Effective Variable-Length-to-Fixed-Length Coding via a Re-Pair Algorithm
Satoshi Yoshida* and Takuya Kida*
*Graduate School of Information Science and Technology, Hokkaido University.

1. Background
VF code
Compression method that splits the input text into variable length substring and then converts them into fixed length codewords.

Strong point
Easy to handle the compressed data.
> Enables fast information retrieval or data mining.

Goal
To develop a VF code that achieves high compression ratio.

2. Grammar-based compression
Construct a grammar that generates the input text only and encodes the constructed grammar.

3. Re-Pair algorithm
Substitute the most frequent bigram into a new symbol until all the bigrams are unique.

4. VF code via Re-Pair algorithm
Encode each symbol with a fixed length codeword.

5. Proposed method
Symbol replacement does not always imply improvement of compression ratio.
> We want to encode with the "optimal" dictionary, which gives highest compression ratio.

6. Experiments
We compared compression ratios, compression speeds, and decompression speeds.

Datasets:
- Japanese text (dazai.utf.txt, 7MB)
- XML document (dblp2003.xml, 90MB)
- DNA sequence (gbhtg119.dna, 87MB)
- English text (reuters21578.txt, 18MB)

Results:
- Compression ratio is better than that of gzip for natural language texts.
- Compression speed is as the original Re-Pair.
- Decompression is faster than that of bzip2.
- Pattern matching is faster than zgrep.

7. Conclusion and future works
We proposed a VF code that uses Re-Pair algorithm and evaluate its performance experimentally in this poster. To develop VF codes that use other grammar-based compression is our future work.