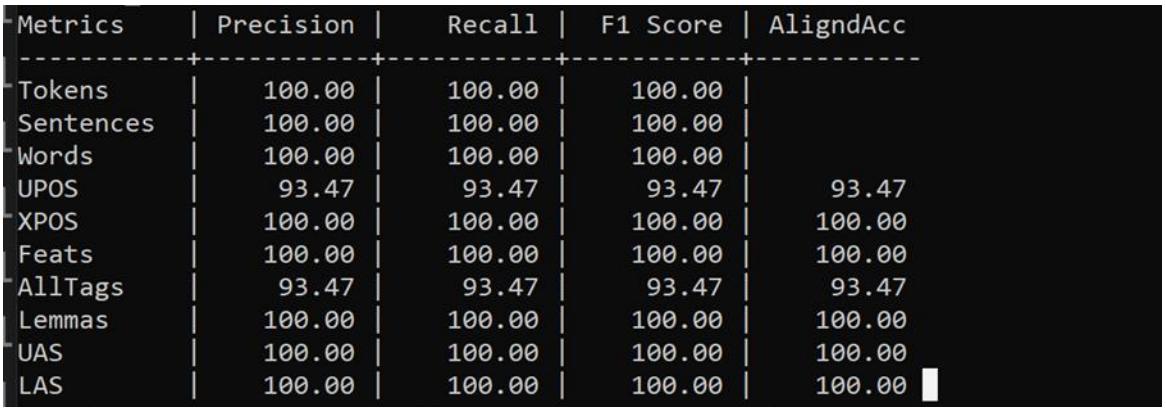
Lab03 – Improve Perceptron Tagger

Kaitlynne Wilkerson

My main attempts at improving the tagger was to see which lines in the getFeatures function played the most important roles in performance and whether there was a rule that existed in the system that, if removed, would cause an increase in the performance score. I was also curious as to whether the system improved the most based on the specific disambiguation rule provided to it or if it was more important to have a certain number of disambiguation rules. To test these ideas, I applied two different strategies. First, check whether the absence of a specific rule in the presence of other rules caused major changes in performance scores. Second, check whether the introduction of a specific rule when no others are present caused major changes in performance scores. These two strategies would provide a picture of how each rule effects the overall performance score and whether the score changes drastically based on the number of rules present. This will, in turn, provide information about how further improvement of the system can be made, including the removal of current rules.

It is important to note what initial performance scores from the system looked like; this will give an idea as to whether performance was generally better or worse when a line was added or removed. It is also important to note that when the term 'large increase' is used, it refers to a change in performance score that is greater than or equal to 10 points.



Metrics	Precision	Recall	F1 Score	AligndAcc
Tokens	100.00	100.00	100.00	
Sentences	100.00	100.00	100.00	
Words	100.00	100.00	100.00	
UPOS	93.47	93.47	93.47	93.47
XPOS	100.00	100.00	100.00	100.00
Feats	100.00	100.00	100.00	100.00
AllTags	93.47	93.47	93.47	93.47
Lemmas	100.00	100.00	100.00	100.00
UAS	100.00	100.00	100.00	100.00
LAS	100.00	100.00	100.00	100.00

Initial Scores from Perceptron-based Tagging

My first attempt was to comment out each line (166-178) to see if any specific rule being removed changed performance in a positive or negative direction. Overall, this strategy did not produce any major changes in performance. The largest drop in performance that occurred due to this strategy happened after commenting out line 171.

Metrics	Precision	Recall	F1 Score	AligndAcc
Tokens	100.00	100.00	100.00	
Sentences	100.00	100.00	100.00	
Words	100.00	100.00	100.00	
UPOS	88.78	88.78	88.78	88.78
XPOS	100.00	100.00	100.00	100.00
Feats	100.00	100.00	100.00	100.00
AllTags	88.78	88.78	88.78	88.78
Lemmas	100.00	100.00	100.00	100.00
UAS	100.00	100.00	100.00	100.00
LAS	100.00	100.00	100.00	100.00

Performance Scores After Commenting Out Line 171

Another close match in performance from this strategy came after commenting out line 166, but the performance scores were still slightly higher than what is shown above.

Metrics	Precision	Recall	F1 Score	AligndAcc
Tokens	100.00	100.00	100.00	
Sentences	100.00	100.00	100.00	
Words	100.00	100.00	100.00	
UPOS	89.62	89.62	89.62	89.62
XPOS	100.00	100.00	100.00	100.00
Feats	100.00	100.00	100.00	100.00
AllTags	89.62	89.62	89.62	89.62
Lemmas	100.00	100.00	100.00	100.00
UAS	100.00	100.00	100.00	100.00
LAS	100.00	100.00	100.00	100.00

Performance After Commenting Out Line 166

This indicates that there was not a specific line in the function that was hindering performance. Since the results from this strategy were not as informative as I had hoped, the next strategy I took was to see if certain pairings of lines or lines themselves contributed more to large increases in performance scores. To establish a baseline for whether a large increase in performance was detected, I commented out every line representing a disambiguation rule (166-178) in the function.

Metrics	Precision	Recall	F1 Score	AligndAcc
Tokens	100.00	100.00	100.00	
Sentences	100.00	100.00	100.00	
Words	100.00	100.00	100.00	
UPOS	55.59	55.59	55.59	55.59
XPOS	100.00	100.00	100.00	100.00
Feats	100.00	100.00	100.00	100.00
AllTags	55.59	55.59	55.59	55.59
Lemmas	100.00	100.00	100.00	100.00
UAS	100.00	100.00	100.00	100.00
LAS	100.00	100.00	100.00	100.00

Baseline Performance of the Perceptron-based Tagger

As expected, the performance score of the perceptron without any disambiguation rules was low. To determine if any specific rules played an important role in large increases in performance, I gradually uncommented each line and checked the new performance score. One interesting result was that when line 166 was the only rule present, it actually led to a massive increase in performance scores. However, when line 166 was commented out initially, it led to only a slight decrease in scores. This large change

could be explained by two possible causes. It is possible that this can be accounted for by the fact that the introduction of any disambiguation rule when none were initially present could lead to large increases in system performance. It is also possible that this specific disambiguation rule was particularly important in parsing the majority of the text correctly.

Metrics	Precision	Recall	F1 Score	AligndAcc
Tokens	100.00	100.00	100.00	
Sentences	100.00	100.00	100.00	
Words	100.00	100.00	100.00	
UPOS	81.47	81.47	81.47	81.47
XPOS	100.00	100.00	100.00	100.00
Feats	100.00	100.00	100.00	100.00
AllTags	81.47	81.47	81.47	81.47
Lemmas	100.00	100.00	100.00	100.00
UAS	100.00	100.00	100.00	100.00
LAS	100.00	100.00	100.00	100.00

Performance Scores After Line 166 is Added from Baseline

To test this, I commented line 166 out and uncommented line 177, so that line 177 was the only disambiguation rule the system had. The performance score that resulted from line 177 was still an increase in system performance, but was dwarfed in comparison to the increase from line 166.

Metrics	Precision	Recall	F1 Score	AligndAcc
Tokens	100.00	100.00	100.00	
Sentences	100.00	100.00	100.00	
Words	100.00	100.00	100.00	
UPOS	66.20	66.20	66.20	66.20
XPOS	100.00	100.00	100.00	100.00
Feats	100.00	100.00	100.00	100.00
AllTags	66.20	66.20	66.20	66.20
Lemmas	100.00	100.00	100.00	100.00
UAS	100.00	100.00	100.00	100.00
LAS	100.00	100.00	100.00	100.00

Performance Score After Line 177 is Added from Baseline

This indicates that the jump in performance score was partially because it was the first rule provided to the system, but that some rules in the system were more important or further encompassing than others. It would, however, be important to continue testing system performance as new rules were added.

With further application of the new strategy, it becomes apparent that no more major increases in performance will occur. Rather, the introduction of a new line either slightly increases or decreases system performance. For example, introducing line 168 made performance drop.

Metrics	Precision	Recall	F1 Score	AligndAcc
Tokens	100.00	100.00	100.00	
Sentences	100.00	100.00	100.00	
Words	100.00	100.00	100.00	
UPOS	80.63	80.63	80.63	80.63
XPOS	100.00	100.00	100.00	100.00
Feats	100.00	100.00	100.00	100.00
AllTags	80.63	80.63	80.63	80.63
Lemmas	100.00	100.00	100.00	100.00
UAS	100.00	100.00	100.00	100.00
LAS	100.00	100.00	100.00	100.00

Performance Scores After Introducing Line 168

Yet, introducing line 171 made performance increase.

Metrics	Precision	Recall	F1 Score	AlignAcc
Tokens	100.00	100.00	100.00	
Sentences	100.00	100.00	100.00	
Words	100.00	100.00	100.00	
UPOS	87.12	87.12	87.12	87.12
XPOS	100.00	100.00	100.00	100.00
Feats	100.00	100.00	100.00	100.00
AllTags	87.12	87.12	87.12	87.12
Lemmas	100.00	100.00	100.00	100.00
UAS	100.00	100.00	100.00	100.00
LAS	100.00	100.00	100.00	100.00

Performance Scores After Introducing Line 171

As we can see from above, adding a new rule does not cause major increases in score, and it does not necessarily cause an increase in score. It also appears that the first couple of rules given to the system cause the largest possible increases in system performance with the size of the improvement being reliant upon the importance of the disambiguation rule. Further application of new disambiguation rules does not tend to cause large changes in performance either. This trend indicates that other potential rules could cause fluctuations in the performance score with large changes being unlikely.

Overall, the strategies employed were not able to improve the system beyond the initial system performance score. However, I was able to demonstrate that each rule was important in creating and maintaining that score, which means that no rule that currently exists in the system significantly lowers the performance score. Furthermore, I was able to provide evidence to support the deduction that the introduction of any new rules would not cause a major fluctuation in performance score.