VQ-Based Model Design Algorithms for Text Compression

Seung P. Kim†
Dept. of Elec. Eng., Polytechnic Univ.
Brooklyn, NY 11201, USA.
e-mail: skim@signal.poly.edu

Xavier Ginesta
Dept. of Elec. Eng., Polytechnic Univ.
Brooklyn, NY 11201, USA.
e-mail: ginesta@signal.poly.edu

Abstract

We propose a new approach for text compression where fast decoding is more desirable than encoding. An example of such a requirement is information retrieval systems. For efficient compression, high-order conditional probability information of text data is analyzed and modeled by utilizing vector quantization concept. Generally, vector quantization (VQ) has been used for lossy compression where input symbol is not exactly recovered at the decoder, hence it does not seem applicable to lossless text compression problems. However, VQ can be applied to high-order conditional probability information so that the complexity of the information can be reduced. First, we represent the conditional probability information of a source in a tree structure where each node in the first level of the tree is associated with respective 1-st order conditional probability and the second level nodes with the 2-nd order conditional probability, etc [1]. For good text compression performances, it is necessary that fourth or higher order conditional probability information be used. If 8-bit ASCII code representation is assumed with 4-th order model, the amount of conditional information is about \((2^8)^4 = 4.3 \times 10^9\) number of probability tables. Even if such a large amount of data can be handled, it is practically impossible to get necessary probability information through training. Therefore, it is essential that the model be simplified enough for training with a reasonable size of training set. We reduce the number of conditional probability tables using the algorithm developed in [2]. We also discuss a semi-adaptive operating mode of the model where the tree is derived through training but actual probability information at each node is obtained adaptively from input data. The performance of the proposed algorithm is comparable to or exceeds other methods such as Prediction by Partial Matching (PPM) but requires smaller memory size.


†Corresponding Author