11-751 Speech Recognition and Understanding

Nikolas Wolfe
Homework 2
October 16th, 2013

Problem 1: Dynamic Programming

Write a program that reads the reference string and the hypothesis string from two files, and calculates the WER and the corresponding alignment between the two strings. You can use any programming language.

Answer: The code will be submitted separately with this homework, but pasting the essential components here…

This is written in JavaScript using JQuery. The HTML markup contains two text input elements containing the reference and hypothesis strings to be evaluated, and a button (with id='calc') to initiate the alignment problem, called by the following function:

`$('#calc').click( …code here… )`

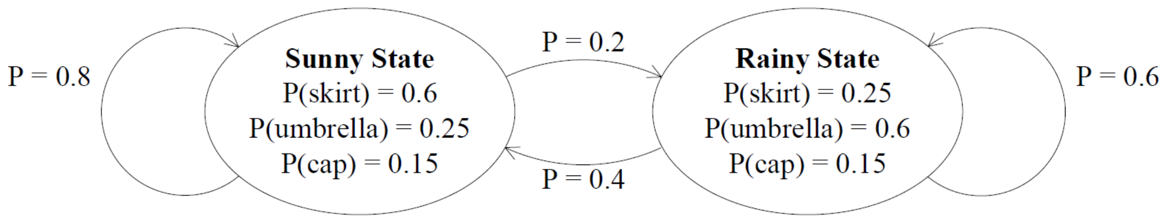
Then the distance calculation is done by a call to function `dist_matrix(ref, hyp) { … }`

```
<script>
$(document).ready(function() {

        // some constants
        var MATCH = "MATCH";
        var DEL = "DEL";
        var INS = "INS";
        var SUB = "SUB";
        var dtw, ref, hyp;

        // hide distance matrix button
        $('#matrix').hide();

        // When the word error rate button is clicked
        $('#calc').click( function() {

                // trim all whitespace and split by word boundaries
                ref = $.trim($('#ref').val().replace(/\s+/g, " ")).split(" ");
                hyp = $.trim($('#hyp').val().replace(/\s+/g, " ")).split(" ");
                dtw = dist_matrix(ref, hyp);

                var d = dtw[dtw.length - 1][dtw[0].length - 1];
                var wer = ((d[0] / ref.length) * 100).toFixed(0);

                $('#wer').html( "WER: " + wer + "%");
                $('#align').html( generate_alignment( dtw, ref, hyp ) );
                $('#matrix').show();
        });

        … code here …

        // calculate the distance matrix
        function dist_matrix(ref, hyp) {

                // initialize the distance array
                var dtw = new Array( ref.length + 1 );
                $.each(dtw, function( i ) {
                        dtw[i] = new Array( hyp.length + 1 );
                        $.each(dtw[i], function( j ) {
                                dtw[i][j] = [0,0,0,""];
                        });
                });

                // initialize edge weights in the array
```

```
			var i, j;
			$.each(dtw, function( i ) {
				dtw[i][0][0] = i;
			});
			$.each(dtw[0], function( j ) {
				dtw[0][j][0] = j;
			});

			/* calculate the distance matrix
			 *
			 * NOTE: this draws inspiration from the Wikipedia article on
			 * Levenshtein Distance. here: http://en.wikipedia.org/wiki/Levenshtein_distance
			 */
			var m, n;
			m = dtw.length;
			n = dtw[0].length;
			for( i = 1; i < m; i++ ) {
				for( j = 1; j < n; j++ ) {

					var r, h;
					r = ref[i - 1].toLowerCase();
					h = hyp[j - 1].toLowerCase();

					if( r == h ) { // match

						// take diagonal distance, save back pointer
						dtw[i][j] = [dtw[i - 1][j - 1][0],(i - 1),(j - 1),"MATCH"];

					} else { // mismatch

						var del, ins, sub, min;
						del = dtw[i - 1][j][0] + 1;		// deletion
						ins = dtw[i][j - 1][0] + 1;		// insertion
						sub = dtw[i - 1][j - 1][0] + 1;	// substitution

						// find the minimum distance, save the back-pointers
						min = Math.min( del, ins, sub );
						if( min == del ) {
							dtw[i][j] = [min,(i - 1),(j),DEL];
						}
						else if( min == ins ) {
							dtw[i][j] = [min,(i),(j - 1),INS];
						}
						else {
							dtw[i][j] = [min,(i - 1),(j - 1),SUB];
						}

					}
				}
			}
			return dtw;
		}

		… code here …

	</script>
```

Problem 2: Application of Hidden Markov Models

For Day 1, we know that $P(Q_1 = \text{Sunny}) = 1$ and $P(Q1 = \text{Rainy}) = 0$.



Let O_t be Jessie's appearance on Day t. For example, we can infer that $P(O_1 = \text{umbrella}) = 0.25$.

Questions:

1. What is $P(Q_2 = \text{Rainy})$?

   Because $Q_1 = \text{Sunny}$, we know that the transition probability of switching to Rainy state is P=0.2, so the likelihood that Q_2 is Rainy is $P(Q_2\text{=Rainy} \mid Q_1\text{=Sunny})$, which is 0.2.

2. What is $P(O_2 = \text{Skirt})$?

   In both Rainy and Sunny state we have probabilities for skirts, so assuming we are in Sunny state in Q_1, we have the transition probability of remaining in Sunny state, which is P=0.8, and the probability of observing 'skirt' given a Sunny state is 0.6. This means we have $P(Q_2 = \text{Sunny}$ *and* $O_2 = \text{Skirt})$, which is $P(Q_2\text{=Sunny}) \times P(O_2\text{=Skirt}|Q_2\text{=Sunny})$, which is 0.8 * 0.6, which is 0.48. If we switch to Rainy state, we have a transition probability of 0.2, and a skirt observation probability of 0.25, which is $P(Q_2\text{=Rainy}) \times P(O_2\text{=Skirt}|O_2\text{=Rainy})$, which is 0.2 * 0.25, which is 0.05. So, the overall likelihood of $P(O_2\text{=Skirt})$ is $P(Q_2 = \text{Sunny}$ *and* $O_2 = \text{Skirt})$ *or* $P(Q_2 = \text{Rainy}$ *and* $O_2 = \text{Skirt})$, which is 0.48 + 0.05, which is 0.53.

3. What is $P(Q_2 = \text{Rainy} \mid O_2 = \text{Skirt})$?

   From Question 1, the probability of $P(Q_2 = \text{Rainy}) = 0.2$. The probability of skirt is based on the overall probability of observing skirt in O_2, which we determined in Question 2 = 0.53. So, $P(Q_2 = \text{Rainy} \mid O_2 = \text{Skirt})$ is $P(O_2\text{=Skirt}$ *and* $Q_1\text{=Rainy})/P(Q_2\text{=Skirt})$, which is (0.53 * 0.2)/0.53 = 0.2.

4. What is $P(O_{100} = \text{cap})$?

   All trials are dependent only on the previous trial, so the 100^{th} state probability of 'cap' is equivalent to the overall likelihood of cap, which is the probability of observing cap in O_{100} given that Q_{99} was Sunny, *or* the probability of observing cap in O_{100} given that Q_{99} was Rainy. If Q_{99} was Rainy, then we either transitioned into Sunny in Q_{100} or we remained Rainy in $Q1_{00}$. Likewise if Q_{99} was Sunny, then we either transitioned into Rainy in Q_{100} or we remained in Sunny in Q_{100}.

   First assume Q_{99} was Sunny, i.e. $P(Q_{99} = \text{Sunny}) = 1$.

   a.) $P(Q_{100} = \text{Sunny}$ *and* $O_{100} = \text{Cap}) = P(Q_{100}\text{=Sunny}) \times P(O_{100}\text{=Cap}|Q_{100}\text{=Sunny}) = 0.8 * 0.15 = 0.12$
   b.) $P(Q_{100} = \text{Rainy}$ *and* $O_{100} = \text{Cap}) = P(Q_{100} = \text{Rainy}) \times P(O_{100} = \text{Cap}|Q_{100} = \text{Rainy}) = 0.2 * 0.15 = 0.03$

   c.) $P(O_{100} = \text{Cap} \mid Q_{99} = \text{Sunny})$ = a.) *or* b.) = 0.12 + 0.03 = 0.15

Second, assume Q_{99} = Rainy, i.e. P(Q_{99} = Rainy) = 1.

d.) P(Q_{100} = Sunny *and* O_{100} = Cap) = P(Q_{100} = Sunny) x P(O_{100} = Cap | Q_{100} = Sunny) = 0.4 * 0.15 = 0.06
e.) P(Q_{100} = Rainy *and* O_{100} = Cap) = P(Q_{100} = Rainy) x P(O_{100} = Cap | Q_{100} = Rainy) = 0.6 * 0.15 = 0.09

f.) P(O_{100} = Cap | Q_{99} = Sunny) = d.) *or* e.) = 0.06 + 0.09 = 0.15

Overall, P(O_{100} = Cap) = c.) *or* f.) = 0.15 + 0.15 = 0.30

And, this makes intuitive sense, because regardless of what state we are in, the probability of Cap is always 0.15, so P(Cap|Sunny) or P(Cap|Rainy) = 0.15 + 0.15 = 0.30

5. Let Y_t = P(Q_t = Sunny). For example, Y_1 = 1. Y_{t+1} can be defined inductively from Y_t by an expression Y_{t+1} = a + BY_t. Find the value of *a* and *b*.

Y_t = P(Q_t = Sunny), so Y_1 = 1 => P(Q_1 = Sunny) = 1,
Y_{t+1} = a + BY_t => a + B(P(Q_t = Sunny))

By induction,
$Y_{t+1} = a + BY_t$
$Y_{t+2} = a + BY_{t+1}$
$Y_{t+3} = a + BY_{t+2}$
…
$Y_{t+n} = a + BY_{t+n-1}$

$Y_1 = 1$
$Y_2 = a + B(1)$
$Y_3 = a + B(a + B)$
$Y_4 = a + B(a + B(a + B))$
$Y_5 = a + B(a + B(a + B(a + B)))$
…
$Y_{t+1} = a + B(Y_t)$

And…

We know that Y_1 = 1 = Sunny, which means the probability of remaining in that state in Y_2 is 0.8.

So Y_2 = a + B(1) = P(Q_2 = Sunny) = 0.8
Then Y_3 = a + B(Y_2), where Y_2 = a + B = 0.8
Which is Y_3 = a + B(0.8)

But Y_3 is the probability of remaining in Sunny (0.8) given that Y_2 is Sunny OR the probability that in Y_2 the state was Rainy (0.2 from Y_1) and then switched *back* to Sunny in Y_3, a probability of 0.4. So the probability of P(Q_3 = Sunny |Q_2 = Rainy) = 0.4, which of course has an 0.2 chance of happening *given* that Q_1 = Sunny. So in Y_3 the probability of getting Sunny is the likelihood of the sequence:

Q_1 Q_2 Q_3 = SSS, which is (1 * 0.8 * 0.8 = 0.64)
Or the sequence Q_1 Q_2 Q_3 = SRS, which is (1 * 0.2 * 0.4 = 0.08),
So Y_3 = P(SSS) + P(SRS) = 0.64 + 0.08 = 0.72

Y_2 = a + B = 0.8
Y_3 = a + B(0.8) = 0.72

Two equations, two unknowns...

a = 0.4
B = 0.4

Proof by Induction:

$\mathbf{Y_{t+1} = 0.4 + 0.4Y_t}$

Y_1 = **1**

State Transition Probability Chains…

Q_1 = S = **1**

$Y_2 = 0.4 + 0.4(1) = \mathbf{0.8}$

$Y_3 = 0.4 + 0.4(0.8) = \mathbf{0.72}$

$Y_4 = 0.4 + 0.4(0.72) = \mathbf{0.688}$

$Q_1Q_2 = SS = 1 * 0.8 = \mathbf{0.8}$

$Q_1Q_2Q_3 = SSS = 1 * 0.8 * 0.8 = 0.64$
$Q_1Q_2Q_3 = SRS = 1 * 0.2 * 0.4 = 0.08$

$0.08 + 0.64 = \mathbf{0.72}$

$Q_1Q_2Q_3Q_4 = SSSS = 1 * 0.8 * 0.8 * 0.8 = 0.512$
$Q_1Q_2Q_3Q_4 = SRSS = 1 * 0.2 * 0.4 * 0.8 = 0.064$
$Q_1Q_2Q_3Q_4 = SSRS = 1 * 0.8 * 0.2 * 0.4 = 0.064$
$Q_1Q_2Q_3Q_4 = SRRS = 1 * 0.2 * 0.6 * 0.4 = 0.048$

$0.512 + 0.064 + 0.064 + 0.048 = \mathbf{0.688}$

Etc. and so on… So,

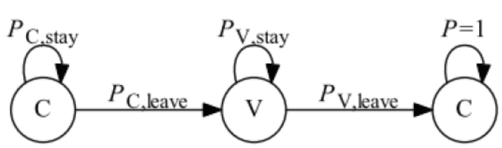
$\mathbf{Y_{t+1} = 0.4 + 0.4Y_t}$

6. Assume that $O_1 = O_2 = O_3 = O_4 = O_5$ = umbrella. What is the most probable sequence of states? (Hint: This can be solved with the Viterbi algorithm, but it would involve a lot of calculations. Try to answer the question with your intuition, and find a way to justify it.)

   Intuition tells me that the most likely state to emit an umbrella in Q_1 is Rainy state, with a probability of 0.6, compared with Sunny state, which has a probability of 0.25. Also, the likelihood of remaining in Rainy state is 0.6 combined with an emission probability of 0.6 for umbrella, so the likelihood of Q_2 emitting an umbrella as well is 0.6*0.6 = 0.36. If the system were to transition to Sunny, probability would be 0.4*0.25 = 0.1. Furthermore, even if we were in Sunny state to begin with, the probability of remaining in Sunny state and seeing an umbrella is 0.8 * 0.25 = 0.2. Any transition to Sunny state involving the viewing of umbrella is going to have a smaller probability of seeing umbrella again than remaining continuously in Rainy state, ergo, intuition tells me that the highest probability of viewing $O_1 = O_2 = O_3 = O_4 = O_5$ = umbrella is for $Q_1 = Q_2 = Q_3 = Q_4 = Q_5$ = Rainy, that is, a succession of Rainy states.

Problem 3: The Three Basic Problems of HMMs

Given the following HMM:



Question 1: The Evaluation Problem

Given the following alpha trellis for Observation {3, 8, 7, 2},

STATE 3 (C)	0.15	0.05	0.07	0.17
STATE 2 (V)	0.07	0.17	0.15	0.05

STATE 1 (C)	0.15	0.05	0.07	0.17
	3	8	7	2

The probability of state (Q_t) transition sequences, where Q_1 = State 1 (C), Q_2 = State 2 (V), and Q_3 = State 3 (C), is as follows, (Note: the sequence cannot go backwards, it can only remain in its current state or go forward, and all transitions are equally likely (0.5)):

Q Sequence	Probability
$Q_1 Q_1 Q_1 Q_1$	1.11563E-05
$Q_1 Q_1 Q_1 Q_2$	3.28125E-06
$Q_1 Q_1 Q_2 Q_2$	7.03125E-06
$Q_1 Q_2 Q_2 Q_2$	2.39063E-05
$Q_1 Q_2 Q_2 Q_3$	8.12813E-05
$Q_1 Q_2 Q_3 Q_3$	3.79313E-05
Sum:	0.000164588

The probability of the observation sequence is: **0.000164588**.

Question 2: The Decoding Problem

Using the same HMM observation sequence as the previous question, use the Viterbi algorithm to find the most probably state sequence given the HMM and observation.

The delta (psi) trellis is as follows:

STATE 3 (C)	0.15 (-)	0.05 (-)	0.00044625 (2)	**8.12813E-05 (2)**
STATE 2 (V)	0.07 (-)	**0.01275 (1)**	**0.00095625 (2)**	2.39063E-05 (1)
STATE 1 (C)	**0.15 (-)**	0.00375 (1)	0.00013125 (1)	1.11563E-05 (1)
	3	8	7	2
	MAX:	0.01275	0.00095625	8.12813E-05

Looking at the maximum delta values in each column the table, we see that there is an optimal state sequence, which is:

$$Q_1 = (1),\ Q_2 = (2),\ Q_3 = (2),\ Q_4 = (3)$$

Question 3: The Learning Problem

Given the same HMM observation sequence as the previous question, the forward-backward algorithm maximizes the likelihood of the observation give the model parameters. Run a manual iteration of the forward-backward algorithm to update the transition and emission probabilities.

The beta trellis:

STATE 3 (C)	0.000074375	0.002975	0.085	1
STATE 2 (V)	0.0010285	0.011225	0.11	1
STATE 1 (C)	0.001256625	0.0121	0.11	1
	3	8	7	2

The gamma trellis:

STATE 3 (C)	0.041069226	0.055879038	0.1973466	0.435897436
STATE 2 (V)	0.265033408	0.716848234	0.547263682	0.128205128
STATE 1 (C)	0.693897366	0.227272727	0.255389718	0.435897436
OBSERVATION	3	8	7	2
SUM:	*1*	*1*	*1*	*1*

The ksi trellis:

STATE 3 > 3	0.041	0.056	0.197
STATE 2 > 3	0.019	0.190	0.423
STATE 2 > 2	0.246	0.527	0.124
STATE 1 > 2	0.527	0.155	0.058
STATE 1 > 1	0.167	0.072	0.197
OBSERVATION	3	8	7
SUM	*1*	*1*	*1*

Since both State 1 and State 3 are the state C, the parameters of the state C should be computed by combining the gamma and ksi values of both State 1 and State 3. Write down the updated parameters:

Transition Probabilities:

$P_{C,stay} =$ 0.49698545246
$P_{V,stay} =$ 0.586671109
$P_{C,leave} =$ 0.628834414
$P_{V,leave} =$ 0.413328891

Adjusted Emission probabilities:

Observation	1	2	3	4	5	6	7	8	9	SUM
P(x|C)	0	0.37	0.31	0	0	0	0.19	0.12	0	1
P(x|V)	0	0.08	0.16	0	0	0	0.33	0.43	0	1

Question 4: Bayesian Decision

Best phoneme sequence = argmax_i P(phoneme sequence | observation)
= argmax_i P(observation | phoneme sequence_i) * P(phoneme sequence_i)

The language model P(phoneme sequence_i), is assumed to be a constant for every phoneme sequence, so this term can be ignored. Assume the acoustic model contains a phoneme sequence consisting of a single C. The HMM of the second phoneme sequence will have only one state C, with a self-loop probability of 1. Given the observation sequence {3, 8, 7, 2}, which of the two phoneme sequences will be recognized by the Bayesian decision rule? Why?

Answer:

HMM 1 = 3 State, CVC = **0.000164588**
HMM 2 = 1 State, C = 0.15 * 0.05 * 0.07 * 0.17 = **0.00008925**

The probability of phoneme sequence **CVC** is greater than the probability of phoneme sequence **C**, so the Bayesian decision rule will accept the **CVC** sequence.