Reverse Mad Libs

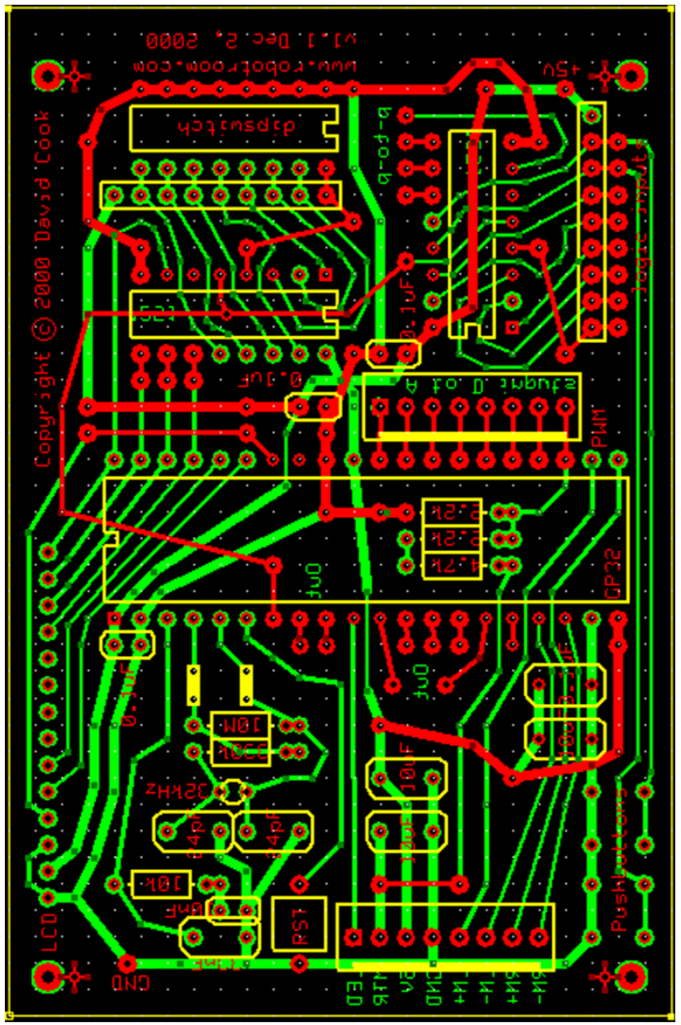
Natural Language Processing

Scott Bing

https://www.quantilus.com/artificial-intelligence-explained-2-2/

Agenda

- Introduction
- What is Natural Language Processing
- A Brief History of NLP
- My Project: Reverse Mad Libs Game
- Resources used
- Questions



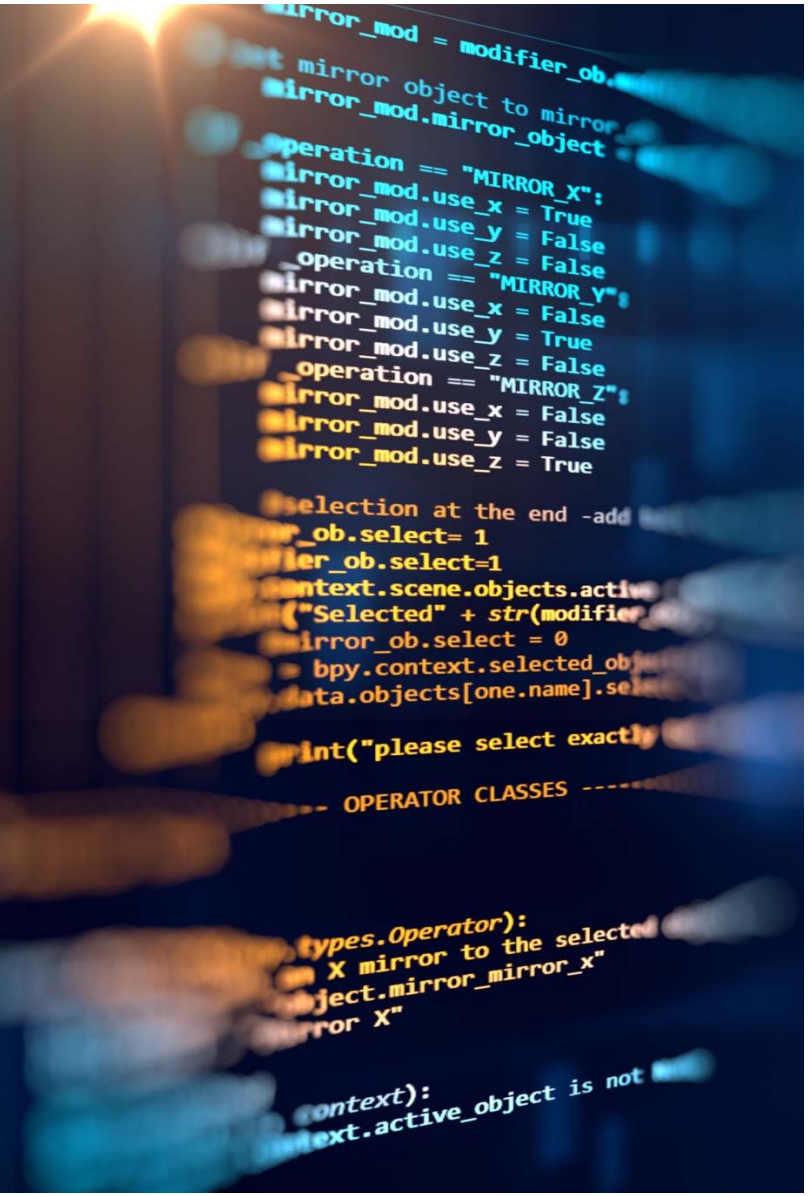
Two Different Worlds

Computer Language

- Structured Data
- ‘1’s and ‘0’s - the language of a computer is Binary
- Adheres to Mathematical Logic and Proofs

Human Language

- Unstructured Data
- Ambiguous
  - Homonyms
  - Homophones
  - Local Idioms
  - Occupational jargon
- Complex
  - Parts of Speech Overlapping
- Its just pain messy



Natural Language Processing

- An attempt to meld human language with computer language.
- Teach a computer how to converse with a human
- Recognize human language
- Computer converse with human
- Final goal – an electronic device can carry on a completely non-rehearsed non-preprogrammed conversation with a human being

NLP Terminology

- NLP – Natural Language Processing – recognizing human language
- NLU – Natural Language Understanding – interpreting human language
- NLG – Natural Language Generation – producing human language

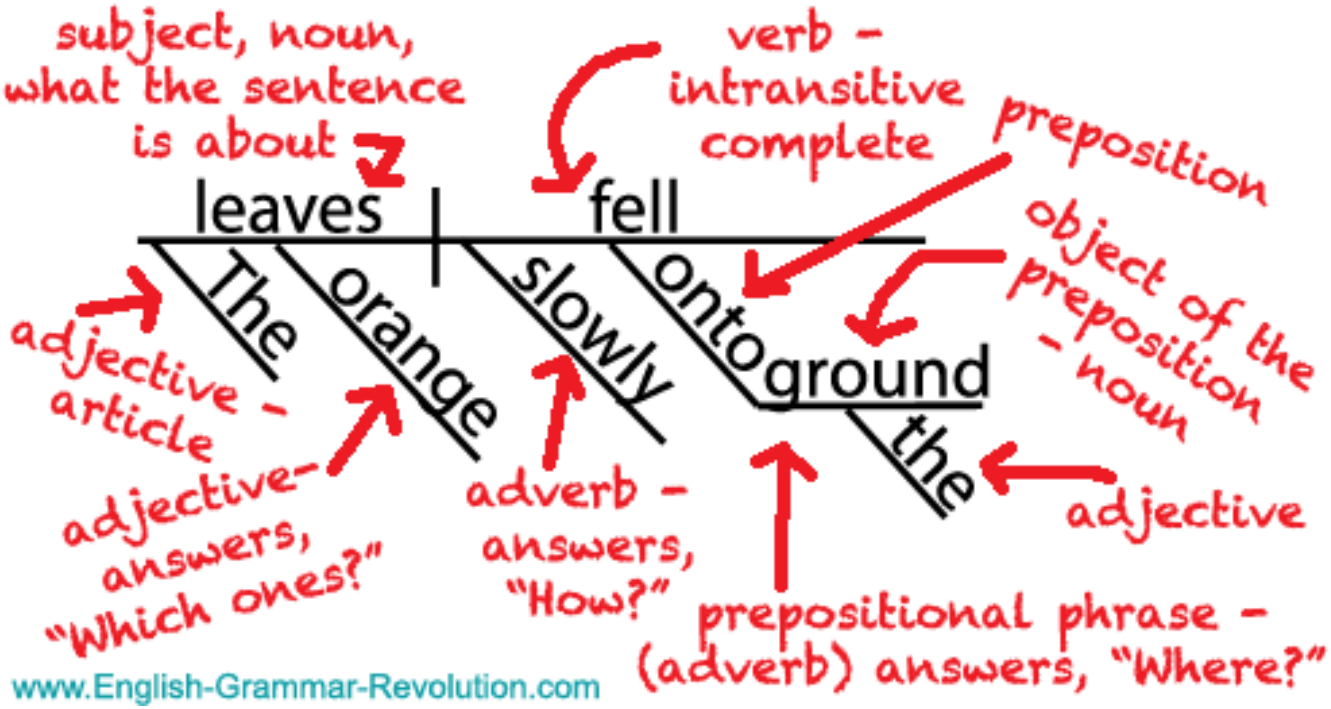
Bert image — sesame street

https://medium.com/@noa.kel/using-bert-with-pytorch-b9624edcda4e

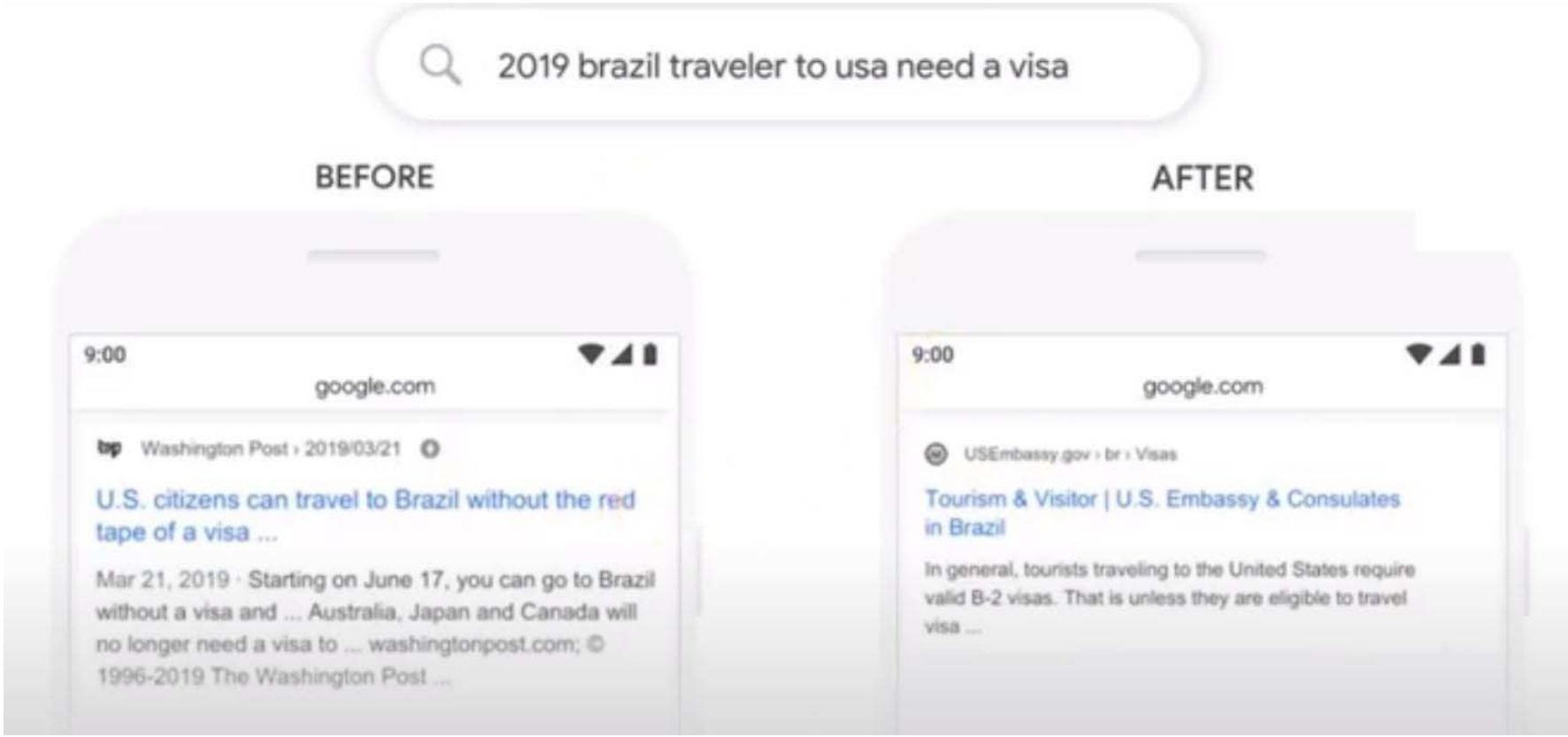
Google BERT

Not this Bert

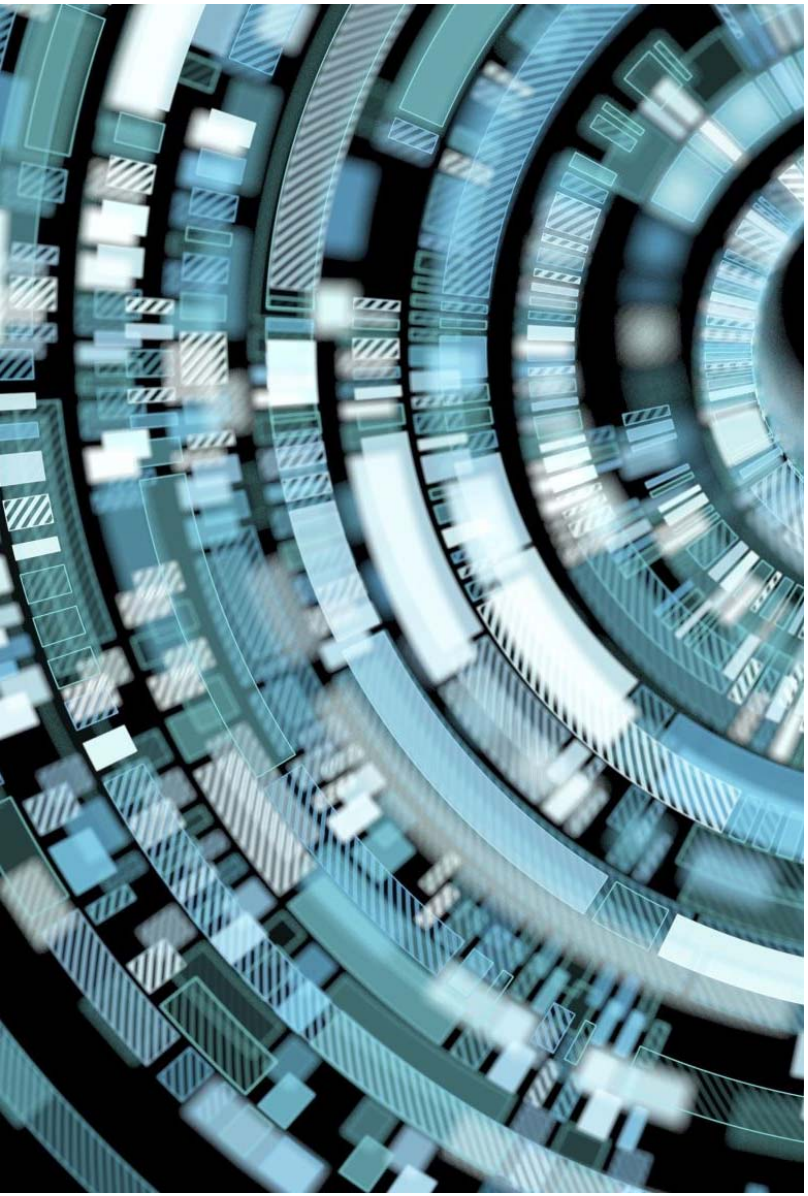
Grade School Nightmare



The Google Search Conundrum



https://www.lockedownseo.com/google-bert-update/



What is BERT?

- Bidirectional Encoder Representations from Transformers
- BERT is a Natural Language Processor facility
- Developed by Google 2018
- A Transformer an object that holds the Encoder and Decoder
- Key innovation is bidirectional scanning procedure

How Bert Works

- Split text to tokens (words)
- Break words into WordPieces
  - ‘calling’ becomes ‘call’, ‘##ing’ ‘preview’ becomes ‘pre##’, ‘view’
- Map words to indexes using a vocab file that BERT provides
- Add special "CLS" and "SEP" tokens (see the readme) CLS is a beginning of the first sentence;.SEP is an end of sentence marker
- Append "index" and "segment" tokens to each input
- Creates a vector representation of each word

Tranformers

- Connects word tokens to context
- Composed of:

  Traditional
  - Encoder - decomposes txt to tokens
  - Decoder – connects words to context

  BERT
  - Encoder - decomposes txt to tokens; connects words to context; makes the final prediction
  - Decoder – presents final result
- Uses a continual internal training model
- Words and sentences are masked forcing the transformers to predict the correct words and prhrases.

Training

- BERT was pre-trained on a large corpus of unlabelled text including the entire Wikipedia(that's 2,500 million words) and book corpus (800 million words). Let's see the 2 training methods that were used to train BERT.[1]

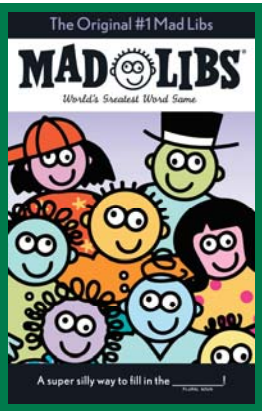
Two BERT Training Models

- **Masked Language Model (MLM)**
  - The model is fed sentences with masked words
  - Goal: predict the masked words
- **Next Sentence Prediction (NSP)**
  - The model is fed a series of two sentences
  - Goal: put the two sentences in the proper order

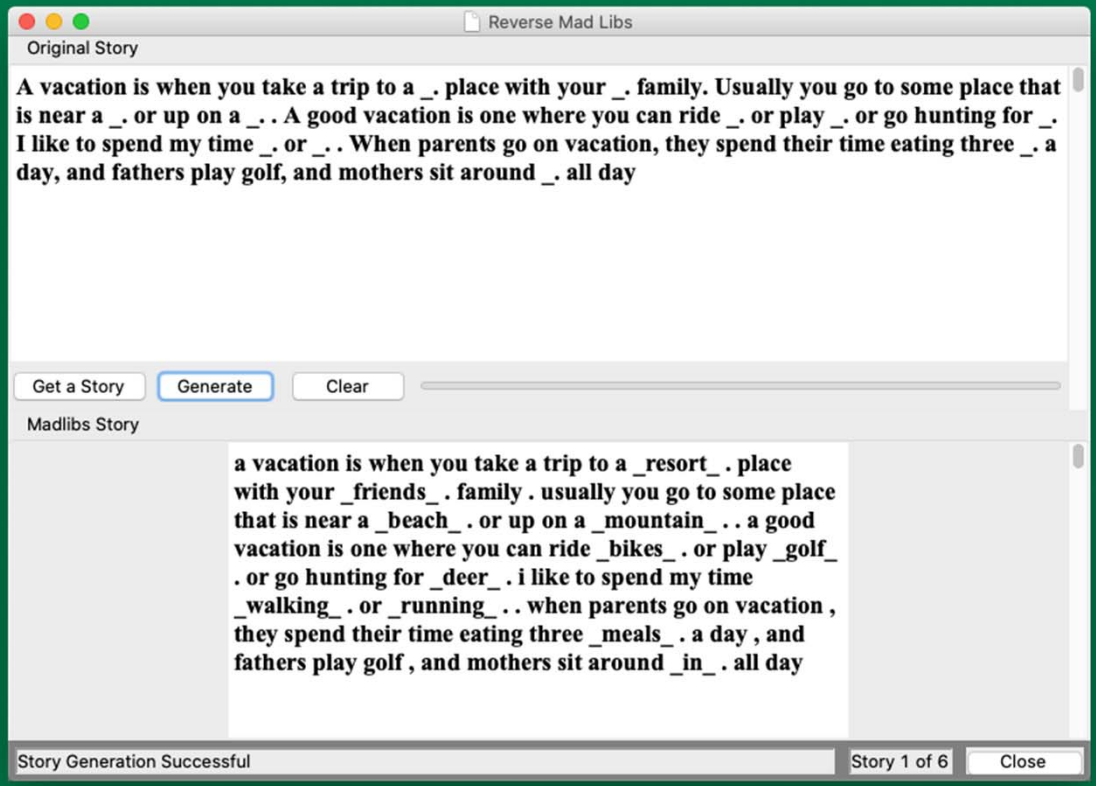
1 Vajpayee Sarthak; Understanding BERT — *(Bidirectional Encoder Representations from Transformers); 2020*

What is Reverse Mad Libs?

- A twist on the popular game
- Player one is asked for a list of arbitrary words
- Player two inserts the words into a story
- Player one typically does not see the story until the words are filled in
- Machine uses NLP Algorithms to predict the missing words



Reverse Mad Libs Application



https://www.amazon.com/Original-1-Mad-Libs/dp/0843100559

Project Resources

Python – Primary implementation Language

Pytorch - Python package for Data Science

Bert – Python NLP Package developed by Google

Tkinter – PythonPackage for GUI development

Improvements to the Application

- Make substitution words more discernable
- Allow the user to select stories from a local file system
- Allow the user to edit the input text file
- Provide a ‘Prepare” routine that randomly selects substitution locations
- Investigate using a more robust trained dataset

Why did I choose Bert

- Ease of installation and use
- Unique Parsing Facility
- Works well with Python

References

- Roni Vorev, BERT Explained: State of the art language model for NLP
- Sarthak Vajpayee, Understanding BERT — (Bidirectional Encoder Representations from Transformers)
- Dipanjan Sarkar, Understanding Feature Engineering: Deep Learning Methods for Text Data
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
- Mohd Sanad Zaki Rizvi, Demystifying BERT: A Comprehensive Guide to the Groundbreaking NLP Framework
- https://www.lockedownseo.com/google-bert-update/

Questions?