Become an expert in any field

Powered by :

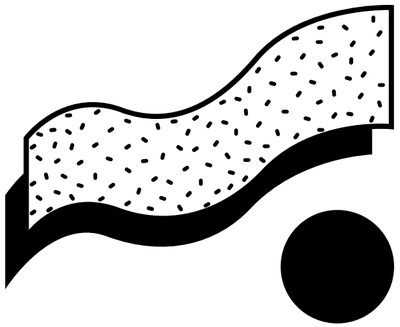
Streamlit

Why ?

Given information is free online,
you could become an expert in any field.
Why is this not happening ?

- **Research papers are really though to read**
- **We make reading papers easier**

Plan



For one paper

Explain every keyword

For many papers

Connect all your papers

What’s Next ?

Explain every keyword

Demo Time !

https://adpt-lrn-phys.streamlit.app/

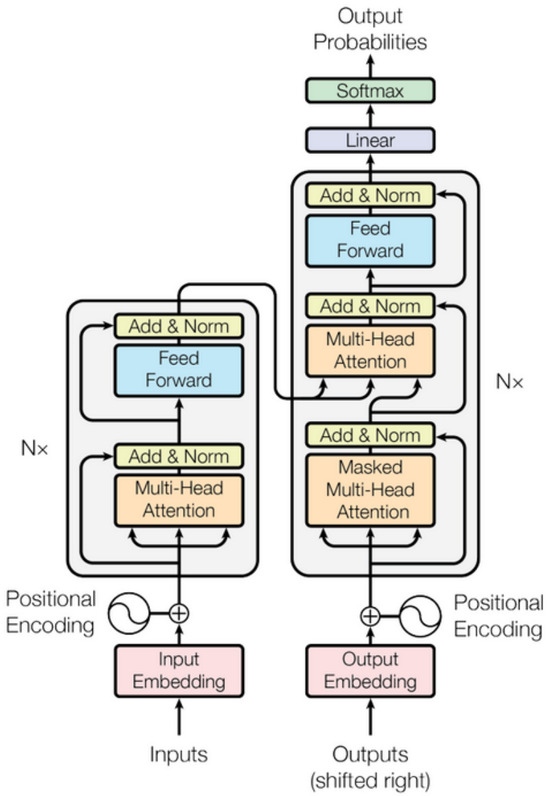


Figure 1: The Transformer - model architecture.

The Transformer follows this overall architecture using stacked self-attention and point-wise, fully connected layers for both the encoder and decoder, shown in the left and right halves of Figure 1, respectively.

3.1 Encoder and Decoder Stacks

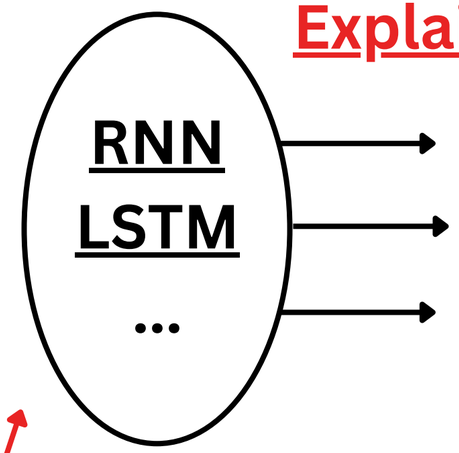
Encoder: The encoder is composed of a stack of $N = 6$ identical layers. Each layer has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise fully connected feed-forward network. We employ a residual connection [11] around each of the two sub-layers, followed by layer normalization [1]. That is, the output of each sub-layer is $\mathrm{LayerNorm}(x + \mathrm{Sublayer}(x))$, where $\mathrm{Sublayer}(x)$ is the function implemented by the sub-layer itself. To facilitate these residual connections, all sub-layers in the model, as well as the embedding layers, produce outputs of dimension $d_{\text{model}} = 512$.

Decoder: The decoder is also composed of a stack of $N = 6$ identical layers. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack. Similar to the encoder, we employ residual connections around each of the sub-layers, followed by layer normalization. We also modify the self-attention sub-layer in the decoder stack to prevent positions from attending to subsequent positions. This masking, combined with fact that the output embeddings are offset by one position, ensures that the

ChatGPT explains it

The introduction provides an overview of recurrent neural networks (RNNs), memory (LSTM), and gated recurrent neural networks. These architectures h used in sequence modeling and transduction problems such as language mo machine translation. The introduction also mentions recent advancements i language models and encoder-decoder architectures.

Keywords extraction



Explain Keywords in parallel

RNNs, or recurrent neural networks, are a t

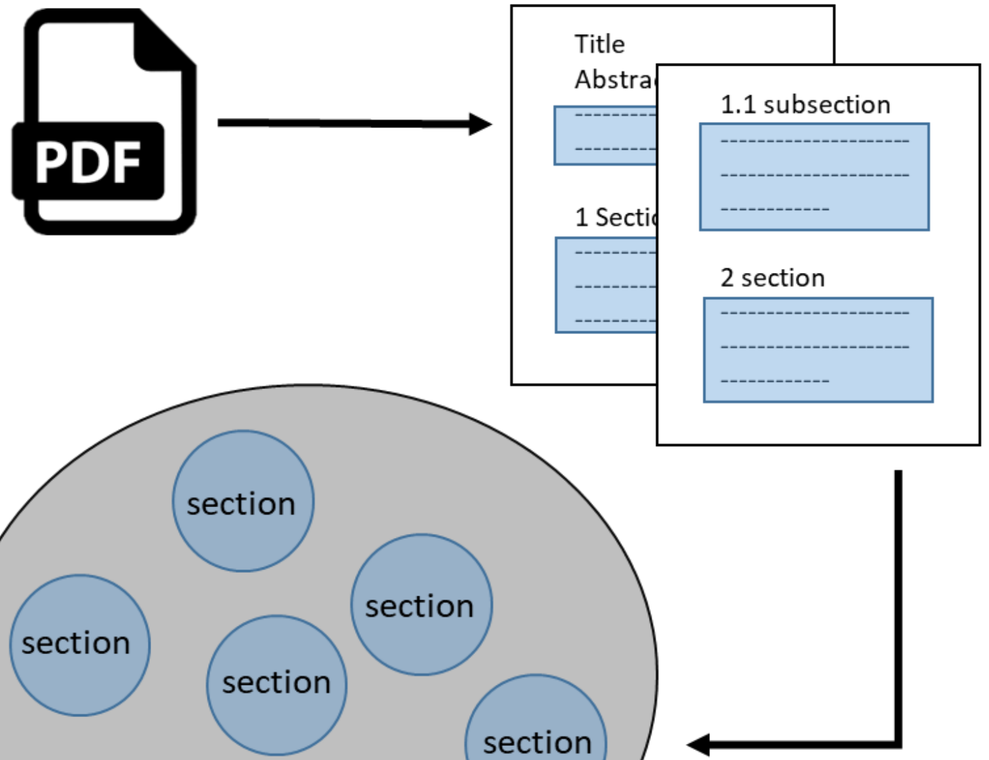
LSTM stands for Long Short-Term Memory, wh

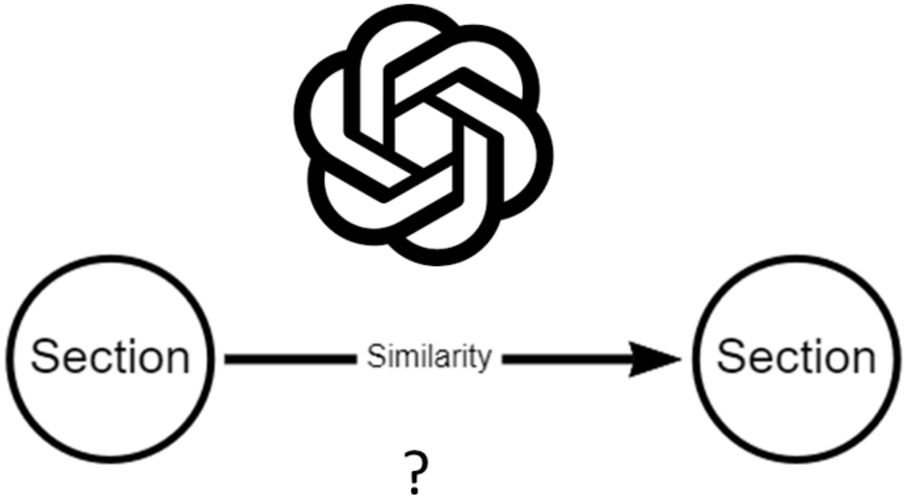
Inject it into the explanation

Generating Graph

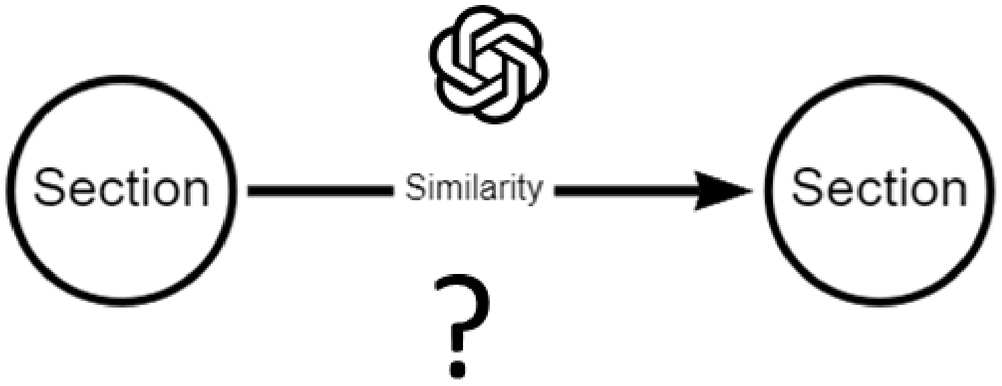
Doc to Nodes

Define Edges



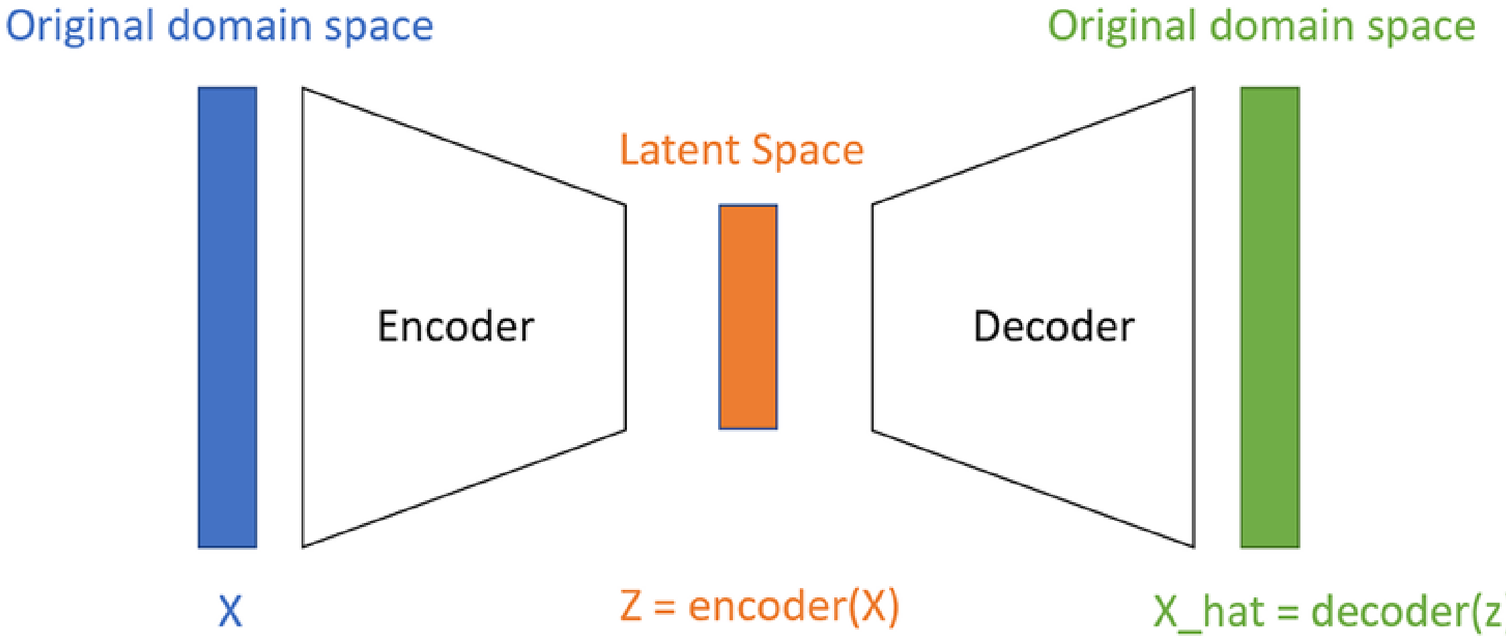


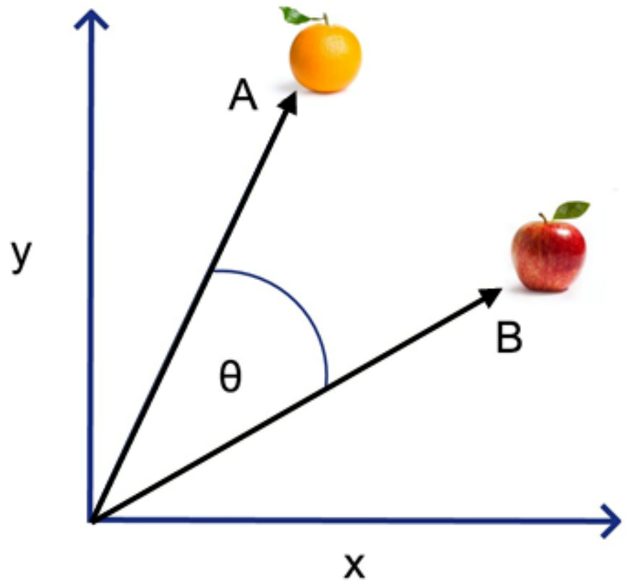
Similarity Measure



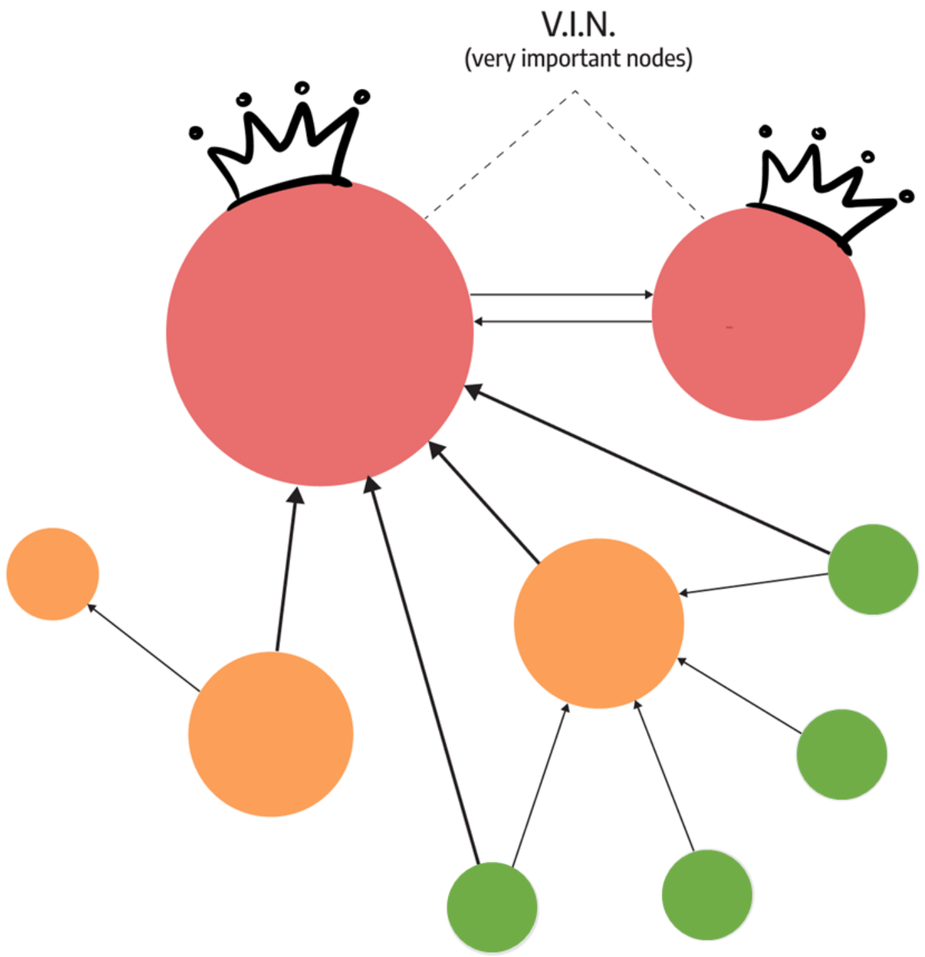
Cosine similarity

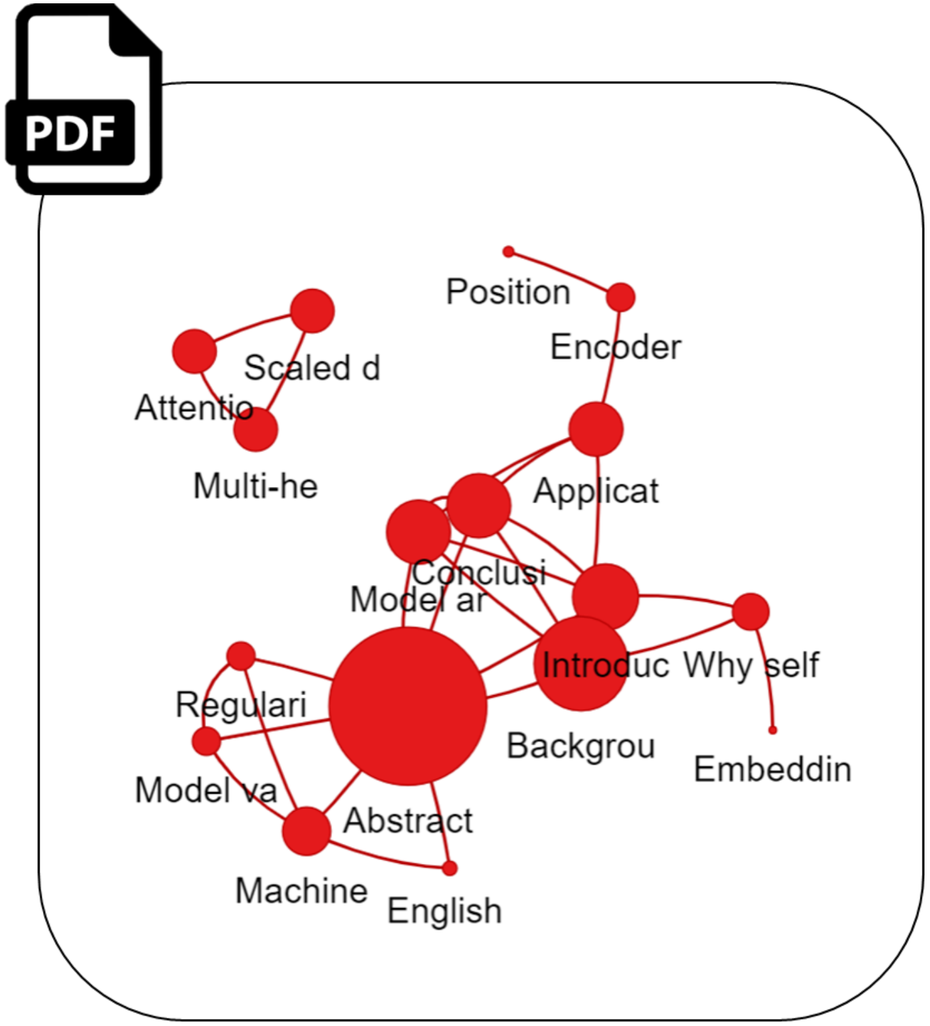
$$Sim(A,B) = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|}$$



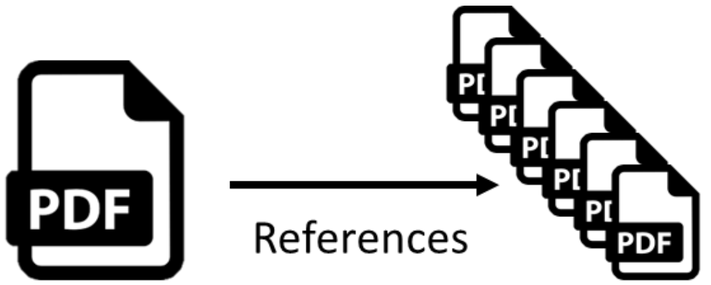


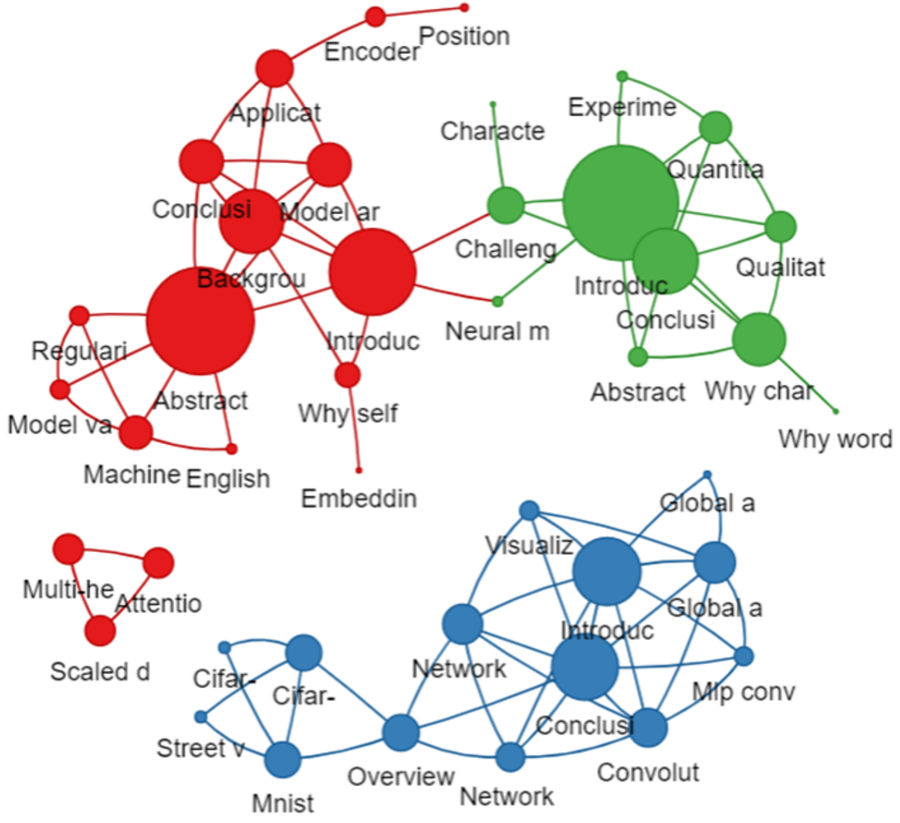
Page rank

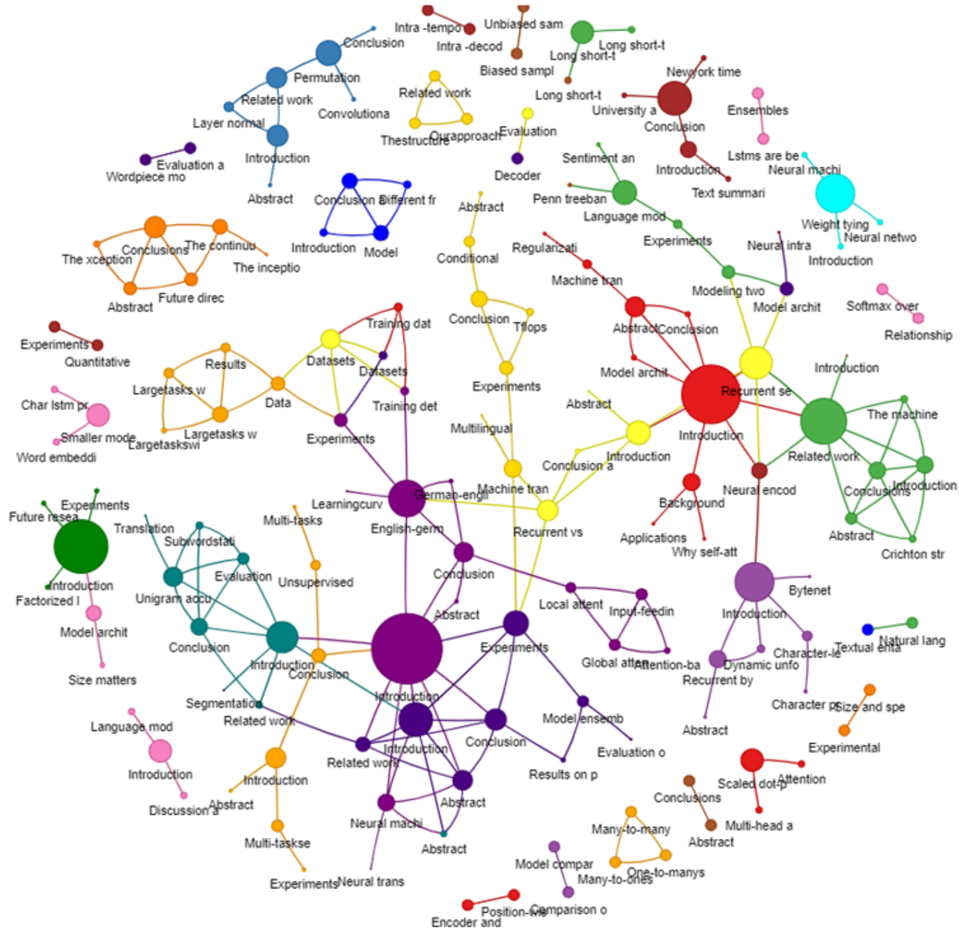

V.I.N.
(very important nodes)


PDF
Position
Encoder
Scaled d
Attentio
Multi-he
Applicat
Conclusi
Model ar
Introduc
Why self
Regulari
Backgrou
Embeddin
Model va
Abstract
Machine
English

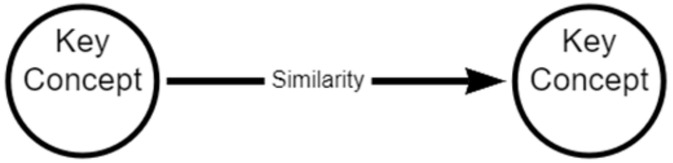

PDF
PDF
PDF


PDF
References
PDF


Encoder
Position
Applicat
Characte
Experime
Quantita
Conclusi
Model ar
Challeng
Backgrou
Introduc
Qualitat
Regulari
Introduc
Neural m
Conclusi
Abstract
Why char
Model va
Abstract
Why word
Machine
English
Why self
Embeddin
Global a
Visualiz
Multi-he
Attentio
Global a
Introduc
Scaled d
Cifar-
Network
Cifar-
Mlp conv
Conclusi
Street v
Overview
Convolut
Mnist
Network

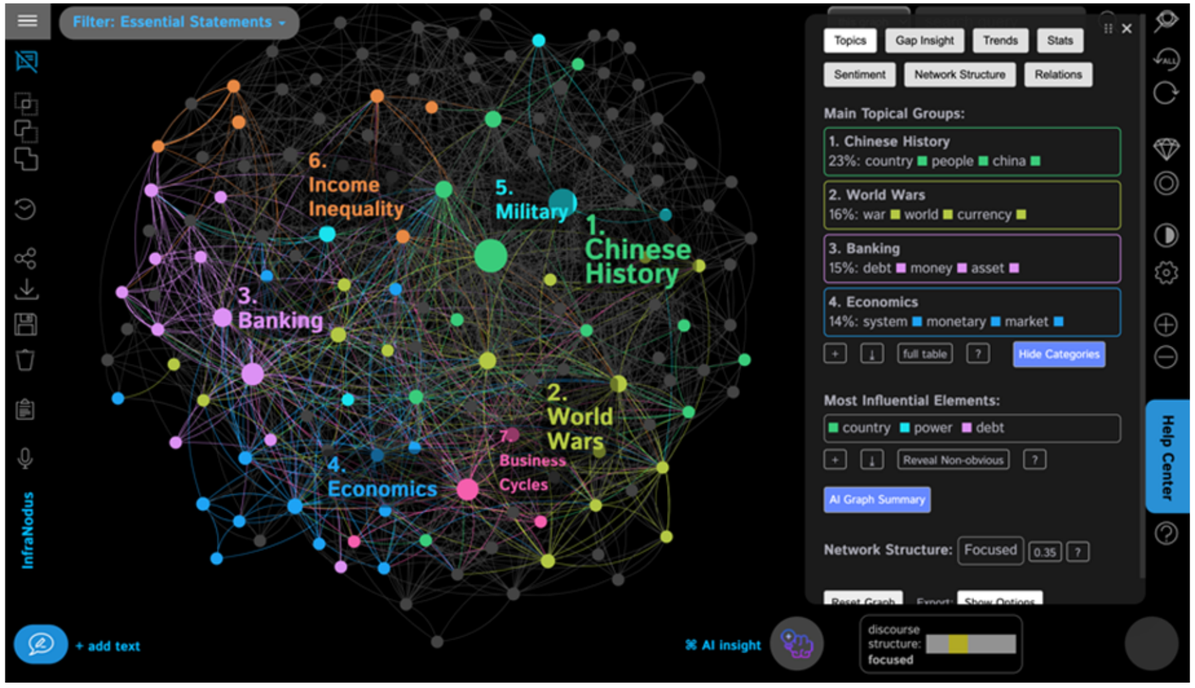

Intra -tempo
Unbiased sam
Intra -decod
Long short-t
Conclusion
Biased sampl
Long short-t
Permutation
Newwork time
Related work
Convolutiona
Long short-t
University a
Layer normal
Thestructure
Chrapproach
Evaluation
Conclusion
Ensembles
Evaluation a
Introduction
Decoder
Sentiment an
Introduction
Lstms are be
Wordpiece mo
Abstract
Conclusion
Different fr
Penn treeban
Text summari
Neural machi
Abstract
Language mod
Weight tying
The xception
Conclusions
The continuo
Introduction
Model
Conditional
Experiments
Neural netwo
The inceptio
Regularizati
Neural intra
Abstract
Future direc
Machine tran
Introduction
Modeling two
Model archit
Softmax over
Experiments
Conclusion
Tflops
Abstract
Conclusion
Relationship
Quantitative
Training dat
Model archit
Introduction
Results
Datasets
Datasets
Recurrent se
Largetasks w
Data
Experiments
Abstract
The machine
Char lstm pr
Training det
Introduction
Related work
Smaller mode
Largetasks w
Multilingual
Introduction
Word embeddi
Largetaskswi
Conclusion a
Introduction
Machine tran
German-engl
Learningcurv
Background
Neural encod
Conclusions
Experiments
Multi-tasks
Recurrent vs
Applications
Why self-att
Abstract
Future resea
Translation
English-germ
Crichton str
Introduction
Subwordstati
Bytenet
Factorized l
Evaluation
Unsupervised
Conclusion
Unigram accu
Abstract
Local attent
Input-feedin
Introduction
Model archit
Conclusion
Experiments
Natural lang
Size matters
Introduction
Conclusion
Global attent
Attention-ba
Character-le
Textual entai
Dynamic unfo
Recurrent by
Segmentatio
Introduction
Character pr
Size and spe
Related work
Model ensemb
Abstract
Language mod
Introduction
Conclusion
Evaluation o
Experimental
Related work
Introduction
Results on p
Scaled dot-p
Attention
Introduction
Abstract
Discussion a
Abstract
Conclusions
Multi-head a
Neural machi
Abstract
Many-to-many
Abstract
Multi-taskse
Abstract
Model compar
One-to-manys
Many-to-ones
Experiments
Neural trans
Position-wis
Comparison o
Encoder and

AI-Powered Visual Text Analysis

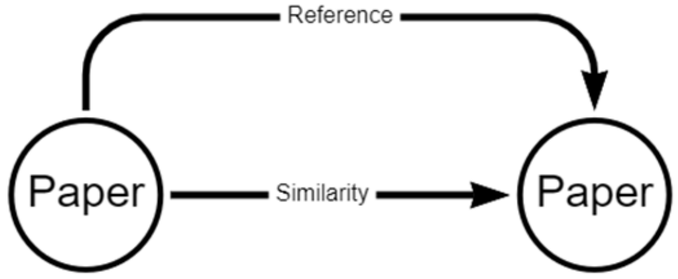


INFRANODUS

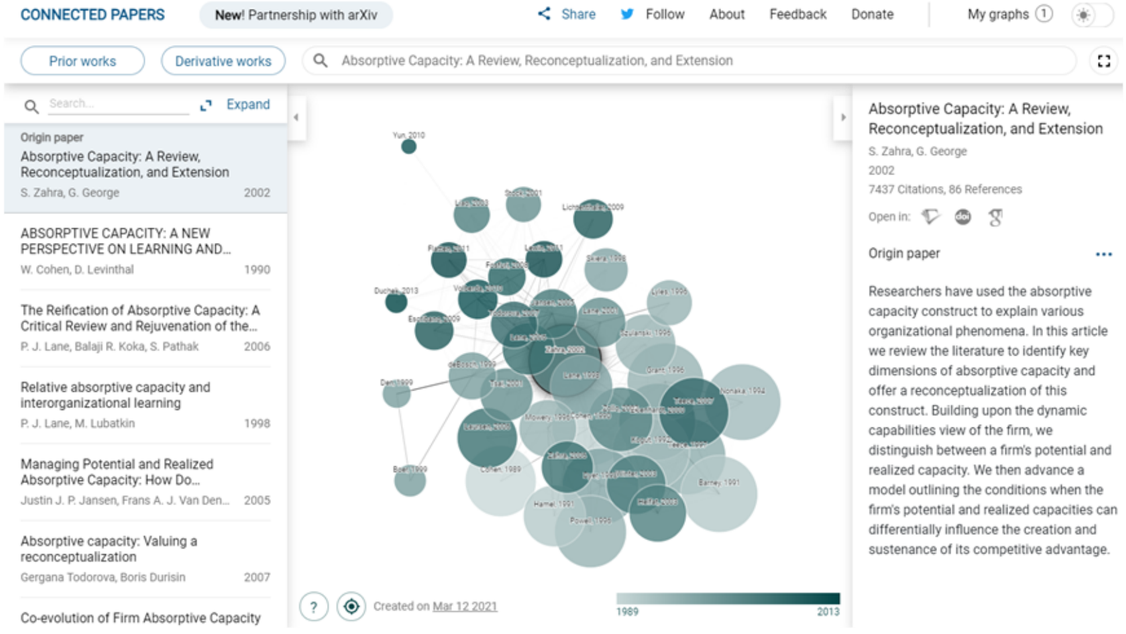
nucliadocs



Connected Papers



CONNECTED PAPERS



What’s next ?

- More features (UI)
  - Learned keyword
  - Study-path recommendation
  - Scientific-writing recommendation
- Improve keyword explanation
- Scalability
- Robustness
- Extend Pdf extraction
  - images
  - tables
- Extend to more than arXiv papers and include book

Contact Information

Anahita Pakiman

p.anahita@gmail.com

Jonathan Abadie

j.abadie.dev@gmail.com