1 Which problems does maxmatch suffer from? (Choose all that apply.)

a) requires comprehensive dictionary +

b) is computationally expensive +

c) is difficult to program -

d) constructs non-grammatical sentences +- (error rate depends on the language)

Detailed answer:

a) yes, the algorithm absolutely depends on the dictionary, for if it lacks a word (sequence of symbols) present in the document, the word will be divided element-wise. If only a shorter sequence of symbols than the actual word is found, than this sequence will undutifully gain the status of a word.

b) yes, it is a greedy algorithm, as every sentence is iterated element-wise across all the dictionary.

c) no, the algorithm is comparatively simple to realise.

d) separation flaws result in non-grammatical sentences, which are prolific in alphabetical languages(multiple letters combine into various words easily), but seem less frequent in case of hieroglyphic scripts.

2 Write a perl/sed substitution with regular expressions that adds whitespace for segmentation around "/"in "either/or"expressions but not around fractions "1/2":

"either/or—> "either / or"

"1/2—> "1/2"

Answer:

3 The text mentions several times that machine learning techniques produce better segmentation than rule-based systems; what are some downsides of machine learning techniques compared to rule-based?

Answer:

1) need enough reference data to train. the more data, the better the results.

2) data-biased, but only if training data are restricted, different text genres need to be processed after training on different data (although it also refers to rule-based segmentors: different texts may require different segmentation rules).

3) liable to crash, for no apparent reason, e.g. if too much data is processed.

4) time- and memory-consuming, concerning training.

4 write a sentence (in English or in Russian) which maxmatch segments incorrectly.

Answer:

We like lying otter-like. —> [We, likely, ingot, ter, like]

5 what are problems for sentence segmentation? provide one example in English or Russian for each that applies.

a) ambiguous abbreviations with punctuation

Yes, periods within sentences make segmentation a tougher task.

Examples in English: e.g., Mr., Mrs., U.S.A, etc. In Russian: и т.д., д. 7,

Other ambiguous abbreviations in English: 'he's' = 'he is' or 'he has', 'he'd' = 'he had' or 'he would'. They could be a problem, because the apostrophe can be interpreted as beginning/end of a quote, but then all the quoted words would pose a problem.

b) sentences containing symbols '!' and '?'

It depends, but these problems seem to be minor (in general case).

'!'

In Russian '!' can be found within a sentence, expressing an emotion but not the end of the sentence, so using '!' as a separation marker would cause a mistake. However, the part of the sentence after '!' begins with a lower-case letter, so a regexp could be used to avoid such mistakes. But it is a rare case in Russian, and I haven't noticed it in English.

'?'

You can imagine such kind of sentence both in English and in Russian: 'Does he want strawberries? Cranberries? Or simply salad?' Although the boundaries are not distinct, it doesn't seem a problem to use '?' as a sentence separation marker. Yes, these sentences are very closely related, but they still can be regarded as sentences.

c) sentences lacking separating punctuation

It depends on the tools we use. Yes, it it a problem in case of sentence separation based on rules, it becomes useless. At the same time, machine learning with pre-processing including POS-tagging might help out.

d) sentences not separated by whitespace

Not a problem, provided punctuation is present.