Question 1c)

All three methods performed well in different ways. The leading method produced the most coherent summary but presented less information from each subsequent sentence as the other methods did (it pretty much just repeated the main point over and over). The orig method was less coherent at times (using words/pronouns that referenced previous sentences, or sentences that were an add on to a previous sentence) but presented more variation in information. The simplified method just repeated one sentence over and over. I don't know if this was right, so I made it delete sentences that had already been said, and it performed similarly to the orig method (I couldn't really tell the difference). I guess the non-redundancy worked because the simplified method just kept repeating itself. Ordering the sentences could be done by considering the position of the sentences in the original texts and arranging the sentences in the summary according to their original positions. For example, if a sentence A came from text 1 and was 20% down the article, and a sentence B came from text 2 and was 50% down the article, then sentence A would appear before sentence B in the summary.

Question 2

This paper introduces new techniques for sentence compression. Previously, sentence compression algorithms focused primarily on word deletion to reduce the length of a text. This study considers other operations such as reordering, substitution, and insertion in order to compress sentences (abstracts rather than extracts). A new corpus was created by getting an annotator to rewrite and compress news articles (two annotators were compared and found to compress in similar amounts and ways). The method used to compress text was formulated as a tree to tree task. Grammar rules were used to find output trees for a given input tree, and a decoder was used to find a maximum over the output trees using the derivation rules of each tree. The model was trained using SVM, which enabled a loss function. The loss function measured the difference between the model prediction and the decoder reference (training attempted to find a derivation that is both high scoring and has high loss).

For the experimental design, a grammar was first extracted from the corpus and a bilingual corpus. The performance of the model was assessed using 385 training sentences, 36 development sentences, and 59 testing sentences. The loss function was chosen using human judges. Evaluation was also done using human judgements. The results showed that the model was good at retaining important information while shortening the sentences. It performed worse, however, at preserving grammaticality. Overall the model performed much worse than the gold standard (human written), in information, grammaticality, and compression rate.

Some limitations of this approach are that models using reordering or substitution can be more prone to producing grammatically incorrect outputs, as opposed to models using deletion. This could be mitigated by hardcoding rules when reordering or substituting that ensure grammaticality. Another limitation is that the compression rate is not fixed. This could be a problem for when you need the text to be reduced to a specific size. This could be fixed by reducing the outputs used to only those that fall within a certain length range.

The grammar extracted from the corpus is represented as a CFG. The process is a form of natural language generation, even though it is a summary of another text, because the new sentences differ in structure and occasionally content as opposed to the old ones (you could say that compression using extraction does not produce any new sentences as the process just trims the old sentences). This study attempts to improve on extractive summary techniques by allowing for other compression operations such as reordering and substitution (abstractive summary).

Questions:

1. What are some practical applications that abstractive summary has been used for?
2. If different languages have different grammar structures, how does using the bilingual corpus help to extract the grammar for the original (one language) corpus?
3. How can this technique be applied to entire texts, instead of just for each sentence? (reordering, substituting, deleting entire sentences)