Text Generation Using Language Model

Problem Statement

- To develop a text generation system which should take in a seed text from the user as input and generate a sequence of a specified number of words.

Problem Statement

- To develop a text generation system which should take in a seed text from the user as input and generate a sequence of a specified number of words.

  - Seed text = "how are", number of words to generate = 2

Problem Statement

- To develop a text generation system which should take in a seed text from the user as input and generate a sequence of a specified number of words.

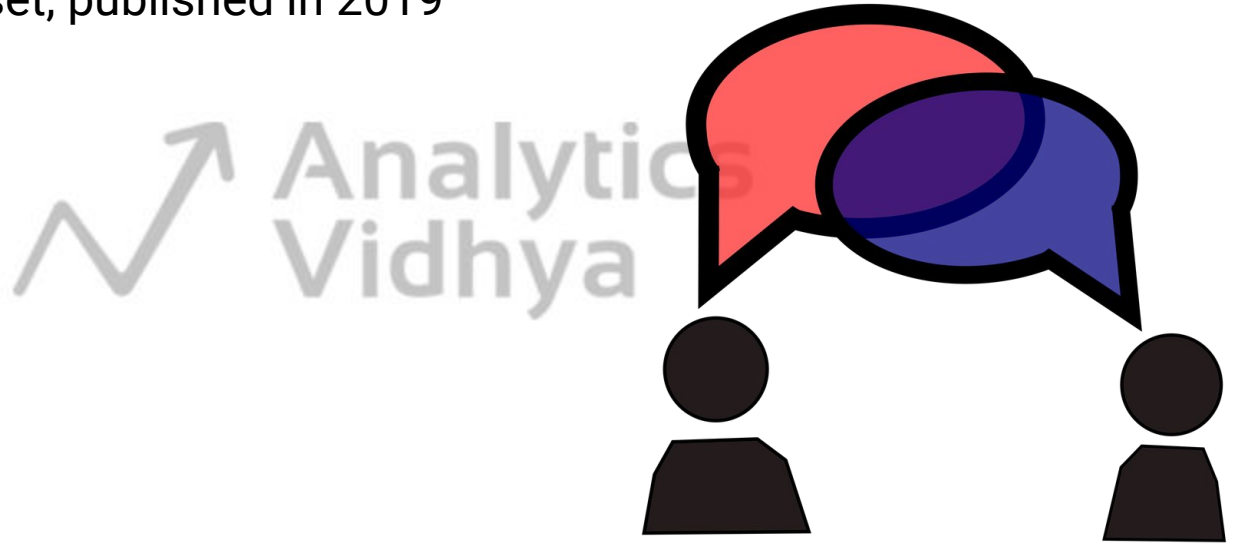 - Seed text = "how are", number of words to generate = 2
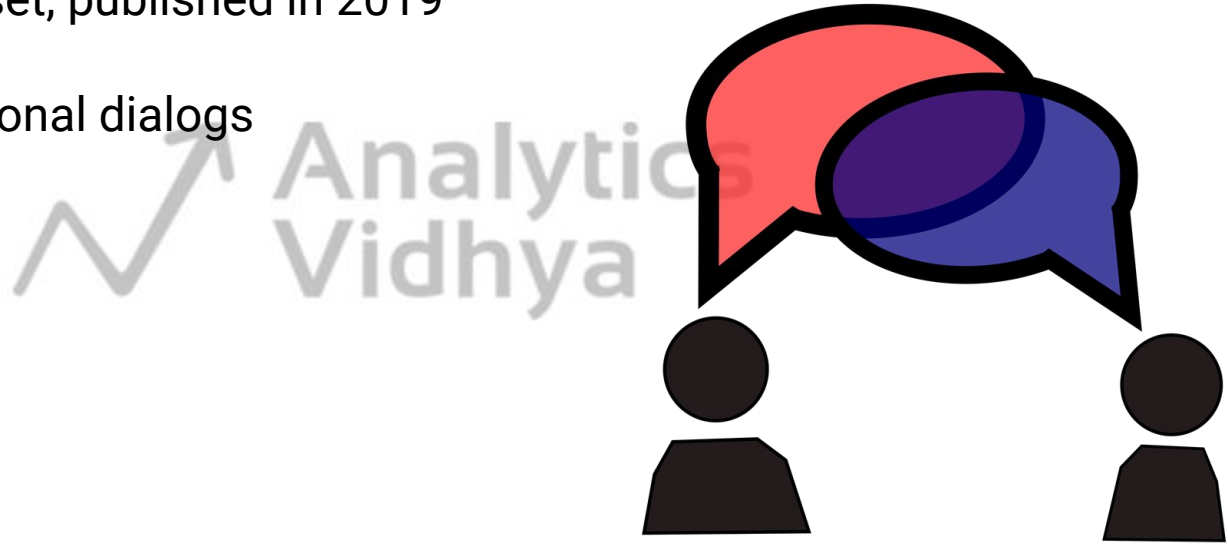
  - Output = "how are you doing"

About the Dataset

- Taskmaster Dataset, published in 2019

About the Dataset

- Taskmaster Dataset, published in 2019

- 64,777 conversational dialogs

About the Dataset

- Taskmaster Dataset, published in 2019

- 64,777 conversational dialogs

- Six domains:
  - ordering pizza,
  - creating auto repair appointments,
  - setting up ride service,
  - ordering movie tickets,
  - ordering coffee drinks and
  - making restaurant reservations

Data Preparation

- ['alright that is perfect',
  'sounds great',
  'what is the price difference']

Data Preparation

- ['alright that is perfect',
  'sounds great',
  'what is the price difference']

x1	x2	y
'alright'	'that'	'is'
'that'	'is'	'perfect'

Data Preparation

- ['alright that is perfect',
  'sounds great',
  'what is the price difference']

- ['alright that is',
  'that is perfect',
  'sounds great',
  'what is the',
  'is the price',
  'the price difference']

Data Preparation

- ['alright that is perfect',
  'sounds great',
  'what is the price difference']

- ['alright that is',
  'that is perfect',
  'sounds great',   ⟹ **length = 2**
  'what is the',
  'is the price',
  'the price difference']

Data Preparation

- ['alright that is perfect',
  'sounds great',
  'what is the price difference']

- ['alright that is',
  'that is perfect',
  'sounds great <pad>',
  'what is the',
  'is the price',
  'the price difference']

Data Preparation

- Vocabulary:

  "<pad>"
  "alright"
  "difference"
  "great"
  "is"
  "perfect"
  "price"
  "sounds"
  "that"
  "the"
  "what"

Data Preparation

- Token to integer mapping:

  "<pad>"   - 0
  "alright"   - 1
  "difference"   - 2
  "great"   - 3
  "is"   - 4
  "perfect"   - 5
  "price"   - 6
  "sounds"   - 7
  "that"   - 8
  "the"   - 9
  "what"   - 10

Data Preparation

- Converting tokens to integers:

"<pad>"	- 0
"alright"	- 1
"difference"	- 2
"great"	- 3
"is"	- 4
"perfect"	- 5
"price"	- 6
"sounds"	- 7
"that"	- 8
"the"	- 9
"what"	- 10

['alright that is',
'that is perfect',
'sounds great <pad>',
'what is the',
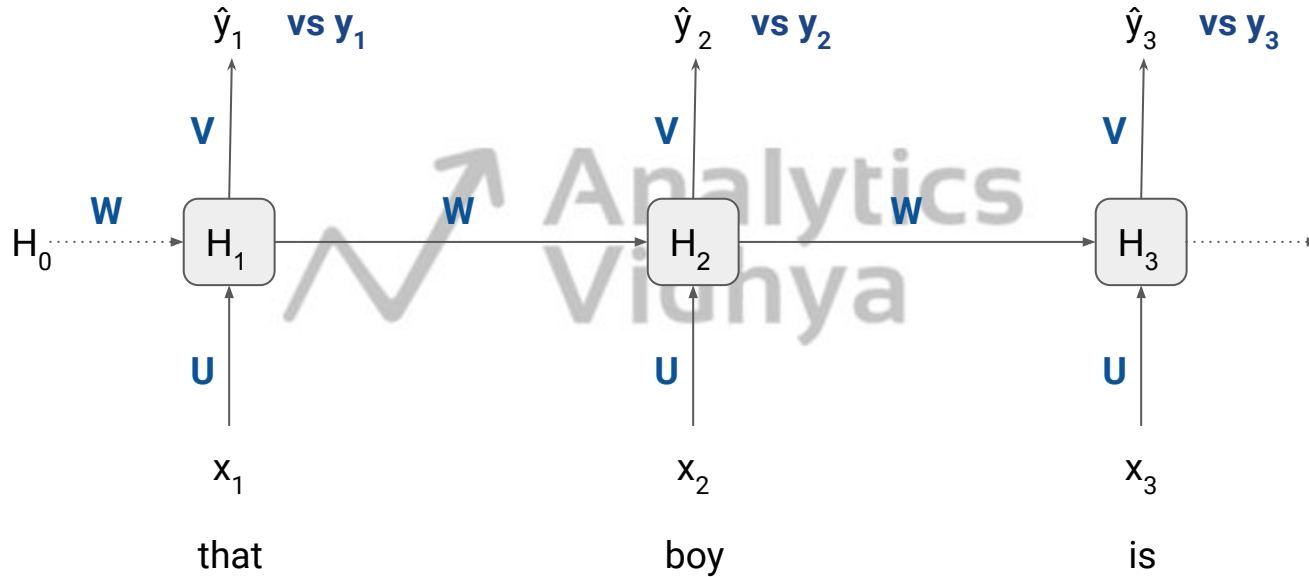'is the price',
'the price difference']

Data Preparation

- Converting tokens to integers:

"<pad>"	- 0
"alright"	- 1
"difference"	- 2
"great"	- 3
"is"	- 4
"perfect"	- 5
"price"	- 6
"sounds"	- 7
"that"	- 8
"the"	- 9
"what"	- 10

[[1, 8, 4],
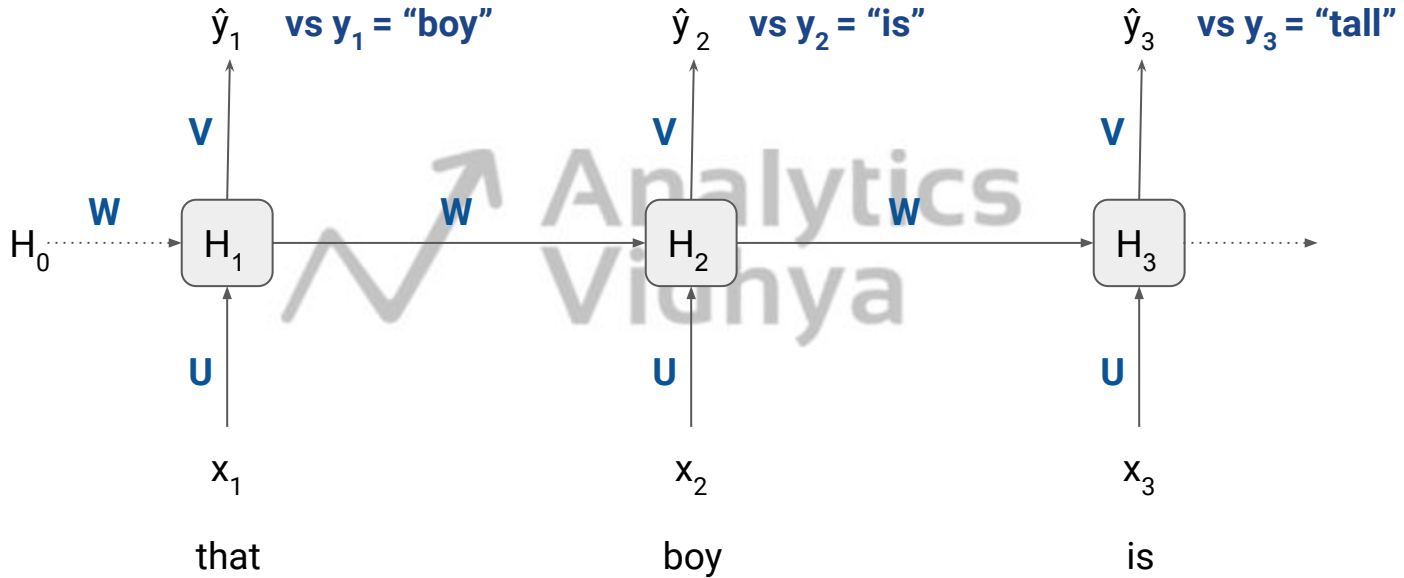[8, 4, 5],
[7, 3, 0],
[10 4 9],
[4 9 6],
[9 6 2]]

LSTM based Language Model

Training Phase

LSTM based Language Model

Training Phase

Data Preparation

- Converting tokens to integers:

"<pad>"	- 0
"alright"	- 1
"difference"	- 2
"great"	- 3
"is"	- 4
"perfect"	- 5
"price"	- 6
"sounds"	- 7
"that"	- 8
"the"	- 9
"what"	- 10

[[1, 8, 4],
[8, 4, 5],
[7, 3, 0],
[10 4 9],
[4 9 6],
[9 6 2]]

Data Preparation

- Converting tokens to integers:
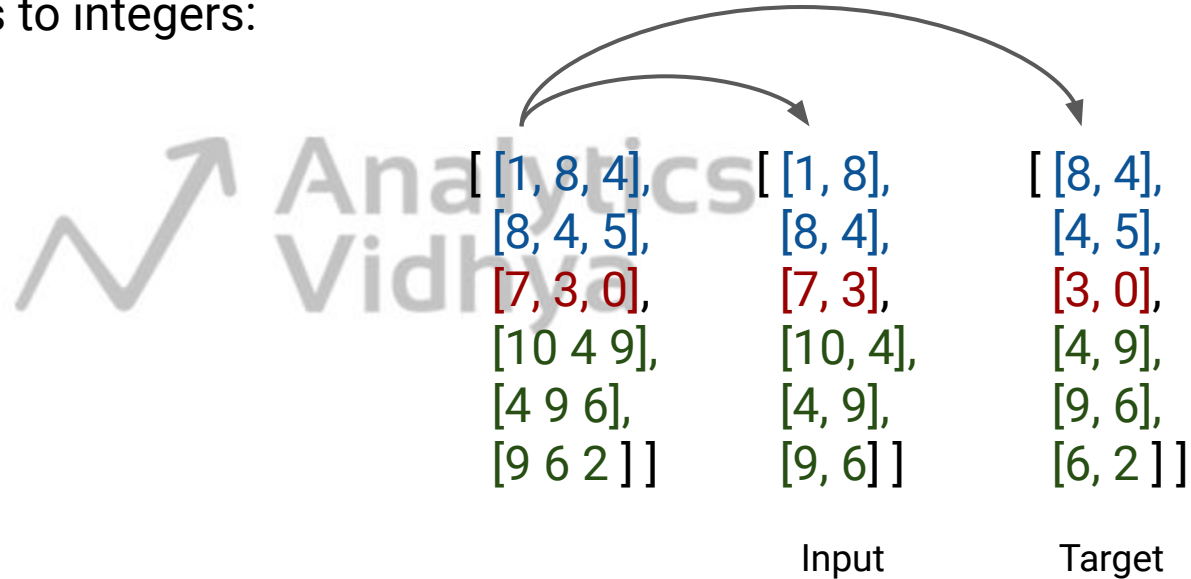
"<pad>"   - 0
"alright"   - 1
"difference"   - 2
"great"   - 3
"is"   - 4
"perfect"   - 5
"price"   - 6
"sounds"   - 7
"that"   - 8
"the"   - 9
"what"   - 10

[[1, 8, 4],   [[1, 8],
[8, 4, 5],   [8, 4],
[7, 3, 0],   [7, 3],
[10 4 9],   [10, 4],
[4 9 6],   [4, 9],
[9 6 2]]   [9, 6]]

Input

Data Preparation

- Converting tokens to integers:

Token	Integer
"<pad>"	- 0
"alright"	- 1
"difference"	- 2
"great"	- 3
"is"	- 4
"perfect"	- 5
"price"	- 6
"sounds"	- 7
"that"	- 8
"the"	- 9
"what"	- 10

[[1, 8, 4],   [[1, 8],   [[8, 4],
[8, 4, 5],   [8, 4],   [4, 5],
[7, 3, 0],   [7, 3],   [3, 0],
[10 4 9],   [10, 4],   [4, 9],
[4 9 6],   [4, 9],   [9, 6],
[9 6 2]]   [9, 6]]   [6, 2]]

              Input   Target

Text Generation

- **User input**:

  - A seed text
  - number of tokens to generate

- **Output**: seed text + model generated tokens

  - Input seed text = "thank you", n = 3
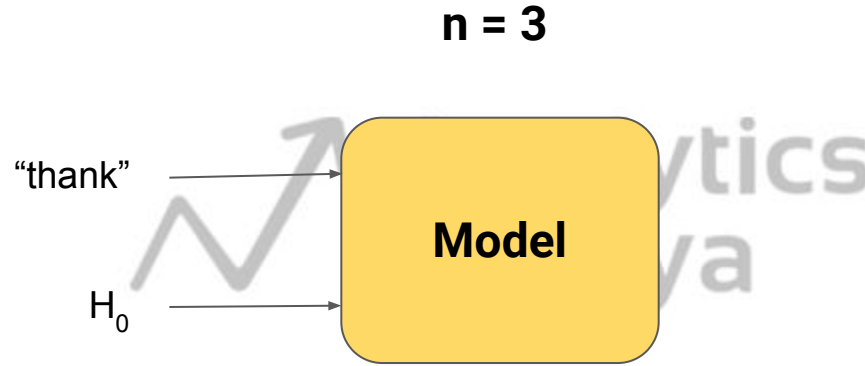 - Output = "thank you very much sir"

Text Generation

n = 3
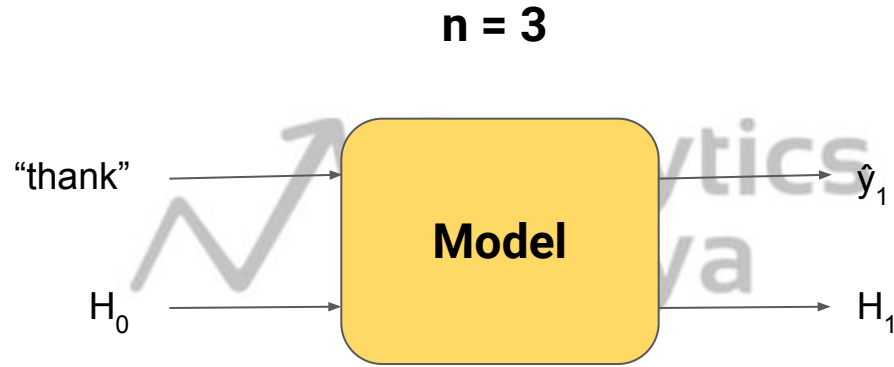

"thank" $\rightarrow$ **Model** $\rightarrow$ $\hat{y}_1$

H_0 $\rightarrow$ H_1

"thank you"

Text Generation

n = 2

"you" → Model → "very"

H_1 → Model → H_2

"thank you"

Text Generation

n = 0
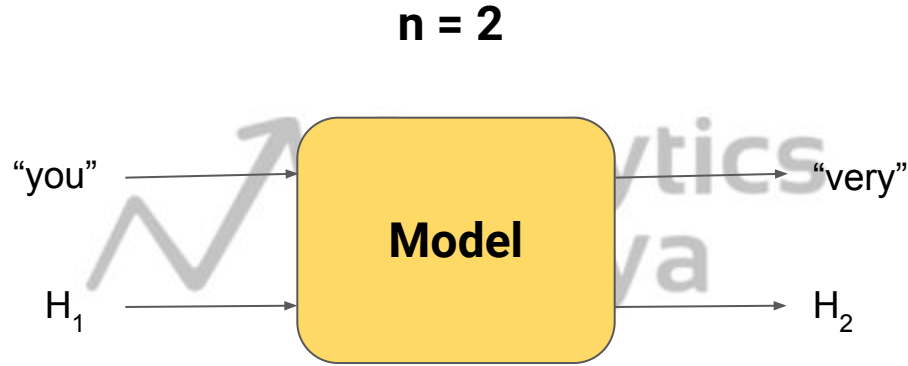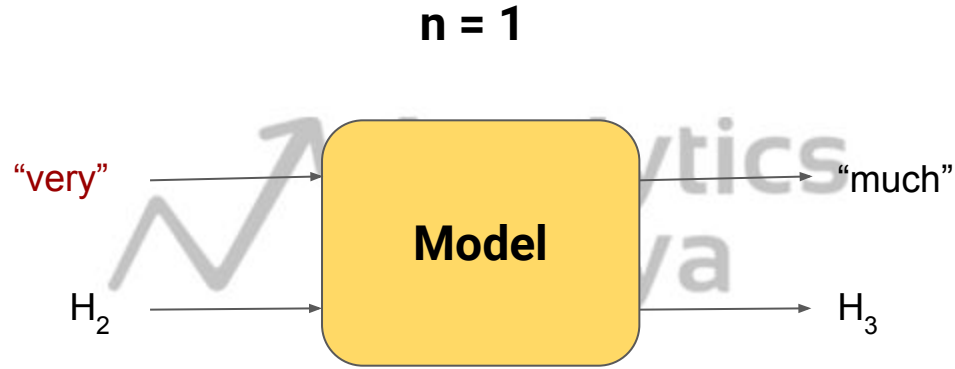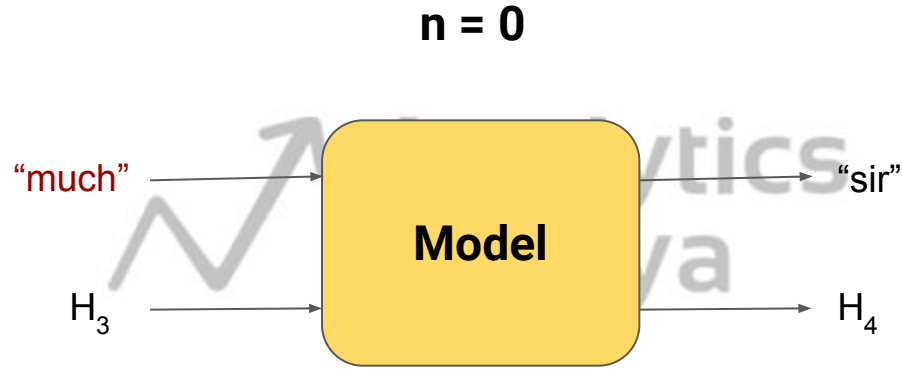
"thank you very much"

Text Generation

n = 3

"thank" → **Model** → $\hat{y}_1$

H_0 → **Model** → H_1

n = 2

"you" → **Model** → "very"

H_1 → **Model** → H_2

n = 1

"very" → **Model** → "much"

H_2 → **Model** → H_3

n = 0

"much" → **Model** → "sir"

H_3 → **Model** → H_4

Thank You