Natural Language Processing

Week 1

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Question 1:

In a corpus, you found that the word with rank 4th has a frequency of 600. What can be the best guess for the rank of a word with frequency 300?

1. 2
2. 4
3. 8
4. 6

Solution

Zipf's Law

A relationship between the frequency of a word (f) and its position in the list (its rank r).

$$f \propto \frac{1}{r}$$

or, there is a constant k such that

$$f.r = k$$

Answer: 3

Solution:

frequency * rank =k [by Zipfs law]

600*4 = 300*r

r = 8

Question 2:

In the sentence, “In Kolkata I took my hat off. But I can’t put it back on.”, total number of word tokens and word types are:

1. 14, 13
2. 13, 14
3. 15, 14
4. 14, 15

Solution

Question 2:

In the sentence, “In Kolkata I took my hat off. But I can’t put it back on.”, total number of word tokens and word types are:

1. 14, 13
2. 13, 14
3. 15, 14
4. 14, 15

Type: unique instances of a word

Tokens: all instances of a word (not unique)

Answer: 1. 14, 13.

Solution: Here, the word “I” is repeated two times so type count is one less than token count.

Question 3:

Let the rank of two words, w1 and w2, in a corpus be 1600 and 400, respectively. Let m1 and m2 represent the number of meanings of w1 and w2 respectively. The ratio m1 : m2 would tentatively be

1. 1:4
2. 4:1
3. 1:2
4. 2:1

Solution

Correlation: Number of meanings and word frequency

The number of meanings m of a word obeys the law:

$$m \propto \sqrt{f}$$

Given the First law

$$m \propto \frac{1}{\sqrt{r}}$$

Answer: 3

Solution:

m1/m2 = sqrt(rank2)/sqrt(rank1) = sqrt(400)/sqrt(1600) = 1:2

Question 4:

What is the valid range of type-token ratio of any text corpus?

1. $TTR \in (0,1]$ (excluding zero)
2. $TTR \in [0,1]$
3. $TTR \in [-1,1]$
4. $TTR \in [0,+\infty]$ (any non-negative number)

Solution

Question 4:

What is the valid range of type-token ratio of any text corpus?

1. $TTR \in (0,1]$ (excluding zero)
2. $TTR \in [0,1]$
3. $TTR \in [-1,1]$
4. $TTR \in [0,+\infty]$ (any non-negative number)

Answer: 1.

Solution: Number of unique words or type ≤ Total number of tokens in text, and both are greater than 1

Question 5:

If first corpus has $TTR_1 = 0.025$ and second corpus has $TTR_2 = 0.25$, where TTR_1 and TTR_2 represents type/token ratio in first and second corpus respectively, then

1. First corpus has more tendency to use different words.
2. Second corpus has more tendency to use different words.
3. Both a and b
4. None of these

Solution

Question 5:

If first corpus has $TTR_1 = 0.025$ and second corpus has $TTR_2 = 0.25$, where TTR_1 and TTR_2 represents type/token ratio in first and second corpus respectively, then

1. First corpus has more tendency to use different words.
2. Second corpus has more tendency to use different words.
3. Both a and b
4. None of these

Answer: b
Solution: Second corpus has more tendency to use different words. If TTR scores are higher then there is more tendency to use different words.

Question 6:

Which of the following is/are true for the English Language?

1. Lemmatization works only on inflectional morphemes and Stemming works only on derivational morphemes.
2. The outputs of lemmatization and stemming for the same word might differ.
3. Output of lemmatization are always real words
4. Output of stemming are always real words

Solution

Answer: 2, 3

Solution: *Stemming* usually refers to a crude heuristic process that chops off the ends of words in the hope of achieving this goal correctly most of the time, and often includes the removal of derivational affixes. *Lemmatization* usually refers to doing things properly with the use of a vocabulary and morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the *lemma* .

Question 7:

An advantage of Porter stemmer over a full morphological parser?

1. The stemmer is better justified from a theoretical point of view
2. The output of a stemmer is always a valid word
3. The stemmer does not require a detailed lexicon to implement
4. None of the above

Solution

Answer: 3

Solution: The Porter stemming algorithm is a process for removing suffixes from words in English. The Porter stemming algorithm was made on the assumption that we don't have a stem dictionary (lexicon) and that the purpose of the task is to improve Information Retrieval performance. Stemming algorithms are typically rule-based. You can view them as a heuristic process that sort-of lops off the ends of words.

Question 8:
Which of the following are instances of stemming? (as per Porter Stemmer)

1. are -> be
2. plays -> play
3. saw -> s
4. university -> univers

Solution

Question 8:
Which of the following are instances of stemming? (as per Porter Stemmer)

1. are -> be
2. plays -> play
3. saw -> s
4. university -> univers

Answer: 2,4
Solution: Stemming cannot convert are->be as it can only convert or chop off word suffixes. Also Porter Stemmer wouldn't chop off if the final outcome is of length 1 as in saw -> s.

Question 9:

What is natural language processing good for?

1. Summarize blocks of text
2. Automatically generate keywords
3. Identifying the type of entity extracted
4. All of the above

Solution

Question 9:

What is natural language processing good for?

1. Summarize blocks of text
2. Automatically generate keywords
3. Identifying the type of entity extracted
4. All of the above

Answer: 4

Solution:

For all the above-mentioned task, NLP can be used

Question 10:

What is the size of unique words in a document where total number of words = 12000. K = 3.71 Beta = 0.69?

1. 2421
2. 3367
3. 5123
4. 1529

Solution

Heaps' Law

Let $|V|$ be the size of vocabulary and N be the number of tokens.

$$|V| = KN^{\beta}$$

Answer: 1

Solution: 3.71 x 12000^0.69 = **2421** unique words. Heap's Law

Natural Language Processing

Week 2

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

QUESTION 1:

According to Zipf's law which statement(s) is/are correct?

(i) A small number of words occur with high frequency.

(ii) A large number of words occur with low frequency.

a. Both (i) and (ii) are correct

b. Only (ii) is correct

c. Only (i) is correct

d. Neither (i) nor (ii) is correct

Solution

QUESTION 1:

According to Zipf's law which statement(s) is/are correct?

(i) A small number of words occur with high frequency.

(ii) A large number of words occur with low frequency.

a. Both (i) and (ii) are correct

b. Only (ii) is correct

c. Only (i) is correct

d. Neither (i) nor (ii) is correct

Correct Answer: a

QUESTION 2:

Consider the following corpus C_1 of 4 sentences. What is the total count of unique bi-grams for which the likelihood will be estimated? Assume we do not perform any pre-processing.

today is Nayan's birthday
she loves ice cream
she is also fond of cream cake
we will celebrate her birthday with ice cream cake

a. 24
b. 28
c. 27
d. 23

Solution

Correct Answer: a

Detailed Solution:

Unique bi-grams are:

<s> today	today is	is Nayan's	Nayan's birthday	birthday <\s>
<s> she	she loves	loves ice	ice cream	cream <\s>
She is	is also	also fond	fond of	of cream
cake <\s>	<s> we	we will	will celebrate	celebrate her
her birthday	birthday with	with ice	cream cake	

QUESTION 3:

A 4-gram model is a _____ order Markov Model.

a. Constant
b. Five
c. Four
d. Three

Solution

More Formally: kth order Markov Model

Chain Rule:

$$P(w_1 w_2 \dots w_n) = \prod_i P(w_i | w_1 w_2 \dots w_{i-1})$$

Using Markov Assumption: only k previous words

$$P(w_1 w_2 \dots w_n) \approx \prod_i P(w_i | w_{i-k} \dots w_{i-1})$$

P(office | about fifteen minutes from)

An N-gram model uses only $N-1$ words of prior context.

- Unigram: P(office)
- Bigram: P(office | from)
- Trigram: P(office | minutes from)

Markov model and Language Model

An N-gram model is an $N-1$-order Markov Model

Correct Answer: d

QUESTION 4:

Which one of these is a valid Markov assumption?

a. The probability of a word depends only on the current word.

b. The probability of a word depends only on the previous word.

c. The probability of a word depends only on the next word.

d. The probability of a word depends only on the current and the previous word.

Solution

QUESTION 4:

Which one of these is a valid Markov assumption?

a. The probability of a word depends only on the current word.

b. The probability of a word depends only on the previous word.

c. The probability of a word depends only on the next word.

d. The probability of a word depends only on the current and the previous word.

Correct Answer: b

QUESTION 5:

For the string **‘mash’**, identify which of the following set of strings have a Levenshtein distance of 1.

a. smash, mas, lash, mushy, hash
b. bash, stash, lush, flash, dash
c. smash, mas, lash, mush, ash
d. None of the above

Solution

QUESTION 5:

For the string **‘mash’**, identify which of the following set of strings have a Levenshtein distance of 1.

a. smash, mas, lash, mushy, hash
b. bash, stash, lush, flash, dash
c. smash, mas, lash, mush, ash
d. None of the above

Correct Answer: c

QUESTION 6:

Assume that we modify the costs incurred for operations in calculating Levenshtein distance, such that both the insertion and deletion operations incur a cost of 1 each, while substitution incurs a cost of 2. Now, for the string **'lash'** which of the following set of strings will have an edit distance of 1?

a. ash, slash, clash, flush
b. flash, stash, lush, blush,
c. slash, last, bash, ash
d. None of the above

Solution

QUESTION 6:

Assume that we modify the costs incurred for operations in calculating Levenshtein distance, such that both the insertion and deletion operations incur a cost of 1 each, while substitution incurs a cost of 2. Now, for the string **'lash'** which of the following set of strings will have an edit distance of 1?

a. ash, slash, clash, flush
b. flash, stash, lush, blush,
c. slash, last, bash, ash
d. None of the above

Correct Answer: d

QUESTION 7:

Given a corpus C_2, the Maximum Likelihood Estimation (MLE) for the bigram "dried berries" is 0.3 and the count of occurrence of the word "dried" is 580. for the same corpus C_2, the likelihood of "dried berries" after applying add-one smoothing is 0.04. What is the vocabulary size of C_2?

a. 3585
b. 3795
c. 4955
d. 3995

Solution

Correct Answer: b

Detailed Solution:

$$P_{MLE}(berries|\ dried) = \frac{C(dried, berries)}{C(dried)}$$

$$0.3 = \frac{C(dried, berries)}{580}$$

$$C(dried, berries) = 174$$

$$P_{Add-1}(berries|\ dried) = \frac{C(dried, berries)+1}{C(dried)+ V}$$

$$0.04 = \frac{174+1}{580+V}$$

$$V = 3795$$

For Question 8 to 10, consider the following corpus C_3 of 3 sentences.

there is a big garden
children play in a garden
they play inside beautiful garden

QUESTION 8:

Calculate **P(they play in a big garden)** assuming a bi-gram language model.

a. 1/8
b. 1/12
c. 1/24
d. None of the above

Solution

Correct Answer: b

Detailed Solution:

P(they | <s>) = 1/3
P(play | they) = 1/1
P(in | play) = 1/2
P(a | in) = 1/1
P(big | a) = 1/2
P(garden | big) = 1/1
P(<\s>|garden) = 3/3
P(they play in a big garden) = 1/3 x 1/1 x 1/2 x 1/1 x 1/2 x 1/1 x 3/3 = 1/12

QUESTION 9:

Considering the same model as in Question 7, calculate the perplexity of **<s> they play in a big garden <\s>.**

a. 2.289
b. 1.426
c. 1.574
d. 2.178

Solution

QUESTION 9:

Considering the same model as in Question 7, calculate the perplexity of **<s> they play in a big garden <\s>.**

a. 2.289
b. 1.426
c. 1.574
d. 2.178

Correct Answer: b

Detailed Solution:

$$perplexity = \sqrt[7]{12} = 1.426$$

QUESTION 10:

Assume that you are using a bi-gram language model with add one smoothing. Calculate **P(they play in a beautiful garden).**

a. 4.472 x 10^-6
b. 2.236 x 10^-6
c. 3.135 x 10^-6
d. None of the above

Solution

Correct Answer: b

Detailed Solution:

|V|=11
P(they | <s>) = (1+1)/(3+11)
P(play | they) = (1+1)/(1+11)
P(in | play) = (1+1)/(2+11)
P(a | in) = (1+1)/(1+11)
P(beautiful | a) = (0+1)/(2+11)
P(garden | beautiful) = (1+1)/(1+11)
P(<\s>|garden) = (3+1)/(3+11)
P(they play in a beautiful garden) = 2/14 x 2/12 x 2/13 x 2/12 x 1/13 x 2/12 x 4/14
= 2.236 x 10^-6

Natural Language Processing

Week 3

Question 1: Which of the following words contains both derivational as well inflectional suffixes:

1. regularity
2. carefully
3. older
4. availabilities

Solution

Question 1: Which of the following words contains both derivational as well inflectional suffixes:

1. regularity
2. carefully
3. older
4. availabilities

Answer: 4
Solution: availabilities = avail(Root word) + able (derivational suffix) + ity (derivational suffix) + es (inflectional suffix).

Question 2: Let's assume the probability of rolling 1 two times in a row of a dice is p. Consider a sentence consisting of N random digits. A model assigns probability to each of the digit with the probability p. Find the perplexity of the sentence.

1. 10
2. 6
3. 36
4. 3

Solution

Question 2: Let's assume the probability of rolling 1 two times in a row of a dice is p. Consider a sentence consisting of N random digits. A model assigns probability to each of the digit with the probability p. Find the perplexity of the sentence.

1. 10
2. 6
3. 36
4. 3

Answer - 3

Solution: Probability of rolling 1 two times in a row is(⅙)*(⅙) = (1/36) . Then perplexity is ((1/36)^N)^-1/N = 36

Question 3: Assume that “x” represents the input and “y” represents the tag/label. Which of the following mappings are correct?

1. Generative Models - learn Joint Probability p(x, y)
2. Discriminative Models - learn Joint Probability p(x, y)
3. Generative Models - learn Posterior Probability p(y | x) directly
4. Discriminative Models - learn Posterior Probability p(y | x) directly

Solution

Question 3: Assume that “x” represents the input and “y” represents the tag/label. Which of the following mappings are correct?

1. Generative Models - learn Joint Probability p(x, y)
2. Discriminative Models - learn Joint Probability p(x, y)
3. Generative Models - learn Posterior Probability p(y | x) directly
4. Discriminative Models - learn Posterior Probability p(y | x) directly

Answer: 1, 4

Solution: Generative classifiers learn a model of the joint probability p(x, y) and make their predictions by using Bayes rules to calculate p(y | x). Discriminative classifiers model the posterior p(y | x) directly, or learn a direct map from inputs x to the class labels y.

Question 4: Which one of the following is an example of the Generative model?

1. Conditional Random Fields
2. Naive Bayes
3. Support Vector Machine
4. Logistic Regression

Solution

Question 4: Which one of the following is an example of the Generative model?

1. Conditional Random Fields
2. Naive Bayes
3. Support Vector Machine
4. Logistic Regression

Answer- 2

Solution: Others model in the option are discriminative model

Question 5: Which of the following morphological process is true for motor+hotel → motel?

1. Suppletion
2. Compounding
3. Blending
4. Clipping

Solution

Question 5: Which of the following morphological process is true for motor+hotel →motel?

1. Suppletion
2. Compounding
3. Blending
4. Clipping

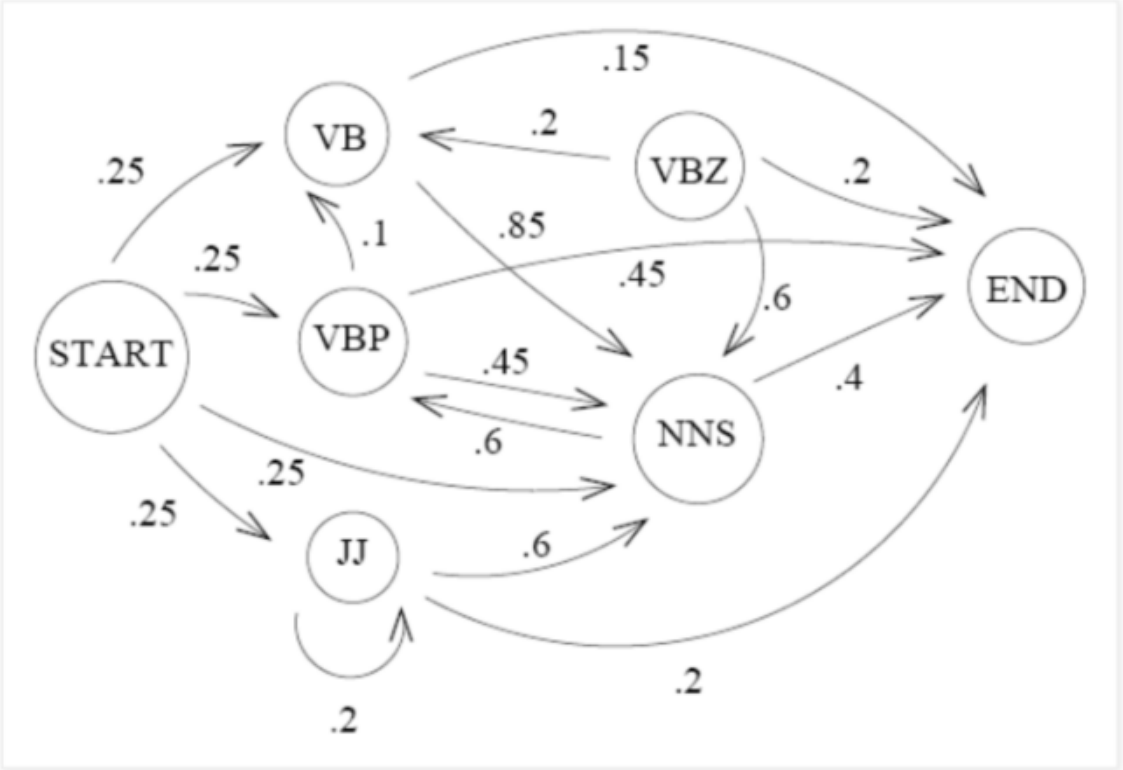
Answer: 3

Solution: Blending combines parts of two different words.

Question 6: Consider the HMM given below to solve the sequence labeling problem of POS tagging. With that HMM, calculate the probability that the sequence of words “free workers” will be assigned the following parts of speech;

VB NNS

	free	*workers*
JJ	**0.00158**	**0**
NNS	**0**	**0.000475**
VB	**0.00123**	**0**
VBP	**0.00081**	**0**
VBZ	**0**	**0.00005**



Solution

P(free workers, VB NNS)

= P(VB|*start*) * P(*free*|VB) * P(NNS|VB) * P(*workers*|NNS)

* P(end|NNS)

= 0.25 * 0.00123 * 0.85 * 0.000475 * 0.4

= $4.96 * 10^{-8}$

Question 7: Which of the following is/are true?

1. Only a few non-deterministic automation can be transformed into a deterministic one

2. Recognizing problem can be solved in linear time

3. Deterministic FSA might contain empty (ε) transition

4. There exist an algorithm to transform each automation into a unique equivalent automation with the least no of states

Solution

Question 7: Which of the following is/are true?

1. Only a few non-deterministic automation can be transformed into a deterministic one

2. Recognizing problem can be solved in linear time

3. Deterministic FSA might contain empty (ε) transition

4. There exist an algorithm to transform each automation into a unique equivalent automation with the least no of states

Answer: 2, 4

Solution:

Every non-deterministic automation can be transformed into a deterministic one.
Deterministic FSA should not contain empty transition.

Question 8: Which of the following are an example of clipping?

1. brunch
2. gator
3. laser
4. ad

Solution

Question 8: Which of the following are an example of clipping?

1. brunch
2. gator
3. laser
4. ad

Answer: 2, 4

Solution: gator (alligator), ad (advertisement)

Natural Language Processing

Week 4

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Question 1: Baum-Welch algorithm is an example of -

a. Forward-backward algorithm

b. Special case of the Expectation-maximization algorithm

c. Both A and B

c. None

Solution

Question 1: Baum-Welch algorithm is an example of -

a. Forward-backward algorithm

b. Special case of the Expectation-maximization algorithm

c. Both A and B

c. None

Answer: C

Question 2: Once a day (e.g. at noon), the weather is observed as one of state 1: rainy, state 2:cloudy, state 3: sunny. The state transition probabilities are :

0.4	0.3	0.3
0.2	0.6	0.2
0.1	0.1	0.8

Given that the weather on day 1 (t = 1) is sunny (state 3), what is the probability that the weather for the next 7 days will be “sun-sun-rain-rain-sun-cloudy-sun”?

a. 1.54 * 10-4

b. 8.9 * 10-2

c. 7.1 * 10-7

d. 2.5 * 10-10

Solution

State 1(S1) = **rainy,** State 2(S2) = **cloudy**, State 3(S3): **sunny**

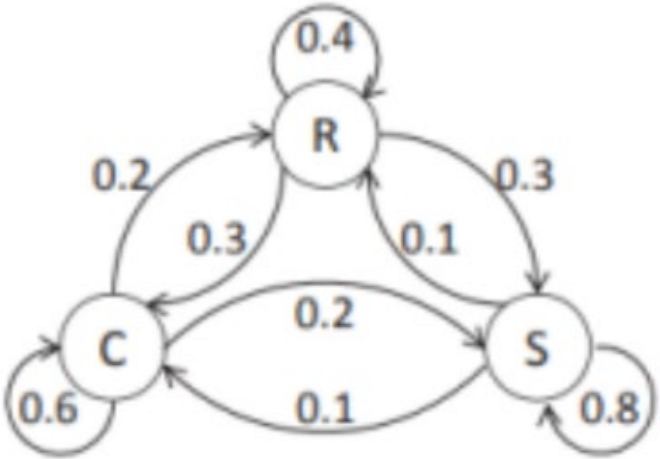
Given: Weather on day 1 (t = 1) is sunny (state 3).

Therefore, P(S3|start) = 1, P(S1|start) = 0, P(S2|start) = 0

O = {sun, sun, sun, rainy, rainy, sun, cloudy, sun}

O = {S3, S3, S3, S1, S1, S3, S2, S3}

	S1(R)	S2(C)	S3(S)
S1(R)	0.4	0.3	0.3
S2(C)	0.2	0.6	0.2
S3(S)	0.1	0.1	0.8



Solution

	S1(R)	S2(C)	S3(S)
S1(R)	0.4	0.3	0.3
S2(C)	0.2	0.6	0.2
S3(S)	0.1	0.1	0.8

O = {S3, S3, S3, S1, S1, S3, S2, S3}

P(O | Model)

= P(S3, S3, S3, S1, S1, S3, S2, S3 | Model)

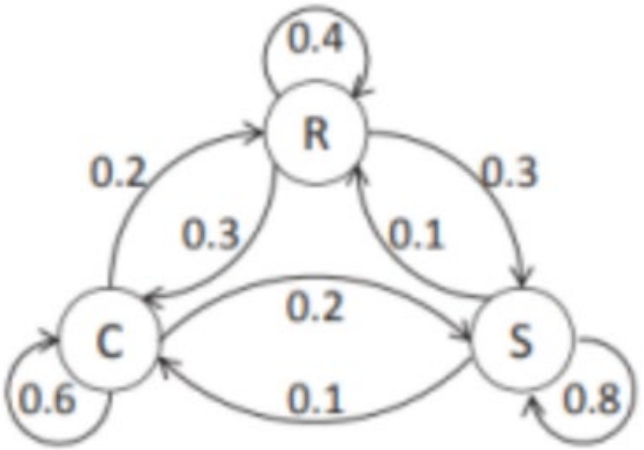
= P(S3|start) P(S3|S3) P(S3|S3) P(S1|S3) P(S1|S1) P(S3|S1) P(S2| S3) P(S3|S2)

= P(S3|start) · a33 · a33 · a31 · a11 · a13 · a32 · a23

= (1)(0.8)(0.8)(0.1)(0.4)(0.3)(0.1)(0.2)

= 1.536 × 10-4

Answer: A



Question 3: We are given the sequence “ki fin yeni!”. Possible POS tags are {T1,T2, T3, T4}. Assume all POS tags are equally likely to be at the starting of a sequence.

	ki	fin	yeni
T1	0.1	0.1	0.8
T2	0.8	0.1	0.1
T3	0.2	0.2	0.6
T4	0.8	0.1	0.1

Table 1: Output Symbol probabilities

	T1	T2	T3	T4
T1	0.18	0.01	0.8	0.01
T2	0.9	0.0	0.05	0.05
T3	0.4	0.5	0.05	0.05
T4	0.4	0.5	0.05	0.05

Table 2: Hidden State transition matrix

Calculate P(x1 = “ki”, x2 = “fin”, y1 = “T1”, y2 = “T2”).

a. 0.000025

b. 0.0001

c. 0.0025

d. None of the above

Solution

Solution: Assume all POS tags are equally likely to be at the starting of a sequence. Possible POS tags are {T1,T2, T3, T4}

Initial probabilities: P(T1|start) = 1/4, P(T2|start) = 1/4, P(T3|start) = 1/4, P(T4|start) = ¼

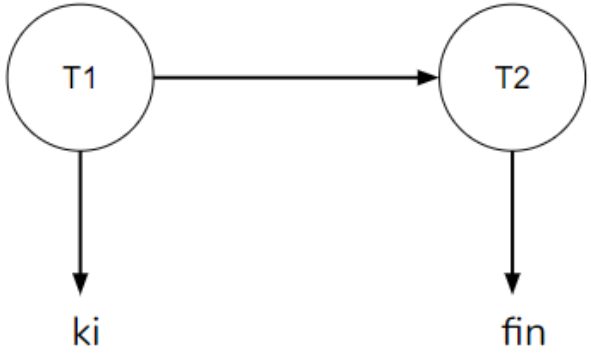
P(x1 = “ki”, x2 = “fin”, y1 = T1, y2 = T2)

= P(T1|start) P(“ki”|T1) P(T2|T1) P(“fin”|T2)

= 0.25 * 0.1 * 0.01 * 0.1

= 0.000025

Answer: A



	ki	fin	yeni
T1	0.1	0.1	0.8
T2	0.8	0.1	0.1
T3	0.2	0.2	0.6
T4	0.8	0.1	0.1

	T1	T2	T3	T4
T1	0.18	0.01	0.8	0.01
T2	0.9	0.0	0.05	0.05
T3	0.4	0.5	0.05	0.05
T4	0.4	0.5	0.05	0.05

Question 4: Let us define an HMM Model with K classes for hidden states and T data points as observations. The dataset is defined as $X = \{x1, x2, \ldots, xT\}$ and the corresponding hidden states are $Z = \{z1, z2, \ldots, zT\}$. Please note that each xi is an observed variable and each zi can belong to one of classes for hidden state. What will be the size of the state transition matrix, and the emission matrix, respectively for this example.

a. $K \times K$, $K \times T$

b. $K \times T$, $K \times T$

c. $K \times K$, $K \times K$

d. $K \times T$, $K \times K$

Solution

Answer: A

Solution: Since there are K hidden states, the state transition matrix will be of size K × K. The emission matrix will be of size K × T, as it defines the probability of emitting an observed state from a hidden state.

Question 5: You are building a model distribution for an infinite stream of word tokens. You know that the source of this stream has a vocabulary of size 1000. Out of these 1000 words you know of 100 words to be stop words each of which has a probability of 0.0019. With only this knowledge what is the maximum possible entropy of the modelled distribution. (Use log base 10 for entropy calculation)

a. 5.079

b. 0

c. 2.984

d. 12.871

Solution

Answer: C

Intuitive Principle

Model all that is known and assume nothing about that which is unknown. *Given a collection of facts, choose a model which is consistent with all the facts, but otherwise as uniform as possible.*

Solution: There are 100 stopwords with each having an occurrence probability of 0.0019. Hence, P(Stopwords) = $100 * 0.0019 = 0.19$

$P(non - stopwords) = 1 - 0.19 = 0.81$

For maximum entropy, the remaining probability should be uniformly distributed.

For every non-stopword w, $P(w) = 0.81/(1000 - 100) = 0.81/900 = 0.0009$.

Finally, the value of the entropy would be,

$H = E(\log(1/p)) = -100(0.0019 * \log(0.0019)) - 900(0.0009 \log(0.0009)) = -2.9841$

Question 6: For an HMM model with N hidden states, V observable states, what are the dimensions of parameter matrices A,B and π? A: Transition matrix, B: Emission matrix, π: Initial Probability matrix.

a. $N \times V$, $N \times V$, $N \times N$

b. $N \times N$, $N \times V$, $N \times 1$

c. $N \times N$, $V \times V$, $N \times 1$

d. $N \times V$, $V \times V$, $V \times 1$

Solution

Answer: B

Solution:

We are given a HMM model with N hidden states, V observable states

Matrix A contains all the transition probabilities and have dimension N × N.

Matrix B contains all the emission probabilities and dimension N × V .

π contains initial probability for all hidden states and have dimension N × 1.

Question 7: Suppose you have the input sentence “Death Note is a great anime”. And you know the possible tags each of the words in the sentence can take.

- Death: NN, NNS, NNP, NNPS
- Note: VB, VBD, VBZ
- is: VB
- a: DT
- great: ADJ
- anime: NN, NNS, NNP

How many hidden state sequences are possible for the above sentence and States?

a. $4 \times 3 \times 3$
b. $4^{3\text{^}3}$
c. $2^4 \times 2^3 \times 2^3$
d. $2^{4\times3\times3}$

Solution

Answer: A

Solution: Each possible hidden sequence can take only one POS tag for each of the words. Hence the total possibility will be a product of the number of candidates for each word.

Number of possible hidden state sequences = 4 * 3 * 1 * 1 * 1 * 3

Question 8: In Hidden Markov Models or HMMs, the joint likelihood of an observed sequence O with a hidden state sequence Q, is written as P(O, Q; θ). In many applications, like POS tagging, one is interested in finding the hidden state sequence Q, for a given observation sequence, that maximizes P(O, Q; θ). What is the time required to compute the most likely Q using an exhaustive search? The required notations are, N: possible number of hidden states, T: length of the observed sequence.

a. Of the order of TN^T
b. Of the order of N^2T
c. Of the order of T^N
d. Of the order of N^2

Solution

Answer: A

Solution: We will need to compute $P(O, Q|\theta)$ for all possible Q. There are a total of N^T possible hidden sequences Q for a sequence of length T. Each individual probability calculation also requires T multiplications.

Natural Language Processing

Week 5

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Question 1: Which of the following are true?

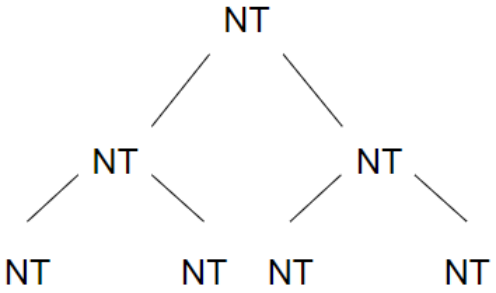
A) Given a CFG and its corresponding CNF, they both produce the same language.

B) For a given grammar, there can be more than one CNF.

C) It requires ‘2n+1’ productions or steps in CNF to generate a string w of length ‘n’.

D) None of the above

Solution

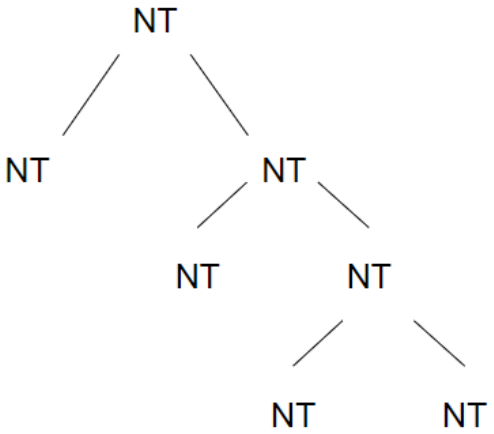


Answer: A, B

Solution: In CNF,

- Either, exactly two non-terminals on the RHS
- Or, 1 terminal symbol on the RHS

Let n be the length of a string. In CNF, each NT can give exactly two NTs.



We start with the (non-terminal) symbol S. Using (n−1) rules of form NT→ NT NT (where NT represents a non−terminal), we can construct a string containing ‘n’ non-terminal symbols. Then on each NT symbol of said string of length ‘n’ we apply a rule of form NT→T. i.e. we apply n rules. In total we will have applied (n−1) + n = 2n−1 rules.

Original Grammar

S → NP VP
S → Aux NP VP
S → VP
NP → Pronoun
NP → Proper-Noun
NP → Det Nominal
Nominal → Noun
Nominal → Nominal Noun
Nominal → Nominal PP
VP → Verb
VP → Verb NP
VP → VP PP
PP → Prep NP
Pronoun → I | he | she | me
Noun → book | flight | meal | money
Verb → book | include | prefer
Proper-Noun → Houston | NWA

Chomsky Normal Form

S → NP VP
S → X1 VP
X1 → Aux NP
S → book | include | prefer
S → Verb NP
S → VP PP
NP → I | he | she | me
NP → Houston | NWA
NP → Det Nominal
Nominal → book | flight | meal | money
Nominal → Nominal Noun
Nominal → Nominal PP
VP → book | include | prefer
VP → Verb NP
VP → VP PP
PP → Prep NP
Pronoun → I | he | she | me
Noun → book | flight | meal | money
Verb → book | include | prefer
Proper-Noun → Houston | NWA

NPTEL

Question 2: Consider the CFG given below:

$S \rightarrow aSb|D$

$D \rightarrow Dc|\varepsilon$

How many non-terminals should be added to convert the CFG into CNF?

A) 3

B) 2

C) 4

D) 5

Solution

Answer: D

Solution: Given: S → aSb|D, D → Dc|ε

1. Since the start symbol S occurs on the right side of a production rule, we create a new start symbol S' and a new production rule S' → S.

   S' → S, S → aSb|D, D → Dc|ε

1. Remove the null productions: D → ε, S → ε
   a. After removing D → ε: S' → S, S → aSb|D|ε, D → Dc|c
   b. After removing S → ε: S' → S, S → aSb|D|ab, D → Dc|c
2. Remove the unit productions: S → D, S' → S
   a. After removing S → D: S' → S, S → aSb|ab|Dc|c, D → Dc|c
   b. After removing S' → S: S' → aSb|ab|Dc|c, S → aSb|ab|Dc|c, D → Dc|c

Solution

S’ → aSb|ab|Dc|c, S → aSb|ab|Dc|c, D → Dc|c

4. Since the right-side of the production rules contain both terminal and non-terminal, we add the following rules: A → a, B → b, C → c

S’ → ASB|AB|DC|c, S → ASB|AB|DC|c, D → DC|c, A → a, B → b, C → c

5. Handling the cases where there are more than two non-terminals on the right-hand side by adding the rule E → SB:

S′ → AE|AB|DC|c, S → AE|AB|DC|c, **E** → SB, D → DC|c, **A** → a, **B** → b, **C** → c

$S \rightarrow$ NN VP	0.50	$S \rightarrow$ VP NN	0.50
$NP \rightarrow$ NN PB	0.40	$P\,B \rightarrow$ PP NN	0.30
$V\,P \rightarrow$ VB NN	0.30	$V\,P \rightarrow$ VB NP	0.20
$V\,P \rightarrow$ NN VB	0.25	$V\,P \rightarrow$ NN PB	0.15
$P\,P \rightarrow$ with	0.10	$P\,P \rightarrow$ without	0.10
$V\,B \rightarrow$ play	0.30	$V\,B \rightarrow$ enjoy	0.20
$V\,B \rightarrow$ watch	0.25	$NN \rightarrow$ children	0.15
$NN \rightarrow$ cricket	0.15	$NN \rightarrow$ friends	0.20
$NN \rightarrow$ football	0.10	$NN \rightarrow$ music	0.12

Question 3:

Using CKY algorithm, find the probability score for the most probable tree for the sentence S_1 = "**children play cricket with friends**". **[1 mark]**

A) 5.06×10^{-4}
B) 2.73×10^{-3}
C) 1.62×10^{-6}
D) None of the above

Solution

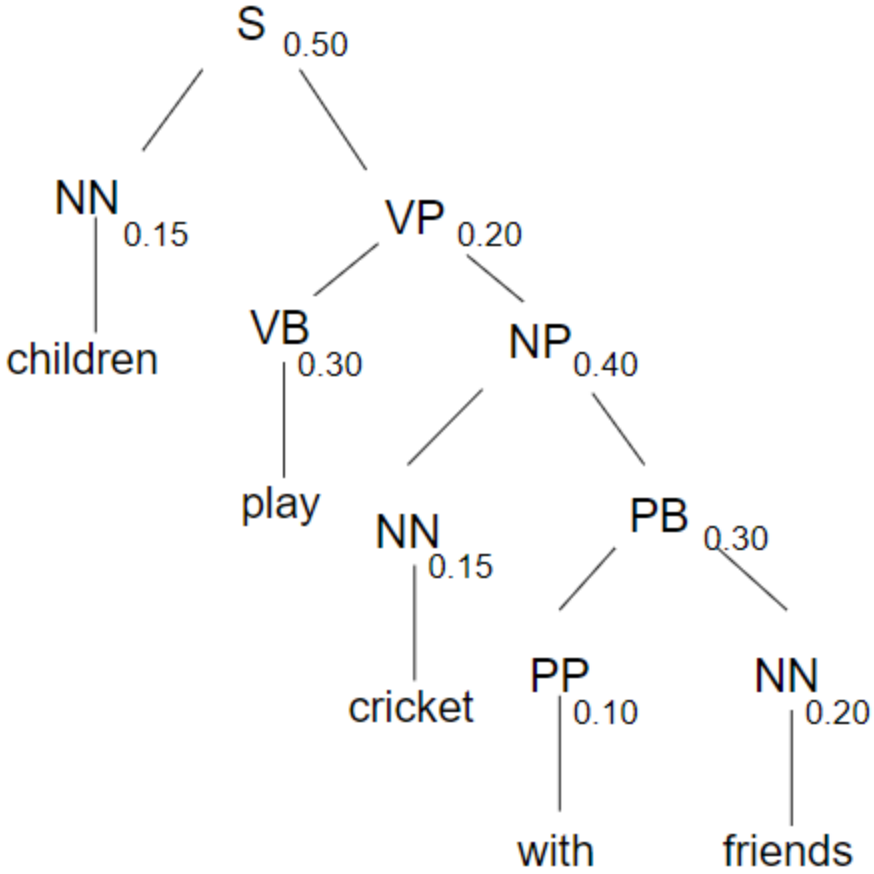
- Let n be the number of words in the input. Think about $n+1$ lines separating them, numbered 0 to n.
- x_{ij} will denote the words between line i and j
- We build a table so that x_{ij} contains all the possible non-terminal spanning for words between line i and j.
- We build the Table bottom-up.

Solution

Answer: C

Solution:

0	1	2	3	4	5
	children	play	cricket	with	friends
	NN	VP	S	-	S
		VB	VP	-	VP
			NN	-	NP
				PP	PB
					NN



Question 4:

Using CKY algorithm, find the number of parse trees for the sentence S_2 = **children enjoy music** and the probability score for the most probable tree. **[2 marks]**

A) $1, 4.95 \times 10^{-3}$
B) $2, 0.36 \times 10^{-3}$
C) $3, 0.99 \times 10^{-3}$
D) $2, 0.54 \times 10^{-3}$

Solution

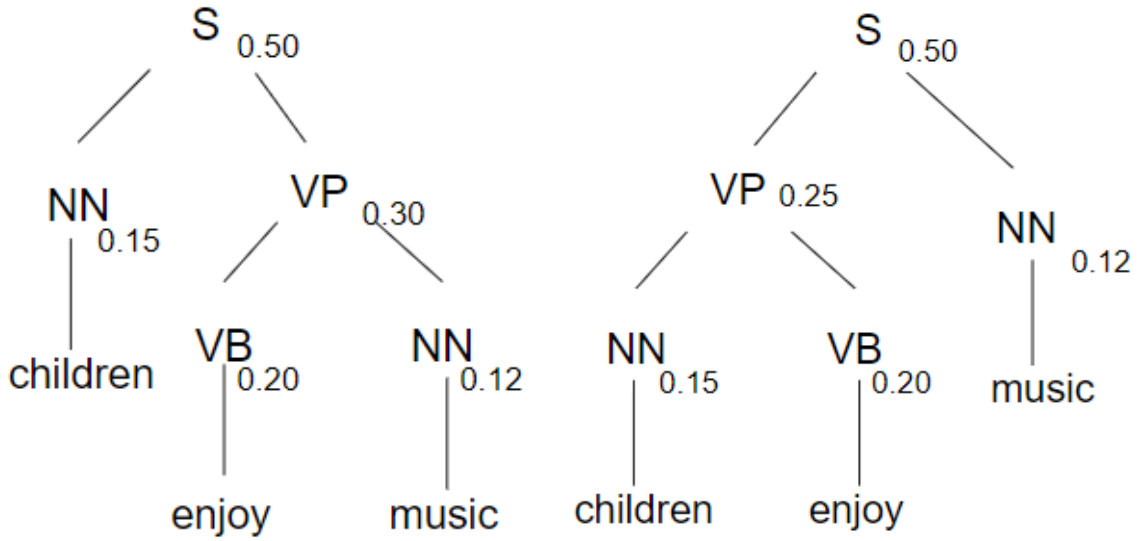
Answer: D

Solution:

There are two parse trees.

$S \rightarrow NN_{11}\, VP_{23} = 0.5 \times 0.15 \times (0.3 \times 0.2 \times 0.12) = 0.54 \times 10^{-3}$ $S \rightarrow VP_{12}\, NN_{33} = 0.5 \times (0.25 \times 0.15 \times 0.2) \times 0.12 = 0.45 \times 10^{-3}$

0	1	2	3
	children	enjoy	music
	NN	VP	S, S
		VB	VP
			NN



Question 5:

Consider the expression below:

P("children watch football enjoy music", $N_{34}|G$) = P_j P("children watch football enjoy music" $|N^j_{34}, G$)

What does the L.H.S. represent? **[1 mark]**

A) Probability of the sentence "children watch football enjoy music", given a grammar G.
B) Probability of the sentence "children watch football enjoy music", given a grammar G and some rule which derives the segment "football enjoy".
C) Probability of the sentence "children watch football enjoy music", given a grammar G and that there is some consistent spanning of the segment "football enjoy", i.e. from word 3 to 4.
D) None of the above

Solution

Question 5:
Consider the expression below:

P("children watch football enjoy music", $N_{34}|G$) = P_j P("children watch football enjoy music" $|N^j_{34}$, G)
What does the L.H.S. represent? **[1 mark]**

A) Probability of the sentence "children watch football enjoy music", given a grammar G.
B) Probability of the sentence "children watch football enjoy music", given a grammar G and some rule which derives the segment "football enjoy".
C) Probability of the sentence "children watch football enjoy music", given a grammar G and that there is some consistent spanning of the segment "football enjoy", i.e. from word 3 to 4.
D) None of the above

Answer: C

Question 6: Which of the following are true with respect to a Top-Down and Bottom-Up Parser?

A) A Top-Down Parser never explores options that will not lead to a full parse.

B) A Bottom-Up Parser never explores options that will not lead to a full parse.

C) A Top-Down Parser never explores options that do not connect to the actual sentence.

D) A Bottom-Up Parser never explores options that do not connect to the actual sentence.

Solution

Answer: A, D

- Top down never explores options that will not lead to a full parse, but can explore many options that never connect to the actual sentence.
- Bottom up never explores options that do not connect to the actual sentence but can explore options that can never lead to a full parse.

Question 7: Consider the CFG given below:

S -> ASA | aB

A -> B | S

B -> b | ε

How many non-terminals need to be added to convert the above grammar into CNF?

A) 1

B) 4

C) 2

D) 3

Solution

Answer: D

Solution: Given: S → ASA|aB, A → B | S, B → b | ε

1. Since the start symbol S occurs on the right side of a production rule, we create a new start symbol S' and add a new production rule S' → S.

   S' → S, S → ASA | aB, A → B | S, B → b | ε

1. Remove the null productions: B → ε, A → ε
   a. After removing B → ε: S' → S, S → ASA|aB|a, A → B|S|ε, B → b
   b. After removing A → ε: S' → S, S → ASA|aB|a|AS|SA|S, A → B|S, B → b
2. Remove the unit productions: S → S, S' → S, A → B, A → S

   S' → ASA|aB|a|AS|SA, S → ASA|aB|a|AS|SA, A → b|ASA|aB|a|AS|SA, B→ b

Solution

S’ → ASA|aB|a|AS|SA, S → ASA|aB|a|AS|SA, A → b|ASA|aB|a|AS|SA, B→ b

4. Handling production rules with more than two non-terminals on RHS:

S’ → AX|aB|a|AS|SA, S → AX|aB|a|AS|SA, A → b|AX|aB|a|AS|SA, B→ b,X → SA

5. Since the right-side of the production rules contain both terminal and non-terminal, we add the following rules: Y → a

S’ → AX|YB|a|AS|SA, S → AX|YB|a|AS|SA, A → b|AX|YB|a|AS|SA, B→ b,
X → SA, **Y** → a

Natural Language Processing

Week 6

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Question 1: Which of the following is/are true about the Chu-Liu-Edmonds Algorithm?

1. Each vertex in the graph greedily selects the incoming edge with the highest weight

2. During the iteration of algorithm it always produce minimum spanning tree

3. During the iteration of algorithm it never produces cycle

4. The running time of the Algorithm is O(EV) where V be the set of nodes and E be the set of directed edges

Solution

Answer: 1, 4

Solution: During the iteration of the algorithm it always produces a Maximum spanning tree and it might produce cycle also.

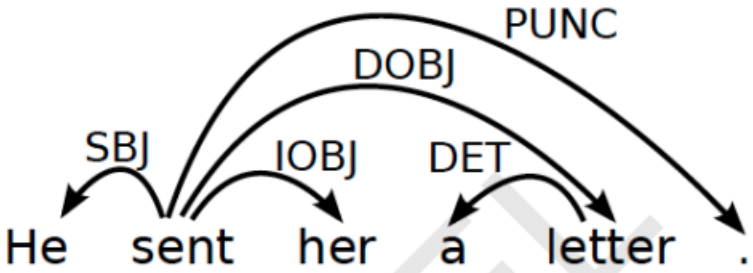
Chu-Liu-Edmonds Algorithm

- Each vertex in the graph greedily selects the incoming edge with the highest weight.
- If a tree results, it must be a maximum spanning tree.
- If not, there must be a cycle.
  - Identify the cycle and contract it into a single vertex.
  - Recalculate edge weights going into and out of the cycle.

NPTEL

Question 2. With respect to a Dependency Structure, which of the following is not a valid criterion for a syntactic relation between a head H and a dependent D in a construction C?

1. The form of D depends on H.
2. The form of H depends on D.
3. H selects D and determines whether D is obligatory.
4. H specifies D.



Criteria for a syntactic relation between a head H and a dependent D in a construction C

- H determines the syntactic category of C; H can replace C.
- D specifies H.
- H is obligatory; D may be optional.
- H selects D and determines whether D is obligatory.
- The form of D depends on H (agreement or government).
- The linear position of D is specified with reference to H.

NPTEL

Solution

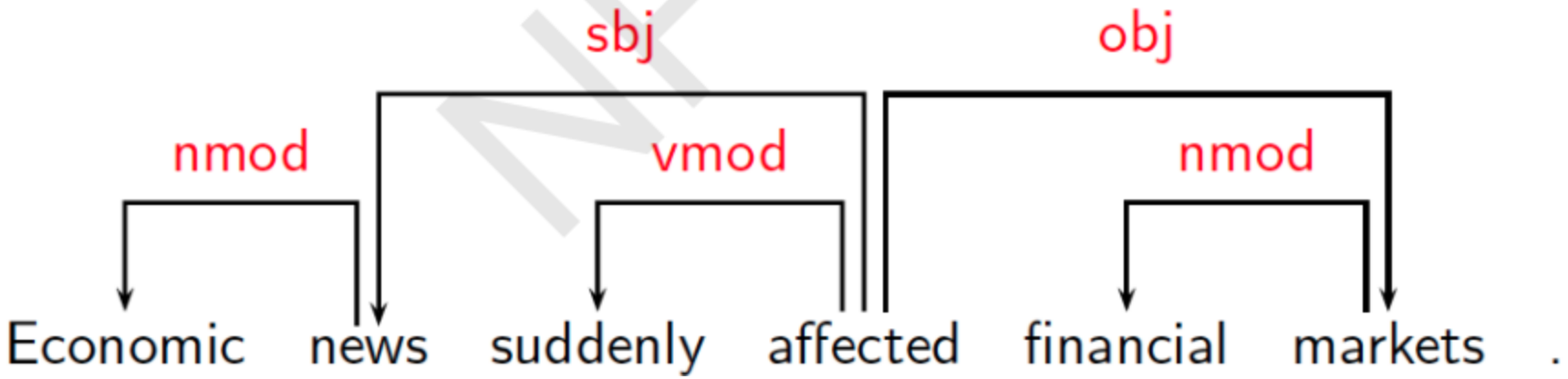
Answer: 2, 4

Question 3: Consider the sentence: “Ramesh scored a brilliant century”. What is the type of the following relation?

century -> brilliant

1. Endocentric
2. Exocentric
3. Both endocentric and exocentric
4. None of the above

Construction	Head	Dependent
Exocentric	Verb	Subject (sbj)
	Verb	Object (obj)
Endocentric	Verb	Adverbial (vmod)
	Noun	Attribute (nmod)



NPTEL

Solution

Answer: 1

Question 4: Which of the following is /are false about data driven deterministic parsing?

1. Deterministic parsing requires an oracle
2. An oracle can be approximated by a classifier
3. A classifier can be trained using treebank data
4. None of the above

Solution

Answer: 4

Solution:

For data driven deterministic parsing 1,2,3 all are true.

Data-driven deterministic parsing:

- Deterministic parsing requires an **oracle**.
- An oracle can be approximated by a **classifier**.
- A classifier can be trained using **treebank** data.

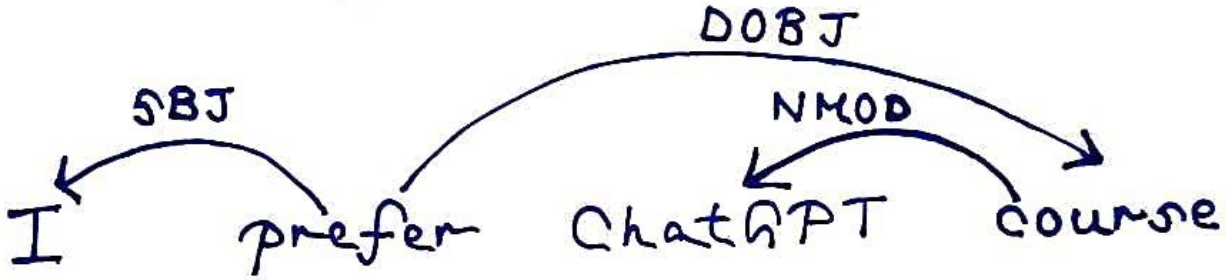
Question 5: Suppose you write down the sequence of actions that generate the parse tree of the sentence “I prefer ChatGPT course" using Arc-Eager Parsing. Assume the sentence is a gold-standard parse in your training data.The number of times you have to use Right Arc, Left Arc, Reduce, Shift is:

Format of the answer is [a, b, c, d] corresponding to the 4 values in the order specified in the query.

1. [3, 0, 2, 1]
2. [1, 2, 1, 3]
3. [1, 2, 0, 3]
4. [1, 2, 0, 2]

Solution

Answer: 3



Question 6: Correct sequence of actions that generates the parse tree of the sentence “I prefer ChatGPT course” using Arc-Eager Parsing is:

Note: Right Arc (RA), Left Arc(LA), Reduce(RE), Shift(SH)

1. SH->LA->SH->SH->LA->RA
2. SH->LA->SH->RE->LA->RA
3. SH->LA->SH->SH->RA->LA
4. SH->LA->RE-->SH->SH->LA

Solution

Answer: 1

Stack	Buffer	Arcs	Transition
[]	[I,prefer,ChatGPT,course]		SH
[I]	[prefer,ChatGPT,course]		LA
[]	[prefer,ChatGPT,course]	I ← prefer	SH
[prefer]	[ChatGPT,course]		SH
[prefer,ChatGPT]	[course]	ChatGPT ← course	LA
[prefer]	[course]	prefer → course	RA
[prefer,course]	[]		

Question 7: Suppose you are training MST Parser for dependency and the sentence, “I like online exam” occurs in the training set. The POS tags for these words are Pronoun, Verb, PropNoun and Noun, respectively. Also, for simplicity, assume that there is only one dependency relation, “rel”. Thus, for every arc from word wi to wj, your features may be simplified to depend only on words wi and wj and not on the relation label. Below is the set of features

f1: pos(wi) = Verb and pos(wj) = Noun|Pronoun

f2: wi = Root | wi occurs before wj in the sentence

f3: wi = Root and pos(wj) = Verb

f4: wj occurs before wi in the sentence

Question 7: The feature weights before the start of the iteration are: [5,20,15,12]

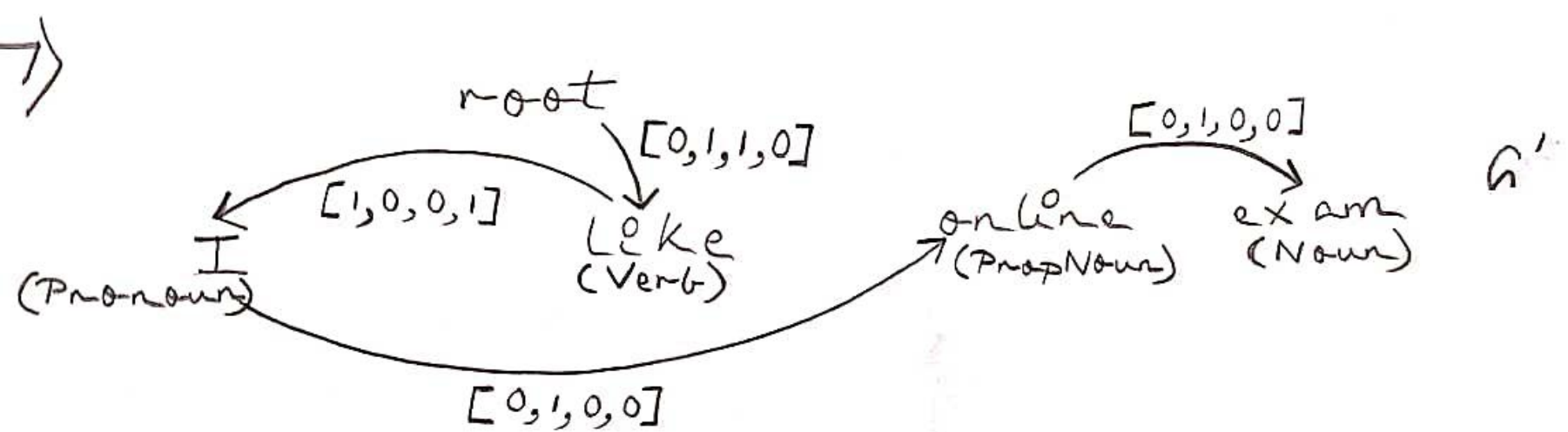
Suppose you are also given that after applying the Chu-Liu Edmonds, you get the following parse tree {Root → like, like → I, I → online, online → exam}

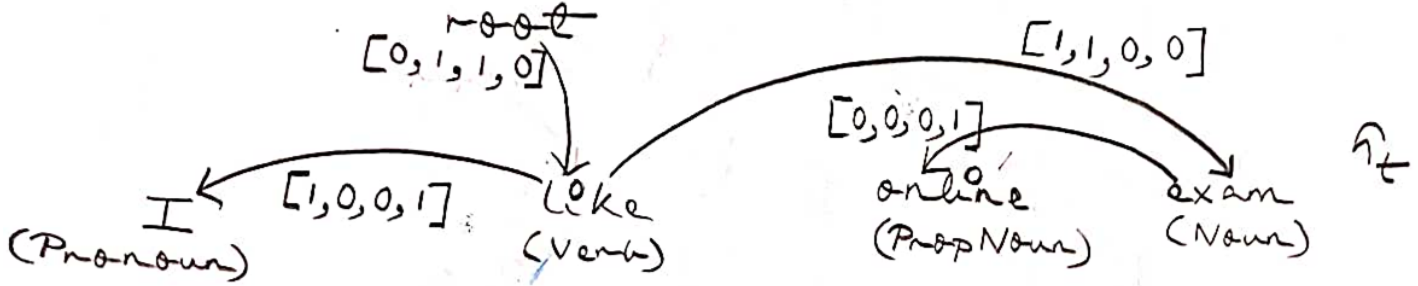
What would be the weights after this iteration?

1. [6, 19, 14, 13]
2. [6, 19, 15, 13]
3. [6, 19, 13, 13]
4. [6, 19, 15, 12]

Solution

Answer: 2





$$w_{old} = [5, 20, 15, 12]$$

$$f(h_t) = [2, 2, 1, 2]$$

$$f(h') = [1, 3, 1, 1]$$

$$f(h_t) - f(h') = [1, -1, 0, 1]$$

$$w_{new} = w_{old} + f(h_t) - f(h')$$

$$= [5, 20, 15, 12] + [1, -1, 0, 1]$$

$$= [6, 19, 15, 13]$$

Question 8: Which of the following is true about the formal conditions on dependency graph ?

1. Graph G is connected and projective
2. G is connected but not acyclic
3. G acyclic and obeys the single head constant
4. Both 1 and 3

- G is connected:
  - For every node i there is a node j such that $i \rightarrow j$ or $j \rightarrow i$.
- G is acyclic:
  - if $i \rightarrow j$ then not $j \rightarrow^* i$.
- G obeys the single head constraint:
  - if $i \rightarrow j$ then not $k \rightarrow j$, for any $k \neq i$.
- G is projective:
  - if $i \rightarrow j$ then $j \rightarrow^* k$, for any k such that both j and k lie on the same side of i.

Connectedness, Acyclicity and Single-Head

- **Connectedness:** Syntactic structure is complete.
- **Acyclicity:** Syntactic structure is hierarchical.
- **Single-Head:** Every word has at most one syntactic head.
- **Projectivity:** No crossing of dependencies.

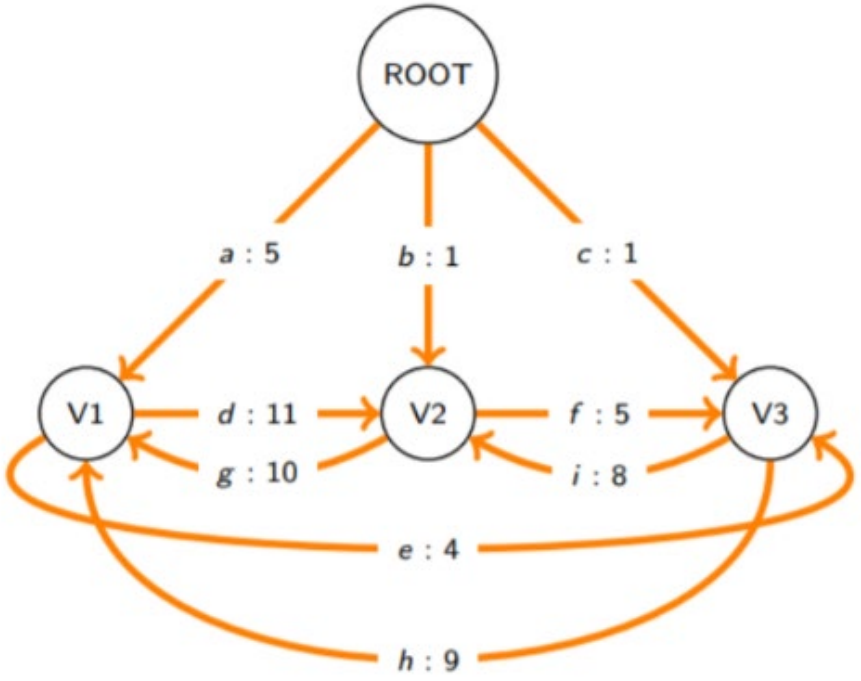
Solution

Answer: 4

Solution:

The formal conditions on dependency graphs are that G is connected, acyclic, projective and should obey single head constraint.

Question 9: Consider the following graph with a root node and 3 other vertices. The edge weights between all the pair of modes have been provided. Suppose you use Chu-Liu-Edmonds algorithm to find the MST for this graph. Which pair of nodes will have to be contracted to form a single vertex during the algorithm in the 1st iteration?



1. (V2, V3)

2. (V1, V3)

3. All these pairs will get contracted at different times in the algorithm

4. (V1, V2)

Solution

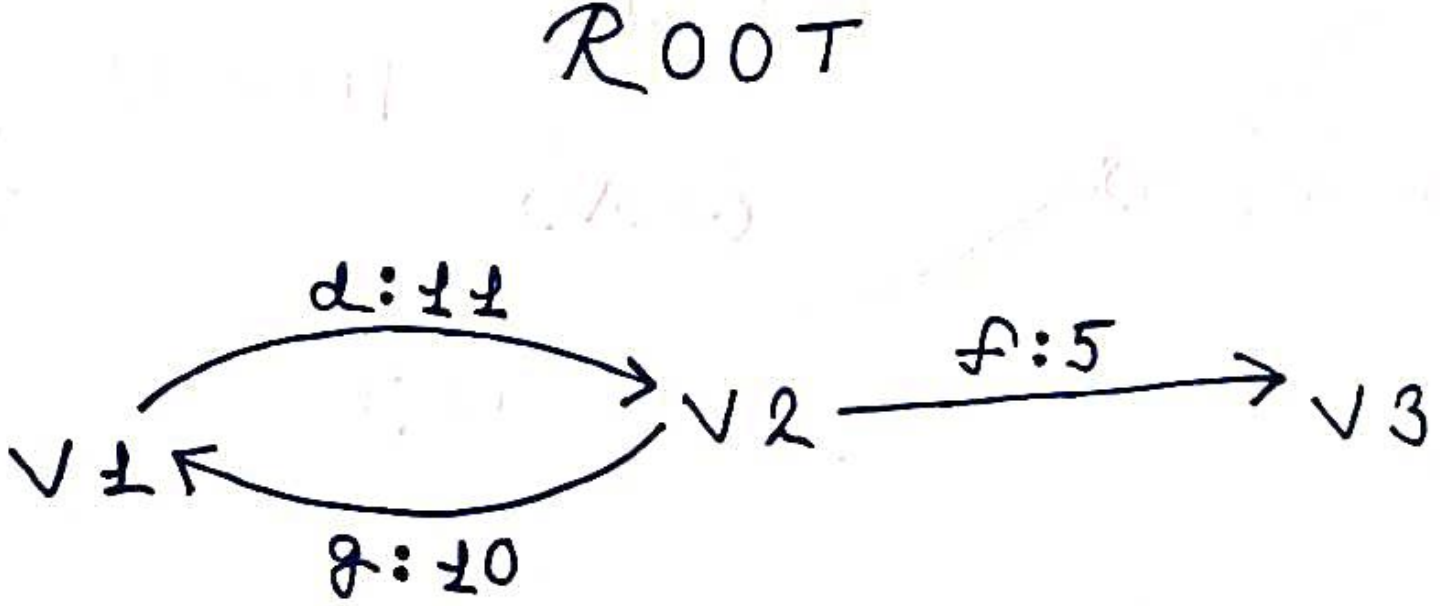
Answer: 4

Solution:

Chu-Liu-Edmonds Algorithm

- Each vertex in the graph greedily selects the incoming edge with the highest weight.
- If a tree results, it must be a maximum spanning tree.
- If not, there must be a cycle.
  - Identify the cycle and contract it into a single vertex.
  - Recalculate edge weights going into and out of the cycle.

NPTEL


ROOT
d:11
f:5
V1
V2
V3
g:10

Natural Language Processing

Week 7

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Question 1: Suppose you have a raw text corpus and you compute word co occurrence matrix from there. Which of the following algorithm(s) can you utilize to learn word representations? (Choose all that apply)

a. CBOW

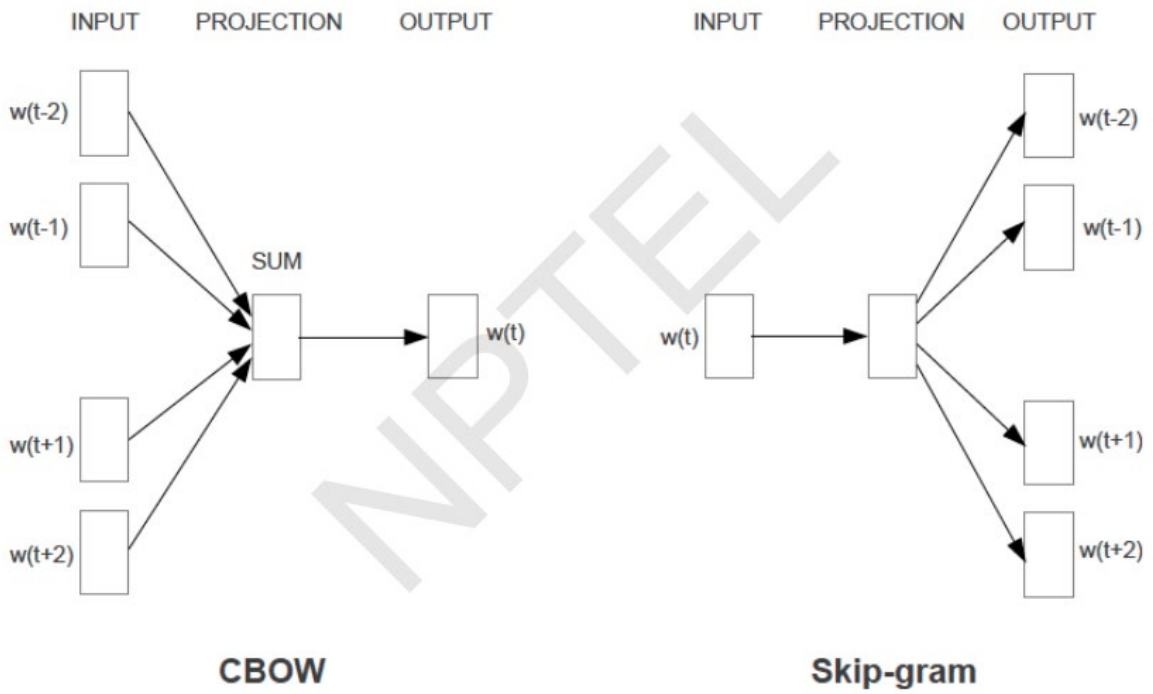
b. SVD

c. PCA

d. GloVe

Solution

Answer: a, b, c, d



NPTEL

Question 2: What is the method for solving word analogy questions like, given A, B and D, find C such that A:B::C:D, using word vectors?

a. vc = va + (vb − vd), then use cosine similarity to find the closest word of vc.

b. vc = va + (vd − vb) then do dictionary lookup for vc

c. vc = vd + (vb − va) then use cosine similarity to find the closest word of vc.

d. vc = vd + (va − vb) then do dictionary lookup for vc.

e. None of the above

Solution

Answer: e

Solution: vd − vc = vb − va

vc = vd + va − vb then use cosine similarity to find the closest word of vc.

Question 3: What is the value of PMI(w1, w2) for C(w1) = 100, C(w2) = 2000, C(w1,w2) = 64, N = 100000? N: Total number of documents.

C(wi): Number of documents, wi has appeared in.

C(wi, wj): Number of documents where both the words have appeared in. Note:

Use base 2 in logarithm.

a. 4

b. 5

c. 6

d. 5.64

$$\overline{PMI}(w_1, w_2) = log_2 \frac{P_{corpus}(w_1, w_2)}{P_{ind}(w_1, w_2)}$$

$$PMI(w_1, w_2) = log_2 \frac{P_{corpus}(w_1, w_2)}{P_{corpus}(w_1) P_{corpus}(w_2)}$$

$$P_{corpus}(w_1, w_2) = \frac{freq(w_1, w_2)}{N}$$

$$P_{corpus}(w) = \frac{freq(w)}{N}$$

Solution:

$P_{corpus}(w1,w2) = 64/100000,$

$P_{corpus}(w1) = 100/100000,$ $P_{corpus}(w2) = 2000/100000$

Solution

Answer: b

Solution:

$$PMI = \log_2 \frac{64 \times 100000}{100 \times 2000} = 5$$

Question 4: Given two binary word vectors w1 and w2 as follows:

w1 = [1010101010]

w2 = [0011111100]

Compute the Dice and Jaccard similarity between them. [2 marks]

a. 6/11, 3/8

b. 10/11, 5/6

c. 4/9, 2/7

d. 5/9, 5/8

Solution

Similarity Measures

Dice coefficient : $\frac{2|X \cap Y|}{|X|+|Y|}$

Jaccard Coefficient : $\frac{|X \cap Y|}{|X \cup Y|}$

X = [1010101010]

Y = [0011111100]

X ∩ Y = [0010101000]

|X ∩ Y| = 3, |X| = 5, |Y| = 6

X ∪ Y = [1011111110],|X ∪ Y| = 8

Answer: a

$$\text{Dice coefficient} = \frac{2 \times 3}{5+6} = \frac{6}{11}$$

$$\text{Jaccard coefficient} = \frac{3}{8}$$

Question 5: Consider two probability distributions for two words be p and q. Compute their similarity scores with KL-divergence. [2mark]

p = [0.20, 0.75,0.50]

q = [0.90, 0.10, 0.25]

Note: Use base 2 in logarithm.

a. 4.704, 1,720

b. 1.692, 0.553

c. 2.246, 1.412

d. 3.213, 2.426

Solution

p = [0.20, 0.75,0.50], q = [0.90, 0.10, 0.25]

Answer: c

Solution:

$$\begin{aligned}
\text{KL-div}(p, q) &= \sum_i p_i \log_2 \frac{p_i}{q_i} \\
&= 0.2 \log \frac{0.2}{0.9} + 0.75 \log \frac{0.75}{0.1} + 0.5 \log \frac{0.5}{0.25} \\
&\approx 2.246 \\
\text{KL-div}(q, p) &= 0.9 \log \frac{0.9}{0.2} + 0.1 \log \frac{0.1}{0.75} + 0.25 \log \frac{0.25}{0.5} \\
&\approx 1.412
\end{aligned}$$

Question 6: Consider the following word co-occurrence matrix given below. Compute the cosine similarity between (i) w1 and w2, and (ii) w1 and w3. [2 mark]

	w4	w5	w6
w1	2	9	4
W2	1	5	6
W3	3	0	1

a. 0.773, 0.412

b. 0.881, 0.764

c. 0.665, 0.601

d. 0.897, 0.315

Solution

Answer: d
Solution:

w1 = [2,9,4], w2 = [1,5,6], w3 = [3,0,1]

$$\text{cosine-sim}\ (\vec{p}, \vec{q}) = \frac{\vec{p} \cdot \vec{q}}{\|\vec{p}\| \cdot \|\vec{q}\|}$$

$$\text{cosine-sim}\ (\text{w1}, \text{w2}) = \frac{2 \times 1 + 9 \times 5 + 4 \times 6}{\sqrt{2^2 + 9^2 + 4^2} \times \sqrt{1^2 + 5^2 + 6^2}} \approx 0.897$$

$$\text{cosine-sim}\ (\text{w1}, \text{w3}) \approx 0.315$$

Natural Language Processing

Week 8

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Assume that you are learning a classifier for the data-driven deterministic parsing and the sentence ‘I prefer ChatGPT course’ is a gold-standard parse in your training data. You are also given that ‘ChatGPT’ and ‘course’ are ‘Nouns’, ‘I’ is a ‘Pronoun’ while the POS tag of ‘prefer’ is ‘Verb’. Obtain the dependency graph for this sentence on your own. Assume that your features correspond to the following conditions:

1. The stack is empty.

2. Top of stack is Noun and Top of buffer is Verb.

3. Top of stack is Pronoun and Top of buffer is Verb.

4. The word at the top of stack occurs before word at the top of the buffer in the sentence

The initial weights of your features are [2,2,2,2 | 3,3,3,2| 2,2,2,2 | 2,2,2,2] where the first four features correspond to LA, and then to RA, SH and RE, respectively. Use this gold standard parse during online learning. What will be the weights after completing two iteration of Arc-Eager parsing over this sentence:

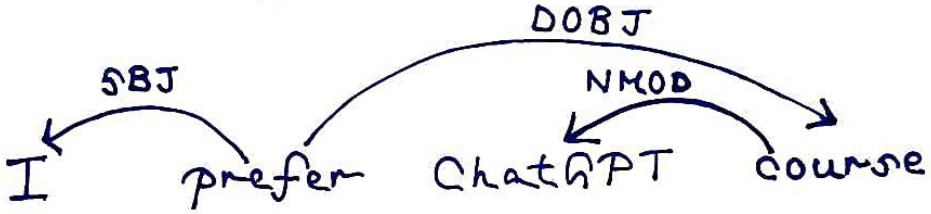
LEARN($\{T_1, \ldots, T_N\}$)

1 $w \leftarrow 0.0$
2 **for** i in $1..K$
3 **for** j in $1..N$
4 $c \leftarrow ([]_S, [w_1, \ldots, w_{n_j}]_B, \{\})$
5 **while** $B_c \neq []$
6 $t^* \leftarrow \arg\max_t w.f(c,t)$
7 $t_o \leftarrow o(c, T_i)$
8 **if** $t^* \neq t_o$
9 $w \leftarrow w + f(c,t_o) - f(c,t^*)$
10 $c \leftarrow t_o(c)$
11 **return** w

Oracle $o(c, T_i)$ returns the optimal transition of c and T_i

NPTEL

Solution



Given - The first four features correspond to LA, and then to RA, SH and RE, respectively.

f(c,t) = [c_1&&LA, c_2&&LA, c_3&&LA, c_4&&LA | c_1&&RA, c_2&&RA, c_3&&RA, c_4&&RA | c_1&&SH, c_2&&SH, c_3&&SH, c_4&&SH | c_1&&RE, c_2&&RE, c_3&&RE, c_4&&RE]

Initial weight: w = [2,2,2,2 | 3,3,3,2| 2,2,2,2 | 2,2,2,2]

Solution

Current configuration = $[]_S$, $[I,prefer,ChatGPT,course]_B$, {}

Conditions: 1. The stack is empty. 2. Top of stack is Noun and Top of buffer is Verb. 3. Top of stack is Pronoun and Top of buffer is Verb. 4. The word at the top of stack occurs before word at the top of the buffer in the sentence.

f(c,t) = [c_1&&LA, c_2&&LA, c_3&&LA, c_4&&LA | c_1&&RA, c_2&&RA, c_3&&RA, c_4&&RA | c_1&&SH, c_2&&SH, c_3&&SH, c_4&&SH | c_1&&RE, c_2&&RE, c_3&&RE, c_4&&RE]

f(c,LA) = [1, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0]

f(c,RA) = [0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0]

f(c,SH) = [0, 0, 0, 0 | 0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0]

f(c,RE) = [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 1, 0, 0, 0]

Solution

w = [2,2,2,2 | 3,3,3,2| 2,2,2,2 | 2,2,2,2]

w * f(c,LA) = w * [1, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0] = 2

w * f(c,RA) = w * [0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0] = 3

w * f(c,SH) = w * [0, 0, 0, 0 | 0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0] = 2

w * f(c,RE) = w * [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 1, 0, 0, 0] = 2

$t^* = RA$, $t_0 = SH$, $t^* \neq t_0$

$w_{new} = w_{old} + f(c, t_0) - f(c,t^*) = w_{old} + f(c, SH) - f(c,RA)$

= [2,2,2,2 | 3,3,3,2| 2,2,2,2 | 2,2,2,2] + [0, 0, 0, 0 | 0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0]
- [0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0] = [2,2,2,2 | 2,3,3,2| 3,2,2,2 | 2,2,2,2]

Solution

Current configuration = $[I]_S$, $[prefer,ChatGPT,course]_B$, {}

Conditions: 1. The stack is empty. 2. Top of stack is Noun and Top of buffer is Verb. 3. Top of stack is Pronoun and Top of buffer is Verb. 4. The word at the top of stack occurs before word at the top of the buffer in the sentence.

f(c,t) = [c_1&&LA, c_2&&LA, c_3&&LA, c_4&&LA | c_1&&RA, c_2&&RA, c_3&&RA, c_4&&RA | c_1&&SH, c_2&&SH, c_3&&SH, c_4&&SH | c_1&&RE, c_2&&RE, c_3&&RE, c_4&&RE]

f(c,LA) = [0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0]

f(c,RA) = [0, 0, 0, 0 | 0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0]

f(c,SH) = [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 1, 1 | 0, 0, 0, 0]

f(c,RE) = [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 1, 1]

Solution

w = [2,2,2,2 | 2,3,3,2| 3,2,2,2 | 2,2,2,2]

w * f(c,LA) = w * [0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0] = 4

w * f(c,RA) = w * [0, 0, 0, 0 | 0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0] = 5

w * f(c,SH) = w * [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 1, 1 | 0, 0, 0, 0] = 4

w * f(c,RE) = w * [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 1, 1] = 4

$t^* = RA$, $t_0 = LA$, $t^* \neq t_0$ $w_{new} = w_{old} + f(c, t_0) - f(c,t^*) = w_{old} + f(c, LA) - f(c,RA)$

= [2,2,2,2 | 2,3,3,2| 3,2,2,2 | 2,2,2,2] + [0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0]
- [0, 0, 0, 0 | 0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0] = [2,2,3,3 | 2,3,2,1| 3,2,2,2 | 2,2,2,2]

Question 1:

Consider the following sentences:

1. I need to write an essay tonight to make sure I get everything right for the upcoming exam.

2. Near the river bank, I sat on the grass and thought I need to visit the bank to deposit a check today.

The lexical relation between the highlighted words in sentences 1, 2 are

a. Homophones, Homonymy

b. Homograph, Synonym

c. Homonymy, Homophones

d. Synonym, Hyponym

Solution

Answer: a

Solution:

What is a lexeme?

Lexeme should be thought of as a pairing of a particular orthographic and phonological form with some sort of symbolic meaning representation.

- Orthographic form, and phonological form refer to the appropriate form part of a lexeme
- Sense refers to a lexeme's meaning counterpart.

Homonymy

Definition

Homonymy is defined as a relation that holds between words that have the same form with unrelated meanings.

Examples

- Bat (wooden stick-like thing) vs Bat (flying mammal thing)
- Bank (financial institution) vs Bank (riverside)

homophones and homographs

homophones are the words with the same pronunciation but different spellings.

- write vs right
- piece vs peace

homographs are the lexemes with the same orthographic form but different meaning. Ex: bass

NPTEL

Hyponymy

One sense is a hyponym of another if the first sense is more specific, denoting a subclass of the other

- *car* is a hyponym of *vehicle*
- *dog* is a hyponym of *animal*
- *mango* is a hyponym of *fruit*

Synonymy

Words that have the same meaning in some or all contexts.

- filbert / hazelnut
- couch / sofa
- big / large
- automobile / car

Polysemy

*Multiple **related** meanings within a single lexeme.*

- The *bank* was constructed in 1875 out of local red brick.
- I withdrew the money from the *bank*.

Are those the same sense?

- Sense 1: “The building belonging to a financial institution”
- Sense 2: “A financial institution”

NPTEL

Question 2:

Consider the following sentences. Which of the following is/are True?

a. Dog is a hyponym of animal.

b. Fruit is a hypernym of apple.

c. Animal is hyponym of dog.

d. Guava is hypernym of fruit.

Solution

Answer: a, b

Solution:

Hyponymy

One sense is a hyponym of another if the first sense is more specific, denoting a subclass of the other

- *car* is a hyponym of *vehicle*
- *dog* is a hyponym of *animal*
- *mango* is a hyponym of *fruit*

Question 3:

Two concepts along with their glosses are given below. Find the similarity score between concepts “currency” and “money” with the Extended Lesk’s algorithm. (Note: Do not consider the stop words.)

currency : the metal or paper medium of exchange that is presently used

money : the most common exchange medium presently used

a. 2

b. 3

c. 6

d. 9

Solution

Answer: c

Solution:

currency : the metal or paper **medium** of **exchange** that is **presently used**

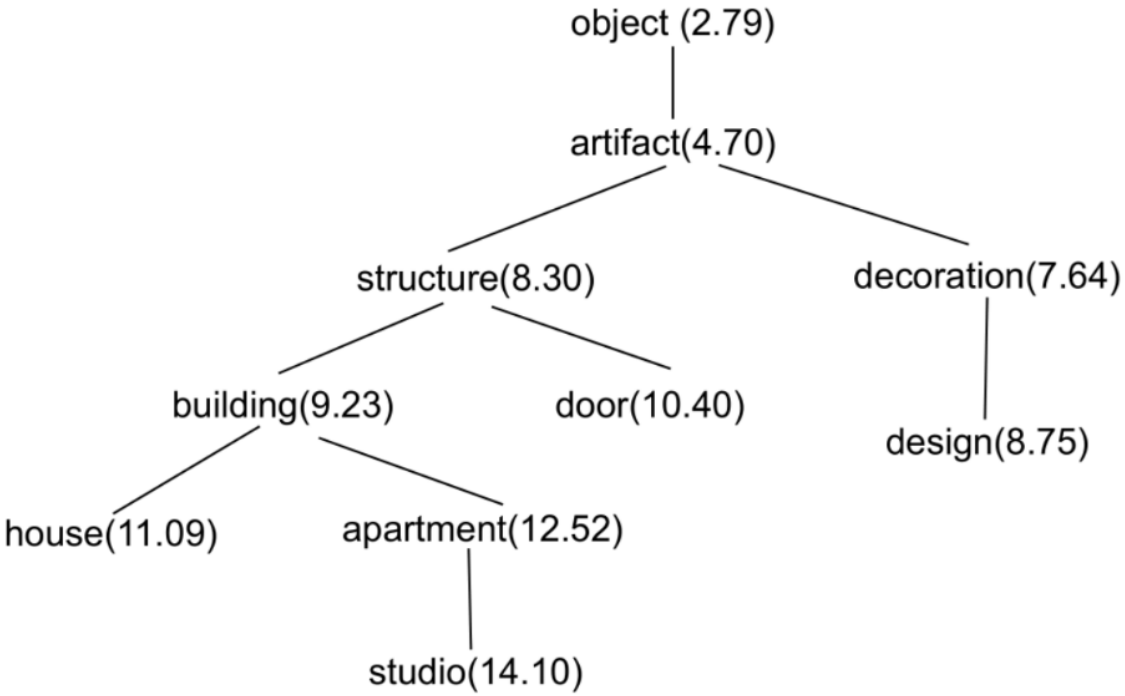
money : the most common **exchange medium presently used**

common words are : medium, exchange, presently used

score = 1^2 + 1^2 + 2^2= 6

For Question 4 to 6, consider a hypothetical wordnet noun taxonomy with their information content as shown in Figure 1. Question 4 carries 2 marks.

Note: Use base 10 in logarithmic calculations



Question 4:

What is the Lin similarity between house and design?

a. 0.564

b. 0.433

c. 0.466

d. 0.473

Solution

Answer: d

Solution: 2*IC(LCS(c_1,c_2))/(IC(c_1) + IC(c_2))

= 2*IC(artifact)/(IC(house) + IC(design)) = (2×4.7)/(11.09+8.75) ≈ 0.473

Information content

- Information content: $IC(c) = -logP(c)$
- Lowest common subsumer : $LCS(c_1, c_2)$: the lowest node in the hierarchy that subsumes (is a hypernym of) both c_1 and c_2

$$sim_{Lin}(c_1, c_2) = \frac{2logP(LCS(c_1, c_2))}{logP(c_1) + logP(c_2)}$$

Question 5:

What is the Resnic similarity between building and door?

a. 11.09

b. 8.30

c. 9.23

d. 4.70

Solution

Answer: b

Solution: IC(LCS(c_1,c_2)) = IC(LCS(building,door) = IC(structure) = 8.30

Resnik Similarity

- Intuition: how similar two words are depends on how much they have in common
- It measures the commonality by the information content of the lowest common subsumer
- $sim_{resnik}(c_1, c_2) = IC(LCS(c_1, c_2)) = -logP(LCS(c_1, c_2))$

NPTEL

Question 6:

What is the Leacock–Chodorow similarity between building and design?

a. 0.398

b. 0.699

c. 0.097

d. None of the above

Solution

Answer: a

Solution:

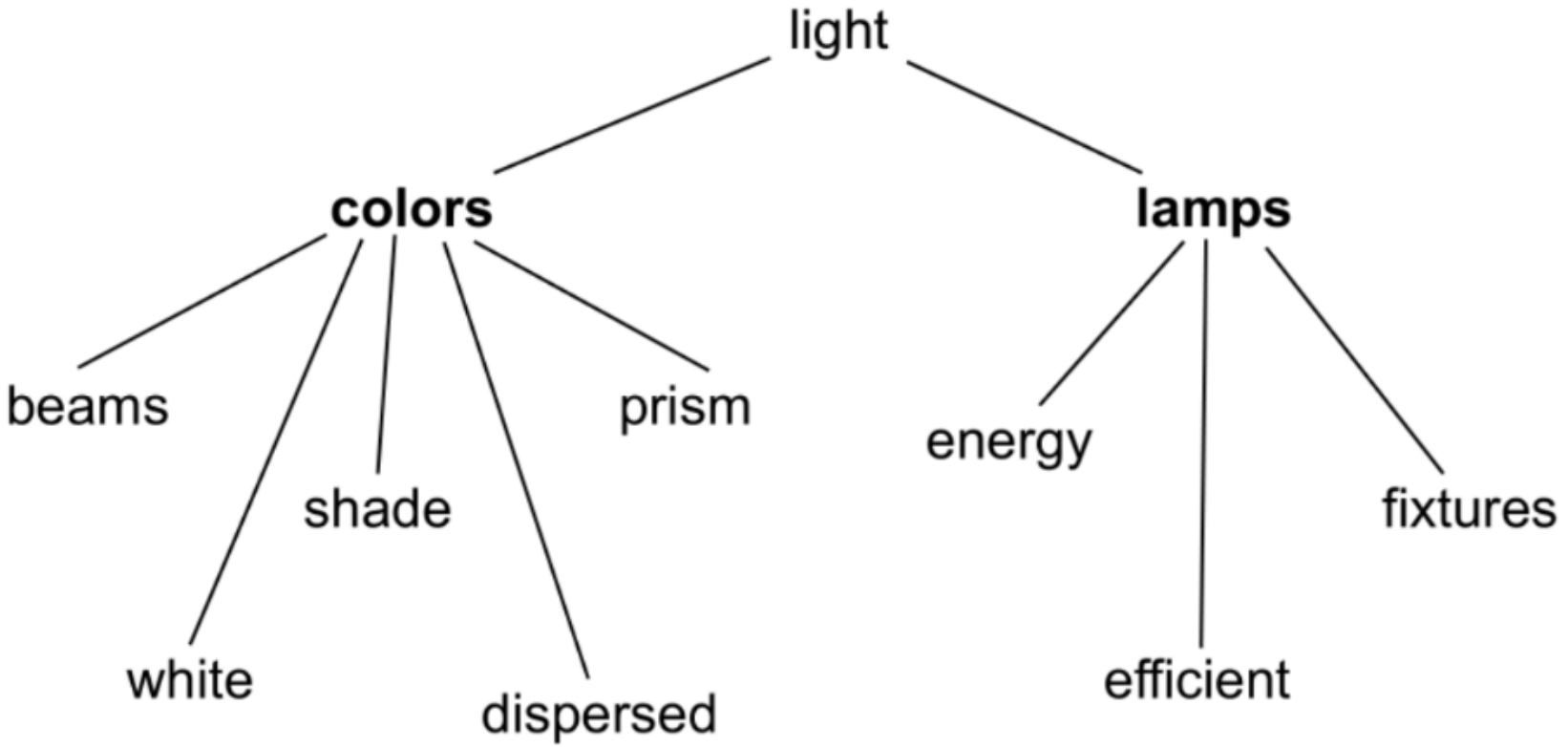
LC similarity = −log pathlen(c1,c2)/2d = −log 4/(2×5) ≈ 0.398

L-C similarity

$$sim_{LC}(c_1, c_2) = -log(pathlen(c_1, c_2)/2d)$$

d: maximum depth of the hierarchy

For Question 7 to 9 consider the network of words for disambiguation of the word "light" as shown in Figure 3. The hubs are "colors" and "lamps". Note: Take the distance between two words as the path length between them.



Question 7:

Compute the scores for (i) the hub “colors” and the component “white” and (ii) the hub “colors” and the component “fixtures”.

a. 0.2, 0.25

b. 1.0, 0.0

c. 0.5, 0.25

d. None of the above

Solution

Answer: d

Solution:

(i) 1/(1+1) = 0.5

(ii) 0 as “colors” is not an ancestor of “fixtures”

$$s_k = \frac{1}{1+d(h_k, w_j)} \text{ if } h_k \text{ is an ancestor of } w_j$$

$$s_i = 0 \text{ otherwise.}$$

Question 8:

What are the scores of the hubs “colors” and “lamps” respectively?

a. 0.6, 0.4

b. 0.20, 0.33

c. 2.5, 1.5

d. None of the above

Solution

Answer: c

Solution: Each component's score is 0.5

Question 9:

Which is the most appropriate sense for the word “light”?

a. colors

b. lamps

c. both colors and lamps are appropriate

d. Not enough data

Solution

Answer: a

Solution: “colors” has the highest score

Natural Language Processing

Week 9

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Question 1: How does Correlated Topic Model create relations among topics?

1. By having lots of general words inside the topics
2. By Removing stop-words
3. By using logistic normal distribution
4. None of the above

Solution

Answer: 3

- The Dirichlet is an exponential family distribution on the simplex, positive vectors that sum to one
- However, the near independence of components makes it a poor choice for modeling topic proportions
- An article about *fossil fuels* is more likely to also be about *geology* than about *genetics*

Using logistic normal distribution

A multivariate normal distribution of a k-dimensional vector $x = [X_1, X_2, \ldots, X_k]$ can be written as

$$x \sim N_k(\mu, \Sigma)$$

with k-dimensional mean vector μ and $k \times k$ covariance matrix Σ

NPTEL

Question 2: Choose the correct statement from below –

I. A low value of alpha will assign fewer topics to each document whereas a high value of alpha will have the opposite effect.

II. A low value of beta will use fewer words to model a topic whereas a high value will use more words, thus making topics more similar between them.

III. LDA cannot decide on the number of topics by itself.

1. (I).

2. (II).

3. III).

4. All of the above.

Solution

Answer: 4

Solution:

All of the above

Question 3. Vikram has lots of documents and he wants to model the content as well as connections.

Which topic modelling technique will be suitable for it?

1. Correlated Topic Model
2. Relational Topic Model
3. Dynamic Topic Model
4. Supervised Latent Dirichlet Allocation

Solution

Question 3. Vikram has lots of documents and he wants to model the content as well as connections.

Which topic modelling technique will be suitable for it?

1. Correlated Topic Model

2. Relational Topic Model

3. Dynamic Topic Model

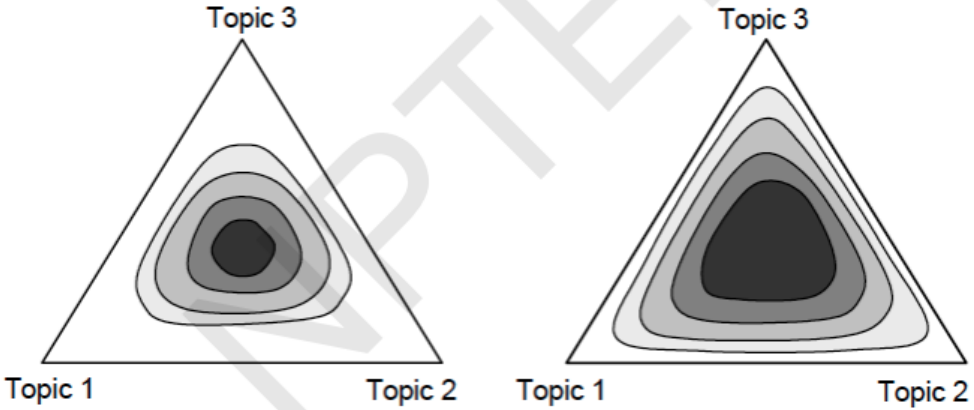
4. Supervised Latent Dirichlet Allocation

Answer: 2

Question 5: You have a topic model with the parameters $\alpha = 0.9$ and $\beta = 0.05$. Now, if you want to have sparser distribution over words and denser distribution over topics, what should be the values for α and β?

1. Both α and β values should be decreased
2. Both α and β values should be increased
3. α should be decreased, but β should be increased
4. α should be increased, but β should be decreased

Solution



Answer: 4

Solution:

α : topic distribution

β : word distribution

Therefore, α should be increased, but β should be decreased.

Question 4: In Topic modeling which hyperparameters tuning used for represents document-topic Density?

1. Dirichlet hyperparameter Beta
2. Dirichlet hyperparameter alpha
3. Number of Topics (K)
4. None of them

Solution

Answer: 2

Solution: alpha is used to represent document-topic intensity

Question 6 :

In Gibbs sampling choose the correct option from below

1. It can not directly estimate the posterior distribution over z
2. It is a form of Markov chain Monte Carlo
3. Here sampling is done in parallel
4. Sampling is stopped before sampled values approximate the target distribution

Gibbs Sampling

- A form of Markov chain Monte Carlo (MCMC), which simulates a high-dimensional distribution by sampling on lower-dimensional subset of variables where each subset is conditioned on the value of all others
- Sampling is done sequentially and proceeds until the sampled values approximate the target distribution
- It directly estimates the posterior distribution over z, and uses this to provide estimates for β and θ

NPTEL

Solution

Answer: 2

Solution: In gibbs sampling, we do sequential sampling until the sampled values approximate the target distribution. This also can directly estimate the posterior distribution over z.

Question 7 : Which of the following is/ are true ?

1. Dirichlet distribution is a family of exponential distribution
2. LDA is impacted by the order of documents
3. In LDA the number of latent clusters are identified automatically
4. All of the above are true

Solution

Answer: 1

Solution:

The order of documents does not matter in LDA, we need to identify the number of latent clusters in advance in the LDA topic model.

Dirichlet Distribution

The Dirichlet distribution is an exponential family distribution over the simplex, i.e. positive vectors that sum to one

For question 8 , 9 and 10 use the following information.

Suppose you are using Gibbs sampling to estimate the distributions, θ and β for topic models. The underlying corpus has 3 documents and 5 words, **{machine, learning, language, nature, vision}** and the number of topics is 2. At certain point, the structure of the documents looks like the following

Doc1: nature(1) language(1) vision(1) language(1) nature(1) nature(1) language(1) vision(1)
Doc2: nature(1) language(1) language(2) machine(2) vision(1) learning(2) language(1) nature(1)
Doc3: machine(2) language(2) learning(2) language(2) machine(2) machine(2) learning(2) language(2)

(number) –number inside the brackets denote the topic no. 1 and 2 denote whether the word is currently assigned to topics t1 and t2 respectively. η = 0.3 and α = 0.3

$$\beta_i^{(j)} = \frac{C_{ij}^{WT} + \eta}{\sum_{k=1}^{W} C_{kj}^{WT} + W\eta} \qquad \theta_j^{(d)} = \frac{C_{dj}^{DT} + \alpha}{\sum_{k=1}^{T} C_{dk}^{DT} + T\alpha}$$

For question 8,9,10 calculate the value upto 4 decimal points and choose your answer

Question 8 : Using the above structure the estimated value of $\beta_{nature}^{(2)}$ at this point is

1. 0.0240
2. 0.02459
3. 0.0260
4. 0.0234

Solution

Answer: 1

Doc1: nature(1) language(1) vision(1) language(1) nature(1) nature(1) language(1) vision(1)
Doc2: nature(1) language(1) language(2) machine(2) vision(1) learning(2) language(1) nature(1)
Doc3: machine(2) language(2) learning(2) language(2) machine(2) machine(2) learning(2) language(2)

Solution:

	t1	t2
machine	0	4
nature	5	0
language	5	4
vision	3	0
learning	0	3

$$\beta_i^{(j)} = \frac{C_{ij}^{WT} + \eta}{\sum_{k=1}^{W} C_{kj}^{WT} + W\eta}$$

$\beta_{nature}^{(2)} = (0+0.3)/(11+5*0.3) = 0.3/12.5 = 0.024$

Question 9 : Using the above structure the estimated value of θ_{t1}^{doc2}

1. 0.6562
2. 0.6162
3. 0.6385
4. 0.50000

Solution

Answer: 2

Solution:

Doc1: nature(1) language(1) vision(1) language(1) nature(1) nature(1) language(1) vision(1)
Doc2: nature(1) language(1) language(2) machine(2) vision(1) learning(2) language(1) nature(1)
Doc3: machine(2) language(2) learning(2) language(2) machine(2) machine(2) learning(2) language(2)

	t1	t2
doc1	8	0
doc2	5	3
doc3	0	8

$$\theta_j^{(d)} = \frac{C_{dj}^{DT} + \alpha}{\sum_{k=1}^{T} C_{dk}^{DT} + T\alpha}$$

θ_{t1}^{doc2} = (5+0.3)/(8+2*0.3) = 5.3/ 8.6 = 0.6162

Question 10 : Using the above structure the estimated value of θ_{t2}^{doc2}

1. 0.6562
2. 0.3975
3. 0.3837
4. 0.3707

Solution

Answer: 3

Doc1: nature(1) language(1) vision(1) language(1) nature(1) nature(1) language(1) vision(1)
Doc2: nature(1) language(1) language(2) machine(2) vision(1) learning(2) language(1) nature(1)
Doc3: machine(2) language(2) learning(2) language(2) machine(2) machine(2) learning(2) language(2)

	t1	t2
doc1	8	0
doc2	5	3
doc3	0	8

$$\theta_j^{(d)} = \frac{C_{dj}^{DT} + \alpha}{\sum_{k=1}^{T} C_{dk}^{DT} + T\alpha}$$

θ_{t2}^{doc2} = (3+0.3)/(8+2*0.3) = 3.3/8.6 = 0.3837

Natural Language Processing

Week 10

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Question 1. Different phases of entity linking are -

A) Candidate Selection -> Reference Disambiguation

B) Reference Disambiguation -> Candidate Selection -> Mention Identify

C) Mention Identify -> Candidate Selection -> Reference Disambiguation

D) All of the above

Solution

Answer: C

Mention Identify -> Candidate Selection -> Reference Disambiguation

Determine "linkable" phrases

mention detection - **MD**

Rank/Select candidate entity links

link generation - **LG**

Use "context" to disambiguate/filter/improve

disambiguation - **DA**

NPTEL

Question 2. The text span s = “Sea” occurs in 600 different Wikipedia articles.

c1 223

c2 161

c3 18

c4 11

No Link 187

Calculate the keyphraseness of “Sea”.

A) 0.232

B) 0.886

C) 0.688

D) 0.976

Solution

C) CF(si) / CF(s) = 223 +161+18+11 / 600 = 413 / 600 = 0.688

keyphraseness(w)

Number of Wikipedia articles that use it as an anchor, divided by the numer of articles that mention it at all.

$$\frac{\mathrm{CF}(w_l)}{\mathrm{CF}(w)}$$

Collection frequency
term w as a link to another Wikipedia article

Collection frequency
term w

NPTEL

Question 3. What is the commonness of (s, c2) in the previous question?

A) 0.765

B) 0.389

C) 0.145

D) 0.910

Solution

B) 161/(223+161+18+11) = 161/413 = 0.389

commonness(w, c)

The fraction of times, a particular sense is used as a destination in Wikipedia.

$$\frac{|L_{w,c}|}{\underbrace{\sum_{c'} |L_{w,c'}|}}$$

Number of links
with target c' and anchor text w

NPTEL

Question 4. Relevant feature/s for a supervised model for predicting the topics to be linked is/are:

A) Disambiguation Confidence

B) Relatedness

C) Link Probability

D) All of the above

Solution

Answer: D

- **Link Probability:** Average as well as maximum of link probability of the link locations – (e.g. Hillary Clinton and Clinton)
- **Relatedness:** Topics which relate to the central thread of the document are more likely to be linked
- **Disambiguation Confidence:** The confidence score of the classifier for disambiguation
- **Generality:** Defined as the minimum depth at which it is located in Wikipedia's category tree. More useful for the readers to provide links for specific topics.
- **Location and Spread:** Where are these mentioned? First occurrence, last occurrence and the spread.

NPTEL

Question 5. Which of the following problem exists in bootstrapping technique for Information extraction are:

A) Sensitiveness towards the seed set

B) High precision

C) Less manual intervention

D) All of the above

Solution

Answer: A

- Target relation: burial place
- Seed tuple : [*Mark Twain, Elmira*]
- Google for “Mark Twain” and “Elmira”

“Mark Twain is buried in Elmira, NY.”
→ X is buried in Y
“The grave of Mark Twain is in Elmira”
→ The grave of X is in Y
“Elmira is Mark Twain’s final resting place”
→ Y is X’s final resting place

- Use those patterns to search for new tuples

NPTEL

Bootstrapping problems

- Requires that we have seeds for each relation
  - Sensitive to original set of seeds
- Generally have lots of parameters to be tuned
- No probabilistic interpretation
  - Hard to know how confident to be in each result

Question 6. Which of the following is an advantage of unsupervised relation extraction:

A) Can work efficiently with small amount of hand-labeled data

B) Not easily generalizable to different relations

C) Need no training data.

D) Always perform better than supervised techniques.

Solution

Answer: C

Question 7. Which of the following is not a Hearst's Lexico Syntactic Patterns for automatic acquisition of hyponyms -

A) X or other Y

B) X and other Y

C) Y including X

D) X but not Y

Solution

Answer: D

Automatic Acquisition of Hyponyms

- Y such as $X((,X)*(,$ and/or$)\ X)$
- such Y as X
- X or other Y
- X and other Y
- Y including X
- Y, especially X

NPTEL

Hearst pattern	Example occurrences
X and other Y	...temples, treasuries, and other important civic buildings.
X or other Y	bruises, wounds, broken bones or other injuries...
Y such as X	The bow lute, such as the Bambara ndang...
such Y as X	...such authors as Herrick, Goldsmith, and Shakespeare.
Y including X	...common-law countries, including Canada and England...
Y, especially X	European countries, especially France, England, and Spain...

NPTEL

Question 8. Advantage of Distant supervision over bootstrapping method

A) Need more data

B) Less human effort

C) Can handle noisy data better

D) No Advantage

Solution

Answer: C

Has advantages of supervised approach

- leverage rich, reliable hand-crafted knowledge
- relations have canonical names
- can use rich features (e.g. syntactic features)

Has advantages of unsupervised approach

- leverage unlimited amounts of text data
- allows for very large number of weak features
- not sensitive to training corpus: genre independent

NPTEL

Question 9. Consider a dataset with a very low number of relations - all of which are very important. For a relation extraction task on that dataset, which of the following is the most useful metric

A) Precision

B) Recall

C) Accuracy

D) F1-Score

Solution

Answer: B

$$P = \frac{\text{Number of correctly extracted relations}}{\text{Total number of extracted relations}}$$

$$R = \frac{\text{Number of correctly extracted relations}}{\text{Total number of gold relations}}$$

Question 10. What is KeyPhraseness (wikipedia)?

A) Number of articles that mention a key phrase divided by the number of wikipedia articles containing it.

B) Number of Wikipedia articles that use it as an anchor, divided by the number of articles that mention it at all.

C) Number of articles that mention a key phrase times by the number of wikipedia articles containing it.

D) Number of Wikipedia articles containing the key phrases times by number of articles mentioning it.

Solution

Answer: B

keyphraseness(w)

Number of Wikipedia articles that use it as an anchor, divided by the numer of articles that mention it at all.

Natural Language Processing

Week 11

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Question 1: Your teacher recommended you to read the book ’Natural Language Processing with Python’. After reading the book, you want to summarize it. What kind of summarization method would you use for this purpose?

1. Abstractive single document summarization

2. Abstractive multi document summarization

3. Extractive single document summarization

4. Extractive multi document summarization

a. 1, 2

b. 3, 4

c. 1, 3

d. 2, 4

Solution

Answer: C

Genres of Summary

- Extract vs. Abstract

  . . .lists fragments of text vs. re-phrases content coherently.

- Single document vs. Multi-document

  . . .based on one text vs. fuses together many texts.

- Generic vs. Query-focused

  . . .provides author's view vs. reflects user's interest.

NPTEL

Question 2: What kind of summarization approach is lexrank?

a. Extractive multi document generic

b. Extractive multi document query specific

c. Abstractive multi document query specific

d. Abstractive multi document generic

Solution

Answer: a

Question 3: Identify whether the following statements are True or False.

1. Maximum Marginal Relevance strives to reduce redundancy while maintaining query relevance.

2. Query-focused summarization can be thought of as a complex question-answering system.

a. True, False

b. True, True

c. False, True

d. False, False

Solution

Answer: b

Query-focused summarization can be thought of as a complex question answering system

Removing Redundant Sentences

Maximal Marginal Relevance

- An iterative method for content selection from a selected list of important sentences

For question 4-8, use the data given in Table 1.

Suppose you have trained a image classifier with 5 classes - cat, dog, lion, tiger, and deer. Consider the confusion matrix shown in Table 1.

Predicted Labels \ Gold Labels	cat	dog	lion	tiger	deer
cat	130	17	9	7	40
dog	15	150	25	10	7
lion	10	45	150	23	5
tiger	15	15	20	120	30
deer	40	30	20	10	155

Table 1

Question 4: What is the macro averaged precision?

a. 0.6696

b. 0.6078

c. 0.6433

d. None of the above

Solution

Answer: c

Solution: Separate confusion matrix for each class is as follows:

class	TP	FP
	FN	TN

cat	130	73
	80	815

dog	150	57
	107	784

lion	150	83
	74	791

tiger	120	80
	50	848

deer	155	100
	82	761

$$\text{recall} = \frac{TP}{TP+FN}$$

$$\text{precision} = \frac{TP}{TP+FP}$$

$$\text{macro averaged precision} = (0.6404 + 0.7246 + 0.6438 + 0.6 + 0.6078)/5 \approx 0.6433$$

Question 5: What is the macro averaged recall?

a. 0.6464

b. 0.6540

c. 0.6190

d. None of the above

Solution

Answer: a

Solution: Separate confusion matrix for each class is as follows:

class	TP	FP
	FN	TN

cat	130	73
	80	815

dog	150	57
	107	784

lion	150	83
	74	791

tiger	120	80
	50	848

deer	155	100
	82	761

$$\text{recall} = \frac{TP}{TP+FN}$$

$$\text{precision} = \frac{TP}{TP+FP}$$

$$\text{macro averaged recall} = (0.6190 + 0.5837 + 0.6696 + 0.7059 + 0.6540)/5 \approx 0.6464$$

Question 6: What is the accuracy of your classifier?

a. 0.6421

b. 0.6536

c. 0.6319

d. None of the above

Solution

Answer: a

Predicted Labels \ Gold Labels	cat	dog	lion	tiger	deer
cat	130	17	9	7	40
dog	15	150	25	10	7
lion	10	45	150	23	5
tiger	15	15	20	120	30
deer	40	30	20	10	155

$$\text{accuracy} = \frac{TP+TN}{TP+FP+TN+FN} = \frac{\text{number of correct predictions}}{\text{the total number of predictions}} = \frac{705}{1098} \approx 0.6421$$

Question 7: What is the micro averaged precision?

a. 0.6915

b. 0.6421

c. 0.6245

d. None of the above

Solution

Answer: b

For micro averaged results, create pooled confusion matrix from all the classes.

class	TP	FP
	FN	TN

cat	130	73
	80	815

dog	150	57
	107	784

lion	150	83
	74	791

tiger	120	80
	50	848

deer	155	100
	82	761

Micro Avg. Table

705	393
393	3999

Micro averaged precision = 705/(705+393) = 0.6421

Question 8: What is the micro averaged recall?

a. 0.6190

b. 0.6535

c. 0.6421

d. None of the above

Solution

Answer: c

For micro averaged results, create pooled confusion matrix from all the classes.

class	TP	FP
	FN	TN

cat	130	73
	80	815

dog	150	57
	107	784

lion	150	83
	74	791

tiger	120	80
	50	848

deer	155	100
	82	761

Micro Avg. Table

705	393
393	3999

Micro averaged recall= 705/(705+393) = 0.6421

Question 9: It is estimated that 20% of ChatGPT generated texts are fake. Some AI system has been applied to filter these fake contents. An AI system claims that it can detect 98% of fake contents, and the probability for a false positive (a real content detected as fake) is 3%. Now if a content is detected as fake, then what is the probability that it is in fact a real content?

a. 0.084

b. 0.109

c. 0.119

d. None of the above

Solution

Answer: b

Solution:

Let, A = Event that a content is detected as fake

B = Event that a generated text is fake

$P(B) = 0.2$

$P(B') = 0.8$

$P(A|B) = 0.98$

$P(A|B') = 0.03$

$$P(B'|A) = P(A|B')P(B')/P(A)$$

$$= P(A|B')P(B')/(P(A|B)P(B) + P(A|B')P(B'))$$

$$= (0.03 \times 0.8)/(0.98 \times 0.2 + 0.03 \times 0.8)$$

$$\approx 0.109$$

Question 10: Consider the system generated summary (S) and the reference summary as follows:

S: ChatGPT is powered by deep learning, a technique that involves training a neural network with extensive data.

R: ChatGPT is deep learning model that uses a neural network to understand language patterns.

What is the ROUGE-1 recall for the given summary with respect to the reference?

a. 0.500

b. 0.571

c. 0.470

d. None of the above

Solution

Answer: b

Solution: ROUGE-1 recall = Matching unigrams in the reference and system generated summary / No. of unigrams in the reference summary = 8/14 = 0.5714

S: ChatGPT is powered by deep learning, a technique that involves training a neural network with extensive data.

R: ChatGPT is deep learning model that uses a neural network to understand language patterns.

Natural Language Processing

Week 12

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Question 1: Which of the following are indicators of Irrealis moods?

1. Words enclosed in quotes.
2. Conditional markers such as “If”.
3. Questions.
4. All of the above

Solution

Answer:

4

Irrealis moods: where the words may not be reliable

- I thought this movie would be as good as the Grinch, but unfortunately, it wasn't.
- This should have been a great movie.

What are the indicators?

- conditional markers (if)
- negative polarity items like 'any' and 'anything'
- certain (mostly intensional) verbs (*expect, doubt*),
- questions
- words enclosed in quotes (which may be factual, but not necessarily reflective of the author's opinion)

NPTEL

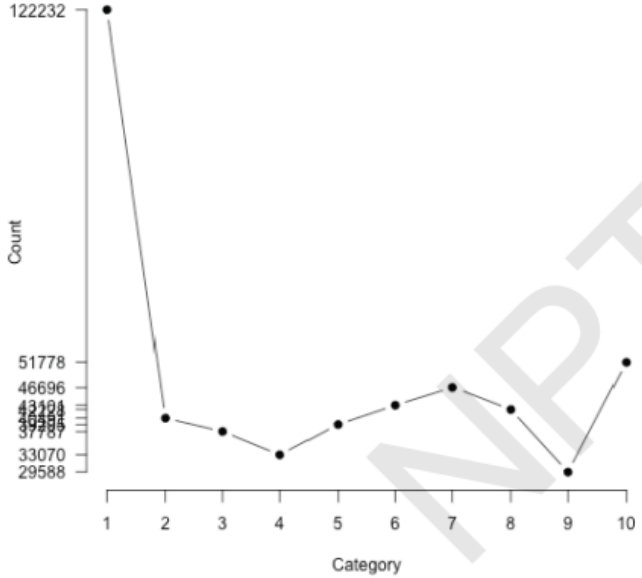
Question 2: Let P(w|c) represent the probability of a word given its rating. Further, let P(w) represent the probability of the word in the overall corpus. While analyzing the polarity of a word in a given corpus, what is the intuition behind dividing P(w|c) by P(w)?

1. To make it comparable across different ratings.
2. To make it comparable across different words.
3. Both (1) and (2).
4. None of the above.

Solution

Answer: 2

- How likely is each word to appear in each sentiment class?
- Let's take count("bad") in 1-star, 2-star, 3-star etc.



NPTEL

- We should use likelihood instead of counts: $P(w|c) = \frac{f(w,c)}{\sum_{w \in c} f(w,c)}$
- Make them comparable between words Scaled likelihood: $\frac{P(w|c)}{P(w)}$

Question 3: Which of the following lexicons are appropriate for valence?

1. Aroused, wide-awake
2. dominant, autonomous
3. happy, satisfied
4. stimulated, excited

Solution

Answer: 3

valence (the pleasantness of the stimulus

- 9: happy, pleased, satisfied, contented, hopeful
- 1: unhappy, annoyed, unsatisfied, melancholic, despaired, or bored

Question 4: Which of the following is/are false?

1. The words ‘bad’, ‘problem’ represents negative emotion
2. The words ‘love’, ‘sweet’ belong to the class of affective processes
3. The words ‘perhaps’, ‘guess’ don’t belong to cognitive processes
4. ‘relaxation’ is an example of low arousal, high pleasure word

LIWC (Linguistic Inquiry and Word Count)

- Home page: `http://www.liwc.net/`
- 2300 words, > 70 classes

Affective Processes

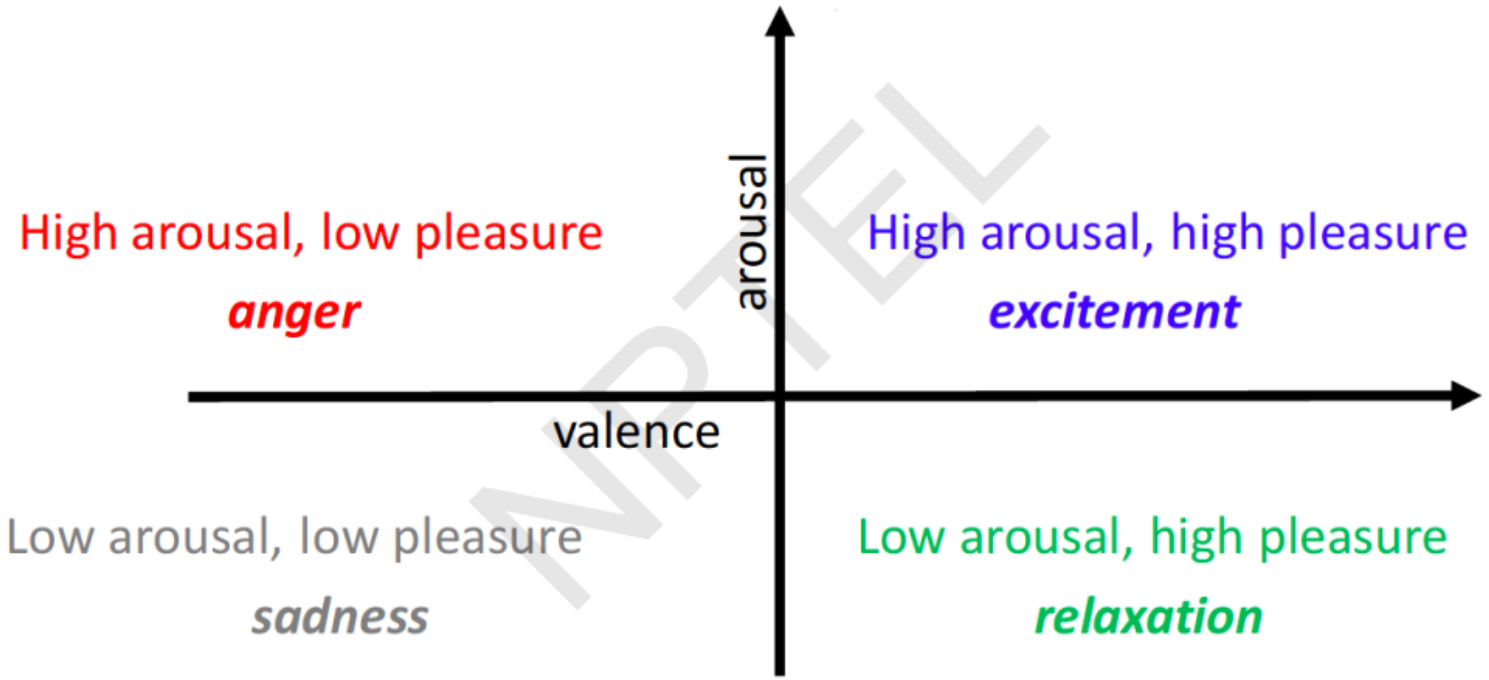
- Negative emotion (bad, weird, hate, problem, tough)
- Positive emotion (love, nice, sweet)

Cognitive Processes

Tentative (maybe, perhaps, guess), Inhibition (block, constraint)

NPTEL

Valence / Arousal Dimenstions



NPTEL

Solution

Answer: 3

Solution: ‘perhaps’, ‘guess’ they are under cognitive processes.

Question 5: Which of the following is/are correct about Turney Algorithm

1. It uses Pointwise Mutual Information measure

2. Jaccard Coefficient also an be used to measure the co-occurence

3. This is used for phrase polarity task

4. It learns polarity of only

a subset of phrase in the corpus

Solution

Answer: 1

Solution:

- Extract a *phrasal lexicon* from reviews
- Learn polarity of each phrase
- Rate a review by the average polarity of its phrases

Question 6: Sentiment lexicons can be learned using intuitions such as:

1. Same polarity words are connected by “and”
2. Opposite polarity words are connected by “but”
3. Both (1) and (2)
4. None of the above

Solution

Answer: 3

Solution:

Basic Intuition

- Adjectives conjoined by “and” have same polarity
  - *Fair and legitimate, corrupt and brutal*
- Adjectives conjoined by “but” do not
  - *fair but brutal*

Question 7: Which of the following are true?

1. Emotions are instinctive and usually short-lived, whereas sentiments are more stable and valid for a longer period of time.

2. Polarity shift technique is performed to handle normalization problems

3. Sentiment lexicons can be learned using intuitions such as same polarity words are connected by “and.

4. 1 & 2

Solution

Answer: 1, 3

Solution: Polarity shift technique is performed to handle negations and intensifiers.

Question 8: Consider the sentence: “The sound was cool; but, the network quality and screen were very dull”. Which of the following are true?

1. Aspect: “sound”, Sentiment: Positive, Opinion Phrase: “cool”.
2. Aspect: “screen”, Sentiment: Negative, Opinion Phrase: “very dull”.
3. Aspect: “were”, Sentiment: Negative, Opinion Phrase: “very dull”.
4. Only (1) and (3)

Solution

Answer: 1,2

Solution: ‘were’ is not a aspect term

Question 9: Consider the sentence: “The environment was great, however rent was very costly”. Which of the following is/are true?

1. Aspect: “environment”, Sentiment: Positive, Opinion Phrase: “great”.
2. Aspect: “rent”, Sentiment: Negative, Opinion Phrase: “very costly”.
3. Aspect: “price”, Sentiment: Negative, Opinion Phrase: “very costly”.
4. Only (a) and (b)

Solution

Answer: 1, 2, 3

Solution: “price” is an implicit aspect.

Natural Language Processing

Good evening, everyone. Welcome to the live session.

Today, we will be practicing problems from the current week’s content.

We will wait 5 minutes for everyone to join in and start at 7:05 pm.

Suppose you have trained a classifier for part-of-speech tagging. Consider the confusion matrix shown below.

		Gold labels			
		noun	**verb**	**adjective**	**adverb**
Predicted Labels	**noun**	90	8	12	15
	verb	15	100	5	25
	adjective	10	20	80	10
	adverb	10	7	30	80

Question 1: Calculate the following:

a. macro averaged precision

b. macro averaged recall

c. accuracy

d. micro averaged precision

e. micro averaged recall

Solution

Separate confusion matrix for each class is as follows:

class	TP	FP
	FN	TN

noun	90	35
	35	357

verb	100	45
	35	337

adjective	80	40
	47	350

adverb	80	47
	50	340

$$\text{recall} = \frac{TP}{TP+FN}$$

$$\text{precision} = \frac{TP}{TP+FP}$$

macro averaged precision = (90/125 + 100/145 + 80/120 + 80/127) /4 = 0.676

macro averaged recall = (90/125 + 100/135 + 80/127 + 80/130)/4 = 0.676

Solution

		Gold labels			
		noun	**verb**	**adjective**	**adverb**
Predicted Labels	**noun**	90	8	12	15
	verb	15	100	5	25
	adjective	10	20	80	10
	adverb	10	7	30	80

accuracy = 350/517 = 0.677

Solution

For micro averaged results, create pooled confusion matrix for all the classes.

class	TP	FP
	FN	TN

noun	90	35
	35	357

adjective	80	40
	47	350

Pooled table	350	167
	167	1384

verb	100	45
	35	337

adverb	80	47
	50	340

$$\text{recall} = \frac{TP}{TP+FN}$$

$$\text{precision} = \frac{TP}{TP+FP}$$

micro averaged precision = 350/517 = 0.677

micro averaged recall = 350/517 = 0.677

Zipf's Law

A relationship between the frequency of a word (f) and its position in the list (its rank r).

$$f \propto \frac{1}{r}$$

or, there is a constant k such that

$$f.r = k$$

More Formally: kth order Markov Model

Chain Rule:

$$P(w_1 w_2 \ldots w_n) = \prod_i P(w_i | w_1 w_2 \ldots w_{i-1})$$

Using Markov Assumption: only k previous words

$$P(w_1 w_2 \ldots w_n) \approx \prod_i P(w_i | w_{i-k} \ldots w_{i-1})$$

P(office | about fifteen minutes from)

An N-gram model uses only $N-1$ words of prior context.

- Unigram: P(office)
- Bigram: P(office | from)
- Trigram: P(office | minutes from)

Markov model and Language Model

An N-gram model is an $N-1$-order Markov Model

Question 2: Consider the CFG given below:

$S \rightarrow aSb|D$

$D \rightarrow Dc|\varepsilon$

How many non-terminals should be added to convert the CFG into CNF?

A) 3

B) 2

C) 4

D) 5

Solution

Answer: D

Solution: Given: S → aSb|D, D → Dc|ε

1. Since the start symbol S occurs on the right side of a production rule, we create a new start symbol S' and a new production rule S' → S.

   S' → S, S → aSb|D, D → Dc|ε

1. Remove the null productions: D → ε, S → ε
   a. After removing D → ε: S' → S, S → aSb|D|ε, D → Dc|c
   b. After removing S → ε: S' → S, S → aSb|D|ab, D → Dc|c
2. Remove the unit productions: S → D, S' → S
   a. After removing S → D: S' → S, S → aSb|ab|Dc|c, D → Dc|c
   b. After removing S' → S: S' → aSb|ab|Dc|c, S → aSb|ab|Dc|c, D → Dc|c

Solution

S’ → aSb|ab|Dc|c, S → aSb|ab|Dc|c, D → Dc|c

4. Since the right-side of the production rules contain both terminal and non-terminal, we add the following rules: A → a, B → b, C → c

S’ → ASB|AB|DC|c, S → ASB|AB|DC|c, D → DC|c, A → a, B → b, C → c

5. Handling the cases where there are more than two non-terminals on the right-hand side by adding the rule E → SB:

S′ → AE|AB|DC|c, S → AE|AB|DC|c, **E** → SB, D → DC|c, **A** → a, **B** → b, **C** → c

Two different kind of relationship among words

Inflectional morphology

Grammatical: number, tense, case, gender

Creates new forms of the same word: *bring, brought, brings, bringing*

Derivational morphology

Creates new words by changing part-of-speech: *logic, logical, illogical, illogicality, logician*

Fairly systematic but some derivations missing: *sincere - sincerity, scarce - scarcity, curious - curiosity, fierce - fiercity?*

NPTEL

$S \rightarrow$ NN VP	0.50	$S \rightarrow$ VP NN	0.50
$NP \rightarrow$ NN PB	0.40	$PB \rightarrow$ PP NN	0.30
$VP \rightarrow$ VB NN	0.30	$VP \rightarrow$ VB NP	0.20
$VP \rightarrow$ NN VB	0.25	$VP \rightarrow$ NN PB	0.15
$PP \rightarrow$ with	0.10	$PP \rightarrow$ without	0.10
$VB \rightarrow$ play	0.30	$VB \rightarrow$ enjoy	0.20
$VB \rightarrow$ watch	0.25	$NN \rightarrow$ children	0.15
$NN \rightarrow$ cricket	0.15	$NN \rightarrow$ friends	0.20
$NN \rightarrow$ football	0.10	$NN \rightarrow$ music	0.12

Question 3:

Using CKY algorithm, find the probability score for the most probable tree for the sentence S_1 = "**children play cricket with friends**". **[1 mark]**

A) 5.06×10^{-4}
B) 2.73×10^{-3}
C) 1.62×10^{-6}
D) None of the above

Solution

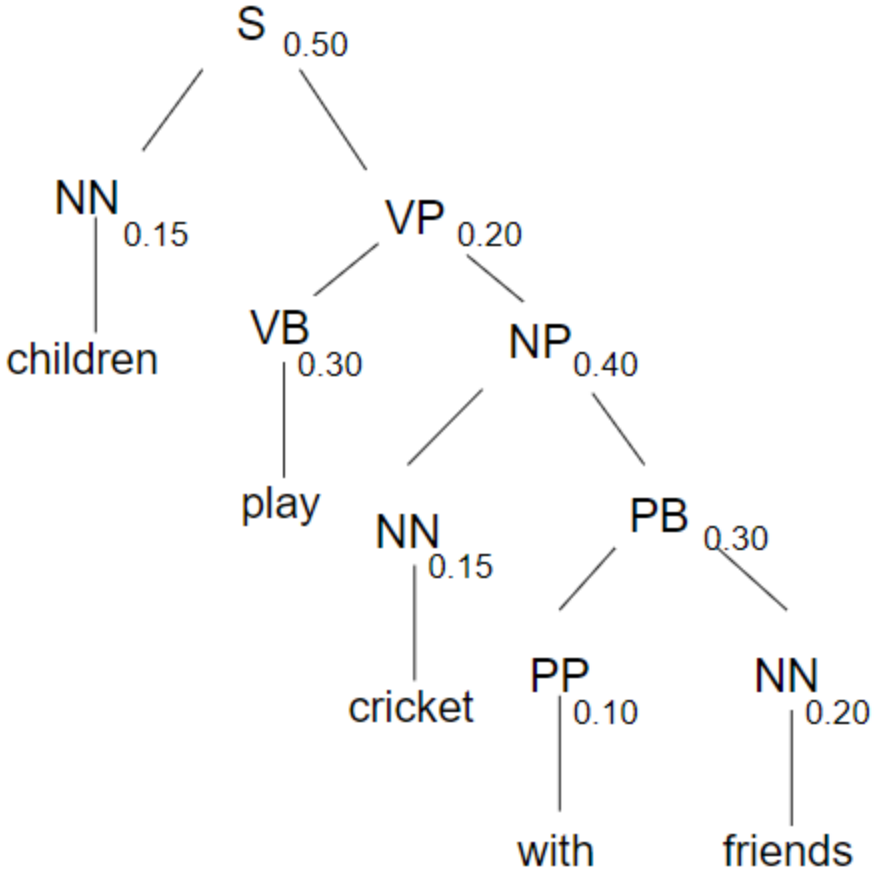
- Let n be the number of words in the input. Think about $n+1$ lines separating them, numbered 0 to n.
- x_{ij} will denote the words between line i and j
- We build a table so that x_{ij} contains all the possible non-terminal spanning for words between line i and j.
- We build the Table bottom-up.

Solution

Answer: C

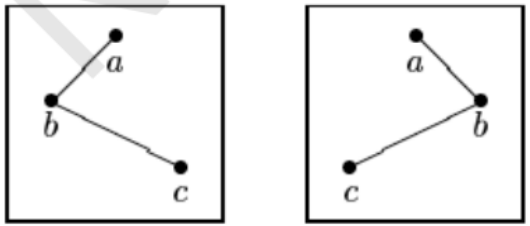
Solution:

0	1	2	3	4	5
	children	play	cricket	with	friends
	NN	VP	S	-	S
		VB	VP	-	VP
			NN	-	NP
				PP	PB
					NN



Formal conditions on Dependency Graphs

- G is connected:
  - For every node i there is a node j such that $i \rightarrow j$ or $j \rightarrow i$.
- G is acyclic:
  - if $i \rightarrow j$ then not $j \rightarrow^* i$.
- G obeys the single head constraint:
  - if $i \rightarrow j$ then not $k \rightarrow j$, for any $k \neq i$.
- G is projective:
  - if $i \rightarrow j$ then $j \rightarrow^* k$, for any k such that both j and k lie on the same side of i.



NPTEL

Formal conditions on Dependency Graphs

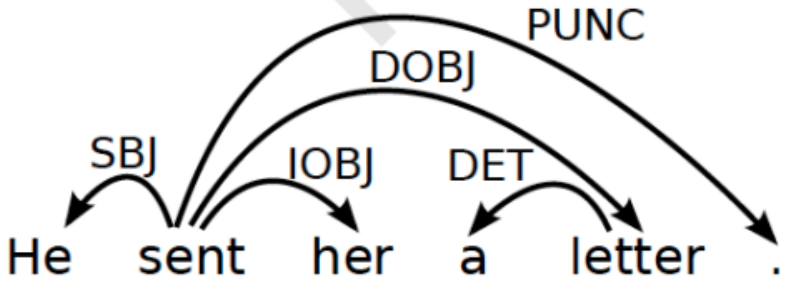
Connectedness, Acyclicity and Single-Head

- **Connectedness:** Syntactic structure is complete.
- **Acyclicity:** Syntactic structure is hierarchical.
- **Single-Head:** Every word has at most one syntactic head.
- **Projectivity:** No crossing of dependencies.

Transition-Based Parsing

Transitions: SH-LA-SH-RA-SH-LA-RE-RA-RE-RA

Stack	**Buffer**
$[\text{sent}, .]_S$	$[\,]_B$



Arcs

He $\xleftarrow{\text{SBJ}}$ sent

sent $\xrightarrow{\text{IOBJ}}$ her

a $\xleftarrow{\text{DET}}$ letter

sent $\xrightarrow{\text{DOBJ}}$ letter

sent $\xrightarrow{\text{PUNC}}$.

NPTEL

Assume that you are learning a classifier for the data-driven deterministic parsing and the sentence ‘I prefer ChatGPT course’ is a gold-standard parse in your training data. You are also given that ‘ChatGPT’ and ‘course’ are ‘Nouns’, ‘I’ is a ‘Pronoun’ while the POS tag of ‘prefer’ is ‘Verb’. Obtain the dependency graph for this sentence on your own. Assume that your features correspond to the following conditions:

1. The stack is empty.

2. Top of stack is Noun and Top of buffer is Verb.

3. Top of stack is Pronoun and Top of buffer is Verb.

4. The word at the top of stack occurs before word at the top of the buffer in the sentence

The initial weights of your features are [2,2,2,2 | 3,3,3,2| 2,2,2,2 | 2,2,2,2] where the first four features correspond to LA, and then to RA, SH and RE, respectively. Use this gold standard parse during online learning. What will be the weights after completing two iteration of Arc-Eager parsing over this sentence:

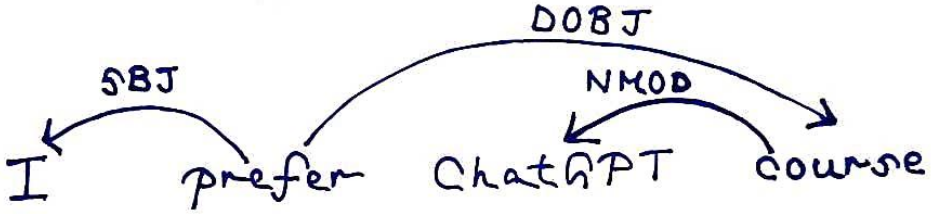
LEARN($\{T_1, \ldots, T_N\}$)

1 $w \leftarrow 0.0$
2 **for** i in $1..K$
3 **for** j in $1..N$
4 $c \leftarrow ([]_S, [w_1, \ldots, w_{n_j}]_B, \{\})$
5 **while** $B_c \neq []$
6 $t^* \leftarrow \arg\max_t w.f(c,t)$
7 $t_o \leftarrow o(c, T_i)$
8 **if** $t^* \neq t_o$
9 $w \leftarrow w + f(c,t_o) - f(c,t^*)$
10 $c \leftarrow t_o(c)$
11 **return** w

Oracle $o(c, T_i)$ returns the optimal transition of c and T_i

NPTEL

Solution



Given - The first four features correspond to LA, and then to RA, SH and RE, respectively.

f(c,t) = [c_1&&LA, c_2&&LA, c_3&&LA, c_4&&LA | c_1&&RA, c_2&&RA, c_3&&RA, c_4&&RA | c_1&&SH, c_2&&SH, c_3&&SH, c_4&&SH | c_1&&RE, c_2&&RE, c_3&&RE, c_4&&RE]

Initial weight: w = [2,2,2,2 | 3,3,3,2| 2,2,2,2 | 2,2,2,2]

Solution

Current configuration = $[]_S$, $[I,prefer,ChatGPT,course]_B$, {}

Conditions: 1. The stack is empty. 2. Top of stack is Noun and Top of buffer is Verb. 3. Top of stack is Pronoun and Top of buffer is Verb. 4. The word at the top of stack occurs before word at the top of the buffer in the sentence.

f(c,t) = [c_1&&LA, c_2&&LA, c_3&&LA, c_4&&LA | c_1&&RA, c_2&&RA, c_3&&RA, c_4&&RA | c_1&&SH, c_2&&SH, c_3&&SH, c_4&&SH | c_1&&RE, c_2&&RE, c_3&&RE, c_4&&RE]

f(c,LA) = [1, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0]

f(c,RA) = [0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0]

f(c,SH) = [0, 0, 0, 0 | 0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0]

f(c,RE) = [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 1, 0, 0, 0]

Solution

w = [2,2,2,2 | 3,3,3,2| 2,2,2,2 | 2,2,2,2]

w * f(c,LA) = w * [1, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0] = 2

w * f(c,RA) = w * [0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0] = 3

w * f(c,SH) = w * [0, 0, 0, 0 | 0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0] = 2

w * f(c,RE) = w * [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 1, 0, 0, 0] = 2

$t^* = RA$, $t_0 = SH$, $t^* \neq t_0$

$w_{new} = w_{old} + f(c, t_0) - f(c,t^*) = w_{old} + f(c, SH) - f(c,RA)$

= [2,2,2,2 | 3,3,3,2| 2,2,2,2 | 2,2,2,2] + [0, 0, 0, 0 | 0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0]
- [0, 0, 0, 0 | 1, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0] = [2,2,2,2 | 2,3,3,2| 3,2,2,2 | 2,2,2,2]

Solution

Current configuration = $[I]_S$, $[prefer,ChatGPT,course]_B$, {}

Conditions: 1. The stack is empty. 2. Top of stack is Noun and Top of buffer is Verb. 3. Top of stack is Pronoun and Top of buffer is Verb. 4. The word at the top of stack occurs before word at the top of the buffer in the sentence.

f(c,t) = [c_1&&LA, c_2&&LA, c_3&&LA, c_4&&LA | c_1&&RA, c_2&&RA, c_3&&RA, c_4&&RA | c_1&&SH, c_2&&SH, c_3&&SH, c_4&&SH | c_1&&RE, c_2&&RE, c_3&&RE, c_4&&RE]

f(c,LA) = [0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0]

f(c,RA) = [0, 0, 0, 0 | 0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0]

f(c,SH) = [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 1, 1 | 0, 0, 0, 0]

f(c,RE) = [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 1, 1]

Solution

w = [2,2,2,2 | 2,3,3,2| 3,2,2,2 | 2,2,2,2]

w * f(c,LA) = w * [0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0] = 4

w * f(c,RA) = w * [0, 0, 0, 0 | 0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0] = 5

w * f(c,SH) = w * [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 1, 1 | 0, 0, 0, 0] = 4

w * f(c,RE) = w * [0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 1, 1] = 4

$t^* = RA$, $t_0 = LA$, $t^* \neq t_0$ $w_{new} = w_{old} + f(c, t_0) - f(c,t^*) = w_{old} + f(c, LA) - f(c,RA)$

= [2,2,2,2 | 2,3,3,2| 3,2,2,2 | 2,2,2,2] + [0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0 | 0, 0, 0, 0]
- [0, 0, 0, 0 | 0, 0, 1, 1 | 0, 0, 0, 0 | 0, 0, 0, 0] = [2,2,3,3 | 2,3,2,1| 3,2,2,2 | 2,2,2,2]

Relations between word meanings

- Homonymy
- Polysemy
- Synonymy
- Antonymy
- Hypernymy
- Hyponymy
- Meronymy

Once a day (e.g. at noon), the weather is observed as one of state 1: rainy, state 2:cloudy, state 3: sunny. The state transition probabilities are :

0.4	0.3	0.3
0.2	0.6	0.2
0.1	0.1	0.8

Given that the weather on day 1 (t = 1) is sunny (state 3), what is the probability that the weather for the next 7 days will be “sun-sun-rain-rain-sun-cloudy-sun”?

a. 1.54 * 10-4

b. 8.9 * 10-2

c. 7.1 * 10-7

d. 2.5 * 10-10

Solution

State 1(S1) = **rainy,** State 2(S2) = **cloudy**, State 3(S3): **sunny**

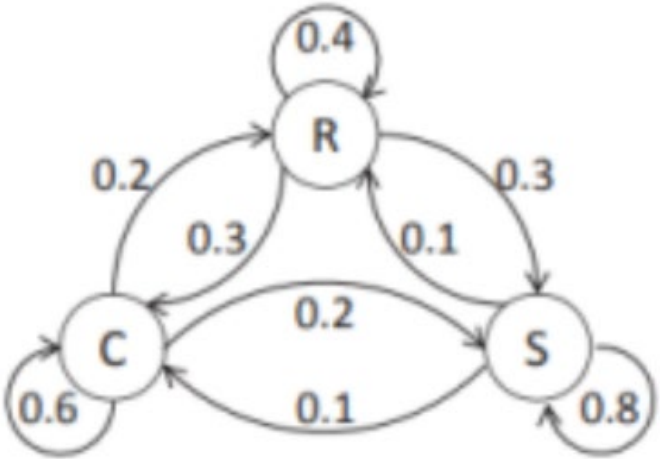
Given: Weather on day 1 (t = 1) is sunny (state 3).

Therefore, P(S3|start) = 1, P(S1|start) = 0, P(S2|start) = 0

O = {sun, sun, sun, rainy, rainy, sun, cloudy, sun}

O = {S3, S3, S3, S1, S1, S3, S2, S3}

	S1(R)	S2(C)	S3(S)
S1(R)	0.4	0.3	0.3
S2(C)	0.2	0.6	0.2
S3(S)	0.1	0.1	0.8



Solution

	S1(R)	S2(C)	S3(S)
S1(R)	0.4	0.3	0.3
S2(C)	0.2	0.6	0.2
S3(S)	0.1	0.1	0.8

O = {S3, S3, S3, S1, S1, S3, S2, S3}

P(O | Model)

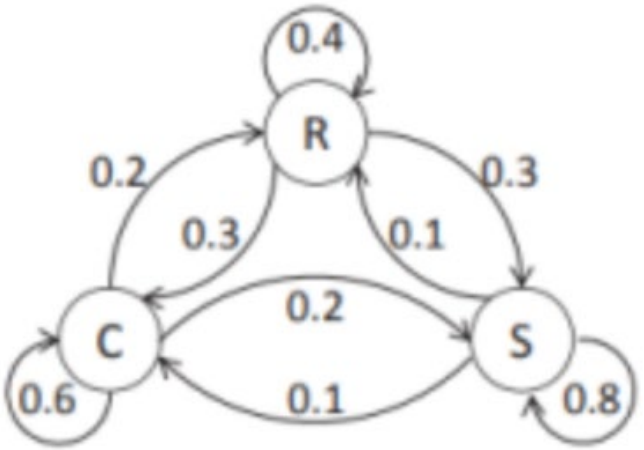
= P(S3, S3, S3, S1, S1, S3, S2, S3 | Model)

= P(S3|start) P(S3|S3) P(S3|S3) P(S1|S3) P(S1|S1) P(S3|S1) P(S2| S3) P(S3|S2)

= (1)(0.8)(0.8)(0.1)(0.4)(0.3)(0.1)(0.2)

= 1.536 × 10-4

Answer: A



Advanced smoothing algorithms

- Good-Turing
- Kneser-Ney