CS11-747 Neural Networks for NLP

Advanced Search Algorithms

Graham Neubig
https://phontron.com/class/nn4nlp2020/

Carnegie Mellon University
Language Technologies Institute

(Some Slides by Daniel Clothiaux)

The Generation Problem

- We have a model of P(Y|X), how do we use it to generate a sentence?

- Two methods:

  - **Sampling:** Try to generate a *random* sentence according to the probability distribution.

  - **Argmax:** Try to generate the sentence with the *highest* probability.

Which to Use?

- We want the best possible single output
  → **Search**

- We want to observe multiple outputs according to the probability distribution
  → **Sampling**

- We want to generate diverse outputs so that we are not boring
  → **Sampling? Search?**

Sampling

Ancestral Sampling

- **Randomly generate** words one-by-one.

```
while $y_{j-1}$ != “</s>”:
  $y_j \sim P(y_j \mid X, y_1, \ldots, y_{j-1})$
```

- An **exact method** for sampling from P(X), no further work needed.

- Any other sampling method is *not* an appropriate way of visualizing/understanding the underlying distribution.

Search Basics

Why do we Search?

- We want to find the **best** output
- What is "best"?
    - The **most accurate** output

$$\hat{Y} = \underset{\tilde{Y}}{\operatorname{argmin}} \operatorname{error}(Y, \tilde{Y})$$

→ impossible! we don't know the reference

- The **most probable** output according to the model

$$\hat{Y} = \underset{\tilde{Y}}{\operatorname{argmax}} P(\tilde{Y}|X)$$

→ simple, but not necessarily tied to accuracy

- The output with the lowest **Bayes risk**

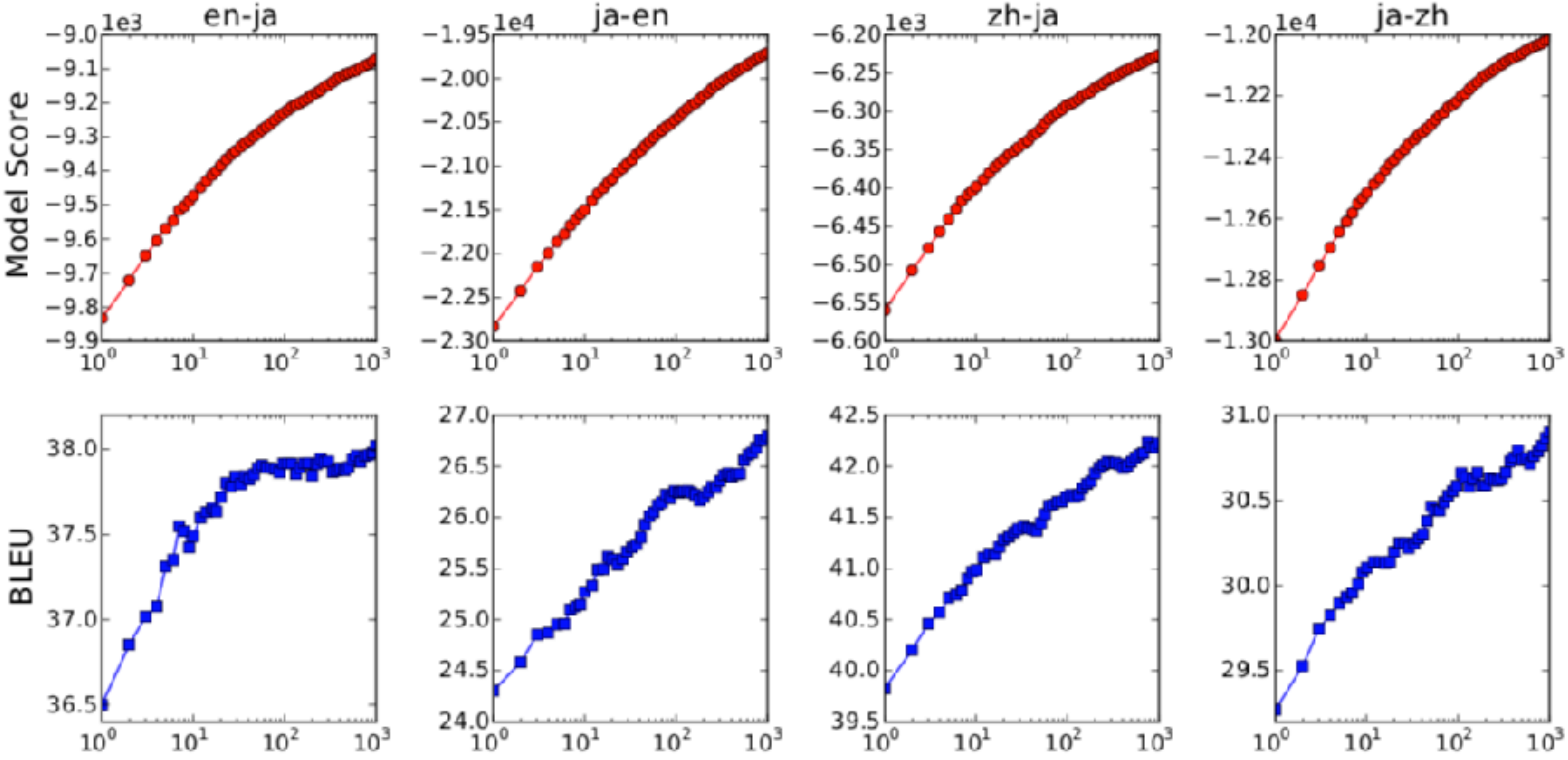
$$\hat{Y} = \underset{\tilde{Y}}{\operatorname{argmin}} \sum_{Y'} P(Y'|X) \operatorname{error}(Y', \tilde{Y})$$

→ which output *looks like* it has the lowest error?

Search Errors, Model Errors

(example from Neubig (2015))

- **Search error:** the search algorithm *fails* to find an output that optimizes its search criterion
- **Model error:** the output that optimizes the search criterion *does not optimize accuracy*



Searching Probable Outputs

Greedy Search

- One by one, pick the single highest-probability word

```
while $y_{j-1}$ != “</s>”:
  $y_j = \text{argmax}\ P(y_j \mid X, y_1, \ldots, y_{j-1})$
```

- **Not exact, real problems:**
  - Will often generate the “easy” words first
  - Will prefer multiple common words to one rare word

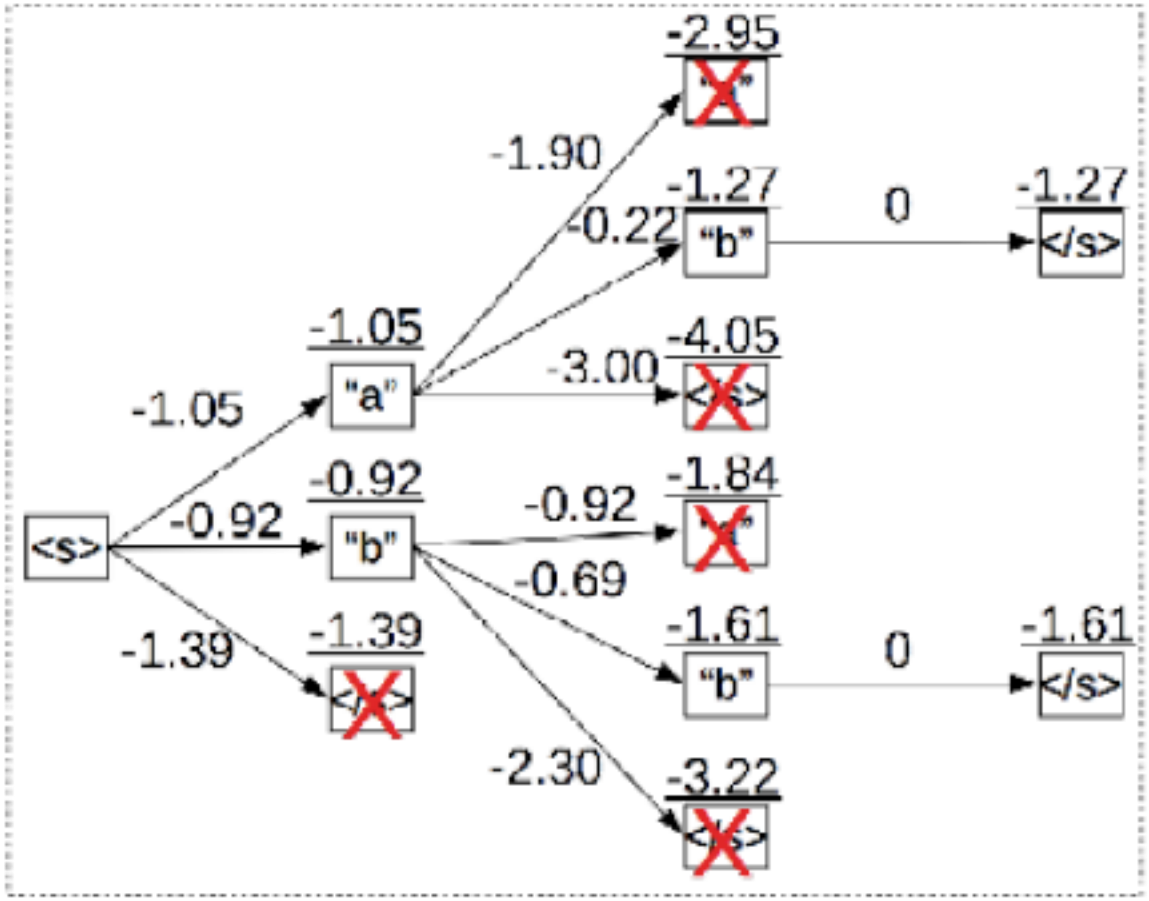
Why will this Help

Next word	P(next word)
Pittsburgh	0.4
New York	0.3
New Jersey	0.25
Other	0.05

Beam Search

- Instead of picking the highest probability/score, maintain multiple paths

- At each time step
  - Expand each path
  - Choose a subset paths from the expanded set



Basic Pruning Methods
(Steinbiss et al. 1994)

- How to select which paths to keep expanding?
- **Histogram Pruning:** Keep exactly *k* hypotheses at every time step
- **Score Threshold Pruning:** Keep all hypotheses where score is within a threshold α of best score s_1

  $s_n + \alpha > s_1$
- **Probability Mass Pruning:** Keep all hypotheses up until probability mass α

What beam size should I use?

- Larger beam sizes will be slower
- May not give better results due to model errors
  - Sometimes result in shorter sequences
  - May favor high-frequency words
- Mostly done empirically -> experiment (range of 5-100 for histogram?)

Problems w/ Disparate Search Difficulty

- Sometimes need to cover specific content, some easy some hard

I	saw	the escarpment
watashi	*mita*	*dangai? zeppeki?* *kyushamen? iwa?*

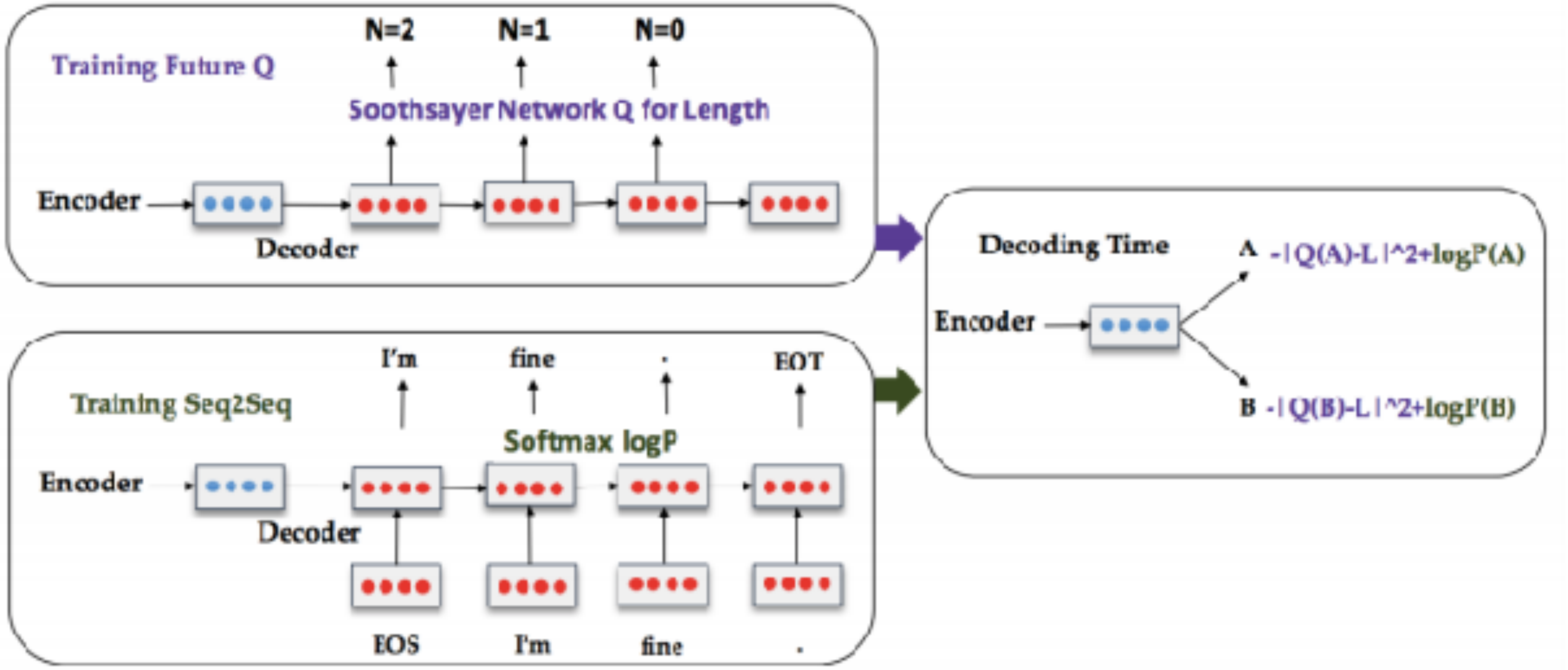
- Can cause the search algorithm to select the easy thing first, then hard thing later

watashi wa dangai wo mita
(I saw the escarpment)

watashi ga mita dangai
(the escarpment I saw)

Future Cost

- also predict how hard it will be to process as-of-yet-unprocessed words, and search for maximum of sum **f(n) = g(n) + h(n)**
  - g(n): cost to current point
  - h(n): estimated cost to goal
- See Koehn (2010 Chapter 6), or Li et al. (2017) for a neural approximation

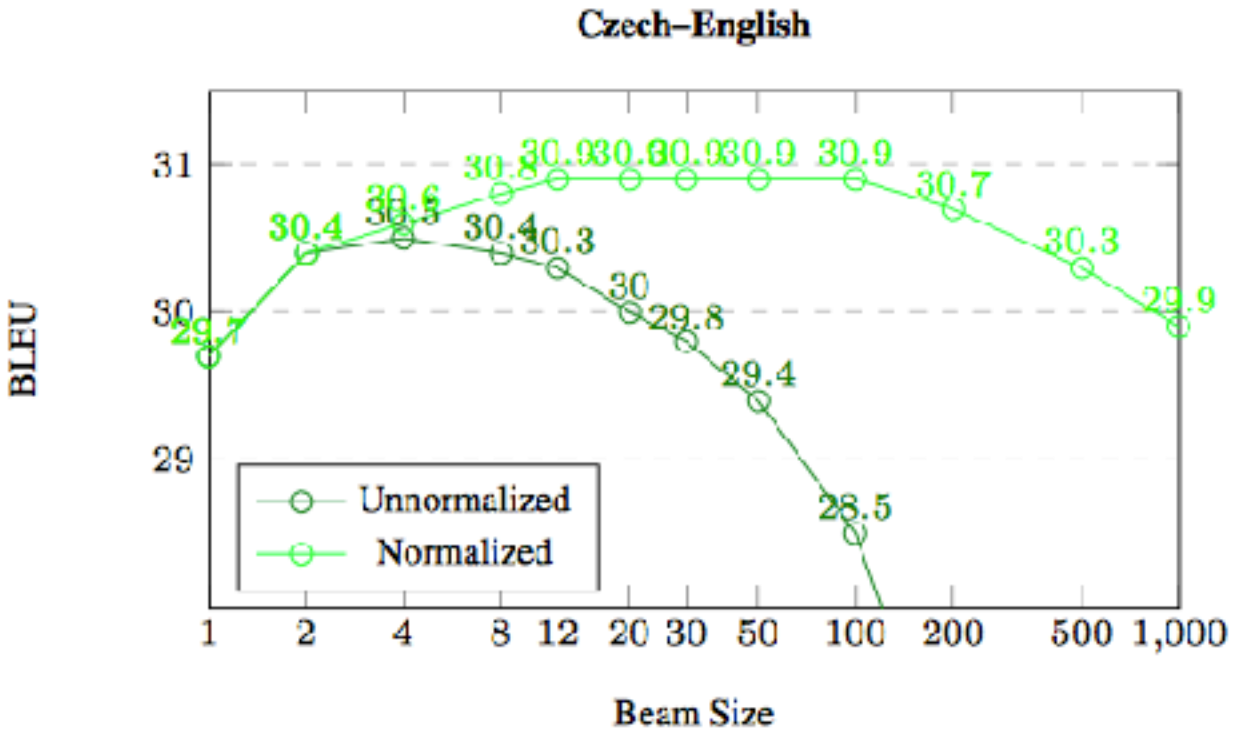


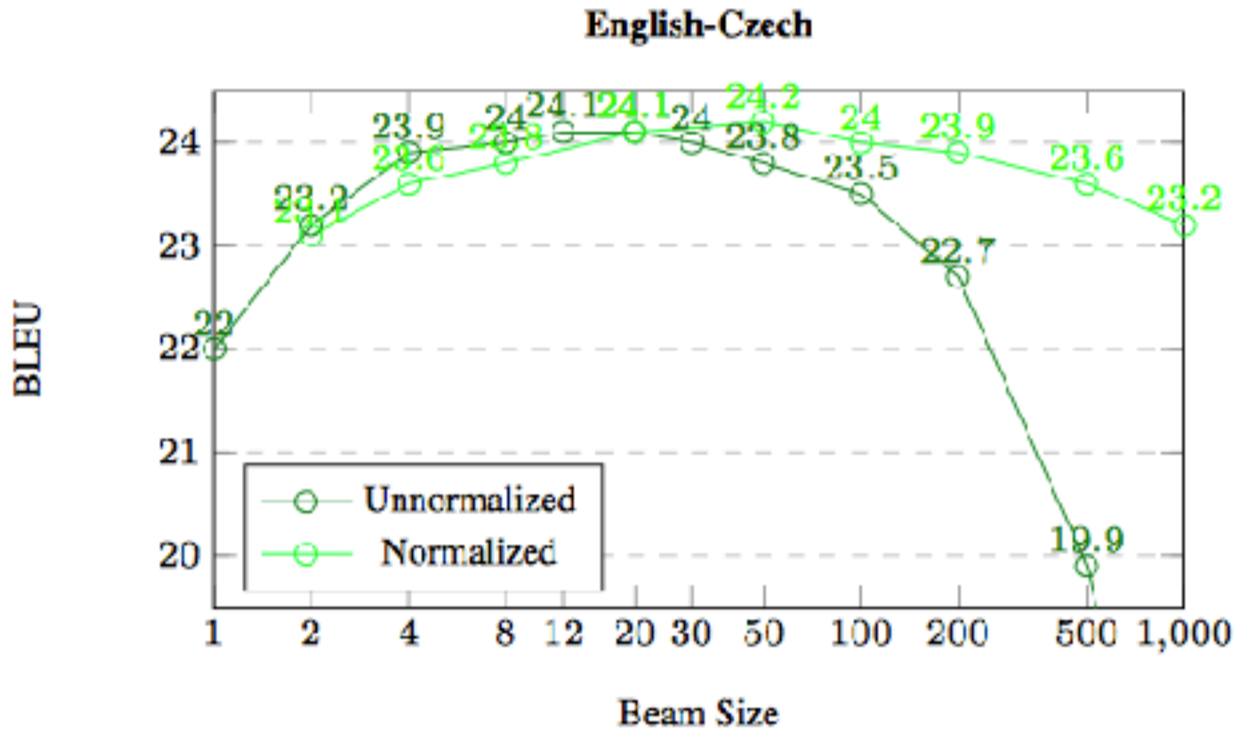
Search and Problems with Modeling

Better Search can Hurt Results!

(Koehn and Knowles 2017)

- Better search (=better model score) can result in worse BLEU score!





- Why? Model errors!

How to Fix Model Errors?

- **Train** the model to maximize accuracy/minimize risk (best!, covered previously)

- **Change the decision rule** to minimize risk (best!)

- **Heuristically modify** the model score post-hoc (OK)

- **Hobble the search algorithm** so it makes more search errors, but the kind of errors you want (meh)

Minimum Bayes Risk Decoding

Basic Concept

- We want outputs that look "safe" given all the other high-probability outputs

Higher in Aggregate

p=0.3	I don't know
p=0.2	My name is Graham
p=0.18	My name is Graham Neubig
p=0.17	My name is Neubig
...	

- Operationalized as searching for hypothesis that minimizes risk

$$\hat{Y} = \operatorname*{argmin}_{\tilde{Y}} \sum_{Y'} P(Y'|X) \text{error}(Y', \tilde{Y})$$

Minimum Bayes Risk Reranking

- Create n-best list
- Create error matrix and probability vector

$$E_{i,j} = \mathrm{error}(Y_i, Y_j) \quad \boldsymbol{p}_i = P(Y_i|X)$$

- Multiply to get the risk

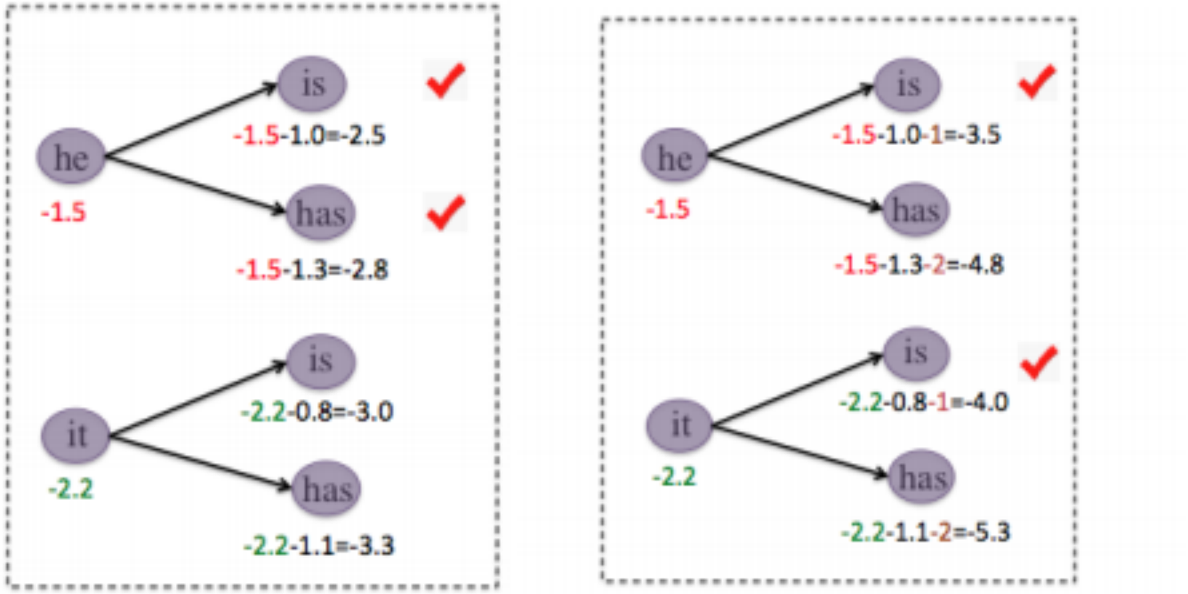
$$\boldsymbol{r} = E\boldsymbol{p}$$

- Find the element with lowest risk

Improving Diversity in top N Choices

(Li et al., 2016)

- Entries in the beam can be very similar
- Improving the diversity of the top N list can help
- Score using source->target and target-> source translation models, language model



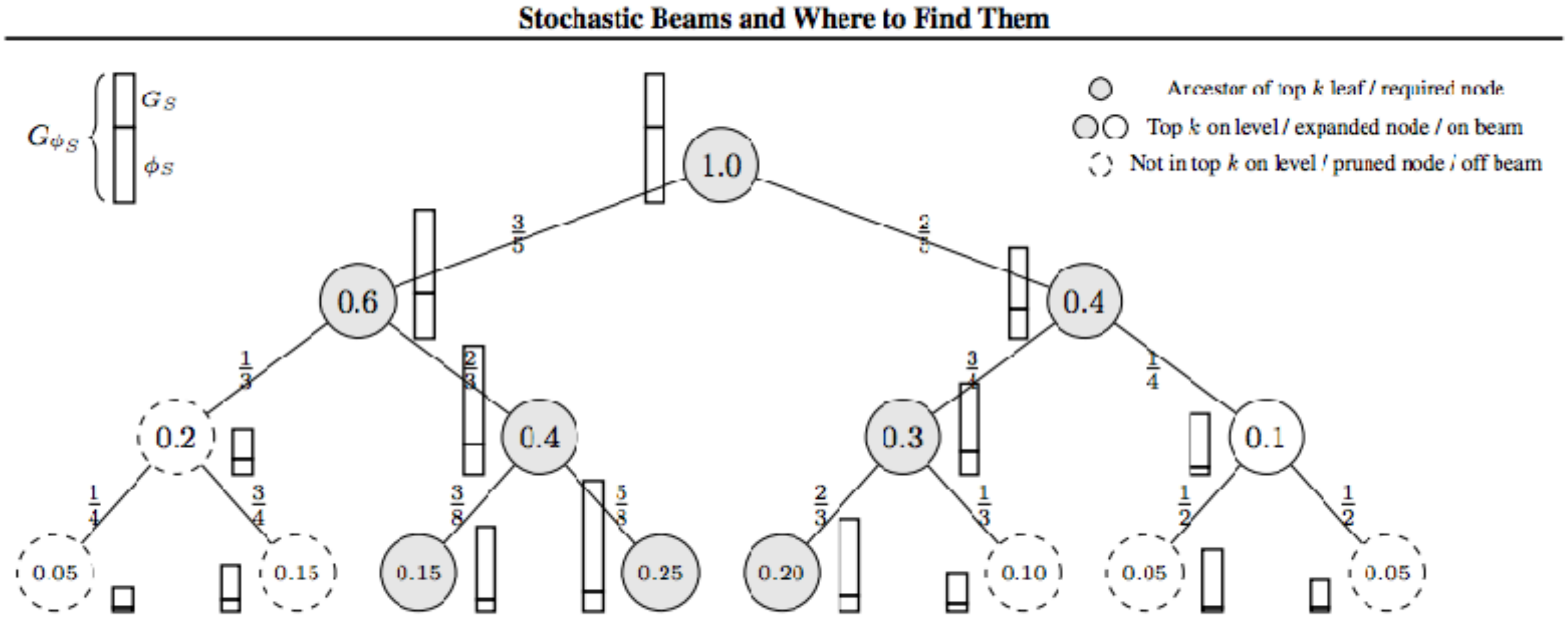
Standard Beam Search

Diversity Promoting Beam Search (γ set to 1)

Sampling without Replacement

(Kool et. al 2019)

- Ancestral sampling samples hypotheses with replacement, how can we do it *without* replacement?
- **Gumbel distribution:** If U is uniform(0,1)
  - G(ϕ) = ϕ − log(− log U)
- Perturbing log probabilities w/ Gumbel noise and find the largest elements = sampling from a categorical distribution without replacement



Heuristic Modifications to Model Score

A Typical Model Error: Length Bias

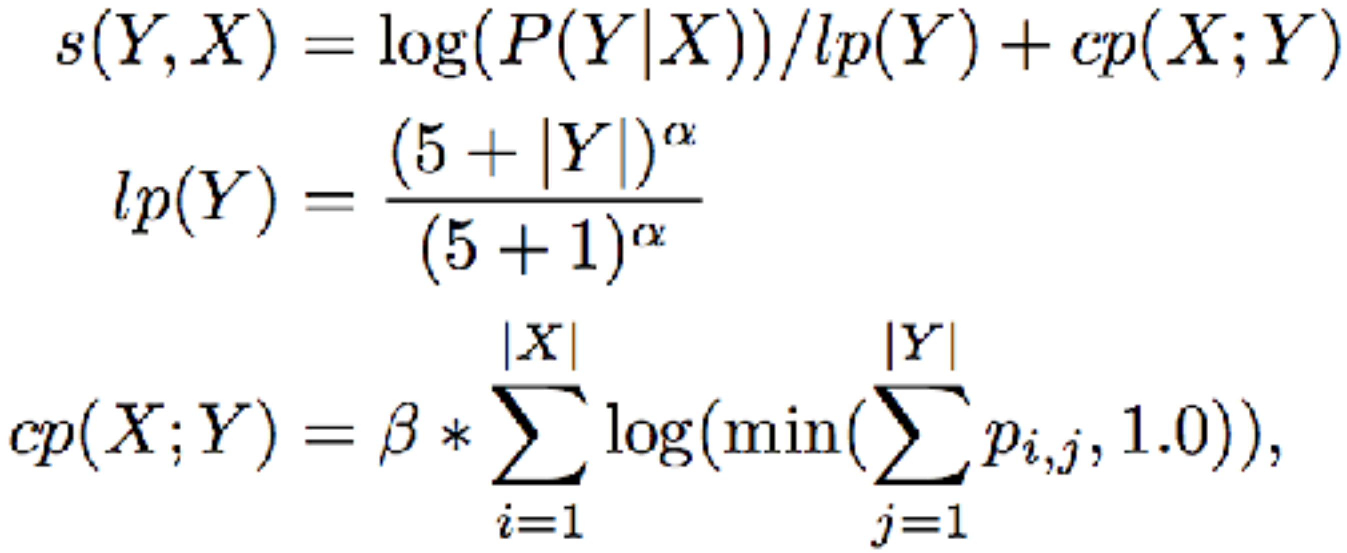
- In many tasks (eg. MT), the output sequences will be of variable length

- Maximum likelihood training+local normalization results in gradually decreasing probability

- Running beam search may then favor short sentences

Length Normalization

- Normalize by the length, dividing by |Y| (Cho et al. 2014)
- More complicated heuristics (Wu et al. 2016)



$$s(Y,X) = \log(P(Y|X))/lp(Y) + cp(X;Y)$$

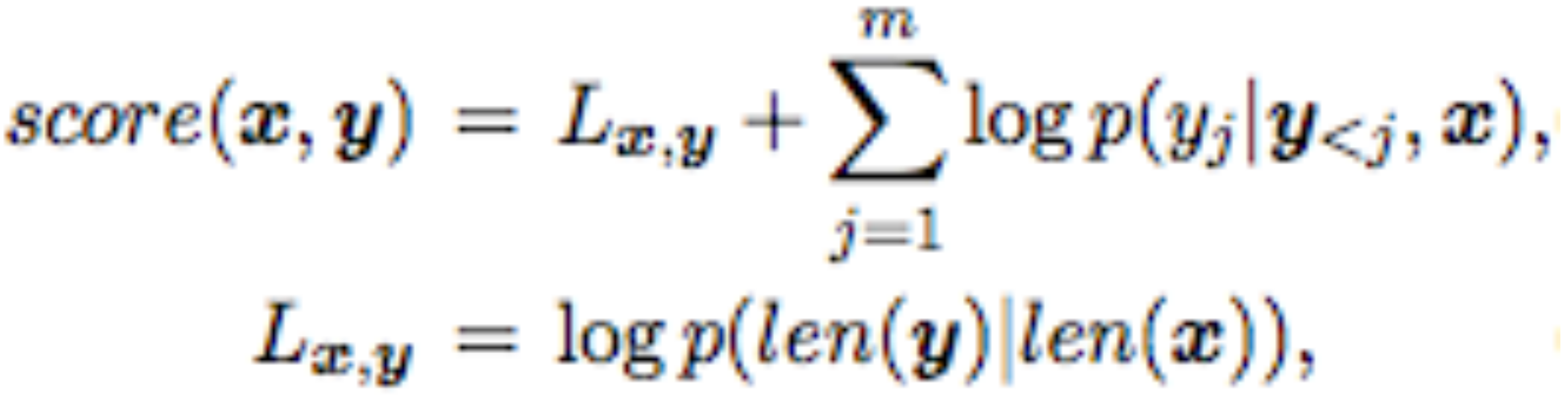
$$lp(Y) = \frac{(5+|Y|)^\alpha}{(5+1)^\alpha}$$

$$cp(X;Y) = \beta * \sum_{i=1}^{|X|} \log(\min(\sum_{j=1}^{|Y|} p_{i,j}, 1.0)),$$

Predict the output length

(Eriguchi et al. 2016)

- Add a penalty based off of length differences between sentences
- Calculate P(len(y) | len(x)) using corpus statistics

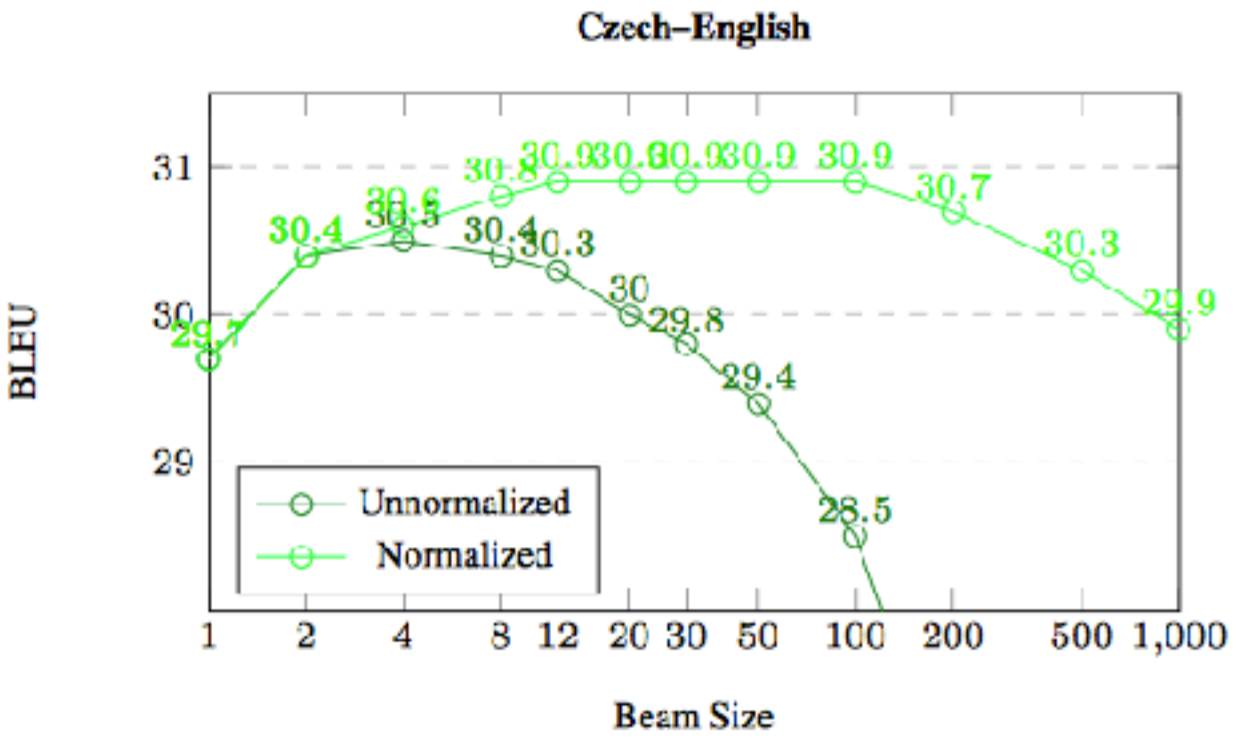


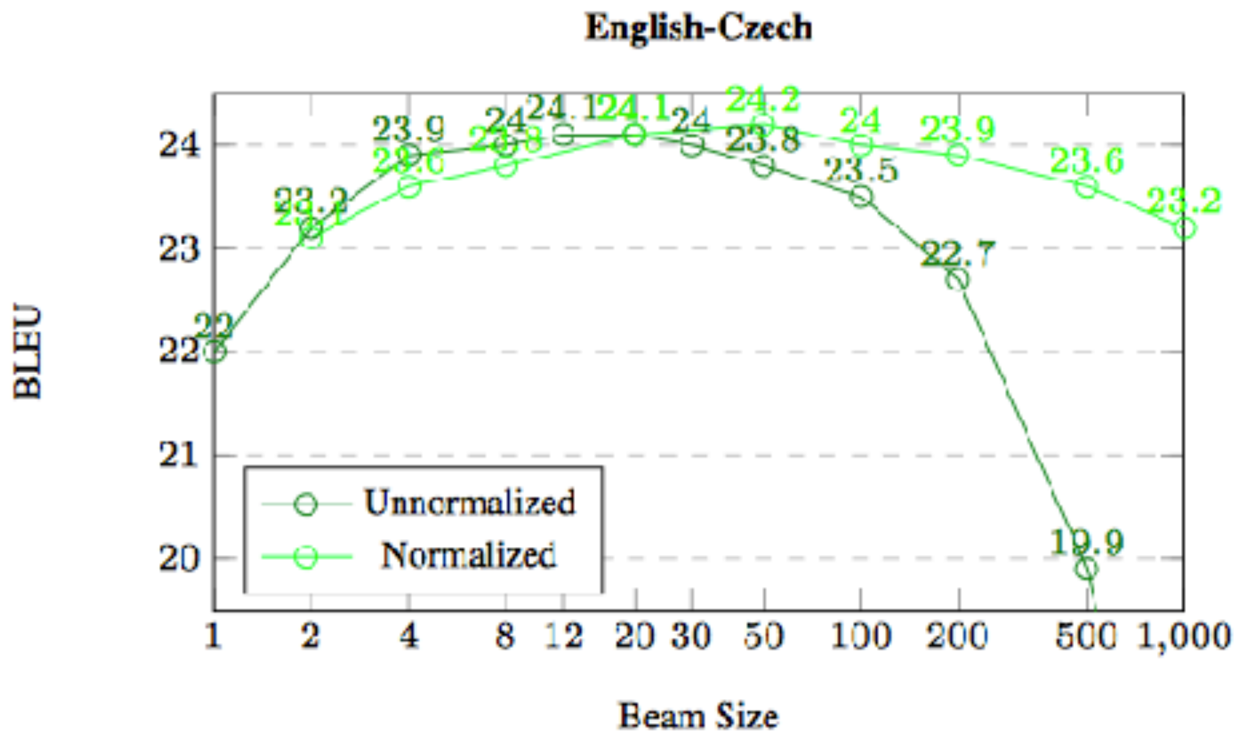
$$score(\boldsymbol{x}, \boldsymbol{y}) = L_{\boldsymbol{x},\boldsymbol{y}} + \sum_{j=1}^{m} \log p(y_j | \boldsymbol{y}_{<j}, \boldsymbol{x}),$$

$$L_{\boldsymbol{x},\boldsymbol{y}} = \log p(len(\boldsymbol{y}) | len(\boldsymbol{x})),$$

Hobbled Search Algorithms

Remember Limited Beam Search Can "Help"

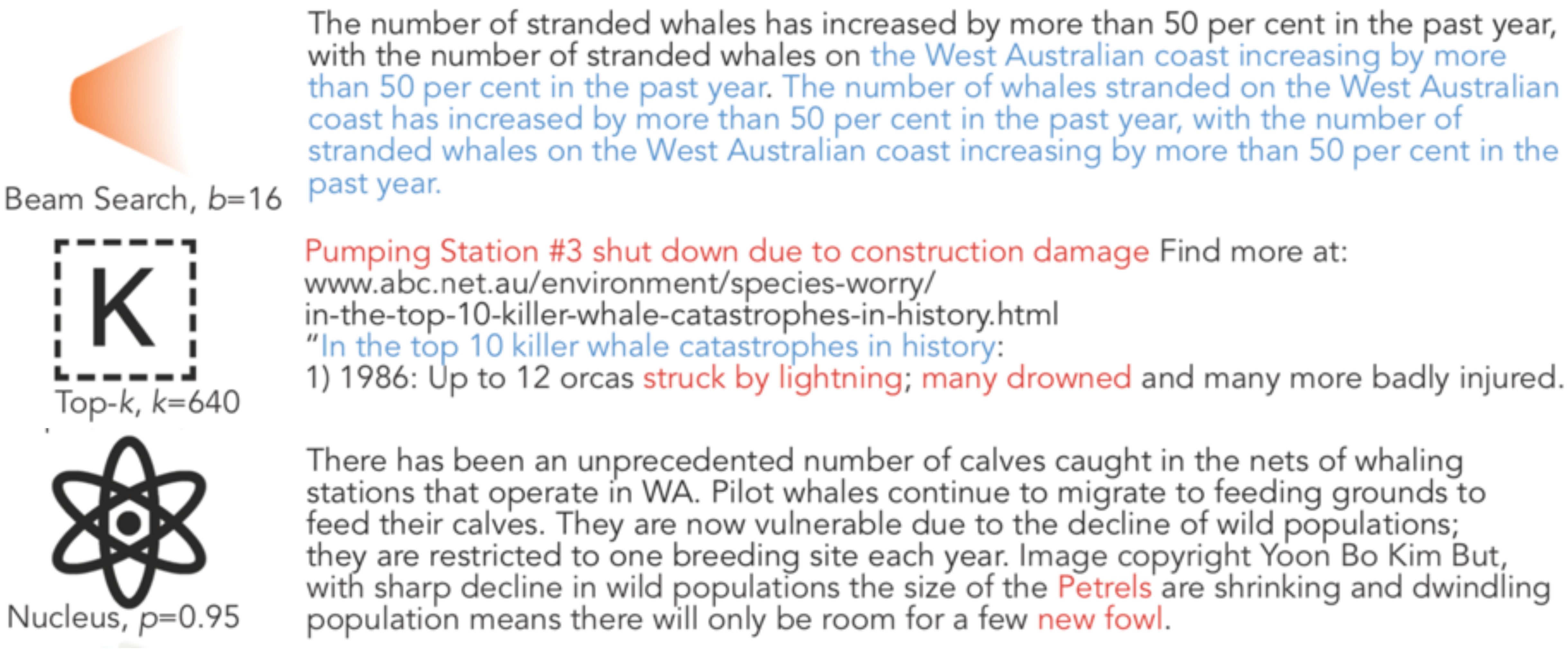




- How else can we modify our search algorithm?

Limited Sampling

- **top-K sampling:** like beam search w/ histogram pruning, but sample from top K instead of enumerate
- **nucleus sampling:** like beam search w/ probability mass pruning, but sample from remaining hypotheses (Holtzman et al. 2020)



Beam Search, *b*=16
Top-*k*, *k*=640
Nucleus, *p*=0.95

The number of stranded whales has increased by more than 50 per cent in the past year, with the number of stranded whales on the West Australian coast increasing by more than 50 per cent in the past year. The number of whales stranded on the West Australian coast has increased by more than 50 per cent in the past year, with the number of stranded whales on the West Australian coast increasing by more than 50 per cent in the past year.

Pumping Station #3 shut down due to construction damage Find more at: www.abc.net.au/environment/species-worry/in-the-top-10-killer-whale-catastrophes-in-history.html
“In the top 10 killer whale catastrophes in history:
1) 1986: Up to 12 orcas struck by lightning; many drowned and many more badly injured.

There has been an unprecedented number of calves caught in the nets of whaling stations that operate in WA. Pilot whales continue to migrate to feeding grounds to feed their calves. They are now vulnerable due to the decline of wild populations; they are restricted to one breeding site each year. Image copyright Yoon Bo Kim But, with sharp decline in wild populations the size of the Petrels are shrinking and dwindling population means there will only be room for a few new fowl.

Cautions about Sampling-based Search

- **Is sampling necessary for diversity?:** questionable, we could do diverse beam search instead.

- **Results are inconsistent from run-to-run:** need to consider variance from this in reporting (in addition to variance in training and data selection)

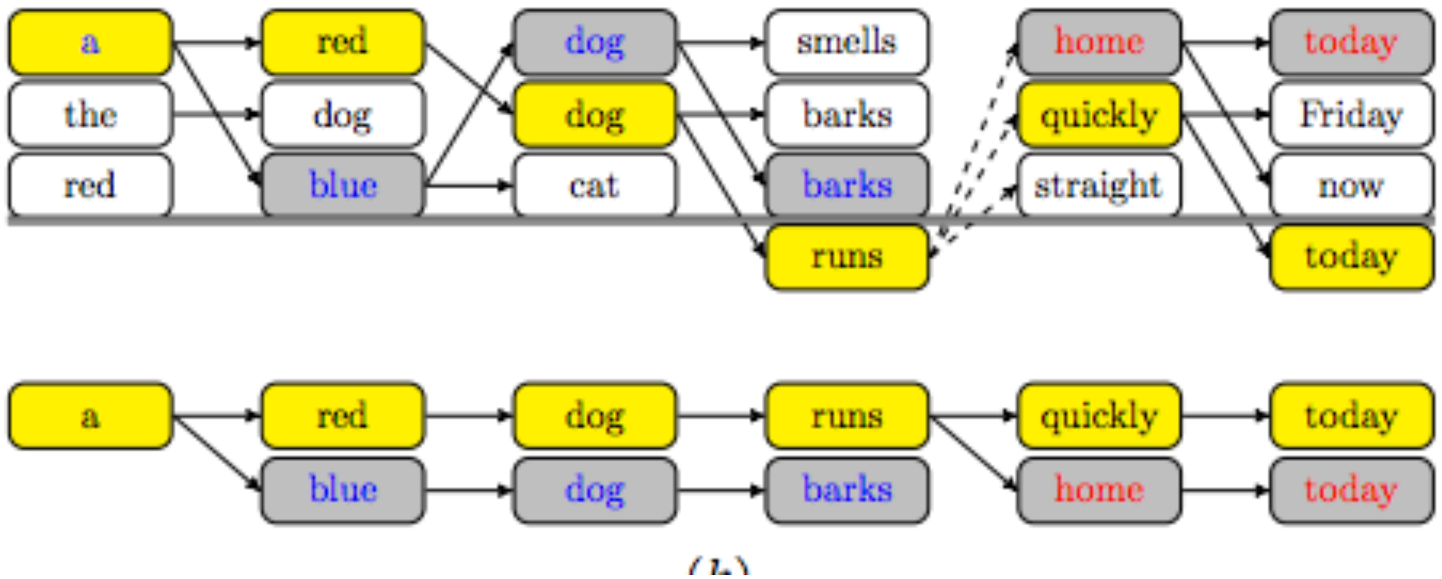
- **Conflates model and search errors:** if you make a *better* model you might get *worse* results, because the search algorithm can't find the outputs your model likes

Search in Training

Using Beam Search in Training

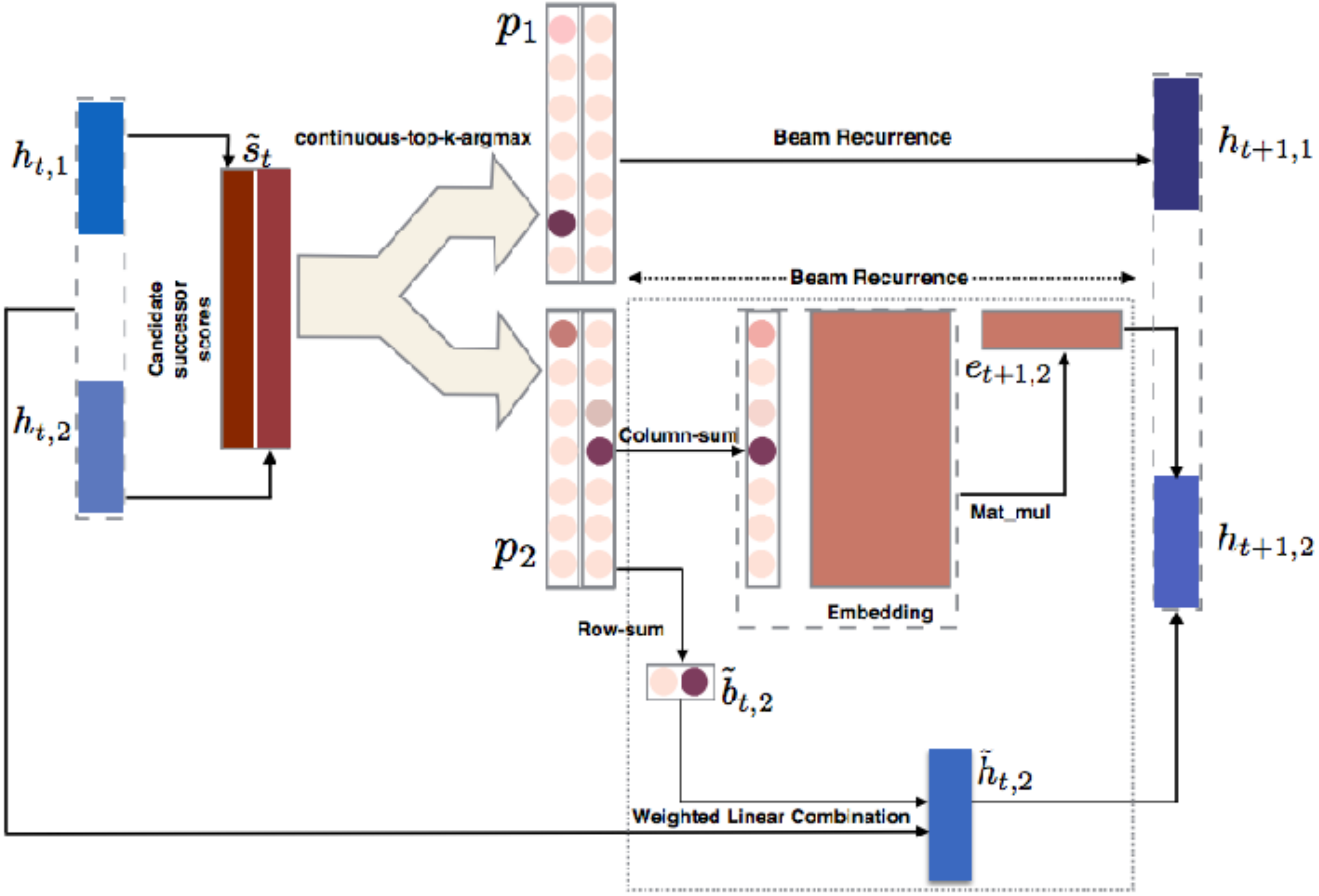
(Wiseman et al., 2016)

- Decoding with beam search has biases
  - Exposure: Model not exposed to errors during training
  - Label: scores are locally normalized
- Possible solution: train with beam search



Continuous Beam Search

(Goyal et al., 2017)



Actor Critic
(Bahdanau et. al., 2017)

- Basic idea:
  - Use Neural Model as an actor that predicts actions (say, the next word)
  - Use a critic to predict final reward (in this case, BLEU) for MT models
  - Actor trained similarly to REINFORCE, critic trained with TD

Actor Critic (continued)

Actor:

$$\frac{\widehat{dV}}{d\theta} = \sum_{k=1}^{M} \sum_{t=1}^{T} \sum_{a \in \mathcal{A}} \frac{dp(a|\hat{Y}^{k}_{1 \ldots t-1})}{d\theta} Q(a; \hat{Y}^{k}_{1 \ldots t-1})$$

- T is the sequence, M in the set of examples, and a the potential next actions, Q reward

Critic:

$$\frac{d}{d\phi} \left(\sum_{t=1}^{T} \left(\hat{Q}(\hat{y}_t; \hat{Y}_{1 \ldots t-1}, Y) - q_t \right)^2 + \lambda_C C_t \right)$$

- C is a measure of reward over average reward similar to REINFORCE style algorithms

Questions?