Statistical structured prediction

Question set (Part 1-B)

Victoria Beltrán Domínguez

January 2022

Contents

1 Theoretical questions

1.1 Question 1 (2 point)

Analyze the behavior of the Inside-Outside estimation algorithm when the initial probabilities associated to the rules are equiprobable. Justify your comments with the example in page 50 from Part I.4 Model Parameter Estimation (P-I.4 from now on).

When the initial probabilities associated to the rules are equiprobable, we find that all the trees following the same structure will have the same probability.

Let us take the example from Part I.4 of slides. If all the rules were equiprobable, the probabilities of the rules would be:

```
1.0 S -> Suj Pre
0.33 Suj -> Art Nom
0.33 Suj -> Art Adj Nom
0.33 Suj -> Art Nom Adj
0.5 Pre -> Verb Nom
0.5 Pre -> Verb
```

```
1.0 Art -> "la"
0.2 Nom -> "vieja"
0.2 Nom -> "ayuda"
0.2 Nom -> "mujer"
0.2 Nom -> "pelea"
0.2 Nom -> "demanda"
0.25 Verb -> "demanda"
0.25 Verb -> "ayuda"
0.25 Verb -> "oculta"
0.25 Verb -> "pelea"
0.5 Adj -> "vieja"
0.5 Adj -> "oculta"
```

In this case, we notice that for example the probability of ”la mujer oculta pelea” and ”la vieja demanda ayuda” is the same (0.0005775). That is because both phrases have similar structures:

- ”La mujer oculta pelea” has two possible structures:
  1. Art + Nom + Verb + Nom
  2. **Art + Nom + Adj** + Verb
- ”La vieja demanda ayuda” also has two possible structures:
  1. Art + Nom + Verb + Nom
  2. **Art + Adj + Nom** + Verb

As we see, even though the order has changed, probabilities are the same because both elements (Art+Adj+Nom and Art+Nom+Adj) come from Suj rules and therefore have the same probability.

1.2 Question 2 (2 points)

Compute the estimation of the rule (Suj → Art Nom) with the example in page 50 from P-I.4 with the Inside-Outside algorithm with the following training sample that includes bracketed samples: D = {(la vieja)(demanda ayuda), la mujer oculta pelea, la vieja ayuda}.

The probabilities of the phrases are:

- P_θ(”(la vieja) (demanda ayuda)”) = 0.00090
- P_θ(”la mujer oculta pelea”) = 0.00090 + 0.01176 = 0.01266
- P_θ(”la vieja ayuda”) = 0.00700

Now, we can compute $\bar{p}$(Suj → Art Nom) as:

$$\bar{p}(\text{Suj} \rightarrow \text{Art Nom}) = \frac{\sum_{x \in \mathcal{D}} \frac{1}{P_\theta(x)} \sum_{t_x} \text{N}(\text{Suj} \rightarrow \text{ Art Nom}, t_x) P_\theta(x, t_x)}{\sum_{x \in \mathcal{D}} \frac{1}{P_\theta(x)} \sum_{t_x} \text{N}(\text{Suj}, t_x) P_\theta(x, t_x)} =$$

$$\frac{\frac{1}{0.0009} \cdot 0.0009 + \frac{1}{0.01266} \cdot (0.0009 + 0.01176) + \frac{1}{0.007} \cdot 0.007}{3} = 1$$

1.3 Question 3 (1 point)

Repeat the previous exercise but using the Viterbi-Score estimation algorithm.

Using Viterbi algorithm instead of the Inside-Outside, only the best derivation tree is considered for each sample. Taking that into account, we can compute $\bar{p}(Suj \rightarrow \text{Art Nom})$ as:

$$\bar{p}(\text{Suj} \rightarrow \text{Art Nom}) = \frac{\sum_{x \in \mathcal{D}} \frac{1}{P_\theta(x)} \sum_{t_x} \text{N}(\text{Suj} \rightarrow \text{ Art Nom}, t_x) P_\theta(x, t_x)}{\sum_{x \in \mathcal{D}} \frac{1}{P_\theta(x)} \sum_{t_x} \text{N}(\text{Suj}, t_x) P_\theta(x, t_x)} =$$

$$\frac{\frac{1}{0.0009} \cdot 0.0009 + \frac{1}{0.01176} \cdot (0.01176 \cdot 0) + \frac{1}{0.007} \cdot 0.007}{3} = \frac{2}{3}$$

1.4 Question 4 (3 points)

Compute the estimation of the rule (Suj $\rightarrow$ Art Nom) with the example in page 49 from P-I.4 with the Inside-Outside algorithm with the following training sample: $\mathcal{D} = \{$la vieja demanda ayuda, la mujer oculta pelea, la vieja mujer oculta demanda ayuda$\}$. You can use the hyp toolkit1 and you have to provide a plot for the last sentence similar to the ones in slide 50 of P-I.4. Compute just one iteration.

In order to compute the estiomation of the rule (Suj $\rightarrow$ Art Nom), we must compute first the prior probability of each sample. Unfotunatelly, "la vieja mujer oculta demanda ayuda" cannot be generated with our actual grammar. For that reason, we will add a new rule, redistributing the probabilities of the Suj rules such that:

```
0.5 Suj -> Art Nom
0.2 Suj -> Art Adj Nom
0.2 Suj -> Art Nom Adj
0.1 Suj -> Art Adj Nom Adj
```

With that grammar defined, we have the following probabilities for each sample:

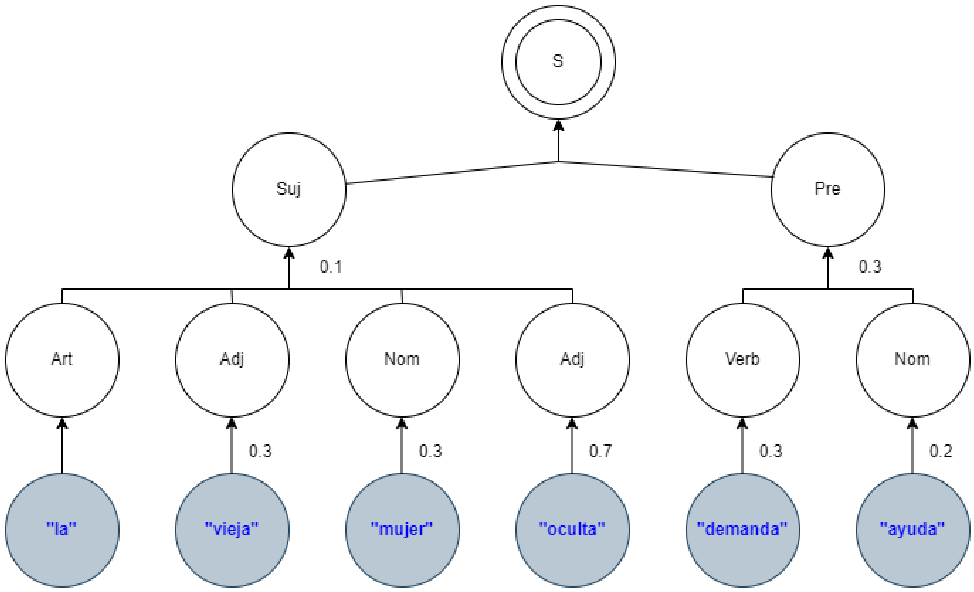
1. P_θ("la vieja demanda ayuda") $= 0.0009 + 0.00168 = 0.00258$
2. P_θ("la mujer oculta pelea") $= 0.0009 + 0.01176 = 0.01266$
3. P_θ("la vieja mujer oculta demanda ayuda") $= 0.0001134$

Then, we can compute the estimation of the rule (Suj $\rightarrow$ Art Nom) as:

$$\bar{p}(\text{Suj} \rightarrow \text{Art Nom}) = \frac{\sum_{x \in \mathcal{D}} \frac{1}{P_\theta(x)} \sum_{t_x} \text{N}(\text{Suj} \rightarrow \text{ Art Nom}, t_x) P_\theta(x, t_x)}{\sum_{x \in \mathcal{D}} \frac{1}{P_\theta(x)} \sum_{t_x} \text{N}(\text{Suj}, t_x) P_\theta(x, t_x)} =$$

$$\frac{\frac{1}{0.00258} \cdot 0.0009 + \frac{1}{0.01266} \cdot (0.0009) + \frac{1}{0.0001134} \cdot 0}{3} = 0.139$$

The derivation tree for the sample "la vieja mujer oculta demanda ayuda" is:



1.5 Question 5 (3 points)

Repeat the previous exercise but using the k-best estimation algorithm with k = 2. You can use the hyp toolkit. Compute just one iteration.

With the k-best estimation algorithm, we only take into account the two most probable derivation trees for each sample. As in the previous example there is no sample with more than two posible derivation trees, results are exactly the same as the previous exercise.

2 Practical assignments

2.1 Question 7 (5 points)

Complete the following table:

After creating different files with the non-terminal symbols specified and varying the number of them, we execute the commands specified in the practice. After one day of execution, we get the following results:

# non-terminal symbols	# rectangle triangles
5	29
10	63
15	61
20	84

2.2 Question 8 (3 points)

Study the classification results depending on the algorithm used for training (Inside-Outside or Viterbi) and the type of samples (bracketed or not). Analyze the results.

First, some experiments have been made with bracketed samples. Results can be seen on Figures 1 and 2:

	equi	isos	righ	Err	Err%
equi	161	442	397	839	83,9
isos	330	355	315	645	64,5
righ	339	309	352	648	64,8

Error: 2132/3000 = 71,07%

Figure 1: Classification results of bracketed samples with Inside-Outside

	equi	isos	righ	Err	Err%
equi	559	379	62	441	44,1
isos	355	366	279	634	63,4
righ	328	347	325	675	67,5

Error: 1750/3000 = 58,33%

Figure 2: Classification results of bracketed samples with Viterbi

The conclusion we can extract from the results is that Inside-Outside takes hours to train, and hasn't enough iterations to distinguish the three types of triangles. On the other hand, Viterbi has lower error, but it fails to ditinguish mostly isosceles and rectangle triangles.

We are also asked to experiment with the algorithms when the samples are not bracketed. For that purpose, we clean the data (taking out all the brackets) and retrain the models. Results can be seen in Figures 3 and 4:

	equi	isos	righ	Err	Err%
equi	705	212	83	295	29,5
isos	442	302	256	698	69,8
righ	384	267	349	651	65,1

Error: 1644/3000 = 54,80%

Figure 3: Classification results of non bracketed samples with Inside-Outside

	equi	isos	righ	Err	Err%
equi	544	456	0	456	45,6
isos	424	261	315	739	73,9
righ	379	256	365	635	63,5

Error: 1830/3000 = 61,00%

Figure 4: Classification results of non bracketed samples with Viterbi

As we can see, both algorithms get low accuracies. Inside-Outside takes lots of times to train and categorizes most of the triangles as equilateral triangles. Meanwhile, Viterbi is a little bit faster but it still fails to learn the features of each triangle.

2.3 Question 9 (3 points)

Study the classification results depending on the algorithm used for learning the PCFG and the size of the training data. Analyze the results.

In order to make experiments, we use the command specified in the exercise to generate new datasets with different seeds for each one. Training data of 500 and 5000 are going to be created to compare the sizes impact on the clasification results. The following figures show the results with different sizes and different algorithms:

	equi	isos	righ	Err	Err%
equi	392	298	310	608	60,8
isos	278	372	350	628	62,8
righ	443	265	292	708	70,8

Error: 1944/3000 = 64,80%

Figure 5: Classification results of Inside-Outside with 500 samples

	equi	isos	righ	Err	Err%
equi	234	449	317	766	76,6
isos	384	429	187	571	57,1
righ	408	319	273	727	72,7

Error: 2064/3000 = 68,80%

Figure 6: Classification results of Inside-Outside with 5000 samples

	equi	isos	righ	Err	Err%
equi	462	320	218	538	53,8
isos	339	297	364	703	70,3
righ	369	299	332	668	66,8

Error: 1909/3000 = 63,63%

Figure 7: Classification results of Viterbi with 500 samples

	equi	isos	righ	Err	Err%
equi	464	166	370	536	53,6
isos	322	311	367	689	68,9
righ	357	332	311	689	68,9

Error: 1914/3000 = 63,80%

Figure 8: Classification results of Viterbi with 5000 samples

It can be seen that Viterbi works better than Inside-Outside in all the cases. Moreover, when we try to get any correlation between number of samples and the error rate, there does not seem to be one. This can be seen by looking at the algorithms independently. In both cases, the error rate is actually higher with more training data. This could be due to the presence of duplicates in the training set.