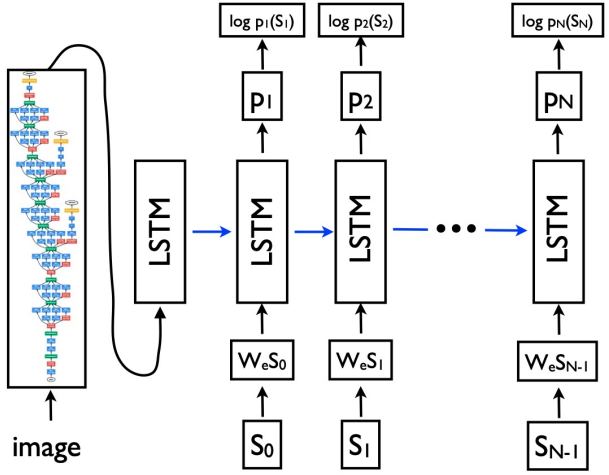
LT2318 H21 AICS: Visual Representations, Caption Evaluation, Decoding and Multi-Modal Transformers

Nikolai Ilinykh[1]
[1]Department of Philosophy, Linguistics and Theory of Science
Centre for Linguistic Theory and Studies in Probability (CLASP)
University of Gothenburg, Sweden
{name.surname}@gu.se

Presented at, December 8, 2021

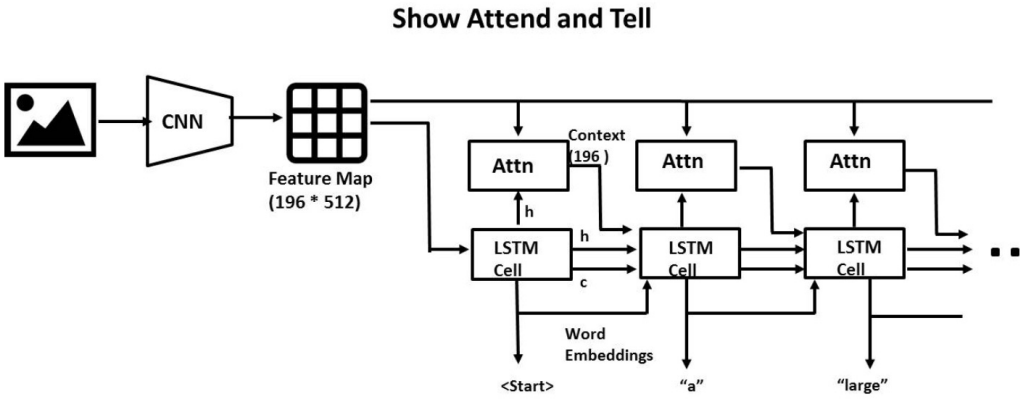
Recap I



1

[1] https://openaccess.thecvf.com/content_cvpr_2015/papers/Vinyals_Show_and_Tell_2015_CVPR_paper.pdf

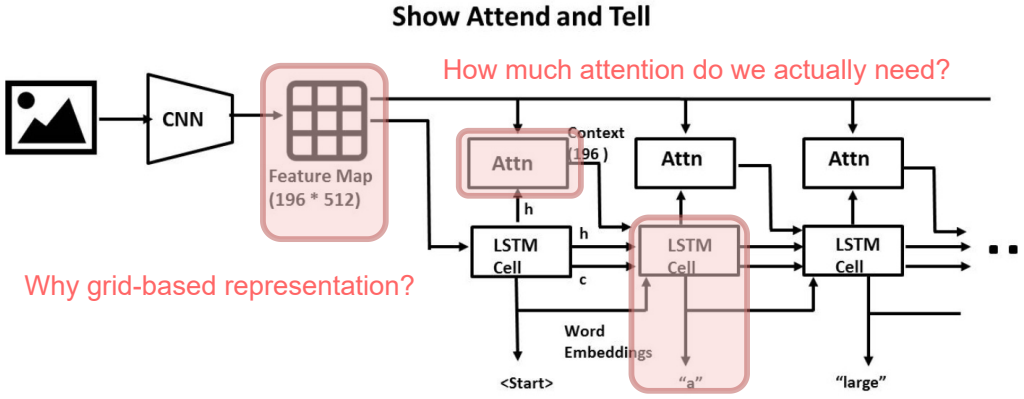
Recap II



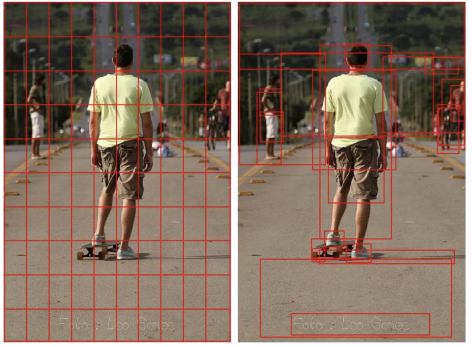
2

[2] https://proceedings.mlr.press/v37/xuc15.html

Challenges


Show Attend and Tell
How much attention do we actually need?
CNN
Feature Map
(196 * 512)
Attn
Context
(196)
Attn
Attn
h
h
c
LSTM
Cell
LSTM
Cell
LSTM
Cell
Word
Embeddings
<Start>
“a”
“large”
Why grid-based representation?
Can we have more control over the generation?

Better Image Representations



- lack of semantic information in grid cells
- humans do not split image into grid cells, they identify objects
- what happens if we change the way we represent the image and use *semantically informed* representations, e.g. bounding boxes of objects?[3]

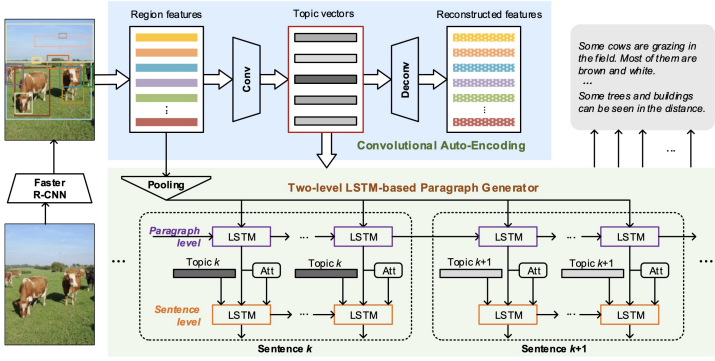
[3] https://openaccess.thecvf.com/content_cvpr_2018/papers/Anderson_Bottom-Up_and_Top-Down_CVPR_2018_paper.pdf

When does semantic information become useful?

- grid cells could be more useful for image classification
- language often diffuses attention of the model across many parts, which is more useful for image captioning
- these findings hold for multi-modal transformers[4]

[4] https://openaccess.thecvf.com/content/ICCV2021/papers/Caron_Emerging_Properties_in_Self-Supervised_Vision_Transformers_ICCV_2021_paper.pdf

Extra: Convolutional Auto-Encoding



- auto-encoding allows to extract better, more dense and compressed visual features
- often useful for modelling longer texts: each sentence has a topic, how do such abstractions realise themselves in images?
- topic distillation with convolutional auto-encoding[5]

[5] https://www.ijcai.org/proceedings/2019/0132.pdf

So, how do we extract bounding boxes?

- we use Faster-RCNN networks to extract bounding boxes[6]
- these networks are designed to identify objects in images and assign classes to them (incl. class attributes)
- the object detector is trained on large annotated datasets, e.g. Visual Genome
- the extraction happens in **two** stages:
  - (i) RPN selects bounding boxes which match the most with the ground-truth bounding boxes
  - (ii) feature maps are extracted from regions of interest (RoI), and these features are passed through the CNN to predict class-specific labels and attributes
- for more information:
  https://panderson.me/images/cvpr18_UpDown_poster.pdf
- the only difference is that now we attend over the detected objects and not grid cells

[6] https://papers.nips.cc/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf

Evaluation Metrics

Evaluation Metrics: BLEU[7]

- the primary task is to compare n-grams of the candidate with the n-grams of the reference translation and count the number of matches; these matches are position-independent; the more the matches, the better the candidate translation is
- the counting of matching n-grams is modified because models tend to "overgenerate" words, which makes precision calculation inaccurate
- the final score is computed over the whole corpus by calculating the geometric mean

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} Count_{clip}(n\text{-}gram)}{\sum_{C' \in \{Candidates\}} \sum_{n\text{-}gram' \in C'} Count(n\text{-}gram')}.$$

[7] https://aclanthology.org/P02-1040/

Evaluation Metrics: ROUGE[8]

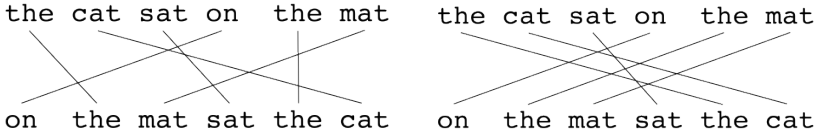
- while BLEU measures precision (how many words from the candidate appear in the reference set), ROUGE measures recall (how many words from the reference set appear in the candidate)
- ROUGE is recall-oriented metric, it computes how much of the reference occurs in the candidate sentence
- the metric is computed on the level of n-grams as well

$$\text{ROUGE-N} = \frac{\sum_{S \in \{ReferemceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)} \quad (1)$$

[8] https://aclanthology.org/W04-1013/

Evaluation Metrics: METEOR[9]

- METEOR is based on explicit word-to-word matching, but it can also match synonyms
- it computes the F-score, weights recall higher than precision
- the metric first computes the alignment between two sentences: every n-gram in the candidate must map to zero or one n-gram in the reference
- we want the lowest number of alignments because it would mean that both candidate and reference match perfectly: it reflects whether the word order is captured as well



[9] https://aclanthology.org/W05-0909/

Evaluation Metrics: METEOR II

- compute precision, recall and f-score
- f-score weights recall higher, e.g., the system is encouraged to cover as many words in the reference as it can

$$P = \frac{m}{w_t} \quad R = \frac{m}{w_r} \quad F_{mean} = \frac{10PR}{R + 9P}$$

Evaluation Metrics: METEOR III

- longer n-gram matches are used to compute a penalty for the alignment
- the more mapping there are that are *not* adjacent in the reference and the candidate sentence, the higher the penalty will be
- thus, METEOR is highly affected by the alignment method
- the final score is calculated over a whole corpus and values are aggregated

$$p = 0.5\left(\frac{c}{u_m}\right)^3 \qquad M = F_{mean}(1-p)$$

Evaluation Metrics: CIDER[10]

- CIDER is, perhaps, one of the few metrics which correlate a lot with human judgements
- it uses tf-idf metric to aggregate statistics for n-grams across the dataset
- intuitively, words present across all captions are less informative; thus, they should be given less weight in the evaluation of similarity

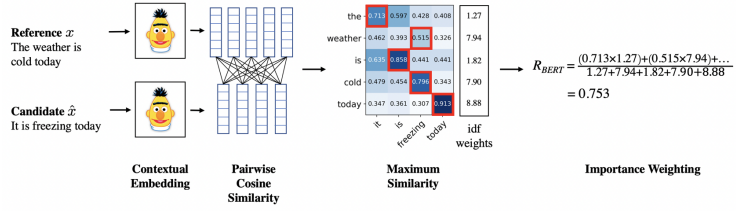
$$g_k(s_{ij}) = \frac{h_k(s_{ij})}{\sum_{\omega_l \in \Omega} h_l(s_{ij})} \log\left(\frac{|I|}{\sum_{I_p \in I} \min(1, \sum_q h_k(s_{pq}))}\right)$$

$$\text{CIDEr}_n(c_i, S_i) = \frac{1}{m} \sum_j \frac{\boldsymbol{g^n}(c_i) \cdot \boldsymbol{g^n}(s_{ij})}{\|\boldsymbol{g^n}(c_i)\| \|\boldsymbol{g^n}(s_{ij})\|}$$

[10] https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Vedantam_CIDEr_Consensus-Based_Image_2015_CVPR_paper.pdf

Some other evaluation metrics worth looking at

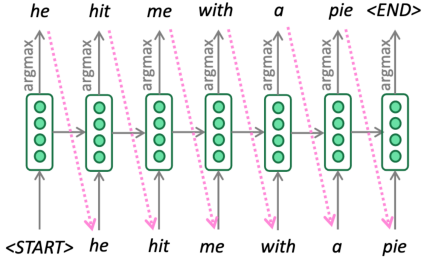
- SPICE[11]: uses semantic propositional content, parses image into a scene graph and can answer questions like "can caption generators count?" or "which caption generator knows more about colors"
- BERTScore[12]: computes cosine similarity using contextual representations (word embeddings)



[11] https://panderson.me/images/SPICE.pdf
[12] https://openreview.net/pdf?id=SkeHuCVFDr

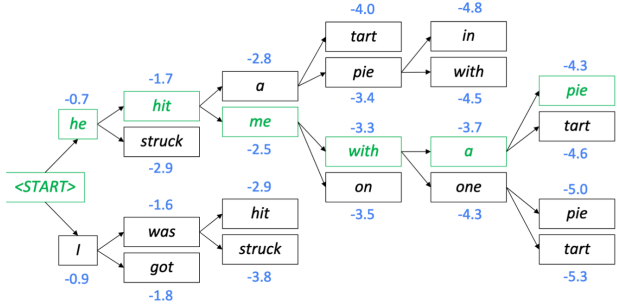
Decoding: from probabilities to words

- at every step during caption generation word by word we get the probability over the whole vocabulary: *how do we choose the predicted word from all these probabilities?*
- practically, decoding is the most important step because bad decoding can mess up even an excellent model
- standard decoding is **greedy**: on each step, take the most probable word (e.g., argmax)
- use that as the next word, and feed it as input on the next step
- keep going until you produce the END token or reach some maximum length
- due to lack of backtracking, the output can be repetitive and unnatural

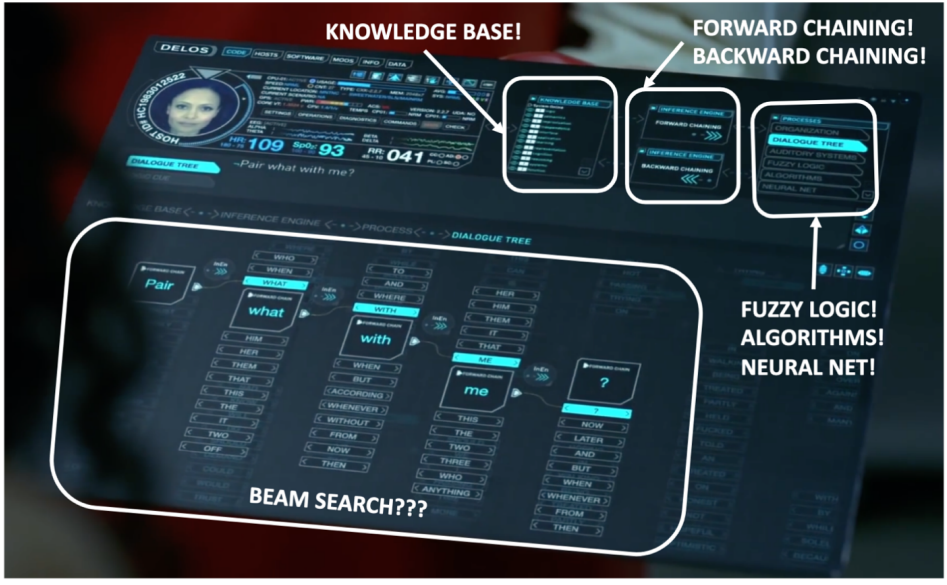


Decoding: Beam Search

- a search algorithm which aims to find a high-probability sequence (not necessarily the optimal sequence, though) by tracking multiple possible sequences at once
- **core idea**: on each step of decoding, keep track of the **k** most probable partial sequences (called hypotheses), **k** is the beam size
- after you reach some stopping criterion, choose the sequence with the highest probability



Decoding: Beam Search in Westworld?!



Decoding: Beam Search II

- smaller **k** is identical to greedy search
- larger **k** means you are considering more hypotheses:
  - it is computationally more expensive
  - large-k beam search tends to produce too-short translations, which decreases evaluation scores[13]
  - in open-ended tasks (e.g., chit-chat dialogue), large-k beam can make output more generic

[13] https://arxiv.org/abs/1706.03872

Decoding: Beam Search III

Human chit-chat partner

Beam size	Model response
1	*I love to eat healthy and eat healthy*
2	*That is a good thing to have*
3	*I am a nurse so I do not eat raw food*
4	*I am a nurse so I am a nurse*
5	*Do you have any hobbies?*
6	*What do you do for a living?*
7	*What do you do for a living?*
8	*What do you do for a living?*

Low beam size:
More on-topic but nonsensical;
bad English

High beam size:
Converges to safe, "correct" response, but it's generic and less relevant

Decoding: Sampling-based decoding

- pure sampling
  - on each step, randomly sample from the probability distribution to get your next word
  - pretty much like greedy decoding, but only sample instead of argmax
- top-n sampling
  - on each step, randomly sample from the probability distribution, restricted to just the top-n most probable words
  - similar to pure sampling, but truncates the probability distribution
  - n=1 is greedy search, n=V is a pure sampling
  - increase n to get more diversity in your output (riskier, less safe, but often more natural)
  - decrease n to get more generic and safe outputs (often more uninteresting)
- **softmax temperature** is a method to control for diversity; you apply softmax to your scores, and while doing so, you normalise your softmax scores by some temperature value; higher temperature makes probability distribution more uniform (probability is spread across vocabulary), while lower temperature makes it more spike (less diverse output, the probability is concentrated on top words)

And now, let's look at some examples of different decoding algorithms!

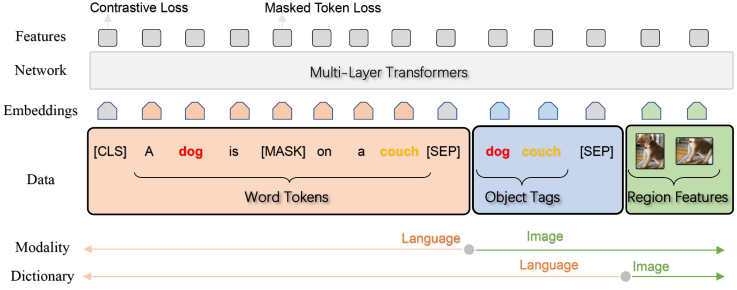
Decoding: Summary

- greedy search is simple, gives low quality output
- beam search searches for high-probability output
  - does better than greedy, but if beam size is too high, can return high-probability but unsuitable output (e.g., genertic, short)
- sampling methods are a way to get more diversity and randomness
  - good for open-ended / creative generation (poetry, stories)
  - top-n sampling allows you to control diversity
  - nucleus sampling[14] is currently possibly the most promising methods which combines the best from both worlds (accuracy and diversity)

[14] https://openreview.net/pdf?id=rygGQyrFvH

Multi-Modal Transformers I: Uni-Stream

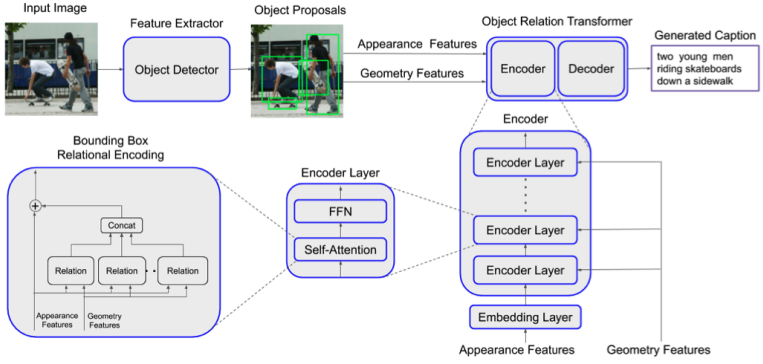
- typically, MMTs are divided into two groups: uni-stream and two-stream (multi-stream) architectures
- uni-stream architectures function similar to BERT, their role is to encode knowledge and use it for tasks like classification
  - examples: UNITER[15], OSCAR[16], VL-BERT[17]
  - uni-stream transformers are *typically* not suited for auto-regressive tasks, e.g. text generation; however, there are tricks to make them generative as well[18]



[15] https://arxiv.org/pdf/1909.11740.pdf
[16] https://arxiv.org/pdf/2004.06165.pdf
[17] https://arxiv.org/pdf/1908.08530.pdf
[18] https://aclanthology.org/2020.inlg-1.39.pdf

Multi-Modal Transformers II: Two-Stream

- two-stream architecture are similar to GPT-2 family od models, the separately attend to each modality and learn to fused them
- these models are mostly used for generation tasks and/or research on information fusion
  - examples: LXMERT[19], Relational Transformer[20]

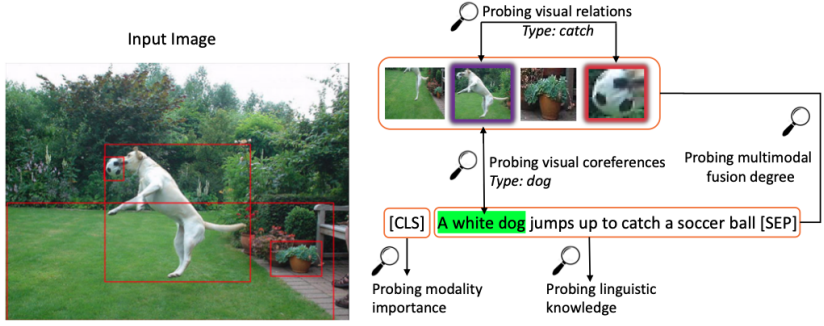


[19] https://arxiv.org/pdf/1908.07490.pdf

[20] https://proceedings.neurips.cc/paper/2019/file/680390c55bbd9ce416d1d69a9ab4760d-Paper.pdf

Areas to explore with multi-modal transformers I

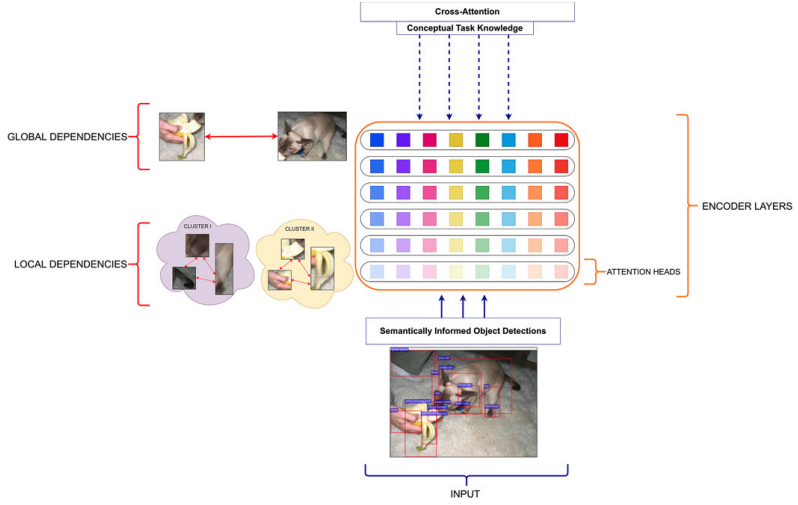
- different probing tasks: number of layers vs multi-modal fusion, the role of each modality in final predictions, which knowledge is encoded in pre-trained models, what can be learned intra-modally (visual relations), differences in captured knowledge with uni-modal architectures[21]



[21] https://arxiv.org/pdf/2005.07310.pdf

Areas to explore with multi-modal transformers II

- the effects of language on vision and vica versa, cross-modal input ablations[22],[23]



[22] https://www.frontiersin.org/articles/10.3389/frai.2021.767971/full

[23] https://arxiv.org/pdf/2109.04448.pdf

Areas to explore with multi-modal transformers III

- spatial relations and grounding in different modalities
- co-reference resolution and learning to align objects with their referential expressions
- a variety of methods to interpret models: for example, inspecting attention values: https://arxiv.org/pdf/1906.04284.pdf