truth-conditional representations.
- Context dependence. Goal: capture how meaning depends on context. Occasion meaning vs standing meaning.

☐ Good-enough compositional data augmentation https://www.aclweb.org/anthology/2020.acl-main.676/

- Rule-based data augmentation scheme
- Background: compositional generation in semantic parsing.

  Grammar-based model (e.g., 90%acc) vs neural model (95% acc)

  What if we put complex data only in test set -- then neural models will be estimated to have much lower acc.

  Neural sequence models struggle with compositional generalization!
- Data augmentation: build inductive bias into NNs by manipulating the data, not the model architecture.

  Check out (Jia and Liang, 2016)
- A new data augmentation scheme: GECA (good-enough compositional data augmentation)

  Aside: this method relies on superficial cues.
- Experiments:

  SCAN dataset (Lake & Baroni 2018)

  Semantic parsing: GeoQuery dataset (Zelle, 1995, Finegan-Dollak et al., 2018)

  Low-resource LM

☐ Returning the N to NLP: towards contextually personalized classification models https://www.aclweb.org/anthology/2020.acl-main.700/

- Contextually personalized models

  Input: representation of user's language

  Context: neighboring users in a network, eventually conversations with neighboring users

  Pretraining output: contextual user representations

Application: interpreting user's text in a new conversation with similar users

☐ Predicting performance for NLP tasks https://www.aclweb.org/anthology/2020.acl-main.774/

- Estimation of performance without actually training
- Pipeline

  Extract dataset features

  Extract language features

  Model features (categorical)

  Use XGBoost to get a predicted score
- A series of baselines

  Model-wise mean value baseline

  Task-wise mean value baseline
- Experiment task: slot filling
- Feature perturbation - datasize
- Application: representative datasets
- Related work: "query performance prediction", "quality estimation" for MT.

☐ Should all cross-lingual embeddings speak English? https://www.aclweb.org/anthology/2020.acl-main.766/

- Background: cross-lingual representations
- Bilingual lexicon induction can evaluate how well the cross-lingual representation is learned.
- Issue with BLI: evaluation: mostly focused within English settings
- Diverse evaluation lexicons: triangulation + filter out mismatched part-of-speech + filter mismatched morphological analysis + remove proper nouns
- Also create new dictionaries extracted from parallel data.
- Experiments:

  Start with fasttext; align in bilingual or multilingual; evaluate LI on diverse dictionaries

  A wide variety of languages

☐ Uncertain NLI https://www.aclweb.org/anthology/2020.acl-main.774/

- To measure NLU: use the RTE task (recognizing textual entailment)

  NLI shifted to a 3-class label set (entailment, neutral, contradiction)
- Problem: the probabilistic nature of NLI
- Propose an UNLI task (uncertain NLI), a regression task.
- Data elicitation

Take existing premise-hypothesis pairs directly from SNLI to form u-SNLI

Give sliding bar to allow a continuous range of labeling.

- Model for example:

  BERT + regressor. Turn softmax into a sigmoid.

  Got good results on NLI when fine-tuned with u-SNLI data.

☐ Treebank embedding vectors for out-of-domain dependency parsing https://www.aclweb.org/anthology/2020.acl-main.778/

- Treebank embedding for multi-treebank models
- Q:

  what treebank ID to use at test time or in production systems?

  Are there better treebank vectors for a given model than the ones in the lookup table?
- Approach:

  Trian parser as usual. T test time: predict the best treebank vector for each test sentence or for a collection of test sentences. Provide this treebank vector as input to parser.

☐ Why is penguin more similar to polar bear than to sea gull? Analyzing conceptual knowledge in distributional models https://www.aclweb.org/anthology/2020.acl-srw.18.pdf

- Linguistic hypotheses (what can we expect from context?)

  Impliedness (Grice 1975, Dale & Reiter 1995)

  Variability (necessary to follow maxim of quantity)

  Afforded actions (Gibson 1945)

  Typicality. Stereotypes: concepts illustrating properties (Veale & Hao 2007, Veale 2013)
- Dataset: property-concept pairs collected from:

  Semantic feature norm datasets

  Lexical resources (wordnet, conceptnet)

  Enriched with GoogleNews w2v model
- This dataset will be released soon.

☐ Non-topical coherence in social talk: a call for dialogue model enrichment https://www.aclweb.org/anthology/2020.acl-srw.17.pdf

- Example: new-topic utterance. They are coherent, but will be classified as "incoherent" from lexical approaches.
- This work:

Proposes annotation strategy for NTUs

A pilot study for annotations.

- Annotation strategy:

  Annotating Content-based discourse relations between utterances. CRs inherited from Disco-SPICE. Add annotations for more CRs not in Disco-SPICE. (e.g., semantic relations from ISO24617-8 and ISO24617-2)

  NTUs are the utterances that bear no CR to the content of prior discourse.

- To analyze: sequence-based social intents

☐ Learning lexical subspaces in a distributional vector space

https://www.mitpressjournals.org/doi/pdf/10.1162/tacl_a_00316

- Lexico-relational semantics

  Synonymy (symmetric-attract)

  Antonymy (symmetric-repel)

  Hypernymy (Asymmetric-attract)

  Meronymy (asymmetric-attract)

- Previous work: distortion of the distributional space has undesired effects.
- Propose Lexizalized Subspaces:

  Learns specialized subspace for each lexical relation.

  Enforces a separation of considerations.

  Synonymy loss, antonymy loss, hypernymy loss, meronymy loss... See paper for details.

- Evaluation

  Intrinsic: similarity vs relatedness, hypernymy tasks

  Extrinsic: NER, SST, SNLI, SQuAD, QQP

  Neighborhood analysis

☐ Decomposing generalization: models of generic, habitual and episodic statements

https://www.mitpressjournals.org/doi/pdf/10.1162/tacl_a_00285

- Motivation

  Generalization. The key is commonsense reasoning.

  Linguistic generalizations should be captured in a continuous multi-label system.

  Framework based on Decompositional Semantics (White et al., 2016)

- Background

  Standard classification: episodic, habitual, stative, generic.

Arguments and predicates do not always fall under such well defined categories as described.

Current corpora: ACE-2, ACE-2005, EventCorefBank, Situational Entities

- Data collection framework

  Decompose arguments and predicates into simple referential properties

  Collect annotations for argument and predicate properties separately, with confidence ratings for each annotation

  Multi-label annotation schema.

- Axes of reference: spatiotemporal, type, Tangble.
- Correct the annotation bias with confidence & binary normalization, to arrive at a single real-valued score.
- Universal Decompositional Semantics-Genericity (UDS-G) dataset. At decomp.io
- Preliminary analysis

☐ Theoretical limitations of self-attentions in neural sequence models

https://nlp.stanford.edu/pubs/hahn2020theoretical.pdf

- Q: What is the computational power of the Transformers? Understanding this is important for (1) designing better models (2) learning something about the nature of language

  Can Transformers model hierarchical structures, with unbounded recursion?

  Can they correctly close brackets? -- using DYCK_2

  Can they evaluate iterated negation? -- using Parity

- Parity:

  Set of bit strings with an even number of 1s.

  RNNs, LSTMs, GRUs, ..., can do this.

  If transformers can't model this, they can't model any non-quasi-aperiodic language.

- DYCK_2

  Correctly bracketed words over (, [,],)

  All context-free languages arise from some DYCK_n through intersection with regular language + letter substitution

  LSTMs can solve this perfectly (at least in theory) using infinite precision

- Can Transformer correctly classify if an input doesn't belong to these two languages?
- Part 1: hard attention

Idea: construct a pair of inputs that are classified the same. But one is from Parity and the other one is not (same for DYCK_2)

- Part 2: soft attention

  Idea: Prove bounds on cross entropy functions

  Change one input symbol. As input length n -> \infty, then |Output_1 - Output_2| = O(1/n).

☐ Does syntax need to grow on trees? Sources of hierarchical inductive bias in S2S networks

https://www.mitpressjournals.org/doi/pdf/10.1162/tacl_a_00304

- Inductive bias
- What properties of a S2S neural network can give it a hierarchical inductive bias?
- Results?

  Sequential RNNs do not have a hierarchical bias

  - Manipulations that seem like they should impart this bias actually do not. E.g., adding syntactic parse information, multi-task learning

  Tree-structured RNNs do have a hierarchical bias.
- Experiment 1 task 1: English question formation

  Hierarchical rule: Move-main. Linear rule: move-first.

  Generate training set with a CFG of 75 rules.
- Experiment 1 task 2: English tense re-inflection

  Hierarchical rule: agree-subject. Linear rule: agree-recent.
- Model tested:

  Sequential RNNs: SRN, GRU, LSTM

  No attention, location-based attention, content-based attention.

  No model perform well on both tasks -> no hierarchical inductive bias.
- Experiment 2: Tree-RNN

  ON-LSTM performs very similar to sequential RNNs, even though they are designed in a way intended to impart a hierarchical bias.

  Tree-GRU from Chen et al., (2017)
- Experiment 3: non-ambiguous training set

  Some on move-main, other data do move-first. Also, some agree-subject, others agree-recent.
- Experiment 4: tree structure vs tree information

New Setting 1: Tree-RNN but with incorrect tree information

Sequential RNN but with brackets added to indicate the parse

No parse info	*Parse information*	
Sequential RNN	Sequential RNN With bracketed input	*No tree structure*
Tree-RNN with incorrect parses	Tree-RNN	*Tree structure*

- Only the model with both architectural tree structure and parse information has a clear hierarchical bias. Adding brackets or just tree-RNN with any trees themselves are not enough.

- Experiment 5: multi-task learning.

  Unambiguously hierarchical vs ambiguous tasks.

  MTL improves hierarchical generalization but only slightly

☐ What BERT is not: lessons from a new suite of psycholinguistic diagnostics for LMs

https://www.mitpressjournals.org/doi/pdf/10.1162/tacl_a_00298

- Tests: identify cases where

  Humans make good predictions on cloze task

  Predictive responses in the brain ("N400") seem to miss key information that would inform word expectations

- Linguistic capacities tested

  Commonsense / pragmatic inference (CPRAG-102)

  Event knowledge and semantic roles (ROLE-88)

  Negation (NEG-136)

- Analysis types:

  Word prediction acc

  Sensitivity tests

  Qualitative analysis

- Results

  Decent on sensitivity to role reversal and differences within semantic category - but seemingly weaker sensitivity than humans

  Great with hypernyms, determiners, grammaticality

Struggles with challenging inference and event-based prediction

Clear insensitivity to contextual impacts of negation

Category: ML for NLP

☐ Fully contextualized language representations for unsupervised tasks

https://www.aclweb.org/anthology/2020.acl-main.76.pdf

- Motivation: is BERT for unsupervised tasks successful? There is an accuracy vs computational complexity trade-off (e.g., BERT has o(n*n^2). Repeats mask-and-predict n times for each token). Can we have the best of both worlds?
- Propose language autoencoding (see paper for details)

  No repetition: predict every token at once using only its surrounding tokens

  Develop new version of "self-masking" of Transformer: hold "self-unknown" property.
- Evaluation

  As fast as UniLM: O(n^2)

  As good as biLM (in downstream tasks)

  Empirical study: unsupervised STS (semantic textual similarity): outperforms even BERT-base-uncased.

☐ Revisiting the context window for cross-lingual word embeddings

https://www.aclweb.org/anthology/2020.acl-main.94/

- Mapping-based cross-lingual word embeddings.

  Assumption: the two embeddings are structurally similar.

  The context window affects the structure of the embedding space.

  How do window sizes affect the cross-lingual embeddings?
- Data: Wikipedia Comparable Corpus
- Results:

  Larger context windows for both source and target embeddings facilitate the alignment of words.

☐ Improving Massively Multilingual Neural Machine Translation and Zero-Shot Translation

https://www.aclweb.org/anthology/2020.acl-main.148.pdf

- Task: Zero-shot translation

  Append a language tag to source sentence to indicate translation direction.

Potential of ~10k translation directions (most zero-shot)

- Problem: more languages included, worse performance delivered.

  Solution: Large-capacity NMT via deep and language-aware modeling

  Language-aware linear transformation, and Language-aware layer normalization

- Problem: inferior performance of zero-shot translation

  Solution: random online back-translation

  Where to back-translate? Sample a random source language.

- Data: collect OPUS-100, a public massive multilingual dataset.

☐ A generative model for joint NLU and generation https://www.aclweb.org/anthology/2020.acl-main.163/

- Natural language (x) -> (NLU) -> Semantic representation (y). The reverse direction is generation.
- Set up a hidden variable z influencing both x and y.

  Inference

  NLG: $p(x|z,y)$, $z \sim q(z|y)$.

  NLU: $p(y|z,x)$, $z \sim q(z|x)$. Use classifiers at different slots.

- Objectives

  When labels are available: max log $p(x, y)$ by minimizing the VAE's ELBO

  When only unlabeled x or y is available: log $p(x)$ or log $p(y)$ with a cascading NLG + NLU path.

- Datasets

  E2E NLG Dataset (Novikova et al., 2017)

  Weather dataset (Balakrishnan et al., 2019)

- Experiment observations

  Z sampled from two posterior distributions show nice clusterings as semantic composition.

  Higher performance when the unlabeled data are leveraged.

☐ BPE dropout: simple and effective subword regularization https://www.aclweb.org/anthology/2020.acl-main.170/

- Task: subword segmentation
- Background: BPE

  Build merge table and vocabulary. Initialize vocab with characters. Then repeat.

- Previous work: Kudo (2018). Refuse from BPE; complicated
- BPE-dropout:

  At each step: some merges are randomly dropped.

  Produces multiple segmentations of the same word.
- Properties:

  Uses rare subwords more often

  Understands rare tokens better

  Is more robust to mis-spellings

☐ Negative training for neural dialogue response generation

https://www.aclweb.org/anthology/2020.acl-main.185/

- Setting: S2S attention dialogue response models trained from scratch. Use greedy decoding during generation.
- Problem: malicious or boring response.
- Method: negative training: don't say that. Negating the gradient from MLE training (of the malicious or boring responses identified.)
- Show negative training is effective in correcting model's behavior.

☐ MixText: linguistically-informed interpolation of hidden space for semi-supervised text classification

https://www.aclweb.org/anthology/2020.acl-main.194/

- Motivation: utilize limited labeled data for learning (e.g., text classification)
- Prior work on SSL:

  VAE

  Confident predictions on unlabeled data for self-training

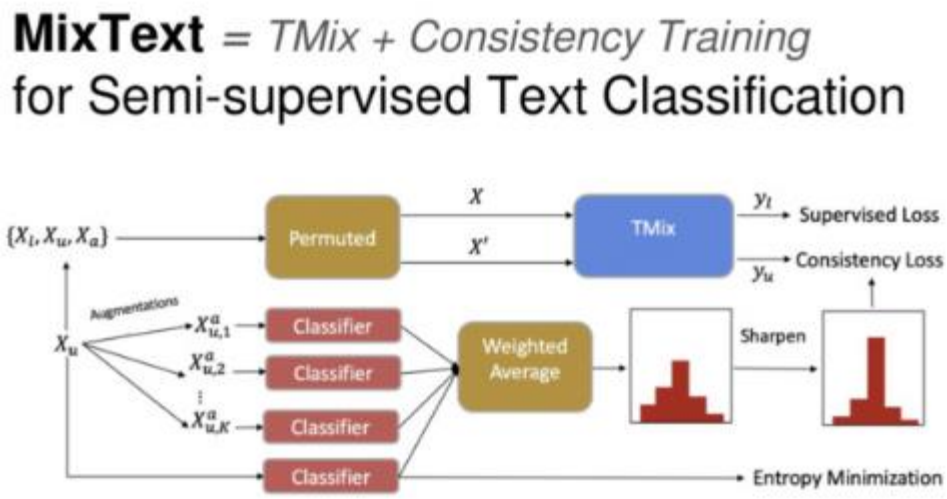
  Consistency training on unlabeled data (e.g., Miyato et al., 2019, 2017; Xie et al., 2019)

  Pre-training on unlabeled data, then fine-tune on labeled data (e.g., BERT)
- Why not enough? Labeled and unlabeled data are treated separately. Models may easily overfit on labeled data while still underfit on the labeled data.
- Propose TextMix:

  Linear interpolations in textual hidden space between different training sentences.

  MixUp (Zhang et al., 2017)

  Interpolate labels.
- Additional question: which layers to mix?
- MixText: Tmix + Consistency Training for Semi-supervised text classification

- 

☐ MobileBERT: a compact task-agnostic BERT for resource-limited devices https://www.aclweb.org/anthology/2020.acl-main.195/

- Model: (see figure in paper)
- Properties of MobileBERT:

  As deep as BERT-large, but are much thinner

  Hard to train a deep & thin network. Design a special teacher network for knowledge transfer (inverted-bottleneck)
- Learning objective: feature map transfer, attention transfer, pre-training distillation
- Training strategies: Auxiliary knowledge transfer (AKT), joint knowledge transfer (JKT), and progressive knowledge transfer (PKT).
- Experiments:

  GLUE results

  SQuAD results

  Ablation study (PD / FMT / AT)
- Take-home message

  It is crucial to keep MobileBERT deep and thin

  Bottleneck / inverted bottleneck structures enable effective layer-wise knowledge transfer

  Progressive knowledge transfer can efficiently train MobileBERT

☐ SMART: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization https://www.aclweb.org/anthology/2020.acl-main.197/

- Propose SMART:

Control model capacity by smoothness-inducing adversarial regularization

Prevent aggressive update by Bregman Proximal Point Optimization

- Smoothness-inducing adversarial regularizer

  If you perturb the input, the output should be also only perturbed by a bit.

  Check the paper for equations.

  The neural network remains similar when input x is slightly perturbed.

- Bregman Proximal Point Optimization

  Prevent aggressive update using trust-region type method.

  Trust-region: search locally in a neighborhood (of model parameters) for each update.

  The neighborhood is induced by Bregman divergence.

  BPP prevents aggressive update. BPP takes task related metric, and can adapt to the information geometry (Raskutti et al., 2015)

- Connection to Mean-Teacher

  The method is also BPP with exponential moving average

- Main results

  Model architecture: RoBERTa_large

  SOTA on GLUE leaderboard

  Ablation study: D_breg and/or R_s

  Evaluate also on MNLI matched Development Set (evaluate the ambiguity)

  Evaluate on Domain Adaptation. Multitask pretraining (MT-DNN) -> SNLI and SciTail

  Robustness on Adversarial NLI (Nie et al., 2019)

☐ Interactive classification by asking informative questions

https://www.aclweb.org/anthology/2020.acl-main.237/

- Task: intent classification

  Natural languages input can be underspecified and ambiguous.

- Methods 1: model the label probability

  Simplifying assumptions: (1) user response depends on the question asked, (2) the model deterministically picks a clarification question given the interaction history.

- Method 2: Question selection

  Max the information gain. Can compute with $p(y|x)$ and $p(r|q, y)$

☐ Pretrained transformers improve out-of-distribution robustness

https://www.aclweb.org/anthology/2020.acl-main.244.pdf

- In reality, test distribution will not match training dist.
- How should models handle?

  Generalize. The ideal case.

  Detect. To alert humans there might be problems.
- Goal: how robust are current NLP models?

  High acc != high robustness. They might use superficial dataset patterns.
- OOD evaluation benchmark by pairing or splitting datasets.

  Sentiment Analysis for restaurant reviews. American -> Chinese, Italian, Japanese.

  Semantic similarity: headlines -> image captions

  RC: CNN -> DailyMail articles

  Textual entailment: Telephone -> Letters

  More datasets in the paper.
- Test on a range of data.
- Main finding:

  Pretrained transformers are more robust.

  Bigger models are not always better.

☐ Curriculum learning for NLU https://www.aclweb.org/anthology/2020.acl-main.542.pdf

- Design a curriculum: distinguish difficult from easy. Arrange from easy to difficult.
- Assign difficulty into several buckets.
- Let the model itself define the difficulties.

  Divide into meta sets. Train a teacher in each set. Use each teacher to score other sets.

  Average n-1 scores as the difficulty score.

  Sampling algorithm: stage I = (1/N bucket 1) U (1/N bucket 2) U … U (1/N bucket i)
- Experiment

  Dataset: GLUE, SQuAD, NewsQA

  Models: BERT base vs + curriculum; BERT large vs + curriculum

  Comparisons to heuristic methods (word rarity, answer length, question length, paragraph length, cross review, etc.) on SQuAD

  Visualization: examples from buckets in SQuAD and SST-2.

☐ The unstoppable rise of computational linguistics in deep learning https://www.aclweb.org/anthology/2020.acl-main.561/

- Historical review.

Early years in finding the nature of language.

Connectionism: Vector spaces formalise distributed representations. Backprop learning is very effective. But some limitations of connectionism gradually arise.

Variable binding. Questions the adequacy of vector spaces.

Systematicity: how can we learn rules which generalise across entities?

- Overview: our understanding of the nature of language from computational linguistics has fundamentally influenced deep learning architectures.
- Inducing features of entities

  Learned vectors replace categorical labels.

  Neural probability replace the occurrence-based counts.
- Inducing relations between entities

  Modeling derivation structures.

  Hand-coded NN model structures reflect the derivation structures.

  Attention induces structures

  Transformers and Systematicity.
- Unbounded generalisation

  Formalising attention's unbounded generalisation.

  Elements in the bag are exchangeable. E.g., in (Jordan 2010)

  Attention-based representations are a nonparametric extension of a vector space.
- Future directions

  What is left to learn?

  E.g., how do we learn the set of entities? The set of levels?

☐ A mixture of h-1 heads is better than h heads https://www.aclweb.org/anthology/2020.acl-main.587/

- MAE: a mixture of attentive experts
- Multihead attention suffers from over-parameterization.

  Previous attempt: prune heads, prune layers
- Our approach:

  activate different heads on different inputs.

  View multihead attention as a mixture of experts, each with constant gates.

  Applicable wherever multihead attention is used.
- Training

Blockwise coordinate descent. Alternate between (1) update the experts by softmax(g_i(x)), (2) update the gating function - weigh the experts by g_i(x)

This should prevent "the rich getting richer"

Motivation: training MoE can generate into the model learning equally weighted experts. BCD is able to make larger jumps in optimization.

The gating function g(.) is MLP

- Experiment evaluations

  Machine translation

  Language modeling. WikiText-103

  Ablation study: mixture of h-2 heads?

☐ Null it out: guarding protected attributes by iterative nullspace projection
https://www.aclweb.org/anthology/2020.acl-main.647/

- Task: controlled representation learning.
- Let classifiers don't condition on e.g., demography.
- Goal: move the gender features into the null space of classifier.
- Method: Iterative Linear Nullspace Projection (INLP):

  Train a classifier predicting Z.

  The classifier's orthogonal projection matrix is a null-space projector P_z.

  Use P_z to remove z-related information from x.

  Accumulate the projections, until Z is not predictable.

- Experiments:

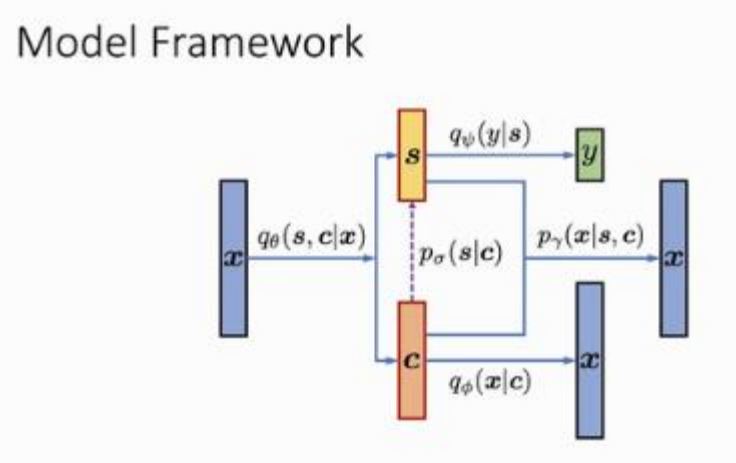
  Linear debiasing word embeddings

  Debiasing contextualized models: biography classification

- Metrics: True Positive Rate gap

☐ Improving disentangled text representation learning with information-theoretic guidance
https://www.aclweb.org/anthology/2020.acl-main.673/

- Model framework:



- Basically the loss terms encourage s to go towards y, encourages c towards c, and pushes s&c apart.
- Use mutual information to derive loss terms to achieve the above purposes.
- Experiments:

  Datasets: Yelp and Personality Captioning

  Metrics to evaluate correctness: ACC, BLEU, S-BLEU, and geometric mean of the previous metrics.

  Additional metrics: style preservation, content preservation, generation quality, and geometric mean

☐ Don't stop pretraining: adapt LMs to domains and tasks

https://www.aclweb.org/anthology/2020.acl-main.740/

- Best paper runner-up
- LMs are training on e.g., trillions of tokens (RoBERTa). What does training on such large tokens give them generalization ability?
- The domains of LM pretraining (e.g., Wikipedia + BookCorpus) vs. Target domain (e.g., Twitter with certain task)
  - Domain granularity matters.
- Explore a few adaptation techniques
  - Domain Adaptive Pretraining (DAPT)
  - Task Adaptive Pretraining (TAPT)
  - Combining DAPT + TAPT
  - Augmenting Data for TAPT
- Domains and tasks to explore

Domain	Pretraining Data	Classification Tasks
Biomedical	S2ORC Papers (7.6B Tokens)	ChemProt RCT
Computer Science	S2ORC Papers (8.1B Tokens)	ACL-ARC SCIERC
Reviews	Amazon Reviews (2.1B Tokens)	Amazon Helpfulness IMDB
News	RealNews Articles (6.7B Tokens)	Hyperpartisan AG News

- DAPT experiments:
  - Doing domain adaptations is very important.
  - Is it about domains or more data? DAPT vs "not DAPT" (adapting to a different domain)
- TAPT experiments:
  - Pretrain LM on data from target task. They do supervised fine-tuning.
  - TAPT can get higher performance with less data (if the task does not have too few resource) than DAPT.
- Can you adapt to any task in a domain? Transfer-TAPT
- Combining DAPT + TAPT?
  - Pretrain on target domain; then target task; then supervised fine-tuning.
  - DAPT+TAPT always outperform DAPT and TAPT
- Augmenting data for TAPT (in low-resource tasks)
  - E.g., human curation; kNN (the VAMPIRE framework; Gururangan et al., 2019)
- Key takeaways: domains may comprise a spectrum. LMs struggle to encode the complexity of a single textual domain, let alone all of language. Important to identify domain-relevant and task-relevant corpora to specialize models.

☐ Efficient contextual representation learning with continuous outputs
https://www.mitpressjournals.org/doi/pdf/10.1162/tacl_a_00289

- Motivation: contextual representation learning is costly.

- Background: softmax layer becomes the speed bottleneck

  Since the W contains 80% of parameters in model
- Proposed method: loss function with a continuous output layer

  L(x, w) = d(x, w), using von mises-fisher loss

  Predict the word embeddings instead of the words.

  Time complexity: O(|V|) -> O(|E|). 80% parameter reduction for ELMo.

  Efficiency improvement of the output layer.
- How to use?

  Open-vocabulary word embeddings. You can adopt to unseen word embeddings.

Category: Other interesting papers

☐ Probabilistically masked LM capable of autoregressive generation in arbitrary word order

https://www.aclweb.org/anthology/2020.acl-main.24/

- Task: text generation in arbitrary order.

  The next token to be predicted could be in any position.

  For example, want to enrich "The quick brown fox jumps over the lazy dog" into a coherent and readable pargraph.
- Related work on non-traditional text generation

  Non-monotonic Sequential Text Generation (Welleck et al., 2019)

  Insertion Trnasformer (Stern et al., 2019)

  Levenshtein Transformer (Gu et al., 2019)
- Propose Probabilistically Masked LM

  Assume the masking prob is a latent variable drawn from a distribution p(r)

  Summing up the probabilities equals collecting the probabilities of permutations of the sentence.
- Use this LM for arbitrarily ordered text generation

  Init with a sequence of blank ([MASK]) tokens

  For each iteration:

  - Specitify a position of the token to be predicted
  - Replace the [MASK] there
- Experiments:

华为诺亚哪吒(NeZha) framework

Data: BookCorpus and Wikipedia

Evaluation: PPL on WikiText, One-Billion Words

Evaluation of latency analysis

Evaluation of language understanding (GLUE test set)

Comparison with XLNet: XLNet is uni-directional, but PMLM is fully bidirectional. XLNet implements two-stream attention, while PMLM employs the traditional simple attention mechanism.

☐ A formal hierarchy of RNN architectures https://www.aclweb.org/anthology/2020.acl-main.43/

- Background:

  Siegelmann and Sontag (1995) showed that "vanilla" (Elman) RNN can simulate any Turing machine, assuming

  - A Turing machine may run for infinite time.
  - The construction also requires infinite precision of the activations.

  In practice, not all RNNs are created equal (Weiss et al., 2018) given tasks. LSTMs can implement a counting mechanism, while GRUs cannot.

  Some RNNs simulate weighted finite automation (WFA), (Peng et al., 2018). QRNN is a rationally recurrent RNN inspired by LSTM. Conjecture: LSTMs are not rationally recurrent.

- Develop hierarchy in two properties:

  Space complexity

  Rational recurrence.

	Rationally recurrent	Non-rationally recurrent
O(n)	RR-complete K-WFAs	Memory networks Stack RNN
O(log n)	Counter-blind QRNN	Counter-aware LSTM
O(1)	Finite state GRU, RNN, CNN	N/A

- Saturated RNNs (Merrill, 2019)

A simplied model for discussing practical capacity of RNN architectures.

Easy to prove that saturated LSTMs can count; saturated GRUs cannot.

- Results

  Proving "irrational" recurrence. Prove: s-LSTMs are not rationally recurrent.

  S-GRUs and Elman s-RNNs are finite state. This indicates that they are rationally recurrent (whereas s-LSTMs were not)

- How does the hierarchy affect language recognition abilities?

  S-LSTM (ok) vs s-QRNN (nope) with a^nb^n.

  Use a WFA on a^nb^n. This works. This shows that the weakness of s-QRNN doesn't come with its rationality!

  Increasing the capacity: s-LSTM and s-QRNN both work.

  "Suffix attack" a^nb^n S*. S-LSTM ok, but s-QRNN (not matter any layer of decoder) do not. This is a fundamental weakness of the s-QRNN compared to the s-LSTM.

- Experiments:

  Recognizing a^nb^n

  With a suffix a^n b^n S*

- Conclusion:

  Develop hierarchy of saturated RNNs in terms of rational recurrence and space complexity.

  Open question: what does saturated theory predict for transformers?

☐ A girl has a name: detecting authorship obfuscation https://www.aclweb.org/anthology/2020.acl-main.203/

- Task: authorship obfuscation
- Adversary's intuition: text transformations in obfuscation increases un-smoothness.

  Intuition analysis: the average occurrence probability decreased.

- Obfusacaiton detection pipeline

  Word likelihood extraction using LM

  Feature representation using binning based features and VGG19 (extracting image-based features).

  Classification model: KNN, ANN, SVM lines RFC, GNB

- Evaluation:

Obfuscators: SN-PAN16 (PEN-CLEF 2016), DS-PAN17 (PSN-CLEF 2017) and Mutant-X (PETS 2019)

Baseline obfuscation detectors: chatacter trigrams (EACL 2012), writeprints (IEEE S&P 2012), GLTR (ACL 2019)

☐ Examining citations of NLP literature https://www.aclweb.org/anthology/2020.acl-main.464/

- This work looks back at the published paper to identify broad trends in their impact on subsequent scholarly work.
- Metrics of research impact (on subsequent scholarly work)

  Often derived from citations

  Citations do not always reflect quality or importance.
- This work:

  Extracted and aligned information from ACL anthology, Google Scholar to create a dataset of citations

  Also use: NLP Scholar paper (Mohammad., LREC 2020)
- Q1-2: EDA about the AA citations

  1.2 million citations as of June 2019

  Do not include AA papers after 2017.
- Q: How well cited are papers from different venues?

  CompLing journal > Top-tier > TACL (pretty new) > workshops ~ other conferences > others
- Q: How well cited are long and short papers? Less than long papers.
- Q: What percentage of papers are cited more than 10 times? ~56%. 6.4% are cited 0 times.
- http://saifmohammad.com/WebPages/nlpscholar.html

☐ NER as dependency parsing https://www.aclweb.org/anthology/2020.acl-main.577/

- Background: flat NER (no nested entities) and nested NER
- Network architectures see the paper
- Results on both nested and flat NER datasets

☐ Let me choose: from verbal context to font selection https://www.aclweb.org/anthology/2020.acl-main.762/

- Goal: can we recommend the fonts by the textual content?
- Subjective characteristics

  There is no strict or universally-accepted rule for choosing fonts

There seems to be enough agreement among human opinions to build reasonably effective models of font properties.

- Collect a dataset. "Short Text Font Dataset", allowing end-to-end trianing.

☐ Smart TODO: automatic generation of TODO items from emails

https://www.aclweb.org/anthology/2020.acl-main.767/

- Data:

  Email corpus: Avocado research email collection

  Identify task sentences: have a "commitment classifier"

  HitApp for annotating human judgements.

- Framework

  Stage 1: (extractive) select 'helpful' sentences

  Stage 2: (abstractive) Seq2seq with copy mechanism

  Generate TODO items.

Tutorial T1: Interpretability and Analysis in neural NLP

Slides

Why should we care about interpretability?

- Deep learning approaches are in a lot of "trial-and-error". Better understanding -> better systems.
- Accountability, trust, and bias in ML. Better understanding -> more accountable systems.
- NNs aid the scientific study of language (e.g., models of human language acquisition, and human languages processing). Better understanding -> more interpretable models

Tools to analyze interpretability

- Structural analysis (Yonatan)
- Behavioral analysis (Ellie)
- Interaction + visualization (Sebastian)
- Other methods

Structural analysis

- Diagnostic classifiers (e.g., Alain and Bengio)
    - The inputs could be machine translation model representations (Belinkov and Glass)
    - Edge probing (Tenney et al)
    - Probe the linguistic structures (e.g., syntax, Hewitt and Manning 2019)
- Needs some control mechanism to probing
    - Baselines: static word embeddings (Belinkov 2017) or random features (Zhang and Bowman 2018)
    - Skylines: SOTA on the task, or a full-fledged model.
    - Hewitt and Liang 2019: proposed a control task. Accuracy vs selectivity tradeoff. Should use the simplest model possible.
    - Pimentel et al., 2020: criticized the probing. Proposed to use the highest-performing model. Analyzed with information theory. (Voita and Titov 2020) also analyzed from information theory.
- Causal probes
    - Probes found correlation. They do not indicate causations.
    - An alternative direction: intervene in the model representations to discover causal effects on prediction.
    - (Giulianelli et al., 2018): Train classifier to predict number from LSTM states. Backprop classifier gradients to change LSTM states, so classifiers can predict number better.
        - Found this intervention improved probing accuracy, but had little effect on LM.
        - But had strong effect on an LM agreement test.
    - (Bau et al., 2019) studies the role of individual neurons in MT.
        - Identify important neurons and intervene in their behavior. Then change their activations based on activation statistics over a corpus.

        - Successfully influence the translation of tense from past to present, but less successful with influencing gender and number.
    - (Vig et al., 2020) use causal mediation analysis to interpret gender bias in LMs.
        - Define interventions via text edit operations and measure counterfactual outcomes.
        - Calculate direct and indirect effects, with mediators as neurons and attention heads.

Behavioral Analysis

Overall

- We usually measure the average-case performance on a test set. This could hide the fact that models perform poorly on "the tail".
- Challenge sets (test suites) aim to cover specific, diverse phenomena. They have systematicity, exhaustivity, and have control over data. They facilitate fine-grained analysis of model performance.
- Key idea: design experiments that allow us to make inferences about the model's representation based on the model's behavior.
- A claim: how a model works should be consistent with *both* physiological (on brain) and behavioral data (on AI models).

Goods and limitations

- Good:
    - Theory agnostic. Avoids prescriptivism.
    - Avoid "squinting at the data". Objective criteria for what counts as "representing" a thing.
    - Interfaces well with linguistics and other fields.
    - Practical -- not whether the model represents a feature, but whether it uses it in the right way.

- Limitations:
    - Whether the model or the data to blame?
    - Tells us that a model did / didn't solve a task, but didn't tell us how.
    - Hard to design.
    - Risk of overfitting to the challenge sets.

Experimental designs: tightly controlled

- Minimal pairs / counterfactuals
    - Gender bias: Rudinger et al., 2018
    - Subject-verb agreements: Marvin and Linzen 2018
    - Veridicality: White et al., 2018

Experimental designs: loosely controlled

- Average over sets with vs without property of interest)
    - FraCas: Cooper et al., 1996
    - GLUE diagnostic set
    - Diverse NLI corpus (DNC) Poliak et al., 2018

Experimental designs: adversarial examples

- Design data sets
    - Jia and Liang 2017
    - Nie et al., 2019: Adversarial NLI

Construction methods

- Source of data
- Example / label generation
- Manual, semi-automatic, fully automatic

Interaction + Visualization

Why do we want visualization systems?

Categorizing research in visualization

Hands-on with a simple attention visualization

Future challenges and limitations

- Interaction + visualization matters at every step: understanding the problem; forming hypotheses; testing the hypotheses.

Other Methods

- Adversarial examples
- Generation explanations
- Formal languages as models of language

Test of Time Award Papers

☐ Centering: A framework for modeling the local coherence of discourse (Barbara Grosz, Aravind Joshi, Scott Weinstein, 1995) https://www.aclweb.org/anthology/J95-2003.pdf

☐ Unsupervised Word Sense Disambiguation Rivaling Supervised Methods (David Yarowsky, 1995) https://www.aclweb.org/anthology/P95-1026.pdf

☐ Distributional Memory: A General Framework for Corpus-Based Semantics (Marco Baroni, Alessandro Lenci, 2010) https://www.aclweb.org/anthology/J10-4006.pdf

☐ Words Representations: A simple and General Method for Semi-Supervised Learning (Joseph Turian, Lev-Arie Ratinov, Yoshua Bengio, 2010) https://www.aclweb.org/anthology/P10-1040.pdf