Song Han
songhan

1. (a) input text: theartrealright and suppose the correct chop is: theatre alright, but greedy algorithm will lead to the art real right if the following cost function is taken:

cost (-BEGIN-, the) = 10

cost (-BEGIN-, theatre) = 100 (pretty high, won't take it in greedy algorithm)

cost (theatre, alright) = 20

cost (the, art) = 10

cost (art, real) = 100 (high in the end)

cost (real, right = 100

if we run the greedy algorithm we'll pick 'the' as the first word instead of 'theartre', and in the end total cost is 220, instead of 130

2(a) input text: h t ystrdy

possible fills: h → he

t → eat, ate

ystrdy → yesterday

cost (-begin-, he) = 10

cost (he, eat) = 10

cost (he, ate) = 20

cost (eat, yesterday) = 100

cost (ate, yesterday) = 40

greedy algorithm will choose "he eat yesterday" with cost 120

but opimal case is "he ate yesterday" with cost 70

3(a)

States: (start location of current word, previous word)

Actions: stride of moving the pointer to the next word

Costs: bigramCost (previous word, current word)

Initial state: (0, -BEGIN-)

isGoad: state[0] == len(query)

iterate over possibleFills (query[state[0]: state[0]+action])

3(c) define $u(w) = \min_{w'} b(w', w)$ thus the cost of the unigram model is smaller or equal to any of the bigram model.

Define the relaxed problem to be: state = index, cost is using the unigram model instead of the bigram model. From the definition of the unigram model, the cost of a state to another is smaller or equal to the orignal problem, thus it's a relaxed problem. Using relaxed problem to estimate the heuristic is guaranteed to be consistant.