Natural Language Processing and Linear Algebra

Contact for full presentation

loganpride525@email.com

What is Natural Language Processing?

Natural Language Processing (NLP) is a tool that falls in the world of Artificial Intelligence (AI). The purpose of NLP is to help aid computers/machines in

Contact for full presentation

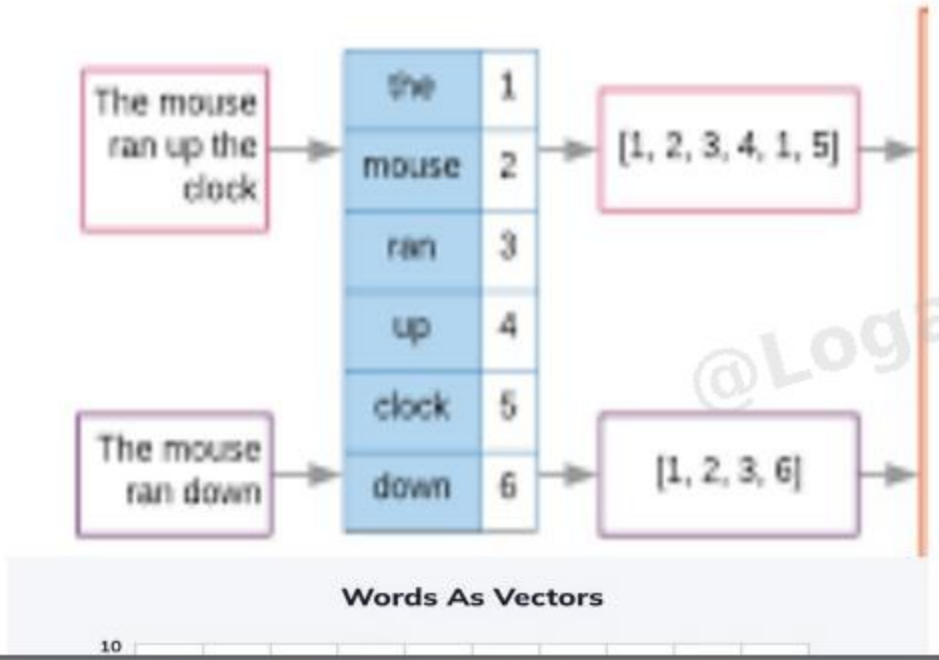
loganpride525@email.com

NLP techniques and Application

NLP techniques: Tokenization, Part-of-speech tagging, Named entity recognition,Word embeddings: and many more!

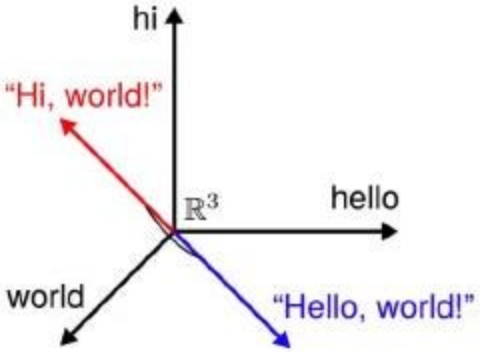
NLP and Linear algebra connection

- Linear algebra concepts such as

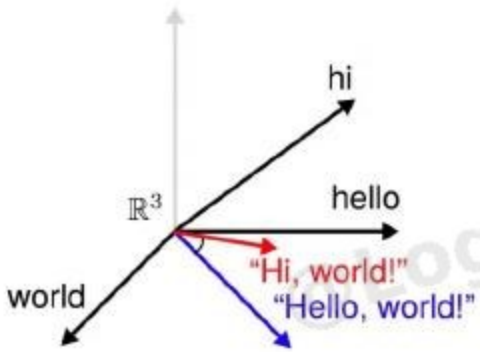


Vectors in NLP

- Vector representation in NLP is represented as a method to store, organize and use data point such as words, sentences and text
- Vectors exhibit properties such as closure under vector addition and scalar multiplication, which are foundational concepts used in operations involving word vectors and sentence vectors in NLP tasks.
  - Vector Addition: When adding two



Cosine Similarity



Soft Cosine Measure

Source: https://github.com/RaRe-Technologies/gensim/blob/develop/docs/notebooks/soft_cosine_tutorial.ipynb

Word Embedding

- Word embeddings transform words into numerical vectors, simplifying their representation for machines to comprehend, store, and manipulate.
- These embeddings, operating in a high-dimensional space, encode semantic meaning and relationships among words, enabling the capture of

Contact for full presentation

loganpride525@email.com

Document Term Matrix

Represents documents as rows and words as columns, where each entry indicates the

Term-Term Co-occurrence Matrix

Represent the frequency or occurrence of each word (term) in relation to other words within a text. Where each row and column correspond to a word, and the cells contain counts or measures of how often those words appear together within a

Term-Document Matrix

Represents terms (words or phrases) arranged as rows and documents are represented as columns. Each cell in the matrix contains a value that represents the frequency or weight of a particular

Contact for full presentation

loganpride525@email.com

Hidden Markov Modeling (HMM)

- Hidden Markov Modeling
  - Type of Markov model where the state of the model are hidden and the output (observation) is observed
  - HMMs are skilled in modeling sequential data, where each observation depends on the

Language Models in NLP

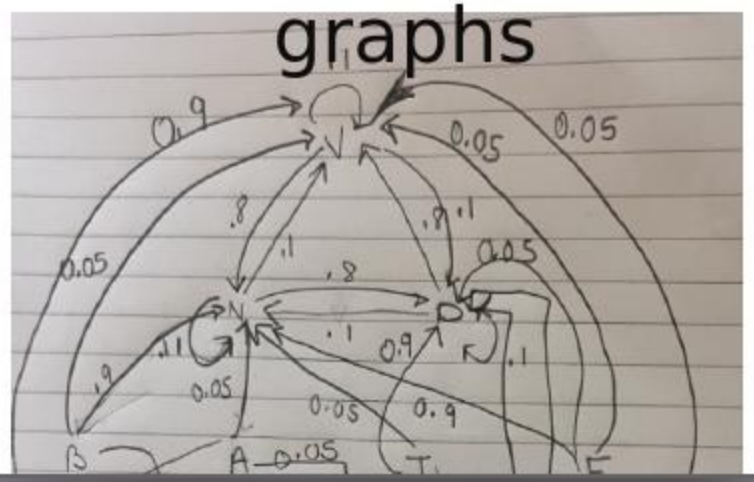
- Language model is
  - Models are computational frameworks and algorithms that purpose is to create to helpful method for compilers understand, process and generate or analyze human

Contact for full presentation

loganpride525@email.com

Defining variables in Hidden Markov Models/Creating Matrices and Vectors

Directed Weighted graphs



Matrices and

```
obs = ['Bob','ate','the','fruit']

# generate new sequences

states = ['Noun','Verb','Determiner']
initial_probs = {'Noun':0.9,'Verb':0.05,'Determiner':0.05}
transition_probs =
{'Noun|Noun':0.1,'Noun|Verb':0.1,'Noun|Determiner':0.8,

'Verb|Noun':0.8,'Verb|Verb':0.1,'Verb|Determiner':0.1,
```


Contact for full presentation

loganpride525@email.com

2. Printing Model Parameters

- This segment prints out matrices for the initial, transition, and emission probabilities

```
print("Initial Distributions")
print initial_probs
```

Visual Look of the matrices that will be performing operations

```
Initial Distributions
{'Verb': 0.05, 'Noun': 0.9, 'Determiner': 0.05}

Transition Probabilities
+-----------+-------+-------+-------------+
| rows      |  Verb |  Noun |  Determiner |
|-----------+-------+-------+-------------|
| Verb      |   0.1 |   0.8 |         0.1 |
| Noun      |   0.1 |   0.1 |         0.8 |
| Determiner|   0.8 |   0.1 |         0.1 |
+-----------+-------+-------+-------------+

Emission Probabilities
```


Contact for full presentation

loganpride525@email.com

Generating probabilities of the sequences

This segment generates the scores using HMM forward algorithm, which involves matrix and vector multiplication.

```
# Generate list of sequences
sequence_length,sequences,sequence_scores = initializeSequences(obs)
```

```
Initialization
Probability Noun    Bob    0.9   0.9   0.81
Probability Verb    Bob    0.05   0.05   0.0025
Probability Determiner    Bob    0.05   0.05   0.0025

Recursion at step 1
Probability Noun    ate     0.81   0.1   0.05     0.0025   0.8   0.05     0.0025   0.1   0.05     0.004125
Probability Verb    ate     0.81   0.8   0.9      0.0025   0.1   0.9      0.0025   0.1   0.9      0.65025
Probability Determiner    ate     0.81   0.1   0.05     0.0025   0.8   0.05     0.0025   0.1   0.05
0.004125

Recursion at step 2
Probability Noun    the     0.004125   0.1   0.05     0.65025   0.8   0.05     0.004125   0.1   0.05
0.026290625
Probability Verb    the     0.004125   0.1   0.05     0.65025   0.1   0.05     0.004125   0.8   0.05
0.026290625
Probability Determiner    the     0.004125   0.1   0.9      0.65025   0.1   0.9      0.004125   0.8   0.9
```


Here is an output example of the forward algorithm put to use just for the sequence of “noun verb determiner deterimer”

- The steps of forward algorithm are:
- Starts with using dynamic programming to compute the probability of observing a sequence of symbols efficiently.
- At each time step, the algorithm calculates the probability of being in each state based on transitions from

Results from the function

5.Printing Best Sequence:

- This part prints out the sequence with the highest score, which represents the most probable sequence of hidden states given the observed sequence

```
# Display sequence scores
for i in range(len(sequences)):
    print("Sequence:%-60s Score:%0.6f" %
(sequences[i],sequence_scores[i]))

# Display the winning score
print("\n Best Sequence")
print(sequences[sequence_scores.index(max(sequence_scores))],max(sequ
ence_scores))
```

- The sequence ['Noun', 'Verb',

```
Scores
Sequence:['Noun', 'Noun', 'Noun', 'Noun']
Score:0.000002
Sequence:['Noun', 'Noun', 'Noun', 'Verb']
```

Codes like the Hidden Markov Model (HMM)

Contact for full presentation

loganpride525@email.com

In Summary

Contact for full presentation

loganpride525@email.com

Resources

- https://medium.com/@ompramod9921/hidden-markov-models-the-secret-sauce-in-natural-language-processing-2d22143c9d04
- https://www.youtube.com/watch?v=RWkHJnFj5rY
- https://medium.com/@postsanjay/hidden-markov-models-simplified-c3f58728caab#:~:text=A%20simple%20example%20of%20an,best%20sequence%20of

Contact for full presentation

loganpride525@email.com