deeplearning.ai

Sequence to sequence models

Error analysis on beam search

Now, beam search is an approximate search algorithm, also called a heuristic search algorithm. And so it doesn't always outputs the most likely sentence. It's only keeping track of B equals 3 or 10 or 100 top possibilities.
So what if beam search makes a mistake ? In this video, you learn how error analysis interacts with beam search and how you can figure out whether it is the beam search algorithm that's causing problem and worth spending time on.
Or whether it might be your RNN models that is causing problems and worth spending time on.

Example

→ RNN

→ Beam Search

BT

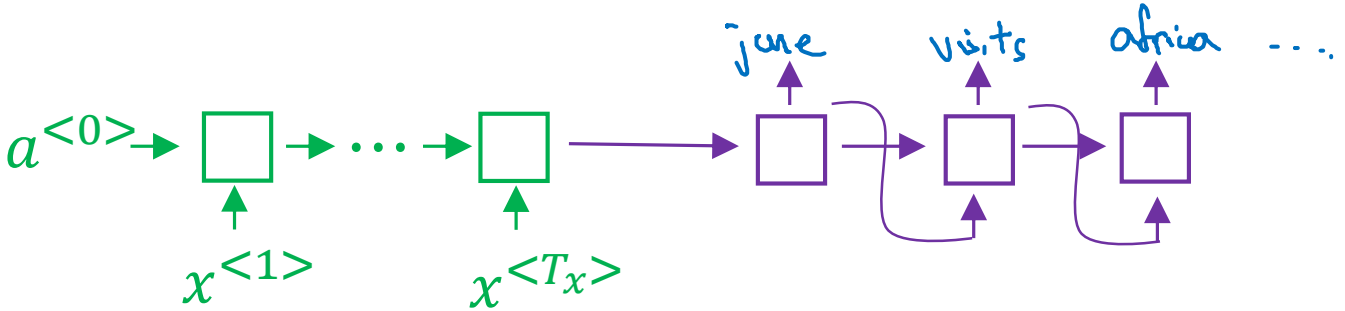
Jane visite l'Afrique en septembre.

But just as getting more training data by itself might not get you to the level performance you want. In the same way ,increasing the beam width by itself might not get you to where you want to go.

Human: Jane visits Africa in September. (y^*)

Algorithm: Jane visited Africa last September. $(\hat{y})$ ←

RNN computes $P(y^*|x) > P(\hat{y}|x)$



Error analysis on beam search

$P(y^*|x)$

$P(\hat{y}|x)$

Human: Jane visits Africa in September. (y^*)

Algorithm: Jane visited Africa last September. ($\hat{y}$)

Case 1: $P(y^*|x) > P(\hat{y}|x)$

$\arg\max_y P(y|x)$

Beam search chose $\hat{y}$. But y^* attains higher $P(y|x)$.

Conclusion: Beam search is at fault.

Case 2: $P(y^*|x) \leq P(\hat{y}|x)$

There are some subtleties pertaining to length normalization that I'm glossing over
And if you're using some sort of length normalization, instead of evaluating these probabilities,
you should be evaluating the optimization objective that takes into account length normalization.

y^* is a better translation than $\hat{y}$. But RNN predicted $P(y^*|x) < P(\hat{y}|x)$.

Conclusion: RNN model is at fault.

Andrew Ng

Error analysis process

Human	Algorithm	$P(y^*\|x)$	$P(\hat{y}\|x)$	At fault?
Jane visits Africa in September.	Jane visited Africa last September.	2×10^{-10}	1×10^{-10}	B R B R R ⋮

Figures out what faction of errors are “due to” beam search vs. RNN model

Andrew Ng

I found this particular error analysis process very useful whenever you have an approximate optimization algorithm such as beam search, that is working to optimize some sort of objective, some sort of cost function that is output by a learning algorithm such as sequence to sequence model or a sequence to sequence RNN that we've been discussing in these lectures.