AI in NLP and Code Generation

Tianyi Zhang

Nov 28, 2022

CS 57700: Natural Language Processing
CS 59300: Human-AI Interaction

Tianyi Zhang

Assistant Professor of Computer Science

Interactive Intelligent Systems Lab

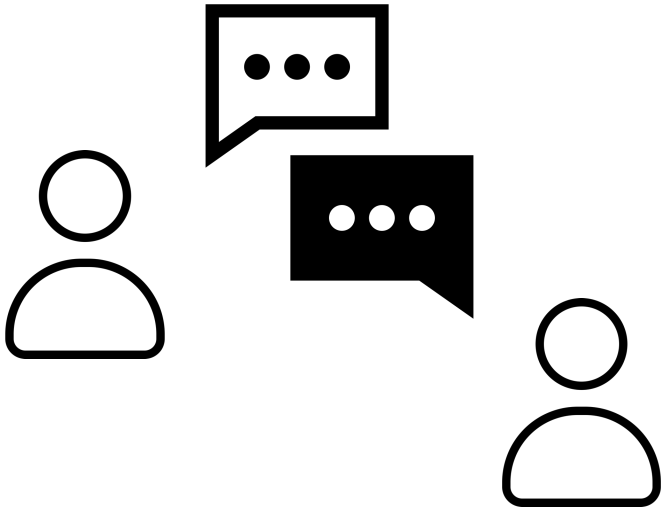
LWSN 3154H • tianyi@purdue.edu • https://tianyi-zhang.github.io/

Research Interests: HCI, SE, AI

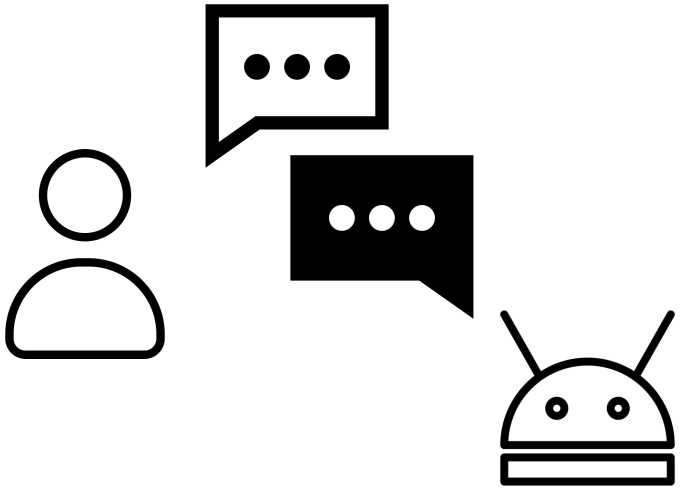
In IIS, we develop interactive intelligent systems to:

- **improve the productivity of programmers and domain experts**
  - Interactive program synthesis and program repair
  - Knowledge acquisition from Stack Overflow
  - Interactive web scraping and data labeling for data scientists
- **improve the robustness, transparency, and fairness of ML**
  - Declarative testing and traffic scene generation for autonomous driving
  - Interactive model debugging for RNNs and transformers
  - Interactive visual analytics for detecting biases and Out-of-Distribution (OOD) errors

Natural Language Processing



Human-Human Communication



Human-Machine Communication

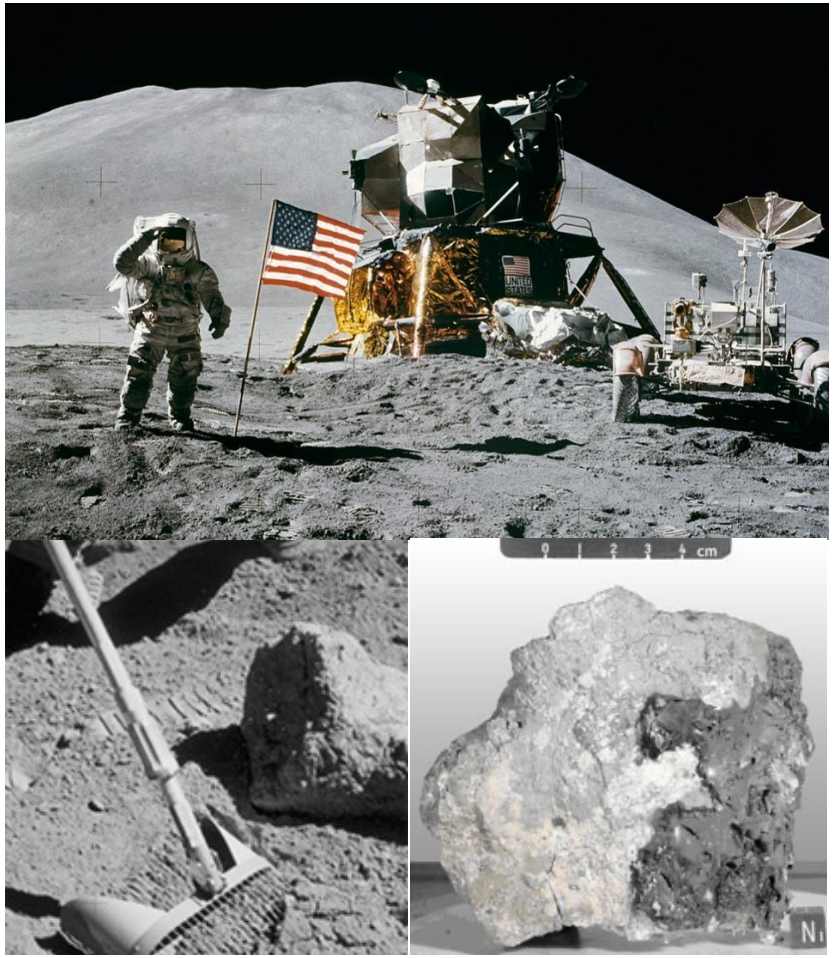
Natural Language Processing

> NLP is a branch of artificial intelligence that concerns with **the interactions between computers and human languages**, in particular how to enable computers **analyze**, **understand**, and **generate** human languages in both written and spoken contexts.

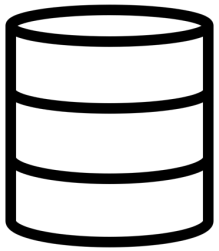
Eliza (1964)

```
  Welcome to
                    EEEEEE  LL      IIII   ZZZZZZ   AAAAA
                    EE      LL       II       ZZ   AA   AA
                    EEEEE   LL       II     ZZZ    AAAAAAA
                    EE      LL       II    ZZ      AA   AA
                    EEEEEE  LLLLLL  IIII ZZZZZZ    AA   AA

  Eliza is a mock Rogerian psychotherapist.
  The original program was described by Joseph Weizenbaum in 1966.
  This implementation by Norbert Landsteiner 2005.


ELIZA: Is something troubling you ?
YOU:   Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU:   They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU:   Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU:   He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU:   It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU:
```

Lunar (1971)

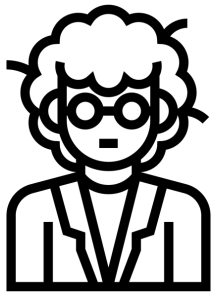


Chemical analysis
of rock samples
from the moon



database

Natural Language Query

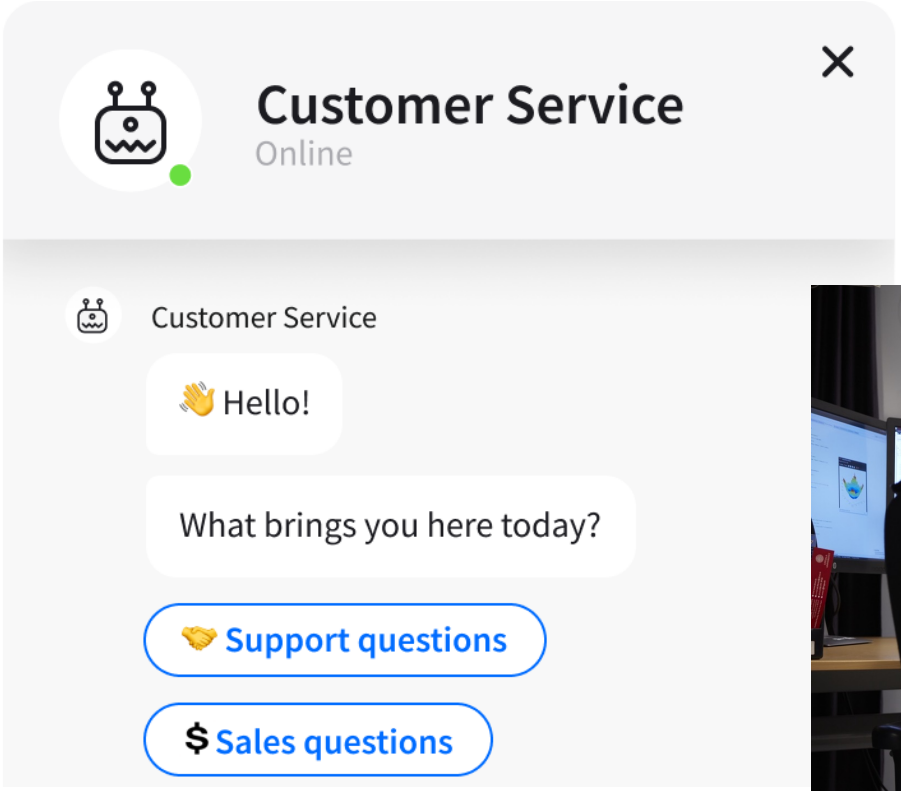


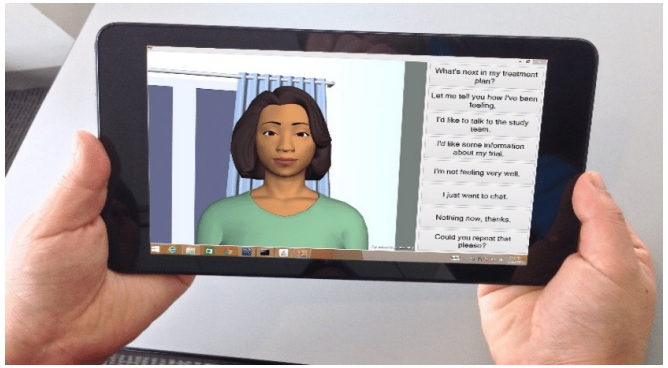
geologist

- *Give me all analyses for Hydrogen in Sample 10046.*
- *In which samples has apatite been identified?*
- *What are the plag analyses for breccias?*
- *What is the average age of the basalts?*
- *…*

Apple Knowledge Navigator (1987)

NLP after 2010





GitHub Copilot: Your AI Pair Programmer

sentiment.ts | write_sql.go | parse_expenses.py | addresses.rb

```python
import datetime

def parse_expenses(expenses_string):
    """Parse the list of expenses and return the list of triples (date, value, currency).
    Ignore lines starting with #.
    Parse the date using datetime.
    Example expenses_string:
        2016-01-02 -34.01 USD
        2016-01-03 2.59 DKK
        2016-01-03 -2.72 EUR
    """
    expenses = []
    for line in expenses_string.splitlines():
        if line.startswith("#"):
            continue
        date, value, currency = line.split(" ")
        expenses.append((datetime.datetime.strptime(date, "%Y-%m-%d"),
                         float(value),
                         currency))
    return expenses
```

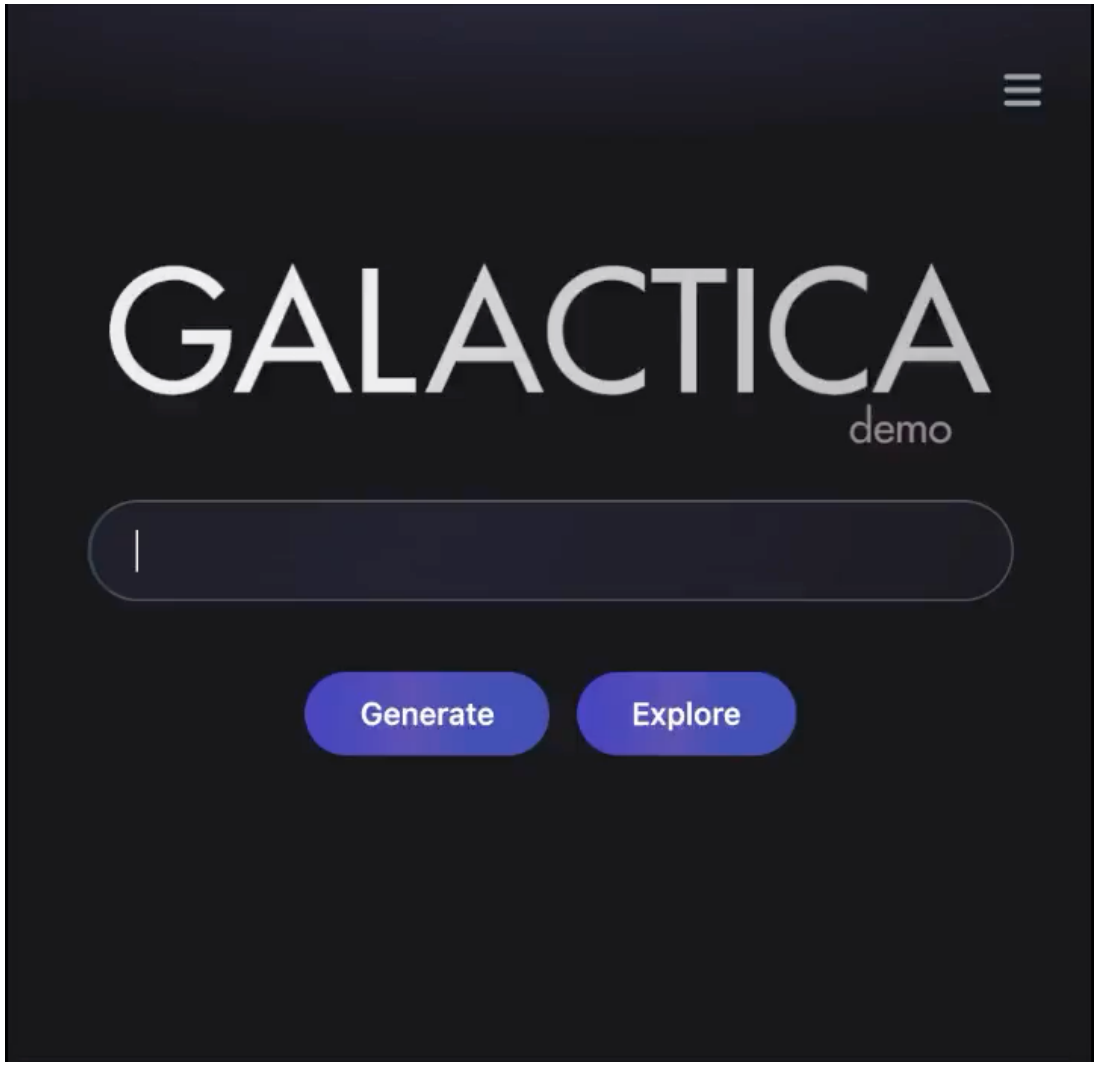
Copilot

OpenAI

Microsoft

GitHub

Galactica: A Language Model for Science



Meta

NLP History

- Symbolic NLP (1950s – early 1990s)
  - Grammars, rules, ontologies, knowledge base, etc.
- Statistical NLP (1990s – 2010s)
  - Learn from a text corpus
  - Hidden markov models, probabilistic grammars, TF-IDF, LDA, SVM, etc.
- Neural NLP (2010s – present)
  - Deep neural networks and representation learning

Tasks and Applications in NLP

Understanding

- Sentiment analysis
- Speech recognition
- Topic modeling
- Text classification
- Natural language inference
- Semantic parsing
- Spam detection
- Named entity recognition
- Relation extraction
- etc.

Generation

- Chatbots
- Question answering
- Text summarization
- Image captioning
- Machine translation
- Natural language interfaces
- Code generation
- Text completion
- Creative writing
- etc.

Fundamental Methods in NLP

- Text preprocessing
  - Lowercase, tokenization, stop words removal, stemming, lemmatization

Original Word	After Stemming
program	program
programs	program
programmed	program
programming	program

Chopping off suffixes based on rules

Original Word	After Lemmatization
is	be
are	be
better	good
programming	program

Reducing each word to its base form

Fundamental Methods in NLP

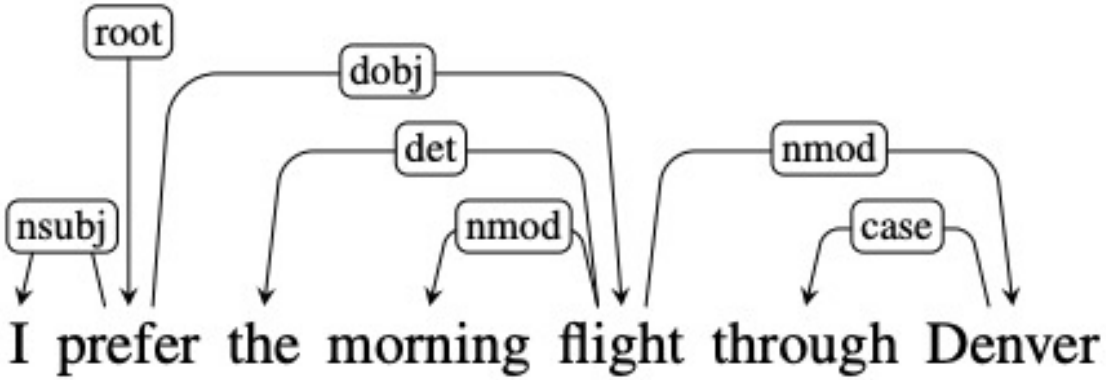
- Part of Speech (POS) Tagging

I	like	programming	in	Python.
↓	↘	↓	↓	↓
Pronoun	Verb	Verb	Preposition	Noun

- Applications of POS tagging
  - Named entity recognition, sentiment analysis, question answering, etc.

Fundamental Methods in NLP

- Dependency Parsing



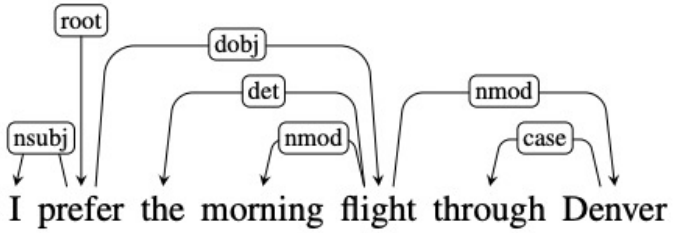
nsubj: nominal subject
dobj: direct object

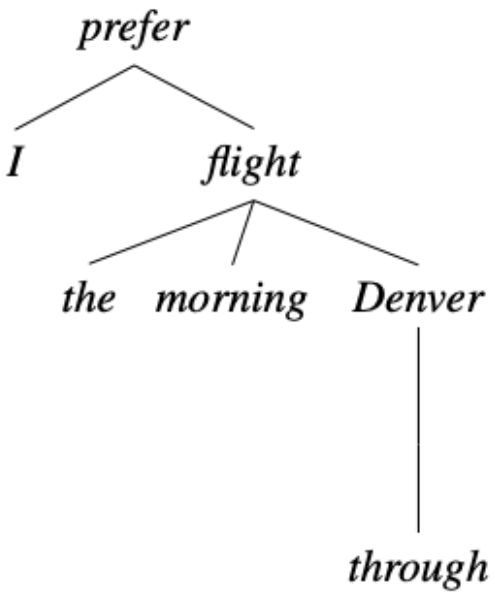
det: determiner
nmod: nominal modifier

case: prepositions, postpositions, and other case markers

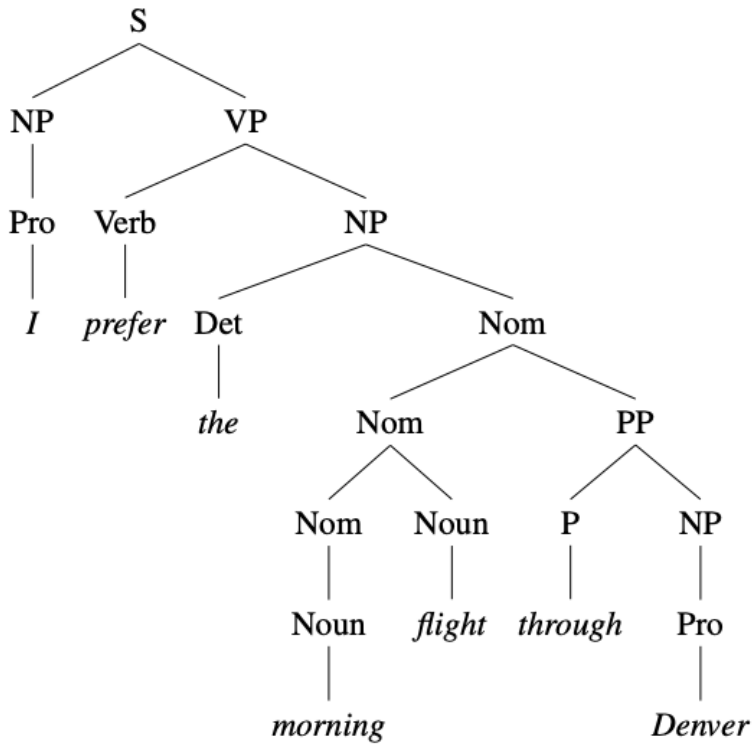
Fundamental Methods in NLP

- Dependency Parsing





Dependency Tree



Phrase-Structure Tree

Fundamental Methods in NLP

- Text preprocessing
  - Lowercase, tokenization, stop words removal, stemming, lemmatization
- Part of Speech (POS) tagging
- Dependency parsing

Commonly used in symbolic NLP

- Vectorization

Commonly used in statistical and neural NLP

Vectorization Methods

- Bag of Words
  - Build a dictionary from a corpus and convert a sentence to an array of 0 and 1

S1: Without music life would be a mistake

S2: Radiohead are a great music band

	without	*music*	*life*	*would*	*be*	*a*	*mistake*	*Radiohead*	*are*	*great*	*band*
S1	1	1	1	1	1	1	1	0	0	0	0
S2	0	1	0	0	0	1	0	1	1	1	1

Vectorization Methods

- TF-IDF
  - Term frequency: how likely to find a word in the corpus?
  - Inverse document frequency: how unique is a word in the corpus?

$$w_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

$tf_{x,y}$: frequency of x in y

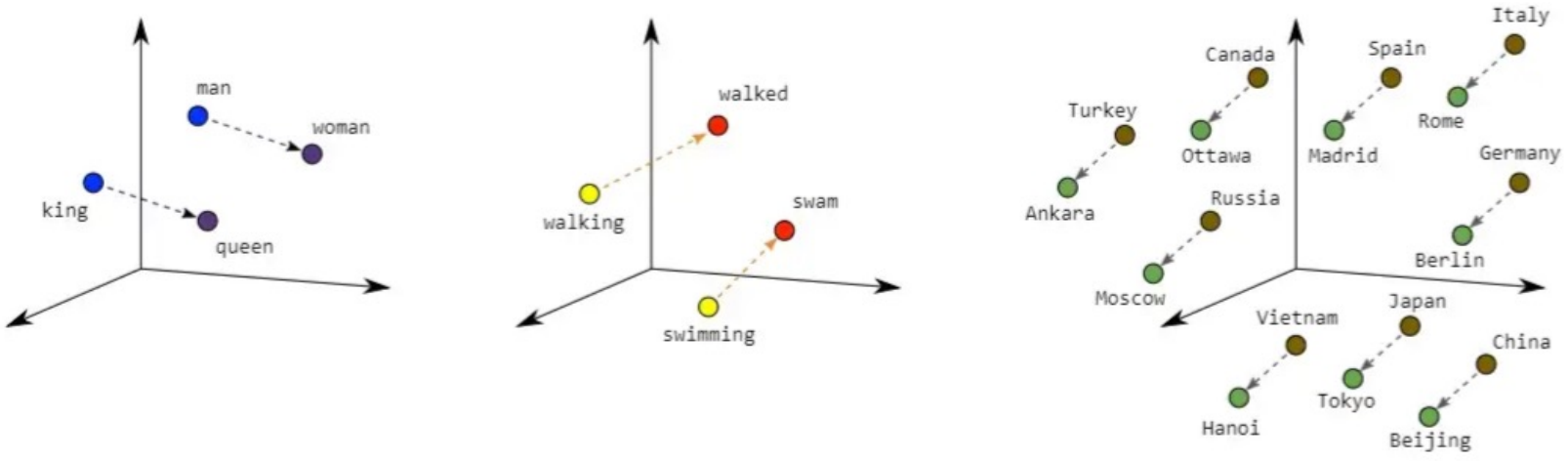
df_x: number of documents containing x

N: total number of documents

	without	music	life	would	be	a	mistake	Radiohead	are	great	band
S1	0.3	0	0.3	0.3	0.3	0	0.3	0	0	0	0
S2	0	0	0	0	0	0	0	0.3	0.3	0.3	0.3

Vectorization Methods

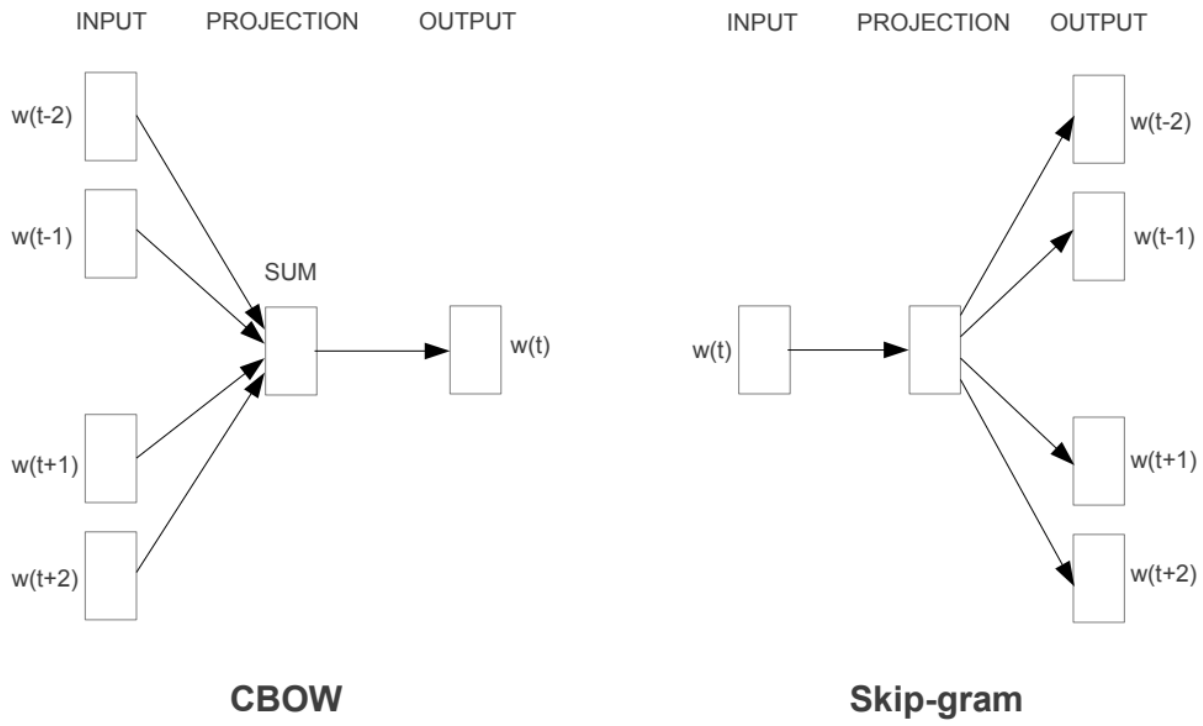
- Word embedding
  - Map words to a high-dimensional space where similar words are close to each other



Male-Female Verb Tense Country-Capital

Vectorization Methods

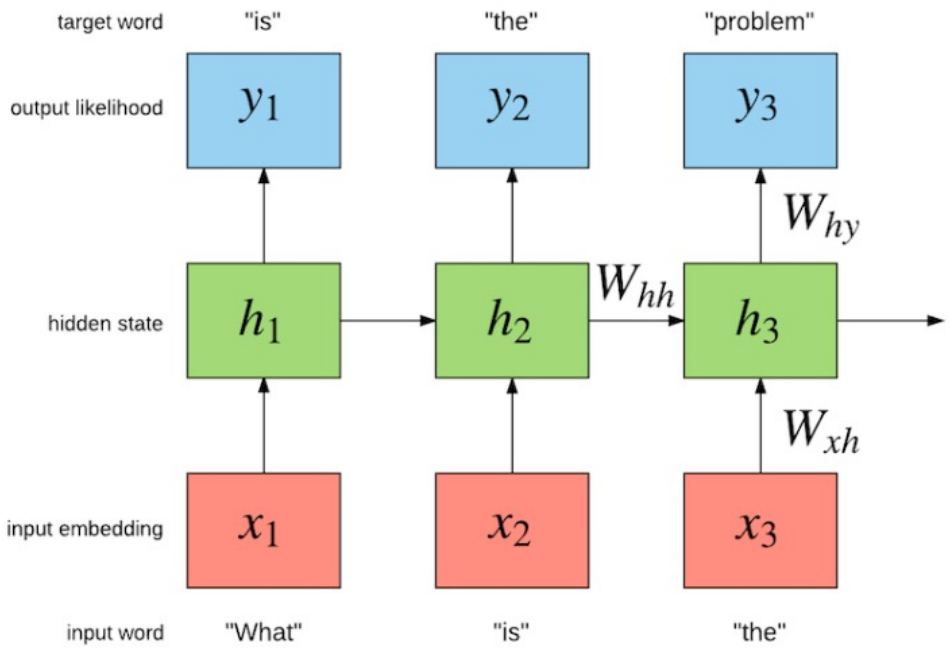
- Learning word embeddings



Mikolov et al. Efficient Estimation of Word Representations in Vector Space. Arxiv 2013.

Vectorization Methods

- Language models (or contextualized word embeddings)
  - Build a vector for each word conditioned on its context



NLP Libraries and Pre-trained Models

- spaCy
- NLTK
- Stanford CoreNLP
- TextBlob
- Gensim
- Hugging Face

Semantic Parsing

- The task of converting a NL utterance to a logical form, e.g., SQL

database

Natural Language Query

geologist

Semantic Parsing

- Slot-filling systems
  - Rule-based intent detection
  - Template-based code generation

Shallow semantic parsing

- Neural machine translation (Seq2Seq)
  - Encoder-decoder model architecture
  - Attention-based, e.g., transformers!

Deep semantic parsing

Grammar-based Semantic Parsing

- Combinatory Categorical Grammar (CCG)
  - A lexicon and a set of grammar rules

syntactic type

semantic type

Utah := $NP : utah$

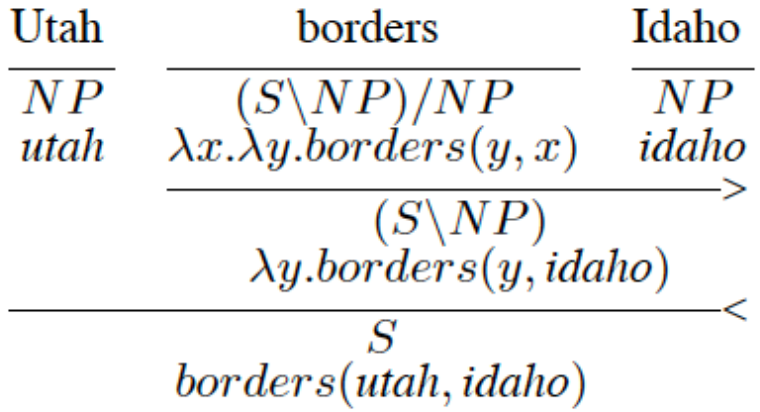
Idaho := $NP : idaho$

borders := $(S\backslash NP)/NP : \lambda x.\lambda y.borders(y,x)$

A simple lexicon

$$
\begin{array}{llll}
A/B : f & B : g & \Rightarrow & A : f(g) \\
B : g & A\backslash B : f & \Rightarrow & A : f(g)
\end{array}
$$

Simple functional application rules



A derivation tree of *"Utah boarders Idaho"*

Azaria et al. *Instructable Intelligent Personal Agent*. AAAI 2016.

CCG Semantic Parsing

- Combinatory Categorical Grammar (CCG)
  - A lexicon and a set of grammar rules

Word	Syntactic Category	Logical Form
set	((S/PP_StringV)/MutableField)	(lambda x y (setFieldFromString x y))
to	PP_StringV/StringV	(lambda x x)
subject	FieldName	subject
send	S/InstanceName	(lambda x (send x))
email	InstanceName	email
set	((S/PP_FieldVal)/MutableField)	(lambda x y (setFieldFromFieldVal x y))
to	PP_FieldVal/FieldVal	(lambda x x)

Azaria et al. *Instructable Intelligent Personal Agent*. AAAI 2016.

PCCG Semantic Parsing

- Extend CCG with a probabilistic model $P(L, T|S)$
  - A conditional distribution over possible *(L, T)* pairs for a given sentence *S*
- Parameterized by $\boldsymbol{\theta}$

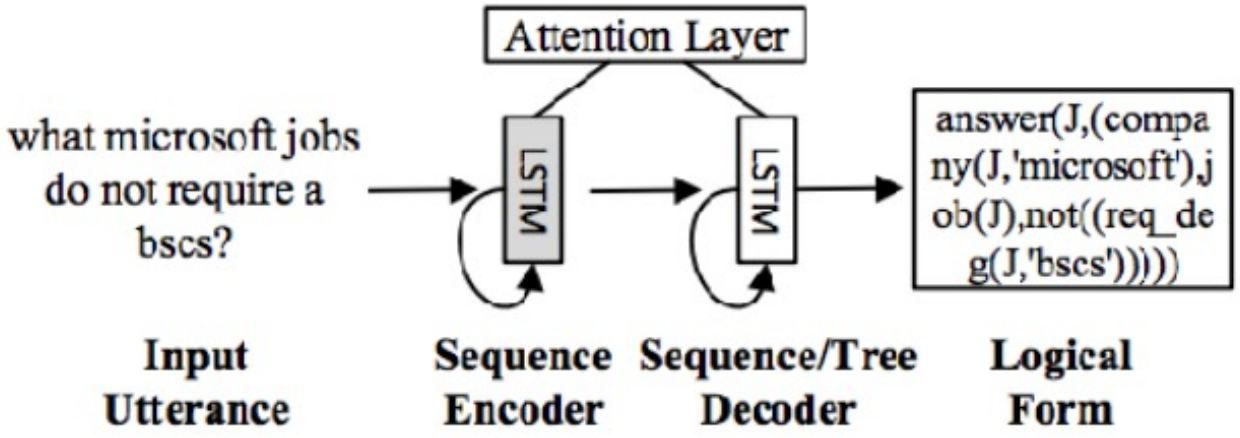
$$P(L,T|S;\bar{\theta}) = \frac{e^{\bar{f}(L,T,S)\cdot\bar{\theta}}}{\sum_{(L,T)} e^{\bar{f}(L,T,S)\cdot\bar{\theta}}} \qquad \arg\max_L P(L|S;\bar{\theta}) = \arg\max_L \sum_T P(L,T|S;\bar{\theta})$$

- Handle ambiguity in natural language

Azaria et al. *Instructable Intelligent Personal Agent*. AAAI 2016.

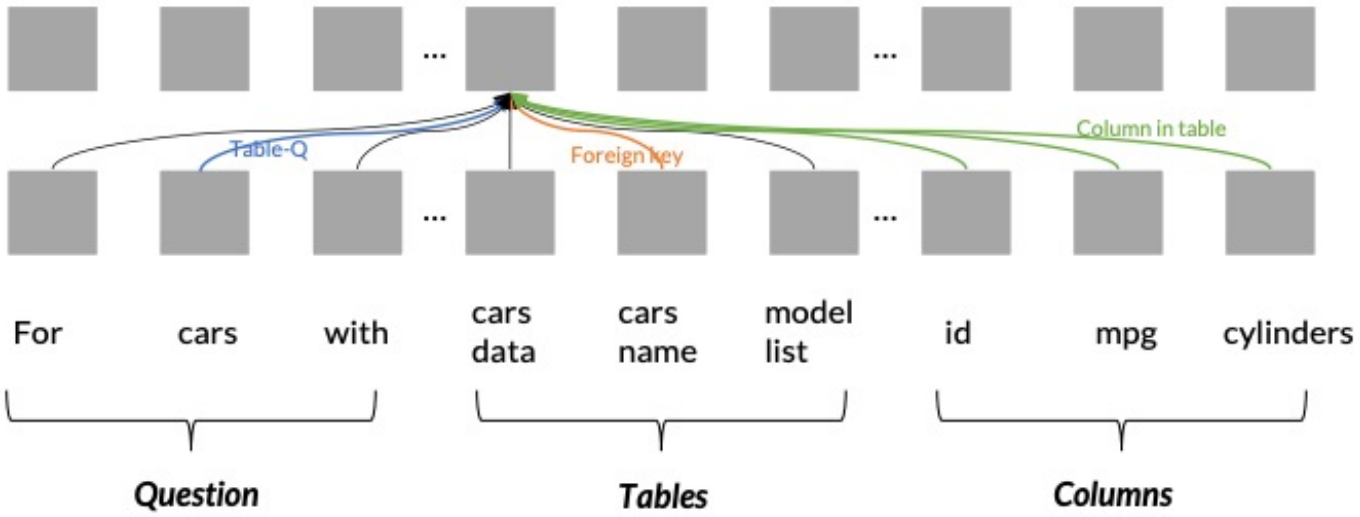
Neural-based Semantic Parsing

- Encoder: understand the meaning of the input sentence
- Decoder: generate the corresponding logic form



Relation-Aware SQL Generation

- Encode database schemas via a relation-aware transformer



$$\alpha_{ij} = \mathrm{softmax}_j \frac{\boldsymbol{q}_i \, \boldsymbol{k}_j^\top}{\sqrt{\mathrm{dim}}}$$

$$\boldsymbol{y}_i = \sum_j \alpha_{ij} \, \boldsymbol{v}_j$$

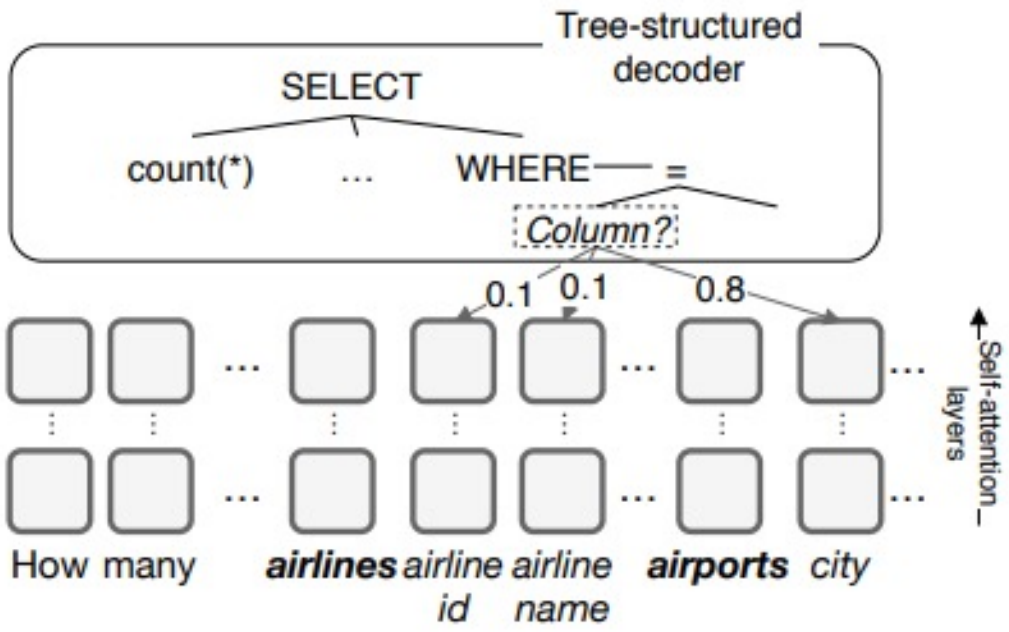
$$\alpha_{ij} = \mathrm{softmax}_j \frac{\boldsymbol{q}_i \left(\boldsymbol{k}_j + \boldsymbol{\beta}_{ij}\right)^\top}{\sqrt{\mathrm{dim}}}$$

$$\boldsymbol{y}_i = \sum_j \alpha_{ij} \left(\boldsymbol{v}_j + \boldsymbol{\varepsilon}_{ij}\right)$$

Wang et al. *RAT-SQL: Relation-Aware Schema Encoding and Linking for Text-to-SQL Parsers.* ACL 2020.

Relation-Aware SQL Generation

- Tree-structured decoder
- Guided by SQL grammar
- Predict a derivation rule at a time, not a token



Wang et al. *RAT-SQL: Relation-Aware Schema Encoding and Linking for Text-to-SQL Parsers.* ACL 2020.