# T-61.5020 Statistical Natural Language Processing

Answers 10 — Speech recognition and language model evaluation
Version 1.1

1. Again, we will use the Viterbi algorithm to find the most probable state sequence from a Hidden Markov Model. There are three differences to the weather model presented in the earlier exercise: The emissions are now done in the state transitions, the model has some null transitions, and the final state is determined.

   a) Let's initialize the grid such that the initial state is $S_1$. We will collect only non-zero probability values.

$$\delta_0(1) = 1$$

**The first observation**

The initial state can lead only the second or fourth state, so let's calculate those probabilities:

$$\begin{aligned}
\delta_1(2) &= a_{12}b_{12}(o_1) = 0.5 \cdot 10^{-1} = 5 \cdot 10^{-2} \\
\psi_1(2) &= 1 \\
\delta_1(4) &= a_{14}b_{14}(o_1) = 0.5 \cdot 10^{-3} = 5 \cdot 10^{-4} \\
\psi_1(4) &= 1
\end{aligned}$$

**The second observation**

From the second state we can go to the third state, and from the fourth state to the fifth state, so there is no choices to be made for those steps.

$$\begin{aligned}
\delta_2(3) &= \delta_1(2)a_{23}b_{23}(o_2) = 5 \cdot 10^{-2} \cdot 1.0 \cdot 10^{-1} = 5 \cdot 10^{-3} \\
\psi_2(3) &= 2 \\
\delta_2(5) &= \delta_1(4)a_{45}b_{45}(o_2) = 5 \cdot 10^{-4} \cdot 1.0 \cdot 10^{-4} = 5 \cdot 10^{-8} \\
\psi_2(5) &= 4
\end{aligned}$$

However, we should notice that the states $S_3$ and $S_5$ can lead to the initial state with a null transition. Thus after the second observation we can go also to $S_1$:

$$\begin{aligned}
\delta_2(1) &= \max(\delta_2(3)a_{31}\,,\ \delta_2(5)a_{51}) \\
&= \max(5 \cdot 10^{-3} \cdot 0.9\,,\ 5 \cdot 10^{-8} \cdot 1.0) \\
&= 4.5 \cdot 10^{-3} \\
\psi_2(1) &= 3
\end{aligned}$$

**The third observation**

Now the possible transitions are from $S_1$ to $S_2$ or $S_4$, and from $S_3$ to $S_4$.

$$\begin{aligned}
\delta_3(2) &= \delta_2(1)a_{12}b_{12}(o_3) = 4.5 \cdot 10^{-3} \cdot 0.5 \cdot 10^{-3} = 2.25 \cdot 10^{-6} \\
\psi_3(2) &= 1 \\
\delta_3(4) &= \max(\delta_2(1)a_{14}b_{14}(o_3)\,,\,\delta_2(3)a_{34}b_{34}(o_3)) \\
&= \max(4.5 \cdot 10^{-3} \cdot 0.5 \cdot 10^{-2}\,,\,5 \cdot 10^{-3} \cdot 0.1 \cdot 10^{-1}) \\
&= 5 \cdot 10^{-5} \\
\psi_3(4) &= 3
\end{aligned}$$

**The fourth observation**

Again, from $S_2$ we can go only to $S_3$ and from $S_4$ to $S_5$.

$$\begin{aligned}
\delta_4(3) &= \delta_3(2)a_{23}b_{23}(o_4) = 2.25 \cdot 10^{-6} \cdot 1.0 \cdot 10^{-3} = 2.25 \cdot 10^{-9} \\
\psi_4(3) &= 2 \\
\delta_4(5) &= \delta_3(4)a_{45}b_{45}(o_4) = 5 \cdot 10^{-5} \cdot 1.0 \cdot 10^{-1} = 5 \cdot 10^{-6} \\
\psi_4(5) &= 4
\end{aligned}$$

**Final state**

In the end we should arrive to the final state $S_1$. With a null transition:

$$\begin{aligned}
\delta_4(1) &= \max(\delta_4(3)a_{31}\,,\,\delta_4(5)a_{51}) \\
&= \max(2.25 \cdot 10^{-9} \cdot 0.9\,,\,5 \cdot 10^{-6} \cdot 1.0) \\
&= 5 \cdot 10^{-6} \\
\psi_4(1) &= 5
\end{aligned}$$

The calculated grid is in the Figure 1. By following the arrows from the end to the beginning, we obtain the most probable sequence $S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4 \rightarrow S_5 \rightarrow S_1$. This corresponds to the word “jaon”.

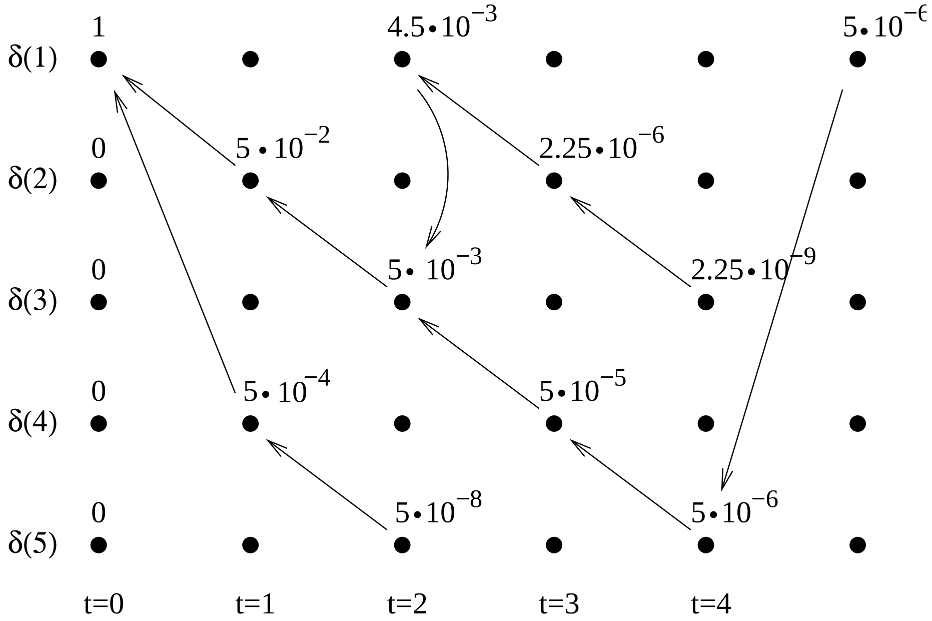

Figure 1: The Viterbi grid after the calculations.

b) In this case we must take into account the probabilities given by the language model. The probability values $\delta$ are calculated conditioned by the different choice of the word $w_j$: $\delta_t(i, w_j)$. The probability for the word is added to the calculations at each point where the word is selected. When we arrive to the initial state again, the selections determine which of the bigram probabilities is used. After that, they can be forgotten, as the language model does not use longer contexts.

Let's initialize the grid as before. We do not select the word yet.

$$\delta_0(1, \_) = 1$$

**The first observation**

The initial state leads to $S_2$ and $S_4$. The second state can start either the word "ja" or "jaon", so both must be taken into account.

$$\begin{aligned}
\delta_1(2, \text{ja}) &= P(\text{ja})a_{12}b_{12}(o_1) = 10^{-2} \cdot 0.5 \cdot 10^{-1} = 5 \cdot 10^{-4} \\
\psi_1(2, \text{ja}) &= 1 \\
\delta_1(2, \text{jaon}) &= P(\text{jaon})a_{12}b_{12}(o_1) = 10^{-5} \cdot 0.5 \cdot 10^{-1} = 5 \cdot 10^{-7} \\
\psi_1(2, \text{jaon}) &= 1 \\
\delta_1(4, \text{on}) &= P(\text{on})a_{14}b_{14}(o_1) = 10^{-2} \cdot 0.5 \cdot 10^{-3} = 5 \cdot 10^{-6} \\
\psi_1(4, \text{on}) &= 1
\end{aligned}$$

**The second observation**

The second state leads only to the third state and the fourth state to the fifth state. In addition, the first state can be reached with a null transition. This is of course possible only for the words that end at this point.

$$\begin{aligned}
\delta_2(3, \text{ja}) &= \delta_1(2, \text{ja})a_{23}b_{23}(o_2) = 5 \cdot 10^{-4} \cdot 1.0 \cdot 10^{-1} = 5 \cdot 10^{-5} \\
\psi_2(3, \text{ja}) &= 2 \\
\delta_2(3, \text{jaon}) &= \delta_1(2, \text{jaon})a_{23}b_{23}(o_2) = 5 \cdot 10^{-7} \cdot 1.0 \cdot 10^{-1} = 5 \cdot 10^{-8} \\
\psi_2(3, \text{jaon}) &= 2 \\
\delta_2(5, \text{on}) &= \delta_1(4, \text{on})a_{45}b_{45}(o_2) = 5 \cdot 10^{-6} \cdot 1.0 \cdot 10^{-4} = 5 \cdot 10^{-10} \\
\psi_2(5, \text{on}) &= 4
\end{aligned}$$

$$\begin{aligned}
\delta_2(1, \text{ja}) &= \delta_2(3, \text{ja})a_{31} = 5 \cdot 10^{-5} \cdot 0.9 = 4.5 \cdot 10^{-5} \\
\psi_2(1, \text{ja}) &= 3 \\
\delta_2(1, \text{on}) &= \delta_2(5, \text{on})a_{51} = 5 \cdot 10^{-10} \cdot 1.0 = 5 \cdot 10^{-10} \\
\psi_2(1, \text{on}) &= 5
\end{aligned}$$

**The third observation**

Possible transitions are from $S_1$ to $S_2$ or $S_4$, and from $S_3$ to $S_4$. The transitions from $S_1$ start new words, so the probabilities from the language model are taken into account. In addition, as we had two possible words in state $S_1$, we can now select the more probable one.

$$\begin{aligned}
\delta_3(2,\mathrm{ja}) &= \max(P(\mathrm{ja}|\mathrm{ja})\delta_2(1,\mathrm{ja})\,,\ P(\mathrm{ja}|\mathrm{on})\delta_2(1,\mathrm{on}))\cdot a_{12}b_{12}(o_3)\\
&= \max(10^{-4}\cdot 4.5\cdot 10^{-5}\,,\ 10^{-2}\cdot 5\cdot 10^{-10})\cdot 0.5\cdot 10^{-1}\\
&= 2.25\cdot 10^{-10}\\
\psi_3(2,\mathrm{ja}) &= 1\\
\delta_3(4,\mathrm{on}) &= \max(P(\mathrm{on}|\mathrm{ja})\delta_2(1,\mathrm{ja})\,,\ P(\mathrm{on}|\mathrm{on})\delta_2(1,\mathrm{on}))\cdot a_{14}b_{14}(o_3)\\
&= \max(10^{-2}\cdot 4.5\cdot 10^{-5}\,,\ 10^{-4}\cdot 5\cdot 10^{-10})\cdot 0.5\cdot 10^{-2}\\
&= 2.25\cdot 10^{-9}\\
\psi_3(4,\mathrm{on}) &= 1\\
\delta_3(4,\mathrm{jaon}) &= \delta_2(3,\mathrm{jaon})a_{34}b_{34}(o_3)\\
&= 5\cdot 10^{-8}\cdot 1.0\cdot 10^{-1} = 5\cdot 10^{-10}\\
\psi_3(4,\mathrm{jaon}) &= 3
\end{aligned}$$

**The fourth observation**

From the second state we can only to the third state, and from the fourth state only to the fifth state. Also the first state can be reached with a null transition.

$$\begin{aligned}
\delta_4(3,\mathrm{ja}) &= \delta_3(2,\mathrm{ja})a_{23}b_{23}(o_4) = 2.25\cdot 10^{-10}\cdot 1.0\cdot 10^{-3} = 2.25\cdot 10^{-13}\\
\psi_4(3,\mathrm{ja}) &= 2\\
\delta_4(5,\mathrm{on}) &= \delta_3(4,\mathrm{on})a_{45}b_{45}(o_4) = 2.25\cdot 10^{-9}\cdot 1.0\cdot 10^{-1} = 2.25\cdot 10^{-10}\\
\psi_4(5,\mathrm{on}) &= 4\\
\delta_4(5,\mathrm{jaon}) &= \delta_3(4,\mathrm{jaon})a_{45}b_{45}(o_4) = 5\cdot 10^{-10}\cdot 1.0\cdot 10^{-1} = 5\cdot 10^{-11}\\
\psi_4(5,\mathrm{jaon}) &= 4
\end{aligned}$$

$$\begin{aligned}
\delta_4(1,\mathrm{ja}) &= \delta_4(3,\mathrm{ja})a_{31} = 0.9\cdot 2.25\cdot 10^{-10} = 2.025\cdot 10^{-13}\\
\psi_4(1,\mathrm{ja}) &= 3\\
\delta_4(1,\mathrm{on}) &= \delta_4(5,\mathrm{on})a_{51} = 1.0\cdot 2.25\cdot 10^{-9} = 2.25\cdot 10^{-10}\\
\psi_4(1,\mathrm{on}) &= 5\\
\delta_4(1,\mathrm{jaon}) &= \delta_4(5,\mathrm{jaon})a_{51} = 1.0\cdot 5\cdot 10^{-10} = 5\cdot 10^{-11}\\
\psi_4(1,\mathrm{jaon}) &= 5
\end{aligned}$$

The grid after the final step is in Figure 2. The different word choices are drawn with different arrows. The most probable of the three paths that have led to the final state is $\delta_4(1,\mathrm{on})$. When we follow the arrows backwards in time, we get the most probable sequence $S_1 \to S_2 \to S_3 \to S_1 \to S_4 \to S_5 \to S_1$. This corresponds to the two-word sequence “ja on”.

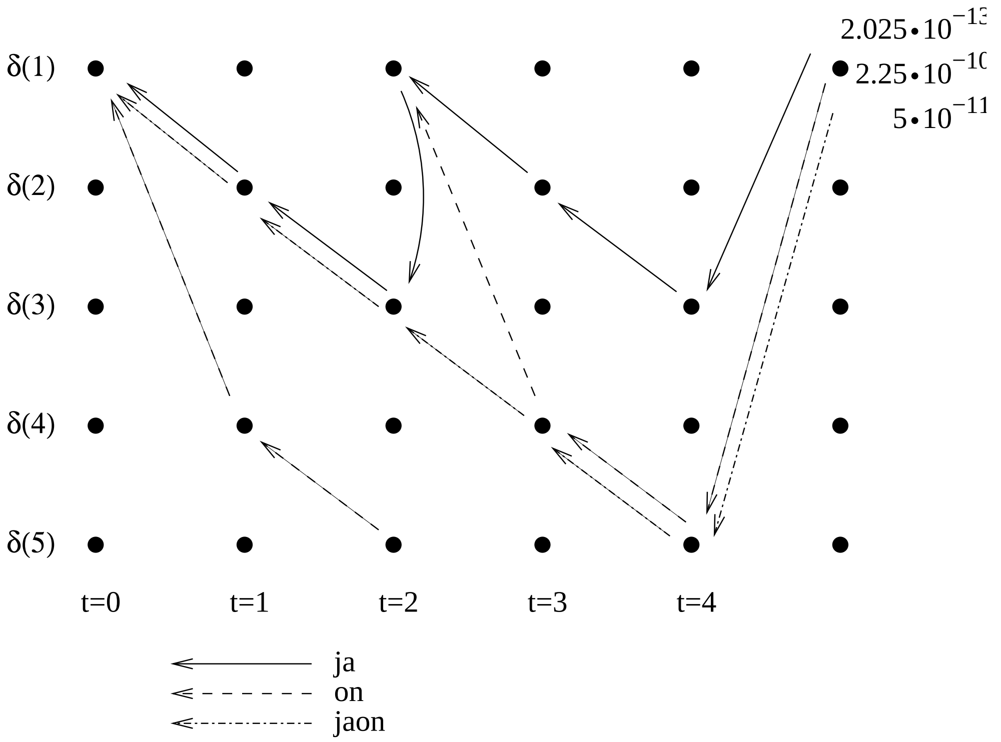

Figure 2: The grid after reaching the final state.

2. The models build with the units from segmentation B have about three times as many unit types as models build from segmentation A. The tokens in A are smaller on average, and thus the evaluation data includes more of them. The tokenwise cross-entropies cannot be compared directly because of this. For example, if the text was segmented to individual letters, the tokens would be quite easy to predict on average, but the likelihood of the whole data is not likely to be very high.

   Instead of direct comparison, we can first normalize the entropies so that they are based on words. The cross-entropy of test data $D$ could be calculated as

   $$H_M(D) = \frac{1}{n}\sum_{i=1}^{n} \log P_M(D_i) = \frac{1}{n} \log P_M(D). \tag{1}$$

   If we divide the logarithm of data likelihood $P_M(D)$ by the number of words in the data, $W_D$, instead of the number of tokens in the data, $n$, we get the normalized, word-based entropy:

   $$H_M^W(D) = \frac{1}{W_D} \log P_M(D). \tag{2}$$

   As we know $H_M(D)$, $n$ and $W_D$, we can calculate the normalized entropy as follows:

   $$H_M^W(D) = \frac{n}{W_D} H_M(D). \tag{3}$$

   Let's convert the given entropies to word-based estimates:

   $$\begin{aligned}
H_{A1}^W(D) &= \frac{344\,960}{100\,000} \cdot 4.54 = 15.66 \\
H_{A2}^W(D) &= \frac{344\,960}{100\,000} \cdot 4.39 = 15.14 \\
H_{A3}^W(D) &= \frac{344\,960}{100\,000} \cdot 4.31 = 14.87 \\
H_{B1}^W(D) &= \frac{301\,271}{100\,000} \cdot 5.19 = 15.64 \\
H_{B2}^W(D) &= \frac{301\,271}{100\,000} \cdot 5.02 = 15.12 \\
H_{B3}^W(D) &= \frac{301\,271}{100\,000} \cdot 4.93 = 14.85
\end{aligned}$$

   It seems that the entropies with the segmentation B are somewhat better in models of all magnitudes. However, as the differences are small, and models B have larger models, the exact sizes must be taken into account. The comparison is easy if we draw plot the results to size–entropy coordinates; see Figure 3.

   The break-line that connects the points of the segmentation A is nearer to the left-down corner that the lines of connecting B, which means better accuracies for the models of same size.

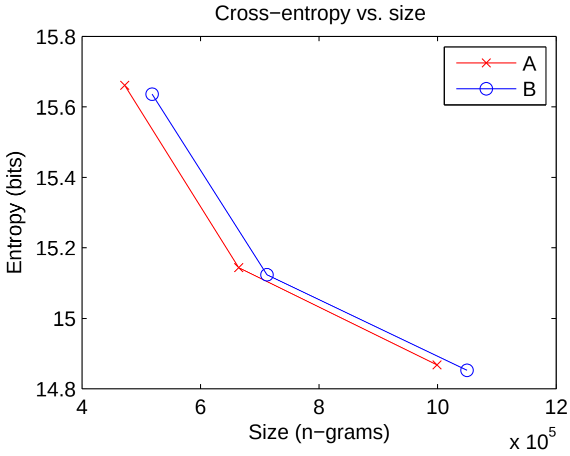

Figure 3: Normalized cross-entropies.

Next we will take a look at the recognition results. The error rates have been calculated per words, so there is no need for normalization. The word error rates (WER) are plotted against model sizes in Figure 4. We see that the results are mixed for the small and large models: Segmentation A works better for the small models, but B seems to outperform it after the size grows over 900 000 n-grams.

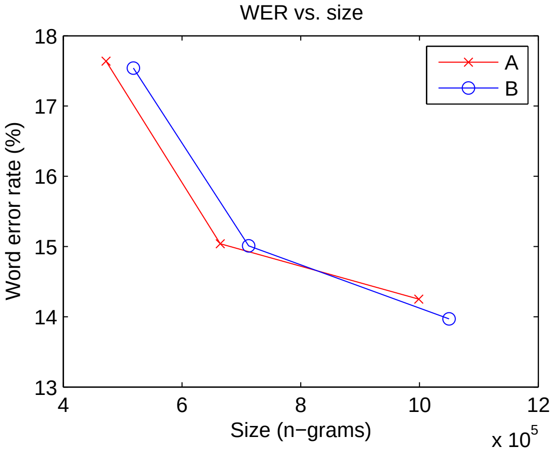

Figure 4: Word error rates.

It seems to be quite clear that the models based on segmentation A are better than those based on B, if the model size is small. For larger models, the results are very close. In addition, the performance is not known for models smaller than half million or larger than one million n-grams. To get more reliable results, we would need more measurement points and test the statistical significance between the values (e.g. with Wilcoxon signed-rank test).