E0-270
Machine Learning

<u>Attention Based Models for Text Summarization</u>

Date:27-04-2019

Koushik Sen
Rahul Dev
Shah Manan Jayant
Upasana Doley

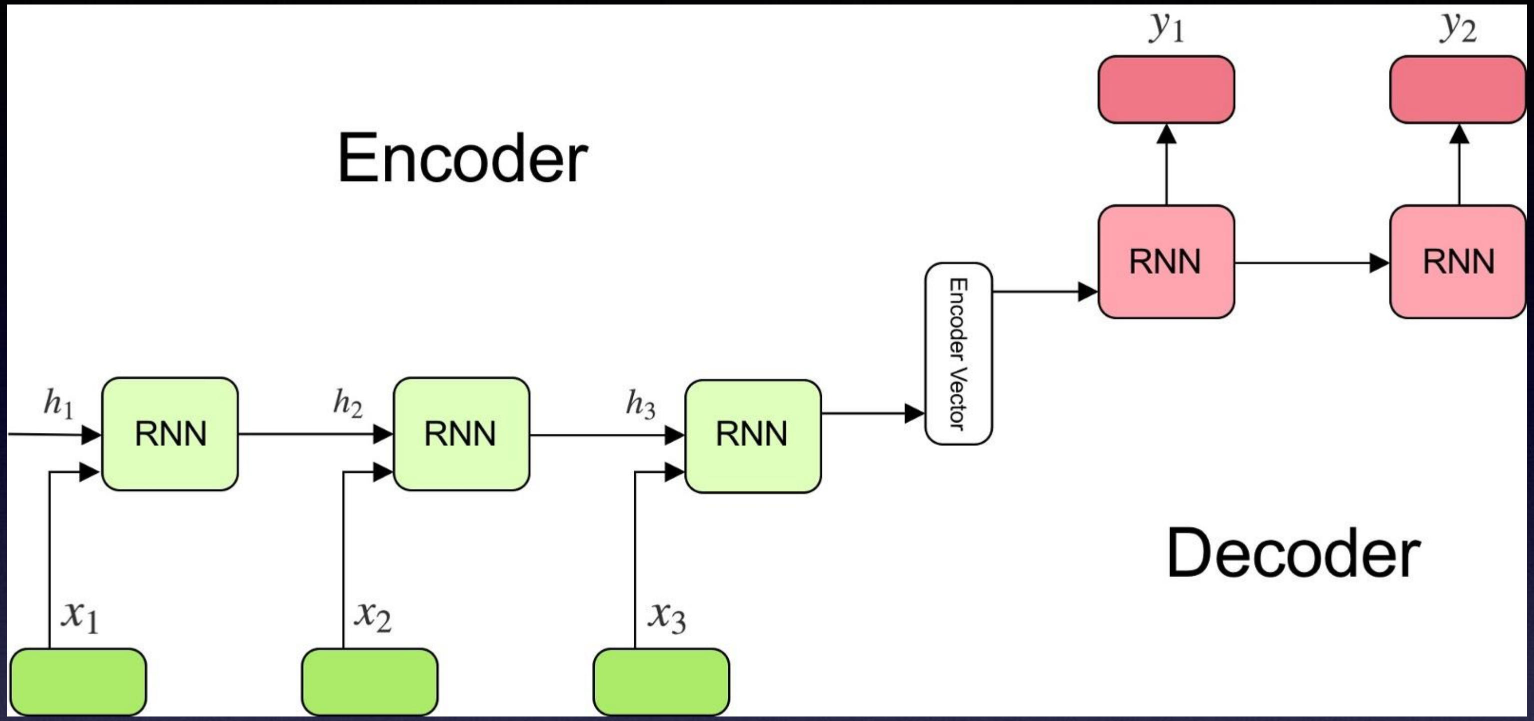
MOTIVATION

Automatic summarization with capability to generate new phrases and sentences like humans for better understanding of the content in large documents.

INTRODUCTION

- Text summarization means creating a smaller version of original text that highlights the important points.
- Extractive methods generates summaries directly from the source text.
- Abstractive method can generate novel words and phrases which are not present in the source text.
- Attention based models struggles in case of OOV words as it generates summaries only from the fixed-length vocabulary.

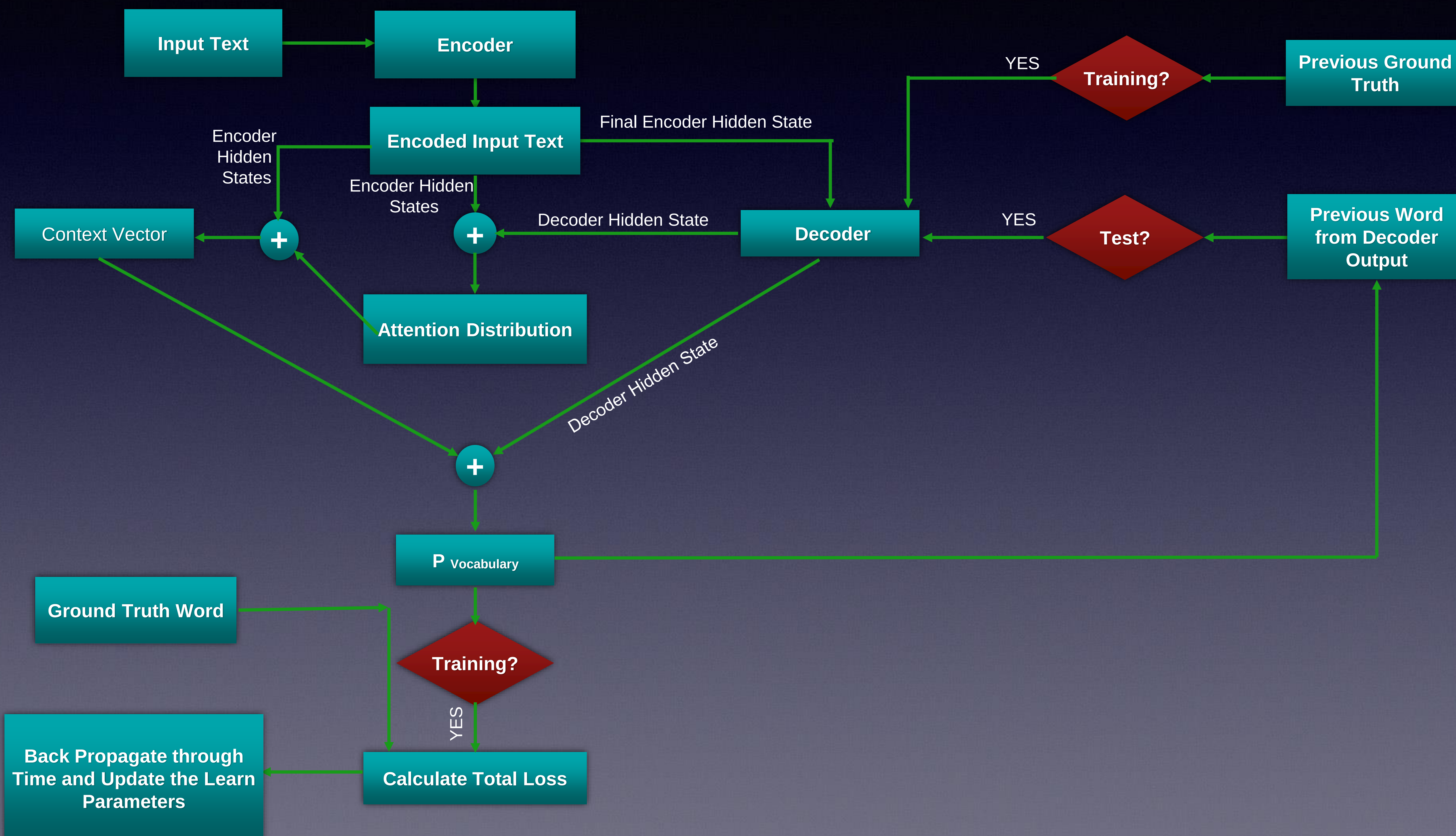
SEQUENCE TO SEQUENCE MODEL



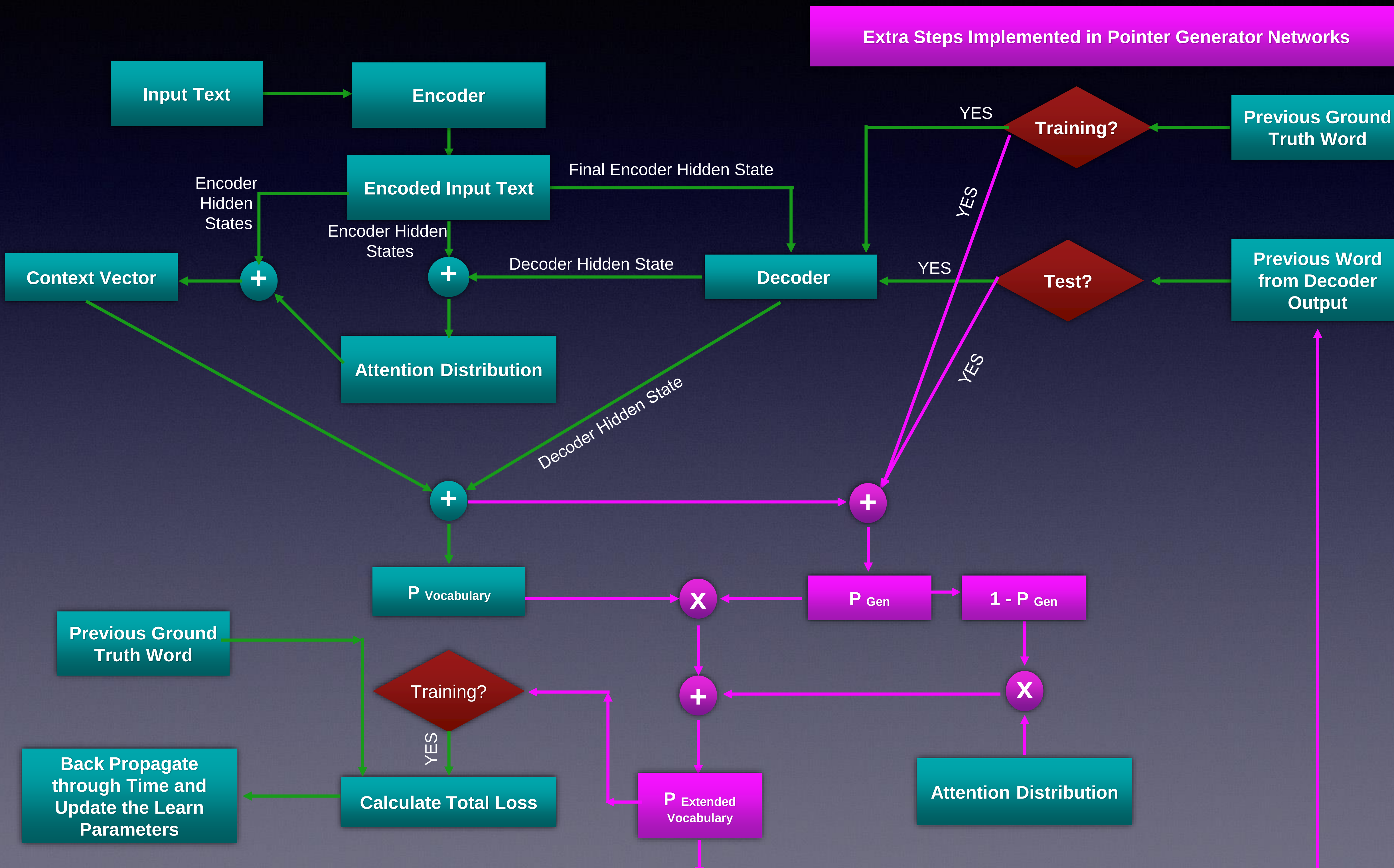
- Consists of an **encoder** and a **decoder** as its main components.
- Encoder: A RNN network which encodes the embeddings of the original source text. It passes the last state of its recurrent layer as an initial state to the first recurrent layer of the decoder part.
- Decoder: It generates the summary from the source text based on the previous hidden states and the previously generated word by the decoder. The decoder takes the last state of encoder’s last recurrent layer and uses it as an initial state to its first recurrent layer .

MODELS IMPLEMENTED

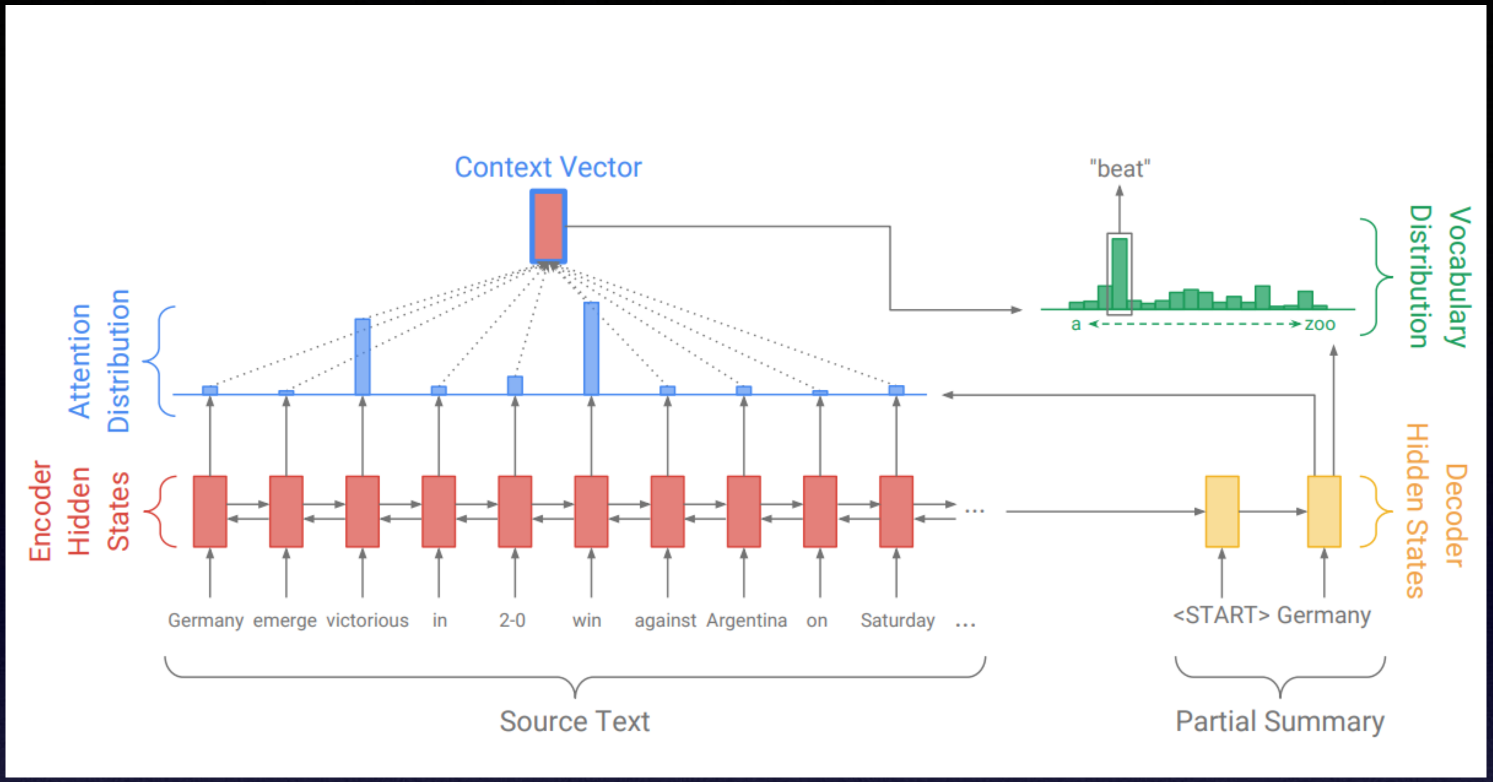
1. Sequence-to-Sequence + Attention Mechanism



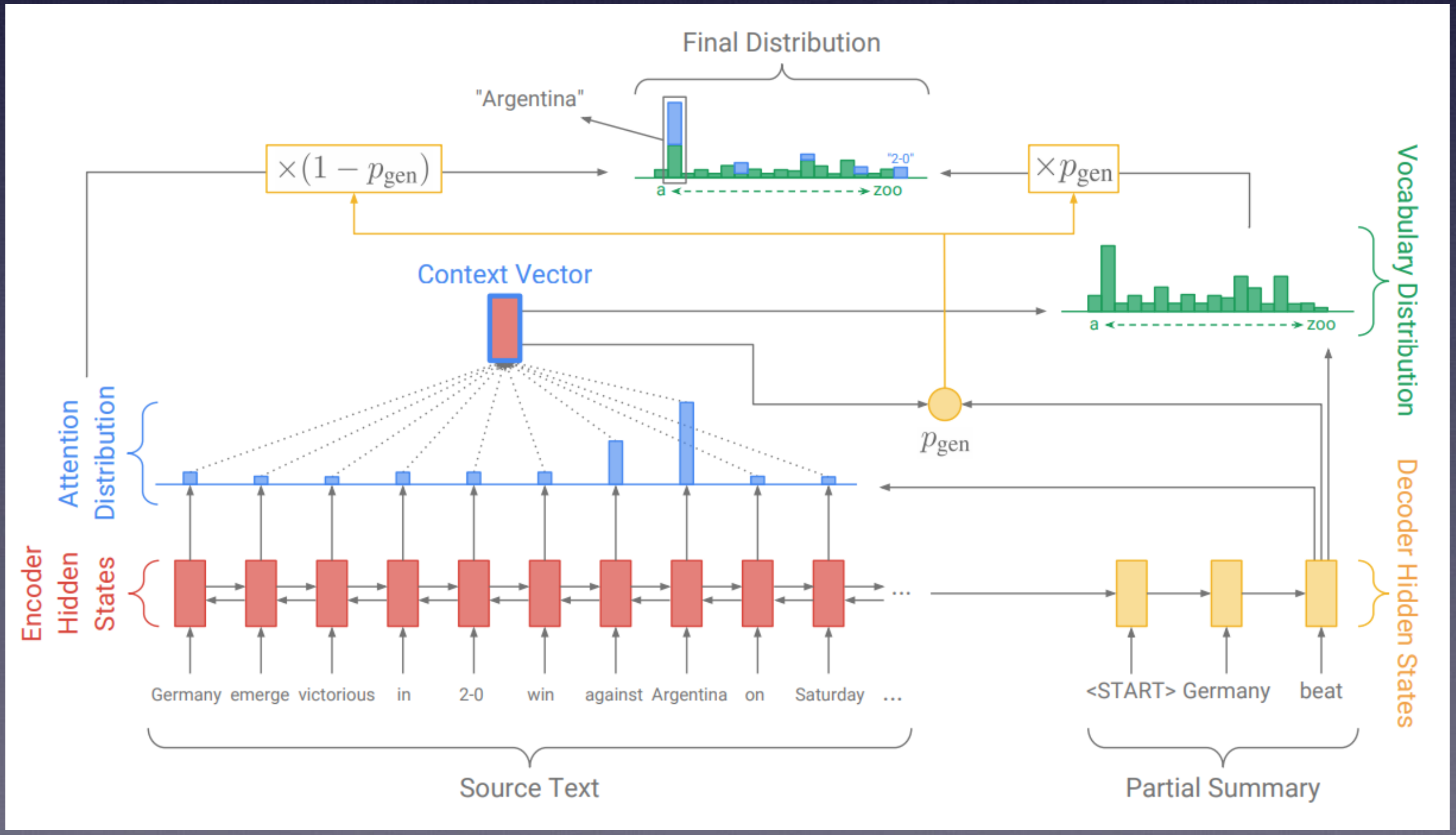
2. Sequence-to-sequence + Attention mechanism + Pointer-Generator Networks



1. Sequence-to-Sequence + Attention Mechanism



2. Sequence-to-sequence + Attention mechanism + Pointer-Generator Networks



Reference : Get to the point: Summarization with pointer-generator networks

ATTENTION-VARIANTS:

- Attention distribution : $a^t = \mathit{softmax}(e^t)$, at time t
- $e^t \in R^N$ can be computed as the following ways

1. Dot-product attention : $e_i^t = s_t^T h_i^t \in R$

2. Multiplicative Attention : $e_i^t = s_t^T W h_i^t \in R$

3. Additive Attention : $e_i^t = v^T \tanh(W_1 h_i^t + W_2 s_t) \in R$

where, $h_i^t \in R^{d1}$ are hidden states of encoders

$s_t \in R^{d2}$ is the decoder state

Model Parameters

Model Parameters	
Batch Size	512
Training Set Examples	1 Million
Input Sequence Length	50
Output Sequence Length	15
Encoder Hidden State Dimension	200
Vocabulary Size	40000
Epochs trained	15
Word Vector Dimension	50
For Pointer Generator Models	
Extended Vocabulary Size	40030

Fig 1:Model Parameters for Attention
Based Networks and Pointer Generator Networks

Comparison of Variants of Attention Mechanism

- **Input Text** : julius berger won nigeria 's challenge cup after they beat the katsina united with a golden goal at the ##th minute in a match held in the national stadium in lagos on saturday

- **Ground Truth** : julius berger wins nigeria 's challenge cup

- **Dot product** : < unk > wins men 's cup final <\s>

- **Multiplicative attention** : < unk > wins men 's < unk > <\s>

- **Bahdanau Attention** : berger wins men with cup <\s>

Comparison of Dot-product Attention vs Pointer Generator

- **Input Text**: julius berger won nigeria 's challenge cup after they beat the katsina united with a golden goal at the ##th minute in a match held in the national stadium in lagos on saturday

- **Ground Truth** : julius berger wins nigeria 's challenge cup

- **Dot-product Attention** : < unk > wins men 's cup final <\s>

- **Pointer Generator Network** : julius berger win in world cup <\s>

ROUGE SCORES

1) Test set 1: consisting of 10 examples with less number of OOV words
2) Test set 2: consisting of 9 examples with more number of OOV words

	F1-Score		
	Test Set 1		
	1	2	L
Dot Attention	31.171	4.011	28.271
Multiplicative Attention	25.47	2.666	22.119
Bahdanu Attention	28.41	1.33	25.73
Pointer Generator with Bahdanu Attention	33.458	5.523	29.813
F1 score in (See et al)	36.441	5.66	33.42

Fig 1: F1-Score with Test Set-1

	F1-Score		
	Test Set 2		
	Rouge 1	Rouge 2	Rouge L
Dot Product Attention	23.542	3.931	22.521
Pointer Generator with Bahadnu Attention	27.457	8.635	25.297

Fig 2: F1-Score with Test Set-2

CONCLUSION

- Attention Model variants generated some unknown <UNK> tokens as it cannot point directly to the OOV words if needed for summary.
- Pointer Generated Models outperformed all the three types of attention models as it can point(copy) the OOV words from the original source text if needed for summary.

REFERENCES

- Bahdanau, D., Cho, K., and Bengio, Y. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473, 2014.
- Nallapati, R., Zhou, B., Gulcehre, C., Xiang, B., et al. Abstractive text summarization using sequence-to-sequence rnns and beyond. arXiv preprint arXiv:1602.06023, 2016.
- See, A., Liu, P. J., and Manning, C. D. Get to the point: Summarization with pointer-generator networks. arXiv preprint arXiv:1704.04368, 2017.
- Pennington, J., Socher, R., and Manning, C. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL https: //www.aclweb.org/anthology/D14-1162.