**B. Data sets**

(tähän juttua miksi valittiin Daily Mail artikkelien käyttö sen alkuperäisen CNN/DM data setin sijaan)

We tested 73 different documents fetched from the dailymail.co.uk website. The links to the documents were stored in a text document and automatically processed with a Python script.

For the automatic processing of the data set, we developed a Python script that creates the summaries using both PyTLDR and the newly developed named-entity summarizer. This process bypassed the use of the GUI for faster processing of large amount of data. The script also measured the time each algorithm took to produce the summaries.

When testing the data, we created summaries that were of equal length to those of the human written reference texts, measured in the amount of sentences.

**Named entity summarizer algorithm**

The new text summarization algorithm we developed uses the Python library spaCy for Named Entity Recognition and the library TextBlob to extract the sentences from the original article.

The spaCy model used for named entity recognition is en\_core\_web\_sm, which is an English multi-task CNN trained on OntoNotes. [1]

The Named Entities used to create the summary are those labeled as a person or an organization.

The algorithm is given the document text and the desired length of the summary measured in sentences as parameters. All the relevant named entities are searched from the document text and added onto a list. If a named entity appears in the document text n-times, it is also added onto the list n-times. This is later used for weighted scoring of the sentences.

Each sentence is extracted from the article and added onto another list. All the sentences on the list are then evaluated against the list of the named entities. When a listed named entity appears in a sentence, that sentence will get a score of 1 for each appearance. As the named entity list can have one named entity listed several times, the appearance of that particular named entity gives a higher overall score for that sentence. This is deliberate as if a named entity appears several times in the document, it is assumed that it holds more importance compared to the other named entities, and as such should be included in the summary.

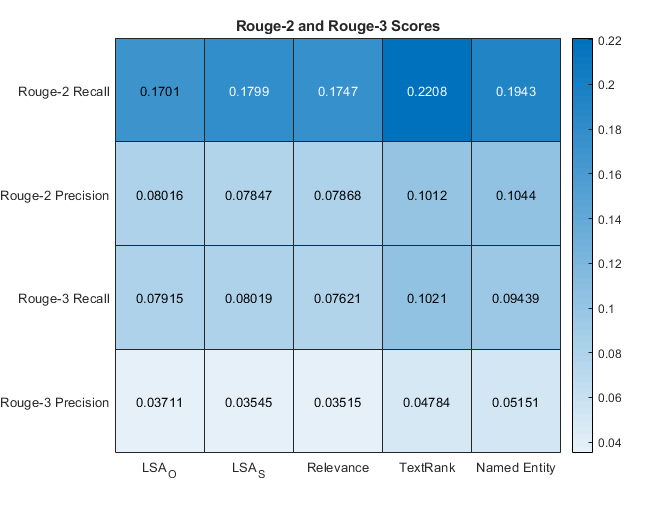
The highest scoring sentences are added into the summary in the order in which they appear in the document as not to disrupt any possible continuation flow in the text.

1. <https://spacy.io/models/en>

**Results**

We tested 73 different documents using 5 different summarizing algorithms; Latent Semantic Analysis by J. Steinberger and K. Jezek, Latent Semantic Analysis by M. Ozsoy, F. Alpaslan and I. Cicekli, Relevance Score Summarization by Y. Gong and X. Liu, TextRank Summarization and finally the named entity summarizing algorithm we developed during this project.

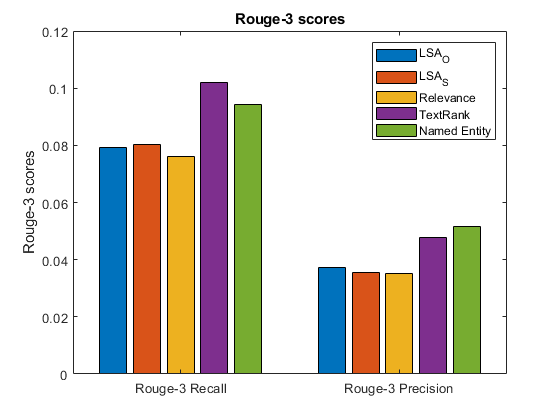
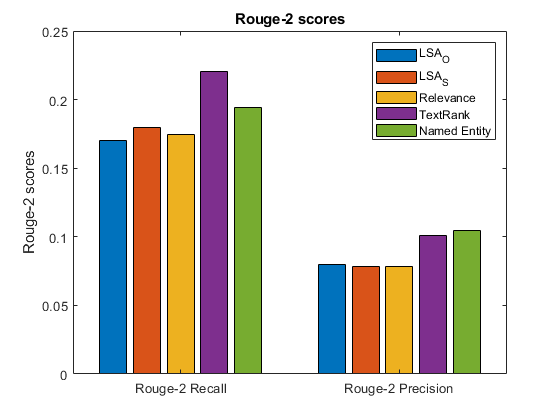
The Rouge-N scores were tested against a human written reference text. The mean length of the document texts was 1264 words. The mean length of the reference texts was 76 words. There was not a large variation between the different algorithms on the length of summary. The mean for the length of the summaries was 165 words.

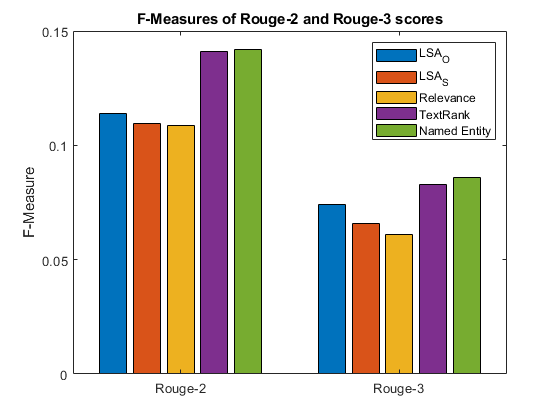


Overall, the results were not very good, but we were still able to compare the different algorithms and their performance against each other.

The best performing algorithm when tested against the Rouge-2 and Rouge-3 Recall was the TextRank algorithm.

Our developed named entity summarizing algorithm performed the best when tested against Rouge-2 and Rouge-3 Precision.

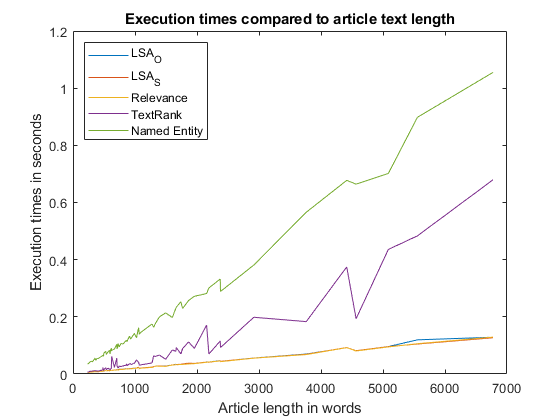




When both Recall and Precision methods were taken into account and calculated into the F-Measure, our named entity algorithm had the best performance out of all five algorithms.

**Efficiency**

The efficiency was measured in how many seconds it took for the algorithm to produce the summary.



The best performing algorithms when measured against the Rouge-N scores were the slowest. The execution time for the TextRank algorithm was almost twice the amount of the most efficient algorithms. The execution time for the named entity algorithm was almost 7.5 times as long. As mentioned before, performance optimization wasn’t the highest priority during this project. The performance could be improved during future development.

**Future development**

As the named entity summarizing algorithm is rather inefficient in comparison to the other algorithms, it should be optimized first before any further future use. For this project, we didn’t consider optimization to be of the upmost importance and prioritized creating a working algorithm we could test against the other algorithms.

In case of successful optimization of the algorithm, it could be useful as part of a product aimed at companies who need to quickly find out what is being written about their product or brand on the Internet.