Master on Artificial Intelligence Universitat Politècnica de Catalunya

Advanced Human Language Technologies

Final Exam

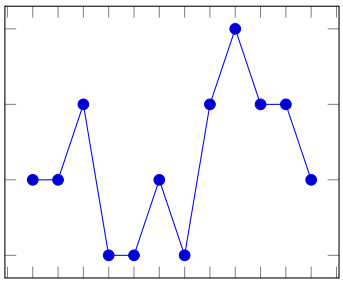
June 11^{th}, 2020

Exercise 1. Smoothing

We want to devise a method to predict stock exchange evolution using n-gram models.
For this, we model the stock value daily behaviour as a sequence of *movements*. Possible movements are in the set $M = \{=, +, -, \wedge, \vee\}$, where:

$=$	no change
$+$	small value increment
$-$	small value decrement
$\wedge$	big value increment
$\vee$	big value decrement

For instance, the sequence $= + \vee = + - \wedge + - = -$ encodes the following stock share behaviour:



We have the following historic data that we want to use as a training set to build a trigram model:

$$= + == ++ = - \wedge - \wedge + == - = + = \vee \vee + =$$

1. Compute the following probability values corresponding to a MLE trigram model, and those corresponding to a MLE model smoothed with Lidstone Law using $\lambda = 0.1$.

   - $P(+ =)$
   - $P(+ = \vee)$
   - $P(+ = +)$
   - $P(+ \mid + =)$
   - $P(\wedge \mid = -)$

   Justify the values chosen for B and N in each case. You can leave the probability value as a fraction. Do not provide just a final numeric result.

2. What is the most likely continuation of the sequence $\vee - = \vee == +$ according to each model? Justify your answer.

SOLUTION

1.

	MLE prob.	Smoothed prob.	Justification
$P(+ =)$	5/21	(5+0.1)/(21+25*0.1)	$N = 21$ observed bigrams; $B = \|M\|^2 = 5^2 = 25$ possible bigrams
$P(+ = \vee)$	1/20	(1+0.1)/(20+125*0.1)	$N = 20$ observed trigrams $B = \|M\|^3 = 5^3 = 125$ possible trigrams
$P(+ = +)$	0/20	(0+0.1)/(20+125*0.1)	$N = 20$ observed trigrams $B = \|M\|^3 = 5^3 = 125$ possible trigrams
$P(+ \mid + =)$	0/5	(0+0.1)/(5+5*0.1)	$N = 5$ occurrences of '$+ =$' $B = \|M\| = 5$ possible values after '$+ =$'
$P(\wedge \mid = -)$	1/2	(1+0.1)/(2+5*0.1)	$N = 2$ occurrences of '$= -$' $B = \|M\| = 5$ possible values after '$= -$'

2. Since it is a trigram model, the probability of the possible continuations of the given sequence is determined by its two last elements, i.e. $= +$.

   The continuation probabilities after $= +$ are:

	MLE prob.	Smoothed prob.
$P(= \mid = +)$	2/3	(2+0.1)/(3+5*0.1)
$P(+ \mid = +)$	1/3	(1+0.1)/(3+5*0.1)
$P(- \mid = +)$	0/3	(0+0.1)/(3+5*0.1)
$P(\wedge \mid = +)$	0/3	(0+0.1)/(3+5*0.1)
$P(\vee \mid = +)$	0/3	(0+0.1)/(3+5*0.1)

   Thus, the most likely continuation is '$=$' in both models.

Exercise 2. Features for log linear sequence annotation models

We are performing PoS tagging for a recently discovered alien language, using a trigram-factored CRF, using tagset $\mathcal{T} = \{\texttt{D}, \texttt{V}, \texttt{N}, \texttt{A}, \texttt{P}\}$, and we defined a history as $h = \langle t_{i-2}, t_{i-1}, w_{[1:n]}, i\rangle$.

1. How many possible histories are there for a given input sequence $\mathcal{X}$ and a fixed value of i? Justify your answer.

2. Which of the following are valid features and which are not? Justify your answer.

$$\mathbf{f}_1(h,t) = \begin{cases} 1 & \text{if } t = \texttt{V} \text{ and } t_{i-1} = \texttt{N} \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{f}_2(h,t) = \begin{cases} 1 & \text{if } t = \texttt{K} \text{ and } w_{i-2} = \texttt{skjkeg} \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{f}_3(h,t) = \begin{cases} 1 & \text{if } t = \texttt{N} \text{ and } t_{i-3} = \texttt{P} \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{f}_4(h,t) = \begin{cases} 1 & \text{if } t = \texttt{V} \text{ and } t_{i+1} = \texttt{A} \text{ and } w_2 = \texttt{wuakla} \\ 0 & \text{otherwise} \end{cases}$$

3. Compute the feature vectors $\mathbf{f}(h,t)$ for each position i, and the global feature vector $\mathbf{f}(\mathcal{X},\mathcal{Y})$ for the input sequence $\mathcal{X}$ = `grufp umdk wuakla du blha skjkeg` and the tag sequence $\mathcal{Y}$ = `P V N D N A`, when using the following features:

$$\mathbf{f}_1(h,t) = \begin{cases} 1 & \text{if } t = \texttt{N} \text{ and } w_i = \texttt{wuakla} \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{f}_2(h,t) = \begin{cases} 1 & \text{if } t = \texttt{N} \text{ and } t_{i-1} \neq \texttt{A} \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{f}_3(h,t) = \begin{cases} 1 & \text{if } t = \texttt{N} \text{ and } t_{i-1} = \texttt{V} \text{ and } w_{i-1} = \texttt{umdk} \\ 0 & \text{otherwise} \end{cases}$$

SOLUTION

1. For fixed $\mathcal{X}$ and i, the only varible elements of the history are t_{i-2} and t_{i-1}. Since each of them may have any value in $\mathcal{T}$, the number of different possible histories is $|\mathcal{T}|^2 = 5^2 = 25$

2. f_1 is valid, since it depends only on t and t_{i-1}, which is included in h.
   f_2 is not valid, since $\texttt{K} \notin \mathcal{T}$.
   f_3 is not valid, since t_{i-3} is not included in h.
   f_4 is not valid, since t_{i+1} is not included in h.

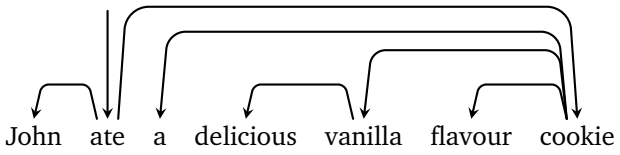
3. for $i = 1$, we have $h = \langle \texttt{START}, \texttt{START}, \mathcal{X}, 1\rangle$, and $\mathbf{f}(h,t) = (\mathbf{f}_1(h,\texttt{P}), \mathbf{f}_2(h,\texttt{P}), \mathbf{f}_3(h,\texttt{P})) = (0,0,0)$
   for $i = 2$, we have $h = \langle \texttt{START}, \texttt{P}, \mathcal{X}, 2\rangle$, and $\mathbf{f}(h,t) = (\mathbf{f}_1(h,\texttt{V}), \mathbf{f}_2(h,\texttt{V}), \mathbf{f}_3(h,\texttt{V})) = (0,0,0)$
   for $i = 3$, we have $h = \langle \texttt{P}, \texttt{V}, \mathcal{X}, 3\rangle$, and $\mathbf{f}(h,t) = (\mathbf{f}_1(h,\texttt{N}), \mathbf{f}_2(h,\texttt{N}), \mathbf{f}_3(h,\texttt{N})) = (1,1,1)$
   for $i = 4$, we have $h = \langle \texttt{V}, \texttt{N}, \mathcal{X}, 4\rangle$, and $\mathbf{f}(h,t) = (\mathbf{f}_1(h,\texttt{D}), \mathbf{f}_2(h,\texttt{D}), \mathbf{f}_3(h,\texttt{D})) = (0,0,0)$
   for $i = 5$, we have $h = \langle \texttt{N}, \texttt{D}, \mathcal{X}, 5\rangle$, and $\mathbf{f}(h,t) = (\mathbf{f}_1(h,\texttt{N}), \mathbf{f}_2(h,\texttt{N}), \mathbf{f}_3(h,\texttt{N})) = (0,1,0)$
   for $i = 6$, we have $h = \langle \texttt{D}, \texttt{N}, \mathcal{X}, 6\rangle$, and $\mathbf{f}(h,t) = (\mathbf{f}_1(h,\texttt{A}), \mathbf{f}_2(h,\texttt{A}), \mathbf{f}_3(h,\texttt{A})) = (0,0,0)$

   Thus the global feature vector is the sum of the factored vectors: $\mathbf{f}(\mathcal{X},\mathcal{Y}) = (1,2,1)$

Exercise 3. Parsing

Given the sentence *John ate a delicious vanilla flavour cookie*,

1. Draw unlabeled dependency trees for the following interpretations
   (a) John ate a cookie with flavour of delicious vanilla
   (b) John ate a delicious cookie with vanilla flavour
   (c) John ate a delicious and flavoured cookie made of vanilla
   (d) John ate a cookie with a delicious flavour of vanilla
2. Given the tree

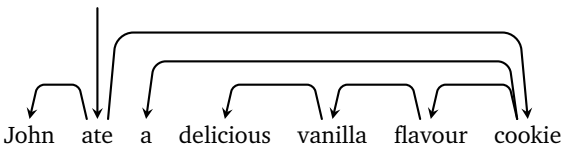


   (a) Explain the interpretation encoded by this tree avoiding any ambiguities.
   (b) Emulate the behaviour that would result in this tree for a transition dependency parser using an arc-standard model (i.e. with operations *shift*, *left-arc*, and *right-arc* between the two topmost stack elements). List the intermediate stack/buffer contents and the required action at each step to obtain the final tree.

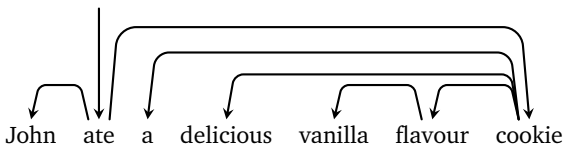
SOLUTION

1.

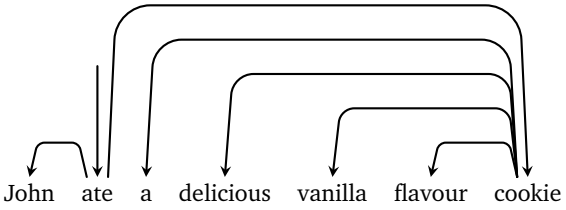
(a)



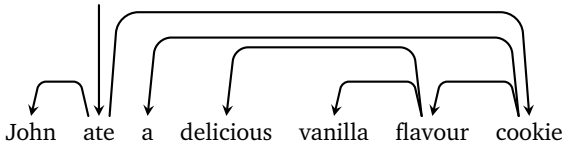
(b)



(c)



(d)



2.

   (a) The tree represents the interpretation where John ate a flavoured cookie made of delicious vanilla.
   (b) The behaviour of an arc-based transition parser to obtain this interpretation would be the following:

Stack	Buffer	Transition	Edges
*	John ate a delicious vanilla flavour cookie	sh	{}
* John	ate a delicious vanilla flavour cookie	sh	{}
* John ate	a delicious vanilla flavour cookie	l-arc	{(2,1)}
* ate	a delicious vanilla flavour cookie	sh	{(2,1)}
* ate a	delicious vanilla flavour cookie	sh	{(2,1)}
*ate a delicious	vanilla flavour cookie	sh	{(2,1)}
* ate a delicious vanilla	flavour cookie	l-arc	{(2,1),(5,4)}
* ate a vanilla	flavour cookie	sh	{(2,1),(5,4)}
* ate a vanilla flavour	cookie	sh	{(2,1),(5,4)}
* ate a vanilla flavour cookie		l-arc	{(2,1),(5,4),(7,6)}
* ate a vanilla cookie		l-arc	{(2,1),(5,4),(7,6),(7,5)}
* ate a cookie		l-arc	{(2,1),(5,4),(7,6),(7,5),(7,3)}
* ate cookie		r-arc	{(2,1),(5,4),(7,6),(7,5),(7,3),(2,7)}
* ate		r-arc	{(2,1),(5,4),(7,6),(7,5),(7,3),(2,7),(0,2)}
*		stop	{(2,1),(5,4),(7,6),(7,5),(7,3),(2,7),(0,2)}

Exercise 4. Vectorial word representations

Given the following term×document matrix:

Term/Doc	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7	Doc8	Doc9	Doc10
car	3	0	0	5	12	0	0	2	8	1
auto	8	6	0	12	0	0	9	1	3	10
best	0	1	7	0	1	5	12	0	2	0

1. Compute the one-hot vector for each of the three words in the vocabulary
2. Compute the TF·IDF score for each word/document.

   NOTE: Remember to normalize the matrix by the maximum value of each row: $\max(\text{car}) = \max(\text{auto}) = \max(\text{best}) = 12$

SOLUTION

1. Since the vocabulary has 3 words, the vector space has dimension 3. One-hot vectors are:

   car $[1, 0, 0]$
   auto $[0, 1, 0]$
   best $[0, 0, 1]$

2. TF·IDF score for each word/document:

   Maximum number of occurrences for each word in a document is:
   $max(\text{car}) = max(\text{auto}) = max(\text{best}) = 12$

   Thus, normalized term frequencies are:

Term/Doc	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7	Doc8	Doc9	Doc10
car	0.25	0	0	0.42	1	0	0	0.17	0.67	0.08
auto	0.67	0.5	0	1	0	0	0.75	0.08	0.25	0.84
best	0	0.08	0.58	0	0.08	0.42	1	0	0.17	0

   And inverse document frequencies:

   Word *car* occurs in 6 out of 10 documents: $df(\text{car}) = 6/10 \longrightarrow idf(\text{car}) = \log_2(10/6) = 0.74$
   Word *auto* occurs in 7 out of 10 documents: $df(\text{auto}) = 7/10 \longrightarrow idf(\text{auto}) = \log_2(10/7) = 0.51$
   Word *best* occurs in 6 out of 10 documents: $df(\text{best}) = 6/10 \longrightarrow idf(\text{best}) = \log_2(10/6) = 0.74$

   (Using a $\log$ base other than 2 is also correct)

   Final TF·IDF scores result of multiplying mormalized term frequencies by their corresponding inverse document frequency, that is:

Term/Doc	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7	Doc8	Doc9	Doc10
car	$0.25 \times 0.74 = 0.19$	0	0	$0.42 \times 0.74 = 0.31$	$1 \times 0.74 = 0.74$	0	0	$0.17 \times 0.74 = 0.13$	$0.67 \times 0.74 = 0.5$	$0.08 \times 0.74 = 0.06$
auto	$0.67 \times 0.51 = 0.34$	$0.5 \times 0.51 = 0.36$	0	$1 \times 0.51 = 0.51$	0	0	$0.75 \times 0.51 = 0.38$	$0.08 \times 0.51 = 0.04$	$0.25 \times 0.51 = 0.13$	$0.84 \times 0.51 = 0.43$
best	0	$0.08 \times 0.74 = 0.06$	$0.58 \times 0.74 = 0.43$	0	$0.08 \times 0.74 = 0.06$	$0.42 \times 0.74 = 0.31$	$1 \times 0.74 = 0.74$	0	$0.17 \times 0.74 = 0.13$	0

Exercise 5. Recurrent Neural Networks

RNNs can be used in a variety of tasks. Consider the following three tasks:

1. Named-Entity Recognition: For each word in a sentence, classify that word as either a person, organization, location, or none. Input: A sentence containing n words.
2. Sentiment Analysis: Classify the sentiment of a sentence ranging from negative to positive (integer values from 0 to 4). Input: A sentence containing n words.
3. Language models: generate text for a chatbot that was trained to speak like you by predicting the next word in the sequence. Input: A single start word or token that is fed into the first time step of the RNN.

For each of the above tasks, describe how could you run a RNN, specifying:

a) How many outputs the RNN produces (i.e. number of times the softmax $\hat{y}(t)$ is called). If the number of outputs is not fixed, state it as arbitrary.

b) What is the set of classes included in the $\hat{y}(t)$ probability distribution.

c) Which input is fed to the RNN and what outputs are produced at each time step.

SOLUTION

1. Named Entity Recognition
   (a) Number of Outputs: n outputs, one per input word at each time step.
   (b) Each $\hat{y}(t)$ is a probability distribution over 4 label values (PER, LOC, ORG, none).
   (c) Each word in the sentence is fed into the RNN and one output is produced at every time step corresponding to the predicted tag/category for each word.
2. Sentiment Analysis
   (a) Number of Outputs: 1 output.
   n outputs is also an valid answer if it is explained that all outputs are averaged.
   (b) Each $\hat{y}(t)$ is a probability distribution over 5 sentiment values (0 to 4)
   (c) Each word in the sentence is fed into the RNN and one output is produced from the hidden states (by either taking only the final, max, or mean across all states) corresponding to the sentiment value of the sentence.
3. Language Models
   (a) Number of Outputs: arbitrary (as many as times the model is called)
   (b) Each $\hat{y}(t)$ is a probability distribution over the vocabulary
   (c) The first input can be a special `<START>` token, and the first output would be the first word of the sentence. Then, the previous output at each step is fed as input for the next time step and produces the next output corresponding to the next predicted word of the generated sentence.

Exercise 6. Convolutional Neural Networks

Consider a neural network that receives as input the following sequence of 2-dimensional embedded vectors:

she	0.2	0.1
told	0.5	0.2
me	-0.1	-0.3
to	0.3	-0.3
remain	0.2	-0.3
in	0.1	0.2
silence	-0.4	-0.4

The first layer of the network consists of a convolution with a single kernel of size 3, with padding=0, stride=1, dilation=2, and the following values:

3	1
-1	2
1	1

Compute the output of the convolution. You do not need to actually perform the numerical computations, just write the additions and multiplications you would need to compute at each position of the output (e.g. $3 \times 0.8 + 2 \times 0.1 + \dots$). Specifiy the kernel position corresponding to each value.

Exercise 7. SOLUTION

position	words	computation	result
1	she, me, remain	$0.2 \times 3 + 0.1 \times 1 + (-0.1) \times (-1) + (-0.3) \times +0.2 \times 1 + (-0.3) \times 1$	0.1
2	told, to, in	$0.5 \times 3 + 0.2 \times 1 + 0.3 \times (-1) + (-0.3) \times 2 + 0.1 \times 1 + 0.2 \times 1$	1.1
3	me, remain, silence	$(-0.1) \times 3 + (-0.3) \times 1 + 0.2 \times (-1) + (-0.3) \times 2 + (-0.4) \times 1 + (-0.4) \times 1$	-2.2