Computational Models for abstractive Text Summarization

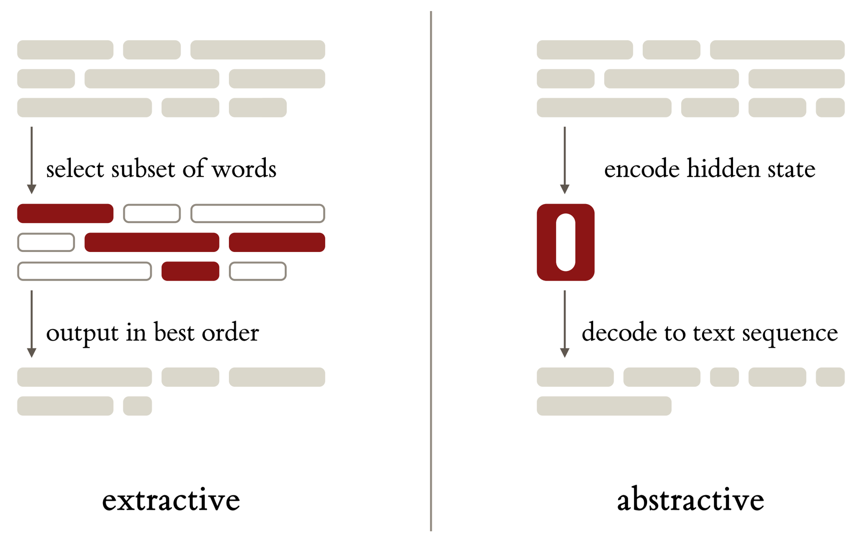
Leonid Keselman & Ludwig Schubert

Task & Datasets

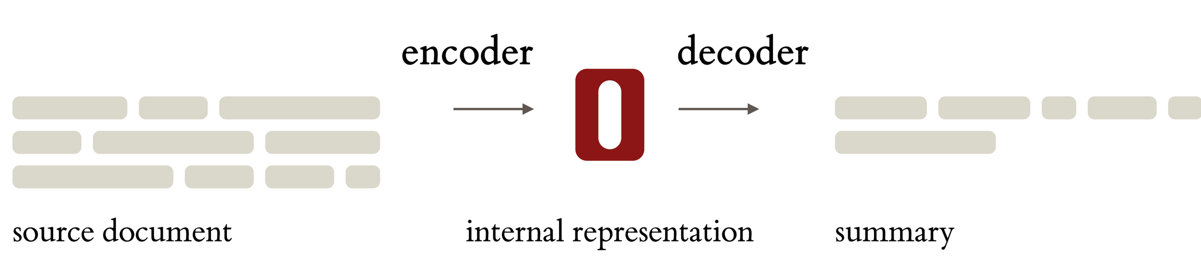
Text summarization is the task of taking a document and creating a shorter version of it while preserving its meaning.

What did you focus on?

Existing summarization techniques can be classified into two categories; extractive, and abstractive:



We focus on building an abstractive model that is able to train faster and scale to larger inputs than traditional *sequence-to-sequence* architectures. We implement a variety of recurrent decoders, paired with efficient feed-forward and convolutional encoders.



Which datasets do you use?

To observe differences in summarization behavior, we trained and tested our models on four datasets:

DUC 2004—432 news articles with 4 model summaries each

NewsIR '16—1M online articles; filtered to media type "News"

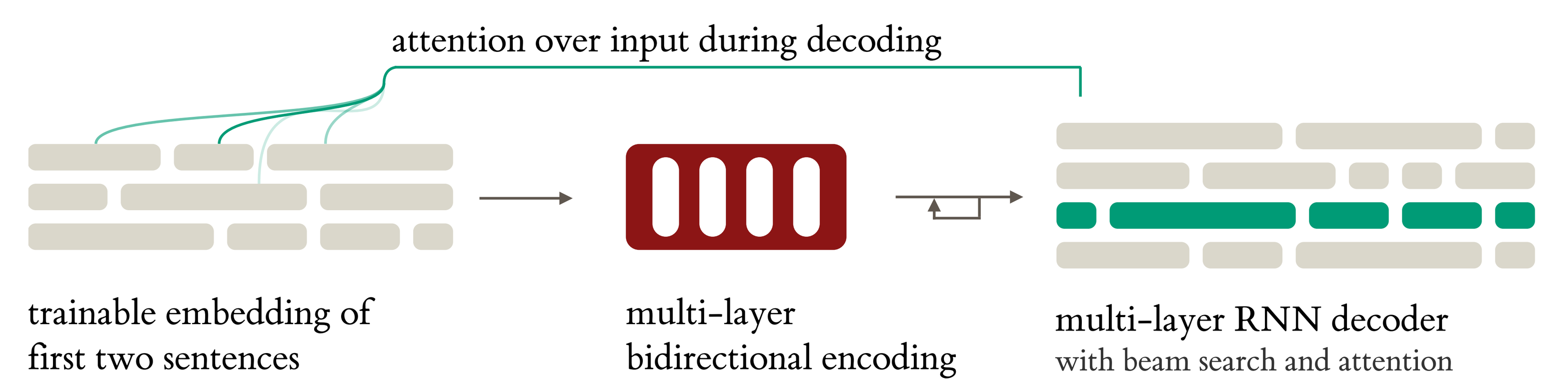
NIPS—all existing NIPS publications; used abstracts and titles

SQuAD—we flipped this question answering dataset to get a rough equivalent of multiple summaries per context paragraph.

All datasets were split 80-10-10 for training, evaluation and testing respectively.

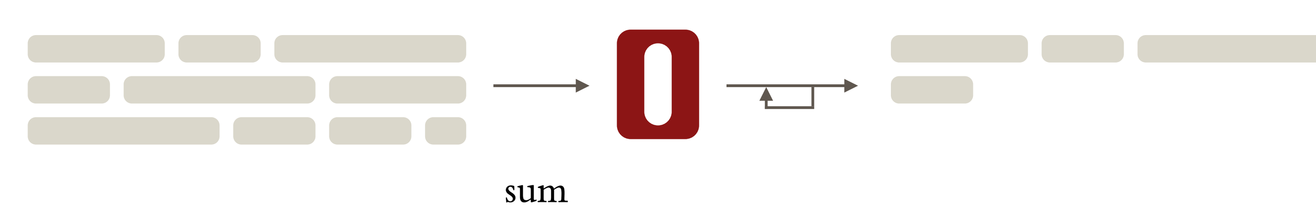
Model Architectures

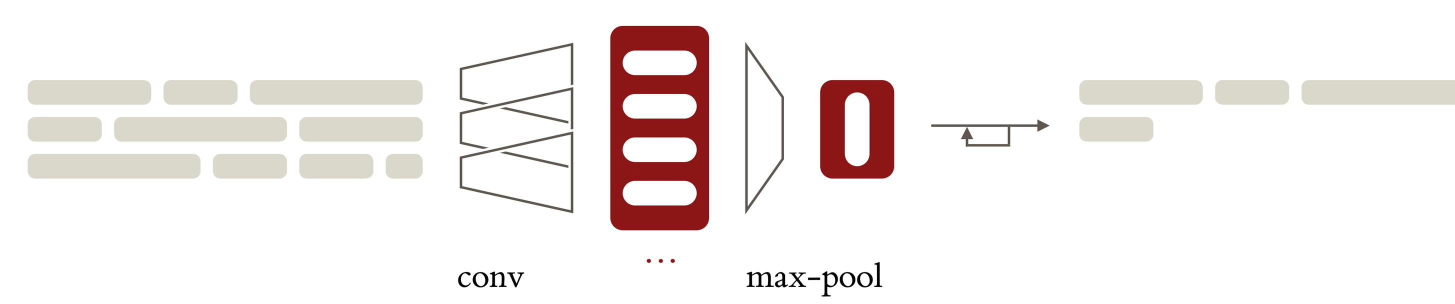
State of the art in text summarization are sequence-to-sequence models with attention:



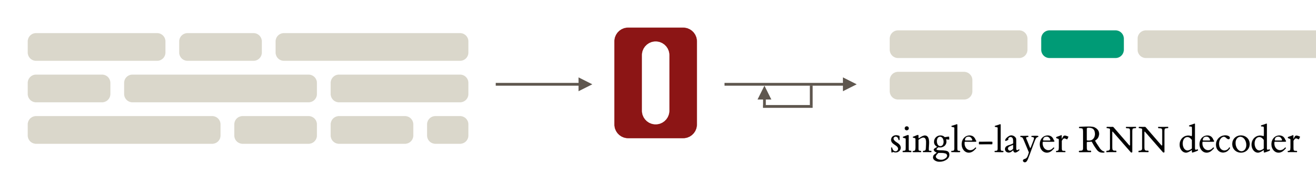
Encoder Architectures

These models are simple & allow for arbitrary length inputs.

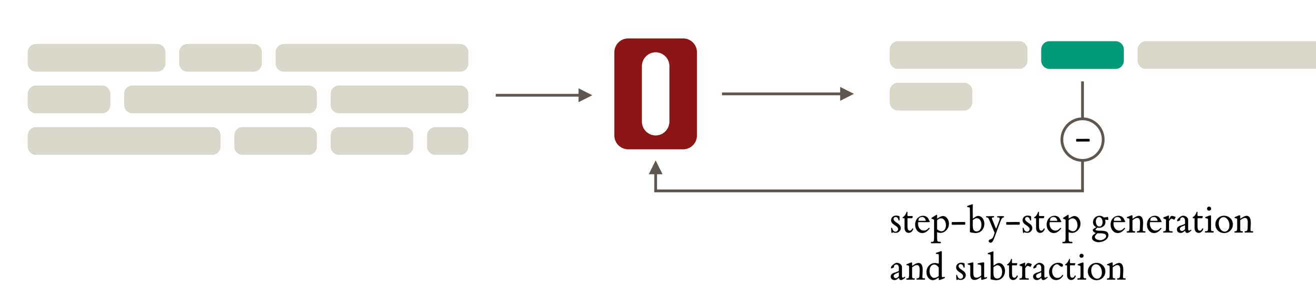


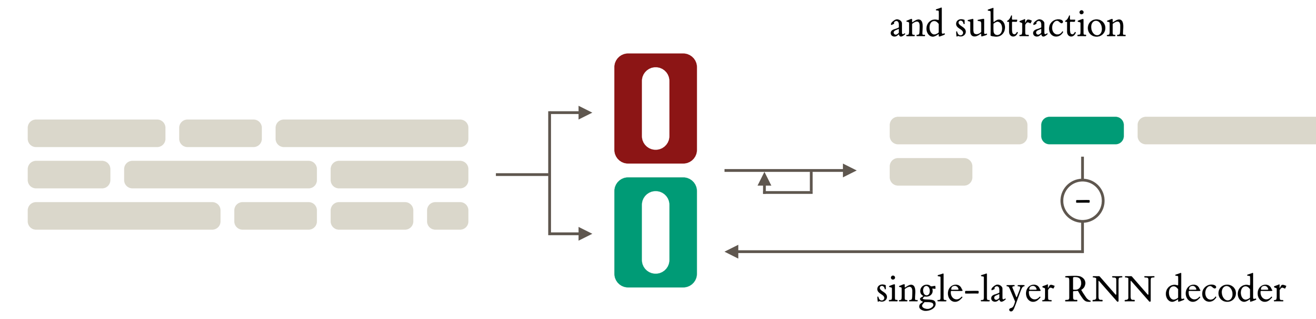


Decoder Architectures



These models aim to increase convergence and ameliorate issues with generated text, such as repetition.





Results & Discussion

Our sum-of-glove model beat our first-sentence baseline on some datasets, while being worse on others.

tested on	Baseline (first sentence)	Sum-of-Glove + RNN (F-scores Rouge-1 & -2), trained on NIPS	SQuAD	NewsIR'16	DUC'04	Textsum (SOA architecture; stopped after 180k steps), trained on NIPS	SQuAD	NewsIR'16	DUC'04
NIPS	3.652 0.060	3.735 0.074	0.052 0	0 0	0 0	3.233 0.165			
SQuAD	3.687 0	0.087 0.015	0.802 0	0.482 0	0 0		4.018 0.237		
NewsIR'16	0.034 0	1.017 0	0.196 0	0.016* 0	0 0			1.810 0	
DUC'04	0.761 0.095	0.631 0	0.020 0	0 0	1.064 0				2.502 0

Example Summary

Generated: **sword of orion sailor missing**

GT: **three yachts missing two dead one sailor missing**

Transfer task: sentiment classification on summaries

One question we were trying to answer was whether summarization could help with other NLP tasks; in our case the answer was "no"—the simple models did not improve the quality of an LSTM-based sentiment model in either case.

tested on \ trained on	full	summaries
full	82.4%	60.1%
summaries	75.5%	65.5%

Conclusion & Future Work

Our models produced summaries that were often on topic but with grammatical issues. On the decoder site we saw promising results from training on identity datasets.a

We could make additional progress both by using larger training data sets and training these systems as autoencoders, where input is expected to match output.

Leonid Keselman, LEONIDK & Ludwig Schubert, SCHUBERL

[ludwig, leonidk]@cs.stanford.edu