A Data Compress Algorithm for Daisy Chain Communication System

Ziyang Xu, Yu Wang*, Ci Gong, Junkang Chen, Jian Zhou, Yi Lu, Xinyu Zhang, Longyu He and Zihao Huang
School of Information Engineering, Wuhan University of Technology, Wuhan, China

*Corresponding author email: wangyu@whut.edu.com

Abstract. In the era of big data, traditional data compression methods cannot complete data compression with features of large volume, rapid growth and complex structure, especially in the processing of real-time data compression. The compression algorithm proposed in this paper is improved on the basis of a lossless bit-swap compression coding algorithm. The coding algorithm is based on VAE model and uses classical ANS compression technology to code multiple potential variables, so as to realize the rapid compression and decompression of data. The experimental results show that the algorithm proposed in this paper can compress data with a high compression rate (47.65%), which is higher than LZAM algorithm and Bit-Swap algorithm.

1. Introduction
With the development of machine learning, the connection between data compression and probability model is more inextricably linked[1]. Huffman coding, arithmetic coding and asymmetric digital systems are classic algorithms for data compression based on statistical models, i.e. data compression is achieved by the frequency of the occurrence of individual characters in statistics[2-3]. Most of the compression tools combine and optimize a variety of compression coding algorithms[4], and it is rare to use a single compression algorithm. For example, the GZIP compression method first uses a variant of the LZ77 algorithm to compress, and then uses static or dynamic Huffman coding method to compress the obtained results[5]; LZAM compression method is an improved and optimized compression algorithm combining the Deflate algorithm and LZ77 algorithm[6].

Figure 1. Overall flow chart of system compression coding
However, in the era of big data, traditional data compression methods cannot complete data compression with features of large volume, rapid growth and complex structure, especially in the processing of real-time data compression[7].

The compression algorithm proposed in this paper is improved on the basis of a lossless bit-swap compression coding algorithm, which is based on VAE model and uses classical ANS compression technology to code multiple potential variables, so as to realize the rapid compression and decompression of data. The overall process of compression coding in this system is shown in Figure 1.

2. Research and Construction of Variational Autocoding Model in Data Compression

In classical data compression based on statistical theory, any data distribution can be converted into lossless coding, in which any data point is encoded into a number of bits equal to the negative logarithmic probability assigned by the model[8]. When the model is matched with the real data distribution, the best expected code length can be obtained.

2.1. The Principle of Variational Autoencoder

Variational autocoding model (VAE) is a generative model that can infer unknown data from known data[9]. VAE mainly solves the problem of how to construct encoders and decoders, so that the data can be encoded into a easy and lossless form to represent, and the data decoding can truly restore the original data, which is somewhat similar to PCA matrix dimensionless reduction, as shown in Figure 2.

![Figure 2. PCA matrix dimension reduction](image)

![Figure 3. Auto-encoder model](image)

$x$ is a matrix, by transformation ($W$) into a lower dimensional matrix $C$. Since this process is linear, $\hat{x}$ can be reduced by the inverse transformation ($W^T$). If $W$ and $W^T$ are replaced by neural network, the model is transformed into a Deep auto-encoder model, as shown in Figure 3.

3. Design and Implementation of Improved Variational Autocoding Communication System

3.1. Problem Analysis

We built a daisy-chain communication system, each device information on the line is 24 bytes, in this case the time to send 24 bytes is 0.02s, when there are more devices in a line, the longer the serial transmission time, the closer the device to the main station power consumption.

In order for equipment to work in a field environment for one year, in addition to the low power consumption of hardware systems, it is necessary to compress data and reduce data transfer time. Compression efficiency and speed are key in the process of compressing data. Bit-Swap coding[10] based on variable self-coding models uses recursive potential variables to solve the problem of initialization that requires a large number of initial bits, which can achieve fast and efficient compression of data and be suitable for this system.

Compared with traditional neural network compression technology, the variable self-coding model based on deep learning has the following advantages[11].

1. The variable self-coding model has a multi-layered network structure, which can be nonlinearly mapped to obtain deeper characteristic values of the data set when training and compressing data.
2. In the variable self-coding model, all data can be removed from a large amount of redundant information and obtained higher compression rate after the sparse processing of neural networks.

The data obtained by each node in this communication system is the tilt of the device and can be processed according to the actual situation. The initial inclination is obtained when the device is enabled,
and when the device status is queried or the device shakes, the data to be compressed is the difference between the actual inclination and the initial inclination. When the device is not wiggling, the actual inclination of the device is zero from the initial inclination in order to achieve maximum compression. The structure of the compression algorithm in this paper is shown in Figure 4.

![Figure 4. Compression algorithm in this paper](image)

3.2. Sample Data Training
First, read the sample data, and the sample data in each one-dimensional vector is trained by neural networks to discrete the mean and variance obtained by the training. Nothing is done when the maximum value of the mean or variance in a set of data is the same as the minimum value, and when the maximum value is different from the minimum, the maximum value is evenly split from the minimum value by the number of feature values, and then the median and last values in the data are obtained. The discretization of the sample data is done as shown in Figure 5.

![Figure 5. Complete training of data set](image)

3.3. Data Compression
The process of data compression on each node is shown in Figure 6. In the process of data encoding, the following processing is needed for the data $x$ containing the fluctuation value of the tilt Angle.

![Figure 6. Data compression process](image)

(1) A sample is made based on the known data to be compressed, first, the sampled data is trained by the neural network to produce the mean and variance, and the corresponding Gaussian distribution $q(z_1|x)$ is further generated. Secondly, the probability mass function is obtained using the probability distribution function, which is used for the initialization of ANS. Third, the asymmetric digital system (ANS) is used to decode $q(z_1|x)$ to get the potential variable $z_1$, at which point the bit rate is $\log q(z_1|x)$, and then the neural network of the sub-generated model is used to process $z_1$ to obtain mean and variance, The corresponding probability distribution function $p(x|z_1)$ and the probability mass function are further obtained, and then the $p(x|z_1)$ is encoded using the asymmetric digital system (ANS), at which point the bit rate is $\log q(z_1|x) - \log p(x|z_1) + \log q(z_2|z_1)$. $Z_2$ is then processed using the generating model neural network to obtain mean and variance, and the probability distribution function $p(z_1|z_2)$ is
obtained. Encode \( p(z_1|z_2) \), at which point the bit rate is \( \log q(z_1|x) - \log p(x|z_1) + \log q(z_2|z_1) - \log p(z_1|z_2) \).

3 Using \( z_2 \) for sampling, the sampled data is trained by the neural network of the reasoning model to obtain mean and variance, and finally obtains \( q(z_3|z_2) \), decoded by ANS to get \( z_3 \), at which point the bit rate is \( \log p(z_2|z_3) \) is obtained.

After getting \( z_3 \), using ANS encode the data. The entire coding process is shown in Figure 7.

3.4. Data Decompression

Obtain the actual tilt Angle of the device, as shown in Figure 9:

The decoding process is the opposite of encoding steps. First of all, ANS is used to decode the potential variable \( z_3 \), according to \( z_3 \) sampling, the data after sampling by the generated model neural network processing to get mean and variance, further get the corresponding probability distribution function \( p(z_2|z_3) \), use ANS to decode the potential variable \( z_2 \), according to the potential variable \( z_2 \) to get the corresponding probability distribution function \( q(z_3|z_2) \), using \( q(z_3|z_2) \) encoding.

After obtaining the potential variable \( z_2 \), in the operation steps of \( z_3 \) above, the \( z_2 \) is done in the same way, and the probability distribution functions \( p(z_1|z_2) \), \( q(z_2|z_1) \), the potential variable \( z_1 \), and further encode \( q(z_2 \ s \ z_1) \) can be obtained. After getting \( z_1 \), the probability distribution functions \( p(x|z_1), q(x|z_1) \) can be obtained, and the whole decoding process is shown in Figure 8.

The background will save the initial inclination information after initial decoding, and the subsequent inclination information needs to be contrasted with the initial inclination to get the real inclination.

4. Data Compression Experiment Results

4.1. Experimental Data

In the compression experiment, use Python for simulation experiment. The training sample was 32x32 pictures, and the number of training sets was 380,000. The data to be compressed was in the same format as defined by the serial port protocol in the system, and a group of 24 bytes was added each time.

4.2. Comparison of Experimental Algorithms

Three algorithms are used in algorithm comparison. The first algorithm is LZAM\(^{[12]} \) algorithm, the
second algorithm is the bit-swap coding algorithm based on variational autocoding, and the third algorithm is the improved variational autocoding algorithm of this system --Processed. The calculation formula of data compression rate is shown in Equation (1):

\[
\text{Compression ratio} = \frac{\text{Bit number of data compression processing}}{\text{Data original bits}}
\]  

(1)

Using 24 bytes of data, after three compression methods, a total of three experiments were carried out, compression information as shown in Table 1. The uncompressed rate is short for UR, the compression rate is short for CR. UR-1, UR-2, UR-3 represents the uncompressed rate of the first, second, and third experiments, respectively.

| Algorithm | UR-1 (%) | CR-1 (%) | UR-2 (%) | CR-2 (%) | UR-3 (%) | CR-3 (%) |
|-----------|----------|----------|----------|----------|----------|----------|
| LZAM      | 89.21    | 10.79    | 89.21    | 10.79    | 89.21    | 10.79    |
| Bit-Swap  | 55.39    | 44.71    | 58.65    | 41.35    | 59.88    | 40.12    |
| Processed | 52.58    | 47.42    | 52.55    | 47.42    | 52.35    | 47.65    |

Twenty-five experiments were carried out in the simulation experiment of the system, and each experiment added a set of 24 bytes to the data to be compressed. The experimental results are shown in figure 10, blue represents the LZAM algorithm, red represents the bit-swap algorithm based on variational autocoding, and yellow represents the improved variational autocoding algorithm.

As can be seen from the figure 10, as the number of compressed bytes increases, the compression rate of the three compression algorithms almost remains unchanged. The LZAM algorithm has a poor compression rate of about 10%, while the Bit-Swap algorithm compression rate is about 40%, and the improved algorithm compression rate is close to 50%. The improved algorithm is always a little higher than the Bit-Swap algorithm compression rate.

5. Conclusion
This paper mainly studies the application of variational autocoding algorithm in lossless compression. First, the sample data were trained to obtain the discrete data, then the data to be compressed was processed by VAE model and encoded by combining ANS technology. Multiple potential variables were used in the encoding process. In this system, combined with the actual application of the system, the data to be compressed is processed by differential processing, and then compressed processing, which can obtain a better compression rate.
Acknowledgments
This work was financially supported by college student innovation fund.

References
[1] Ahfock D C and Astle W J and Richardson S. Statistical properties of sketching algorithms [J]. Biometrika, 2021, 108(2) : 283-297.
[2] Zeinab Tirandaz and Gholamreza Akbarizadeh and Hooman Kaabi. PolSAR image segmentation based on feature extraction and data compression using Weighted Neighborhood Filter Bank and Hidden Markov random field-expectation maximization [J]. Measurement, 2020, 153.
[3] Kumar M V P, Mahapatra S. Pyramid coding based rate control for variable bit rate video streaming [J]. Iet Image Processing, 2016, 10(9):671-680.
[4] Fangliao YANG et al. A Novel Two-Stage Compression Scheme Combining Polar Coding and Linear Prediction Coding for Fronthaul Links in Cloud-RAN [J]. IEICE Transactions on Communications, 2017, E100.B(5) : 691-701.
[5] Zubair Ali Syed and Tariq Rahim Soomro. Compression Algorithms:Brotli, Gzip and Zopfli Perspective [J]. Indian Journal of Science and Technology, 2018, 11(45) : 1-4.
[6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In Advances in Neural Information Processing Systems, pp. 5998–6008, 2017.
[7] Townsend, J., Bird, T., and Barber, D. Practical lossless compression with latent variables using bits back coding. In International Conference on Learning Representations, 2019.
[8] Wallace, C. S. Classification by minimum-message-length inference. In International Conference on Computing and Information, pp. 72–81. Springer, 1990.
[9] Veera Ankalu Vuyyuru and G. Appa Rao and Y. V. Srinivasa Murthy. A novel weather prediction model using a hybrid mechanism based on MLP and VAE with fire-fly optimization algorithm [J]. Evolutionary Intelligence, 2021, : 1-13.
[10] Kingma, Friso and Abbeel, Pieter and Ho, Jonathan. Bit-Swap: Recursive Bits-Back Coding for Lossless Compression with Hierarchical Latent Variables. In proceedings of the 36th International Conference on Machine Learning, PMLR 97:3408-3417, 2019.
[11] LIANG Jie1, CHEN Jiahao2, ZHANG Xueqin2, ZHOU Yue2, LIN Jiajun. One-hot encoding and convolutional neural network based anomaly detection [J]. Journal of Tsinghua University (Science and Technology) 2019,07.
[12] weijia zhu, Adaptive LZMA-BASED Coding FOR Screen Content [J], IEEE,2013:373-375.