Grammarly is an advanced writing assistant that uses NLP to help users

with spelling, grammar, punctuation, and style issues in their writing.

Many people struggle to express their ideas clearly and organize them

effectively when writing emails or documents. Using grammarly will improve

your writing efficiency and keeping content organized.

Grammarly uses NLP and ML to identify key points in your text and predict the reader's

focus. Grammarly uses extractive summarization algorithms like LexRank and TextRank,

inspired by PageRank, to identify important sentences based on their connections

to others in the text. In simpler terms LexRank and TextRank look

at the total words that a sentence has in common with the rest of the sentences

This helps users to focus more on the important parts of their text for improved

communication. It also effectively helps the user to avoid any irrelevent or redundant

information.

Another approach to text summarization using NLP is that they incorperate

BERTSUM. You input the user's text into a large language model to obtain a latent

representation of each sentence, then use labeled training data to train the

model to distinguish between main points and non-essential sentences.

BERTSUM allows the user to quickly understand the main idea of the long-form content.

Grammarly uses semantic features to identify whether a sentence is a main point.

The system features tend to differ These features vary by domain. For example,

in emails people often put the highlighted points at

the start of their message while in a lengthy document, the key points

would be in the title and section headers.

These features enable grammarly to improve relevance of feedback.

To reduce latency, Grammarly optimized its reinforcement learning model using

sampling techniques for longer texts. The approach involves

selecting random groupings of main points and scoring them based on our

features, rather than evaluating the entire set of possibilities, but they needed

more than that to balance latency. Other examples of optimization they used are

tracking the trajectory of weights that the model gives each sentence at each

sampling iteration. Another example would be dropping sentences that are

definitely not the main point. All these techniques reduce inference time

significantly. This basically means that the optimizations lead to a fast

user experience.