

Article

An Efficient Lossless Compression Algorithm for Maritime Safety Information Using Byte Encoding Network

Jiwei Hu , Yuan Gao, Qiwen Jin, Guangpeng Zhao and Hongyang Lu *

School of Information Engineering, Wuhan University of Technology, Wuhan 430070, China; hujiwei@whut.edu.cn (J.H.); 334658@whut.edu.cn (Y.G.); qiwenjin@whut.edu.cn (Q.J.); 276223@whut.edu.cn (G.Z.)

* Correspondence: luhongyang@whut.edu.cn

Abstract: The short message function of the BeiDou satellite system, due to its strong concurrent processing capabilities, can quickly and accurately send information to the target location in emergency situations. However, because of data redundancy and limitations on message length, a single piece of information often requires multiple transmissions through BeiDou short messages to be completed, resulting in limited transmission capacity. To improve the transmission capacity of information, it is necessary to compress the information during transmission using BeiDou's short message communication function for maritime safety information. This paper proposes a Byte Encoding-enhanced Prediction by Partial Matching, variant D (BPPMd) algorithm that is particularly suitable for transmitting maritime safety information. Combined with a maritime safety information encoding algorithm (ME), it further improves compression efficiency, optimizes byte space, reduces information redundancy, and ensures the accuracy of the information. In this study, we constructed a maritime safety information dataset that includes three categories of information: meteorological warnings, navigation warnings, and disaster warnings. Experimental results show that the proposed algorithm is particularly suitable for compressing the maritime safety information dataset and outperforms other benchmark algorithms. Therefore, this study indicates that the proposed lossless compression method can be a feasible and effective solution for BeiDou short message communication.

Keywords: maritime safety information; data compression; byte encoding module; binary encoding



Citation: Hu, J.; Gao, Y.; Jin, Q.; Zhao, G.; Lu, H. An Efficient Lossless Compression Algorithm for Maritime Safety Information Using Byte Encoding Network. *J. Mar. Sci. Eng.* **2024**, *12*, 1075. <https://doi.org/10.3390/jmse12071075>

Received: 4 June 2024
Revised: 17 June 2024
Accepted: 24 June 2024
Published: 26 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The BeiDou Navigation Satellite System [1] (BDS) is an independently developed and operated navigation system by China. It integrates navigation positioning, user detection, standard timing, and short message communication. It is the third mature satellite navigation system [2] after the U.S. GPS [3] and Russia's GLONASS. Compared to other major operational navigation satellite systems [4], the BeiDou system uses an RNSS navigation signal structure similar to that of the U.S. GPS and Europe's Galileo, with an application accuracy close to GPS. The most notable feature of the BeiDou system is its satellite positioning and short message communication capabilities.

Since its launch in 2013, BeiDou's Short Message Communication [5] (SMC) has been instrumental in life-saving rescues [6] and in providing position reporting for maritime fishing vessels. The BeiDou SMC maritime communication system consists of three parts: the onboard terminal, BeiDou satellites, and the receiving platform [7]. The specific process involves the onboard terminal sending short message information to the satellite, which then transmits the short message information to the receiving platform. The receiving platform subsequently sends the short message information to another user terminal. The BeiDou SMC maritime communication system is illustrated in Figure 1.

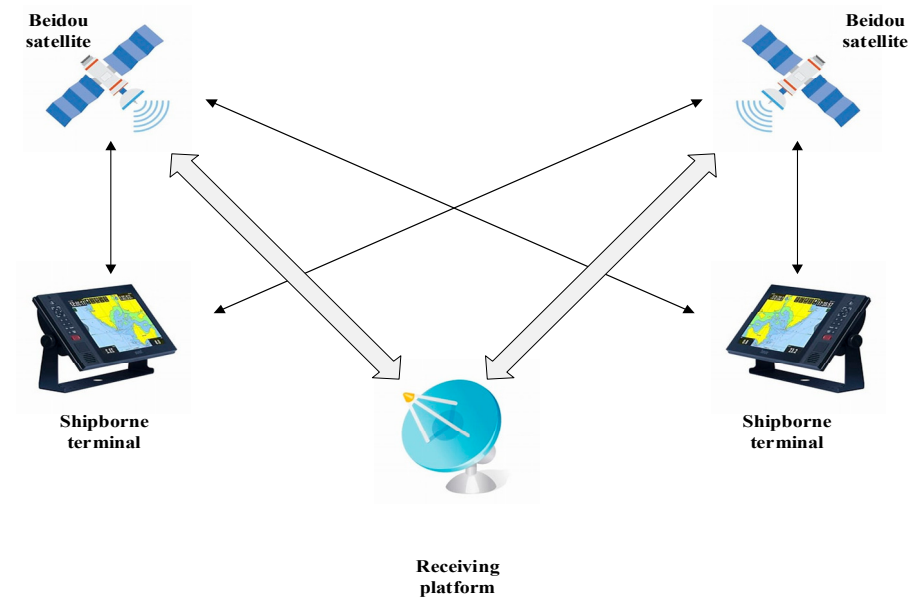


Figure 1. Beidou Short Message Maritime Communication System.

SMC is a communication method that uses the BeiDou Navigation Satellite System (BDS) to transmit short messages [8]. It utilizes intersatellite links for transmission, unaffected by interruptions in ground communication links or signal base stations. This method boasts round-the-clock coverage without blind spots, allowing for the widespread submission of large amounts of safety data in maritime areas [9]. It features high-precision positioning and navigation capabilities, along with strong interference resistance [10] and bidirectional digital message communication capabilities. Currently, it finds widespread applications in various fields such as weather alert dissemination [11], automatic meteorological station data transmission [12], and high-precision marine surveys [13].

Due to the limited transmission capacity of SMC and significant restrictions on the length of information [14], it necessitates packetization for transmission. However, excessive packetization can lead to increased packet loss during transmission. To enhance the transmission capacity of information, it is essential to process short messages by reducing space occupancy [15] to increase the transmitted information. In this paper, we present a method to encode maritime safety information. By organizing various types of data in the dataset into a dictionary, we perform byte-space encoding of maritime safety information based on this dictionary. After encoding, we further improve compression rates and reduce redundancy by combining efficient compression algorithms.

Data compression technology [16] aims to remove redundant information from input data streams, utilizing smaller output data streams to preserve all or some essential information from the input data stream. Data compression technology has been developed for over thirty years, yielding numerous excellent compression algorithms. Present compression algorithms are divided into dictionary coding and entropy coding methods. The former includes algorithms such as LZ77 [17], and LZMA [18], while the latter includes Huffman coding [19], and arithmetic coding [20].

The PPMd algorithm is an efficient data compression algorithm, which is an improved version based on the Prediction by Partial Matching [21] (PPM) algorithm, specifically suitable for compressing text data. The PPMd algorithm primarily achieves compression by establishing a statistical model based on historical data. Its core idea is to use context modeling to predict the probability distribution of the current character and encode characters based on this probability distribution. In the process of compressing maritime safety information, the PPMd algorithm demonstrates superiority over other compression algorithms, showing a significant advantage in compression ratio. It can reduce data size, resulting in faster transmission and better resource utilization. In this paper, we propose an

improvement scheme for the PPMd algorithm, named BPPMd, which incorporates a byte encoding module during the input process, further reducing the compression ratio.

The key contributions of this paper are as follows:

- (i) Based on the maritime safety information from the Maritime Safety Administration of the People's Republic of China and the China Oceanic Information Network, we conducted an analysis of the data and integrated it into three different datasets: meteorological forecasts, navigation warnings, and disaster warnings. Since there was limited data on storms, waves, sea ice, and tsunamis in the disaster warning information, we used a large model to expand it, thereby creating the Maritime Safety Information Dataset.
- (ii) When using BeiDou short messages, the transmission of maritime safety information faces challenges of redundant data and limited short message length. To address these issues, we propose the BPPMd algorithm. At the input stage, we introduce a byte encoding module that performs semantic understanding and feature extraction on the current input. This module predicts the data and updates the network, thereby eliminating redundant information. Combined with a lossless compression algorithm, this approach further reduces the byte space occupancy of text data, ensuring the information's accuracy and reliability.
- (iii) Experimental results show that the proposed algorithm achieves high compression efficiency, making it especially effective to compress maritime safety information. It can serve as a feasible and effective solution for BeiDou Short Message Communication.

2. Materials and Methods

The remainder of this section is structured as follows: In Section 2.1, a concise introduction to the constructed maritime distress safety information dataset is provided. Section 2.2 provides a concise overview of the ME algorithm employed in our study. Additionally, Section 2.3 provides an overview of the research background and recent advancements in several major lossless compression algorithms.

2.1. Maritime Distress Safety Information Dataset

The Maritime Distress Safety Information refers to relevant information provided to ensure the safety of maritime navigation and operations. Maritime safety is one of the most critical considerations in maritime affairs, involving the protection of vessels, crew members, maritime facilities, and the marine environment [22]. The accuracy and timeliness of maritime safety information are crucial for navigational safety. Maritime distress safety information plays a vital role in navigational safety [23], maritime operations planning, disaster warning, and rescue, as well as maritime planning and navigation. They help relevant industries make informed decisions to protect lives and property, enhance efficiency, and improve economic benefits.

In this paper, the Maritime Distress Safety Information dataset we constructed mainly consists of weather dataset, navigation warning dataset, and disaster warning dataset. The disaster warning dataset includes storm surge warning, wave warning, sea ice warning, and tsunami warning information. Each type of maritime distress safety information corresponds to different scenarios of application.

In this paper, weather forecast information [24] specifically refers to weather forecasts for marine and near-shore areas, including river estuaries. Its primary function is to provide accurate weather information for navigation and maritime operations, aiding vessels, marine engineering, and maritime industries in making proper planning decisions. Weather forecast information can provide weather conditions and changes in marine environments in the area where vessels are located, including wind force, wave height, ocean currents, visibility, etc. Vessels can adjust their routes, reduce speed, or avoid adverse weather conditions based on forecast information to ensure the safety of the vessel and crew.

Navigation warnings [25] are notifications or announcements used to provide navigators with warning information about potential dangers, navigation obstacles, weather changes, marine conditions, or other factors that may affect navigational safety. They are mainly used to alert vessels to specific areas or conditions, prompting them to take appropriate measures to ensure the safety of the vessel and crew.

In disaster warning information, storm [26] surge is a phenomenon of abnormal rise in sea level driven by intense winds and low atmospheric pressure, common during storms or severe typhoons. Storm surge warnings provide information about the potential height, arrival time, and affected areas of storm surges. Wave warnings [27] provide information about wave height, direction, period, etc. Sea ice warnings [28] provide information about the presence, distribution, thickness, drift, etc., of sea ice. Tsunami warnings [29] provide information about possible tsunamis, including the epicenter, magnitude, and potentially affected coastal areas, allowing vessels and maritime personnel time to prepare for evacuation or avoidance, thereby preventing accidents such as vessel instability or personnel falling overboard.

Compared to weather forecast information and navigation warning information, we have access to less disaster warning information. To enhance the robustness of subsequent compression testing, we utilized large models to train the disaster warning dataset and expand its data content.

2.2. Maritime Safety Information Encoding Algorithm

The BeiDou SMC plays a crucial role in areas such as the ocean where there is no mobile signal coverage [30]. In such situations, devices equipped with BeiDou terminals can use SMC for emergency communication [31]. In the field of maritime operations, the China Shipbuilding Industry Systems Engineering Research Institute has conducted various developments in the BeiDou series systems, such as the BeiDou distress survival terminal and BeiDou-based “smart ship” communication and navigation systems and electronic customs clearance systems, enabling SMC to play a greater role. To address issues such as slow transmission rates and excessive redundancy during SMC transmission [32], byte-level encoding can be applied, which offers better compression efficiency and transmission effectiveness compared to common compression algorithms.

After analysis, it was found that the warning information in the Maritime Distress Safety Information dataset follows relatively fixed information rules. Therefore, a simple and efficient pattern-matching [33,34] method can be adopted for information extraction. Pattern matching is a basic operation in string manipulation in data structures [35], aiming to identify all occurrences of a given substring within a string. Rule-based pattern matching tends to yield satisfactory results.

Given the current state of maritime safety information, there is no widely applicable byte-level encoding algorithm. Leveraging pattern matching, we propose an ME algorithm, enabling the encoding of various maritime distress safety information types. This algorithm effectively reduces the byte space of messages and enhances the transmission capacity of BeiDou SMC.

2.3. Lossless Compression Algorithm

After encoding maritime safety information, we can further improve the compression ratio by combining efficient lossless compression algorithms [36], thereby enhancing the transmission capacity of SMC.

Arithmetic coding is a type of entropy coding employed for lossless data compression. It compresses the input file into a long real number between 0 and 1, representing all the information of the input file based on its statistical properties. This information is lossless. Arithmetic coding allows the compression ratio to approach the entropy value of the data, achieving theoretically optimal compression rates. During encoding, arithmetic coding divides the current interval into several subintervals according to the frequency of symbol occurrences. It selects the corresponding subinterval for the current symbol in

the input stream to replace the current interval and proceeds to the next round of division. This process continues until the input data stream ends. Finally, a random floating-point number within the interval is outputted, serving as the compressed encoding of the data stream information.

The encoding starts from an initial interval $[0, 1)$, with low set to 0 and high set to 1. Continuously reading characters from the original data, the algorithm finds the interval $[L, H)$ where the character belongs.

$$\begin{cases} \text{low} = \text{low} + (\text{high} - \text{low}) \times L \\ \text{high} = \text{low} + (\text{high} - \text{low}) \times H \end{cases} \quad (1)$$

Ultimately, any decimal number within the interval $[\text{low}, \text{high})$ is converted to binary form to produce the encoded data.

The PPM algorithm [37] has shown remarkable effectiveness in the field of text compression. It was initially proposed by J. Cleary and I. Witten and further developed and implemented by A. Moffat. This algorithm constructs a context index tree [38] to record the continuous relationships between compressed characters and their corresponding frequency information. Based on this information, the PPM algorithm can accurately predict the probabilities of various characters appearing next. Due to the efficiency and accuracy of probability prediction in the PPM algorithm, many researchers have been attracted to optimize the algorithm, resulting in numerous variants. These variants are all based on the PPM algorithm and aim to address the issue of handling zero-frequency context characters. Since there is a significant number of escaping characters throughout the entire compression process, the accuracy of predicting the escape character probabilities directly impacts the compression ratio of the algorithm. Therefore, different variants of the PPM algorithm have been developed to address different escape character prediction probabilities, such as PPMa [39], PPMb [40], PPMc [41], and PPMd. The emergence of these variant schemes enriches the application fields of the PPM algorithm and improves its performance and applicability in the field of data compression.

With the rise of artificial intelligence and research, it has been discovered that neural networks also have significant effects in data compression applications [42]. Existing neural network models are based on the concept of context prediction. They treat the data to be compressed as sequence data and consider these sequence data as a correlated sequence of context, meaning that characters are not independent of each other. Therefore, based on a certain length of context, there will be a prediction of the probability distribution of the next character appearing. Common neural-network-based compression techniques include CMIX [43], Deep-Zip [44], and NNCP [45].

The dual-stage compression method based on BeiDou SMC [46] aims to effectively compress maritime safety data and minimize data redundancy to achieve more efficient compression results. This approach utilizes a binary encoding method (MBE) specifically tailored for the byte space of primary-level short messages. Additionally, a compression algorithm named XH is proposed, which effectively compresses maritime safety data through hash dictionaries.

However, the above algorithms still do not achieve optimal compression for maritime safety information. To address the high redundancy and limited information length issues during BeiDou SMC transmission, this paper proposes the BPPMd algorithm tailored for maritime distress safety information.

3. Method

We encounter issues of data redundancy and message length constraints when transmitting maritime distress safety information using the BeiDou SMC system. A single piece of information often requires multiple transmissions of BeiDou short messages to be completed. To enhance the transmission capability of information, we introduce an ME algorithm in this paper. In this algorithm, we employ a pattern-matching-based method for information extraction, conducting dictionary searches for each message field to reduce

space redundancy. Additionally, we propose the BPPMd algorithm tailored for maritime distress safety information. We introduce a byte encoding module in the input phase and utilize an LSTM network to predict the current input and historical information. We use the predicted values for encoding and updating the network. This algorithm further improves compressibility while minimizing text data redundancy, with relatively minor sacrifices in compression time.

In this section, we first introduce the ME algorithm. Following that, we present the BPPMd lossless compression algorithm that we proposed, as illustrated in Figure 2.

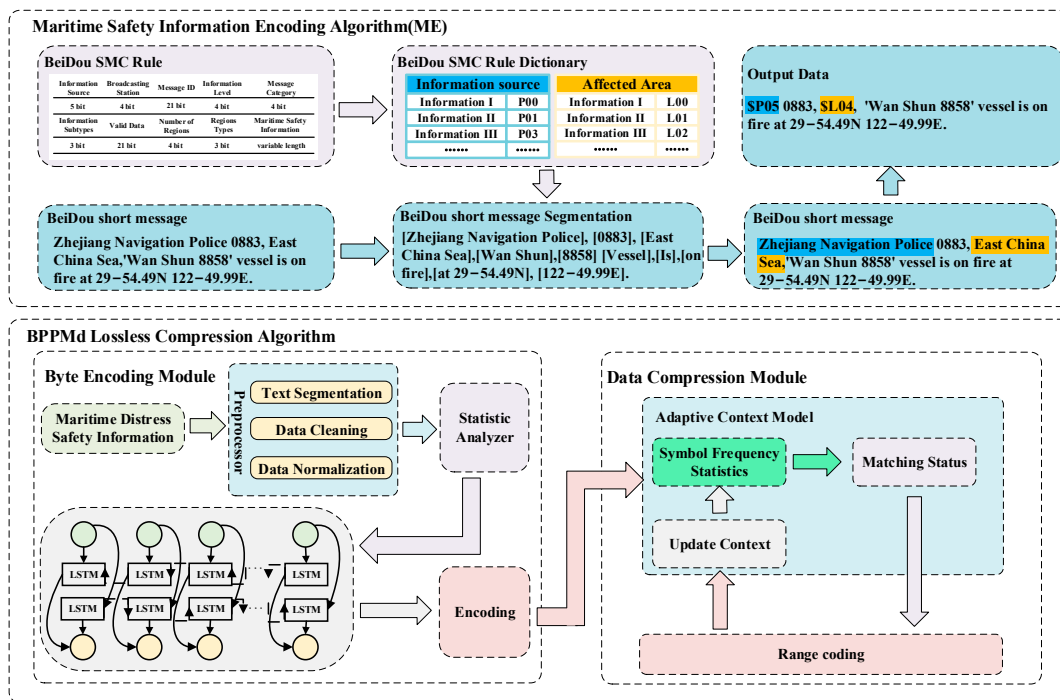


Figure 2. Diagrams of ME algorithm and BPPMd lossless compression algorithm. In the ME algorithm, the blue and orange colors represent the contents from different dictionaries in the BeiDou SMC rules. The figure illustrates how we encode the content of BeiDou short messages based on these dictionaries to reduce space redundancy.

3.1. ME Algorithm of Maritime Safety Information

After analyzing the characteristics of the Maritime Safety Information dataset, we found that the format of maritime distress safety information exhibits significant regularity. In natural language processing, for information with fixed rules, a simple and efficient pattern-matching method based on rules [47] can be adopted for information extraction.

The rule-based method refers to a technique where a large amount of text is examined, and patterns of rules existing in the text are analyzed. These patterns are systematically parsed and matched for information extraction [48]. Scholars have proposed various approaches based on rules, including a rule-based knowledge element attribute extraction method and a web information extraction method based on regular expressions [49]. Although the rule-based method has lower automation and universality, it offers higher accuracy and good flexibility for natural language text. It is straightforward to operate during the extraction process but heavily relies on formulated rules (string patterns) and is suitable for structured text. In this method, regular expressions are used to describe sets of regular expressions, composed of ordinary characters and special characters like wildcards. They are commonly employed for text content searches and can match text based on certain algorithms, facilitating the extraction of substrings from strings. Regular expressions form the foundation of rule-based pattern-matching methods for extracting text information.

For ordinary direct regex matching, if the matching rules are too loose and the boundaries are weak, there may be numerous matching results, many of which are irrelevant to the task objectives. Therefore, we use a regular matching method with a keyword triggering mechanism. The process is as follows:

- (1) For a string s , given a regular expression R_A for a keyword, obtain the match result set A .
- (2) Obtain the set F of starting positions of s in the match result set A .
- (3) Set the search range to x characters based on the task's target string length. Let the i -th element in set F be F_i . If the number of elements in F is n , then split the string s into n substrings, where each substring ranges from $[F_i, F_i + x]$. Denote the set of substrings as $S_{children}$.
- (4) Given a target regular expression R_B , use R_B to match each element in $S_{children}$ to obtain the final result set.

In summary, keyword-based range regular expression matching first locks onto a small part of the complete string based on keyword position information. Then, it performs regular expression matching within this specified range, yielding more precise results and avoiding excessive matches. The specific implementation of the ME algorithm is illustrated in Figure 3.

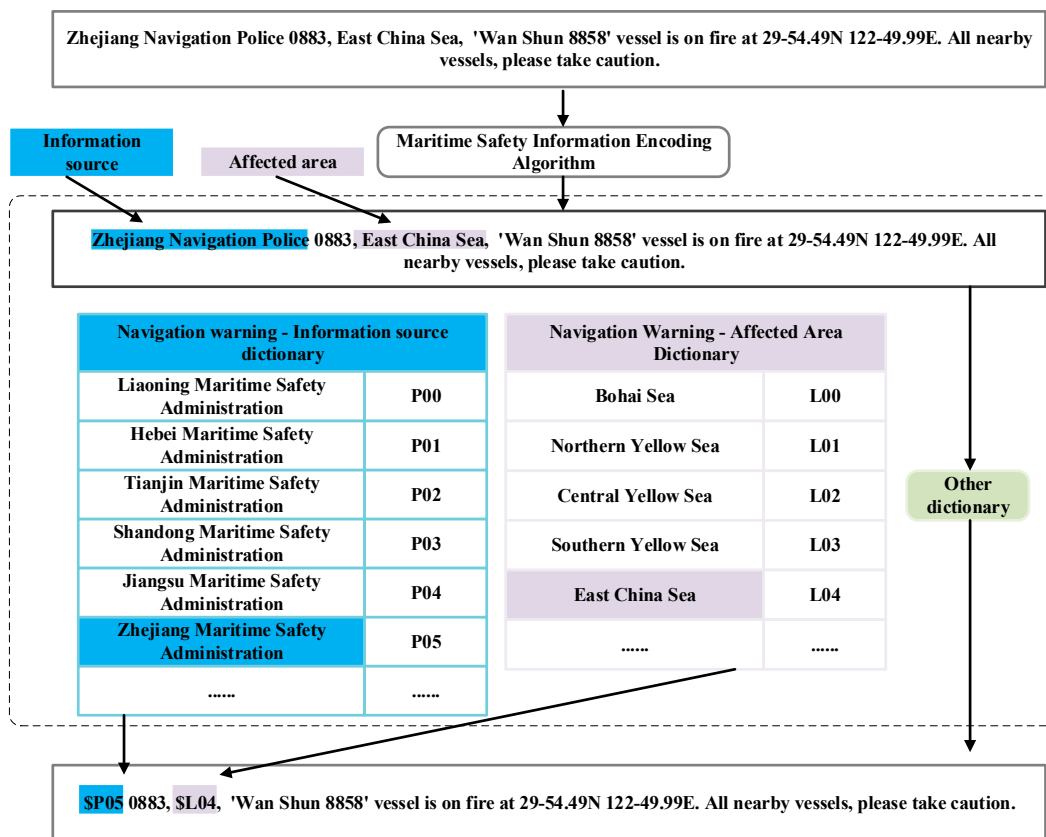


Figure 3. Diagrams of ME algorithm implementation.

Zhejiang Navigation Police is one of the issuing authorities defined by us, and “East China Sea” is one of the affected areas defined by us. Taking the issuing authority and affected area as examples, Zhejiang Navigation Police is encoded as \$P05 according to the dictionary, and “East China Sea” is encoded as \$L04. This is performed to reduce the byte space of the information and improve transmission efficiency.

3.2. BPPMd Lossless Compression Algorithm

After encoding maritime safety information with a specialized algorithm, we can further process the data using efficient lossless compression algorithms to reduce the space occupancy of textual data. The traditional LZ77 algorithm, as shown in Figure 4a, divides the entire sliding window into two regions: the dictionary area on the left and the area to be encoded on the right. The encoder searches in the dictionary area until it finds a matching string. However, the LZ77 algorithm performs poorly in compressing maritime distress safety information. The PPMd algorithm, which has shown excellent results in text compression, is depicted in Figure 4b. It constructs a complex context indexing tree, recording the order and frequency information between the compressed data streams. It predicts the probability of each character in the data stream to be compressed based on the stored information and finally encodes the predicted probability values using an interval encoder. The PPMd algorithm has already achieved good results on the maritime distress safety information dataset, but there are still aspects that can be improved in multimodel fusion. Further research on this could lead to even better results.

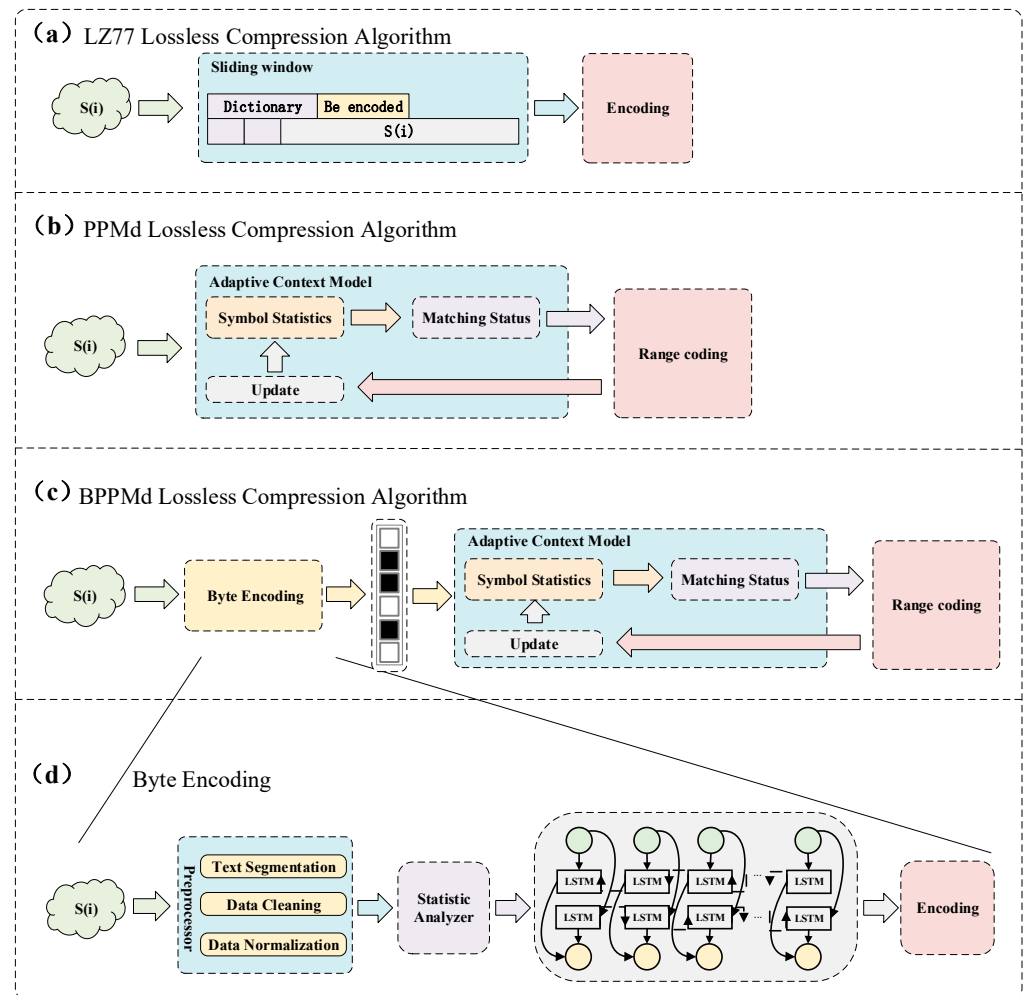


Figure 4. Diagrams: (a) is the LZ77 lossless compression algorithm, (b) is the PPMd lossless compression algorithm, (c) is the proposed BPPMd lossless compression algorithm, and (d) represents the byte encoding module. BPPMd integrates the byte encoding module into the design space and implementation of the PPMd algorithm. It enhances semantic feature comprehension, reduces redundancy, preserves critical structural features, and yields a more compact representation. Consequently, it effectively reduces the byte space occupancy of textual data. S(i) refers to the input text. In Figure (a), S(i) is encoded from right to left, forming a dictionary based on the input content and encoding it accordingly.

LSTM is a form of recurrent neural network designed to capture long-term dependencies. This makes it adept at retaining contextual information and understanding relationships between words when processing text data, rather than merely handling each word individually. By doing so, we can achieve a more nuanced representation of the text data, which is derived from learning the data itself. This representation is typically more compact than the original text data because it incorporates semantic understanding and feature extraction, removing some redundant information while preserving essential semantic and structural features. Once we have obtained this more compact and expressive representation, we can apply traditional compression algorithms to it. Since this representation has already been processed by the LSTM model, traditional compression algorithms can usually handle it more effectively, thereby achieving better compression of the text data.

Based on LSTM's text processing capabilities, we incorporate it as a byte encoding module into the PPMd algorithm. As shown in Figure 4c, our method is called BPPMd. It aims to further enhance compression performance while minimizing the sacrifice of compression speed. The byte encoding module, illustrated in Figure 4d, comprises a preprocessor, statistical analyzer, predictor, and encoder.

The preprocessor is used to perform preprocessing tasks such as tokenization, data cleaning, and normalization on the input data to prepare it for statistical analysis. Tokenization, as mentioned by us, involves reading input data byte by byte and further segmenting it into bits. Data cleaning entails removing unnecessary characters based on data characteristics or handling padding bytes at the end of files. Normalization involves standardizing the data to facilitate more effective processing in subsequent steps. The statistical analyzer then performs statistical analysis on the preprocessed data to determine the frequency characteristics and distribution of words in the data. Based on the results of this statistical analysis, a vocabulary is constructed. This vocabulary provides the predictor with all possible input symbols or data values needed for prediction, helping the predictor better understand the features and patterns of the input data, thereby improving the accuracy of predictions and the efficiency of encoding.

The predictor includes a prediction method and an update method. The predictor first updates an accumulated bit pattern based on the input bits. If the number of bits exceeds a set threshold, it invokes the prediction method. This method involves iterating through the neuron layers and performing forward propagation on each layer. The output of the current layer is copied to the input buffer of the next layer. Next, it computes the output value of each neuron in the output layer. The raw output values of the output layer are then exponentiated and normalized to represent a probability distribution.

The update method is then called, which involves performing backpropagation training for each neuron layer at each time step. It calculates and updates the output layer error for the current time step and updates the network weights based on the errors of the output layer and the hidden layers. Following this, the probability values of each possible byte are updated based on the vocabulary and the predictor's output. For each byte in the vocabulary, if it is part of a word, the corresponding probability value is extracted from the predictor's output.

The flowchart for the BPPMd algorithm is illustrated in Figure 5.

In the byte encoding stage, this byte encoding module begins by preparing input data through preprocessing steps such as text segmentation and data cleaning. It constructs a vocabulary based on the characteristics of the data, which initializes a predictor trained on symbol frequencies from the input data to generate a compression model. During the compression phase, the input data are read byte by byte and converted into a bitstream. The encoder uses the predictor's output and encoding algorithms to encode each bit, resulting in a more compact and expressive representation of the text data after semantic understanding and feature extraction. This facilitates efficient data compression when combined with context models and range coders. The encoded data are written to a file via an output stream, while encoding rates and other statistical information are recorded. The entire process integrates LSTM networks for prediction and encoding, ensuring effective

compression and decompression of data. Post byte encoding module, the input data are more compact compared to the original text data, removing redundant information and effectively reducing the byte space occupancy of the text data.

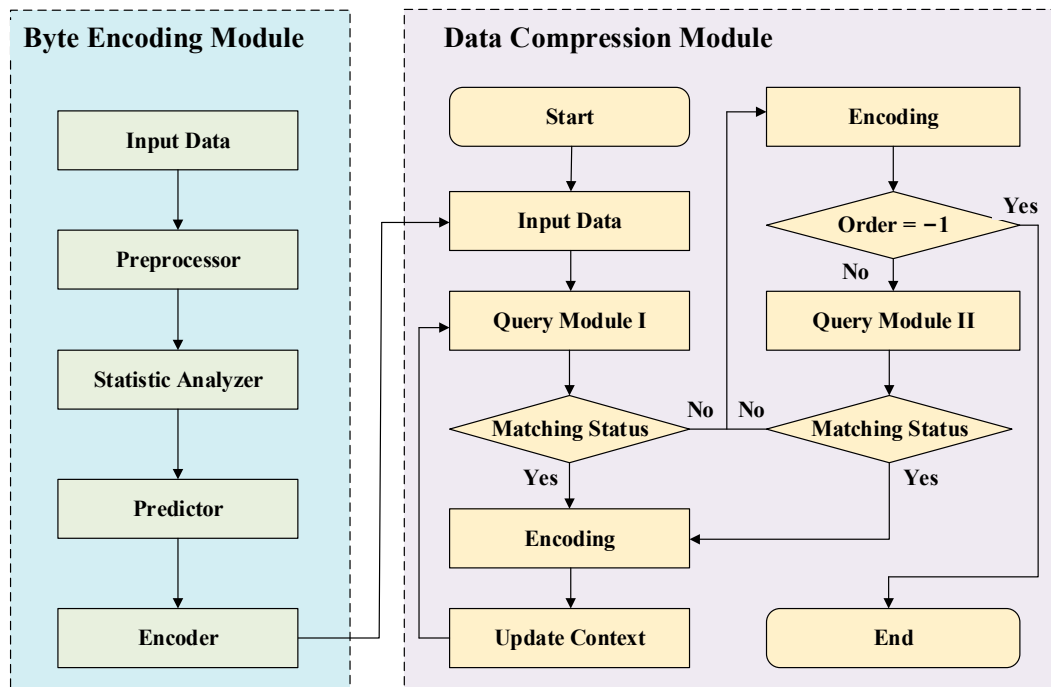


Figure 5. Flowchart of BPPMd algorithm.

During the processing in the data compression stage, when the data enter the compression module, they first go through the query module (I) for character querying. If the character to be compressed is matched at the current level, the match is successful. The character is then encoded based on the frequency information of the matched character, and the context tree is updated before continuing to compress the next byte. If the match is unsuccessful, an escape occurs at the current level. The escape character is encoded based on its frequency, and the process falls back to the previous level using query module (II) to continue matching operations. When the current order is -1 , indicating an escape at the top level, the compression process ends. The difference between query module I and query module II lies in the way escape characters are predicted. The former uses a fixed escape character frequency (frequency of 1), while the latter uses secondary escape estimation (SEE) to predict the probability of escape characters.

The escape module accurately predicts the probability of escape characters by establishing an escape context model. The context model established by the escape module is simpler than the context model and prediction module established by the PPM algorithm. Additionally, the prediction process of the escape module is straightforward, directly producing prediction results by only querying one layer of context information. The escape module utilizes various contextual information, including the total frequency information of the current context, the number of characters, and the order of the context level, as well as the parent context information and child context information, to more accurately predict the probability of escape characters.

The prediction update module passes the predicted probability information to the encoder module. The encoder utilizes range coding algorithm to encode the probabilities into corresponding bitstream information, which is then outputted through the output interface. During the prediction process of the PPMd algorithm, two types of prediction information are generated: probability information for binary contexts (having only one

successor character) and probability information for multisymbol contexts (having more than one successor character).

When selecting the binary model, the algorithm requires only one multiplication operation, while selecting the multisymbol encoder requires one division operation and two multiplication operations. These calculations narrow down the range of probability mapping. When the range of the interval becomes sufficiently small, the algorithm needs to perform interval expansion operations, enlarging the range of the interval by 256 times, while outputting the upper 8 bits of the original boundary data through the output buffer module. The flowchart of the range coding algorithm is shown in Figure 6.

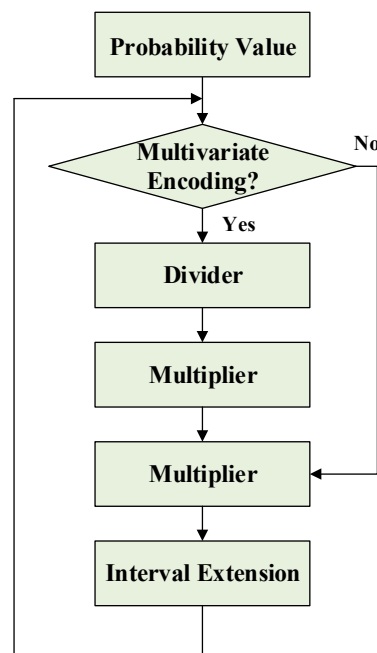


Figure 6. Flowchart of range coding algorithm.

4. Experiment and Result

4.1. Data Source and Evaluation Metrics

The data used in the experiments of this paper are all sourced from multiple maritime institutions, including the Maritime Safety Administration of the People’s Republic of China and the China Oceanic Information Network. With authorization from the China Maritime Safety Administration, we obtained relevant maritime safety information data. We organized, categorized, and augmented these data into six distinct datasets, which now serve as benchmark datasets for compressing and transmitting maritime safety information. The datasets can be categorized into weather forecast dataset, navigation warning dataset, storm warning dataset, wave warning dataset, sea ice warning dataset, and tsunami warning dataset. Table 1 below reflects the characteristics of the data in the datasets.

Table 1. The data characteristics of the datasets.

Dataset	Amount	Data Size/Byte	Average Size/Byte
Weather	4253	714–1929	1962
Navigation	6000	32–2194	378
Storm	1000	371–3490	1184
Wave	1501	174–3220	907
Sea ice	206	1577–2246	1883
Tsunami	180	92–885	467

Regarding the data characteristics in Table 1, we conducted a statistical analysis of the distribution of file sizes. The specific distribution of file sizes is illustrated in Figure 7.

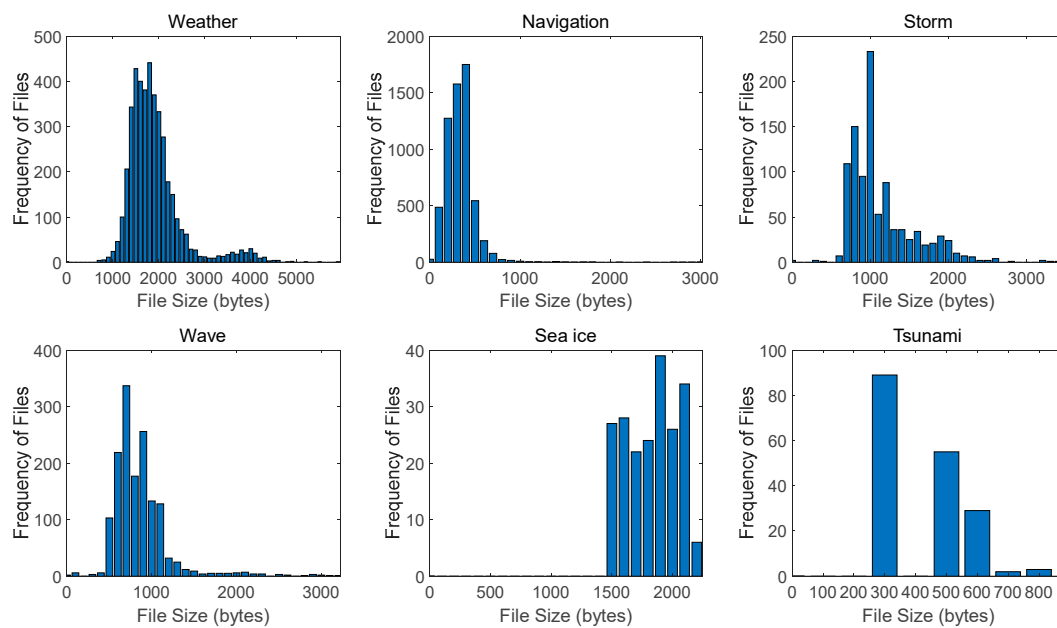


Figure 7. Distribution of file sizes for different datasets.

The weather forecast dataset plays a crucial role in maritime safety. Weather forecast information provides weather conditions and changes in the marine environment in the area where ships are located, including wind force, wave height, ocean currents, visibility, etc. This information is essential for the safe navigation of ships. Ships can adjust their routes, reduce speed, or avoid adverse weather conditions based on forecast information to ensure the safety of the vessel and crew. Depending on the geographical area, the forecasts are divided into three main parts: Northern Sea Area, Eastern Sea Area, and Southern Sea Area.

The data in the navigation warning dataset are primarily used to alert ships to specific areas or conditions, prompting them to take appropriate measures to ensure the safety of the vessel and crew. Through navigation warnings, ships and crew members can obtain critical navigation information to make informed decisions and take necessary actions to ensure safe navigation. Navigation warnings are typically issued and updated by national maritime authorities, marine meteorological agencies, port authorities, and international organizations to ensure the comprehensiveness and authority of the warnings.

Storm surge warnings are crucial for residents, fishermen, and vessels in coastal areas. They allow for emergency measures to prevent floods and coastal erosion caused by storm surges, ensuring the safety of lives and property. Wave warnings assist vessels and maritime operations in planning routes, selecting appropriate speeds, and determining sailing times. Additionally, wave forecasts and warnings are essential for the safety of offshore engineering, fisheries, and marine recreational activities. For vessels and maritime personnel, understanding and adhering to wave forecasts and warnings help mitigate risks, reducing the occurrence of accidents such as vessel instability or crew falling overboard.

Sea ice warnings are essential for vessel navigation, maritime safety, and planning and decision making for offshore operations. Vessels can adjust routes, avoid ice areas, or take anti-icing measures based on sea ice forecasts and warnings to reduce the risk of hull damage and collision with icebergs.

Tsunami warnings are critical for residents, fishermen, and vessels in coastal areas. They provide time for preparation and evacuation, minimizing the harm to lives and property caused by tsunamis.

When performing experiments with these datasets, practical application requirements are taken into account, and the proposed algorithm needs to be evaluated based on compression performance, reliability, and other relevant factors.

The compression ratio is defined as the ratio of the size of uncompressed data to the size of compressed data. It is generally regarded as the most critical metric for assessing compression effectiveness. The formulas for calculating the compression ratio and the average compression ratio are provided below:

$$CR = \frac{\lambda}{\eta} \quad (2)$$

$$mCR = \frac{\sum_{j=1}^M CR_j}{M} \quad (3)$$

where λ denotes the memory space occupied by the original data, η denotes the memory space occupied by the compressed data, and M represents the number of datasets of a specific type. CR is the compression ratio of the data. mCR is the average compression ratio of the data.

Unbiased standard deviation is used to estimate the dispersion of sample data and provides a more accurate estimate of the population standard deviation by adjusting the denominator to $n - 1$. The standard error measures the dispersion of sample means, i.e., the degree of fluctuation in sample means if multiple samples are drawn from the same population. It reflects the precision of the sample mean relative to the population mean. The calculation of standard error is usually based on sample data and is inversely proportional to the sample size. Both the unbiased standard deviation and standard error are statistical measures describing the data distribution and are often considered important indicators for assessing the stability and reliability of compression ratios. The formulas for standard deviation and standard error are as follows:

$$S = \sqrt{\frac{\sum_{i=1}^N (x_i - \bar{x})^2}{N - 1}} \quad (4)$$

$$SE = \frac{S}{\sqrt{N}} \quad (5)$$

where N is the size of the dataset, x_i is the value of the i -th data point, \bar{x} is the mean of the dataset, S is the standard deviation, and SE is the standard error.

4.2. Experiments and Analysis

The experiments were performed on an Ubuntu system, with Python and C++ as the programming languages. The integrated development environments (IDEs) used were PyCharm and Visual Studio Code. Python was used for byte-level encoding of datasets, while C++ was used as the compilation language for algorithms. The experiments were executed on a system equipped with an Intel i7-9900 processor and 32 GB of RAM. In the subsequent experiments, each observation was repeated five times to mitigate potential sources of randomness in the experimental results. To ensure an objective presentation of the experimental results, we used statistical analysis methods to analyze the data.

Due to the relatively small volume of content in the Maritime Distress Safety Information dataset, the average size of the tsunami warning dataset is only 467 bytes, while the average size of the navigation warning dataset is even smaller at only 378 bytes. Therefore, given its characteristics of small text, we chose the well-known Calgary Corpus as the public dataset for testing in Table 2. The Calgary Corpus, commonly used for data compression, has smaller content volume and is more suitable for small-text characteristics. During testing, we did not choose algorithms based on deep learning such as cmix or Dzip, as they require longer processing times for file compression, which does not meet

our requirements. Instead, we selected several widely used algorithms for comparative testing. To demonstrate specific compression effects, each algorithm was configured for exceptional compression ratios: lz4-9 [50], gzip-best [51], xz-9-e [52], zstd-ultra-22 [53], and Brotli-q11 [54]. Comparative results suggest that BPPMd, xz, and Brotli demonstrate outstanding compression performance. Moreover, in all instances, BPPMd outperforms LZ4, gzip, zstd, xz, PPM, and PPMd in terms of performance. In some cases, BPPMd’s compression efficiency is lower than xz and Brotli.

Table 2. Different compression algorithms’ compression ratios on the Calgary Corpus. Best results in bold.

Dataset	LZ4	gzip	xz	zstd	Brotli	x3 [55]	PPM	PPMd	BPPMd
bib	2.7963	3.1880	3.6355	3.4847	3.9204	3.0854	3.2659	4.6314	5.1270
book1	2.1390	2.4618	2.9412	2.9079	2.9992	2.9166	3.0670	3.6618	3.8769
book2	2.5947	2.9630	3.5961	3.5134	3.6963	3.2186	3.2401	4.3786	4.688
geo	1.1956	1.4968	1.9260	1.5823	1.9352	1.6264	1.1062	1.8532	1.8541
news	2.2822	2.6116	3.1714	3.0735	3.3382	2.6673	2.6003	3.6545	3.9227
obj1	1.7384	2.0837	2.2741	2.1812	2.3021	1.8583	1.4920	2.2828	2.2833
obj2	2.5497	3.0438	4.0161	3.5113	3.7853	2.7231	2.3402	3.6928	3.7032
paper1	2.3062	2.8669	3.0743	3.0173	3.4393	2.6116	2.7153	3.6521	4.1051
paper2	2.2962	2.7707	3.0149	2.9829	3.3078	2.6975	2.8691	3.6814	4.1542
pic	7.7471	9.7978	12.8754	11.825	12.536	8.9855	7.9067	10.536	10.570
progc	2.3057	2.9870	3.1507	3.0960	3.4092	2.5405	2.5939	3.6400	3.8622
progl	3.4812	4.4324	4.7816	4.7285	5.1165	3.4485	3.5737	5.5565	5.9158
progp	3.4642	4.4144	4.7718	4.7244	4.9979	3.4310	3.6143	5.4641	5.6797
trans	4.0755	4.9674	5.6131	5.4165	6.0833	3.6220	3.8776	6.5143	6.8450

From the average compression ratio results in Table 2, the BPPMd algorithm outperforms baseline algorithms in most cases, except for experiments on the pic file where it falls short of xz, and on the geo, obj1, and obj2 files where it falls short of Brotli. The BPPMd algorithm constructs more context trees using the context model, incurring additional space costs, which lead to higher compression ratios. Analyzing the results in Table 3, BPPMd underperforms compared to Brotli only for the navigation warning dataset. As shown in Table 1 this dataset has significantly smaller memory sizes and contains very few repetitive fields, resulting in poor compression performance by BPPMd for this type of data.

Table 3. Results of experiments for Maritime Distress Safety Information dataset. Best results in bold.

Dataset	Acc	x3	xz	Brotli	PPMd	BPPMd
Weather	1.5601	1.9991	2.6787	3.4716	3.3271	4.0009
Navigation	1.0298	1.0004	1.1966	1.7552	1.6006	1.7033
Storm	1.5349	1.4549	1.741	2.6153	2.1825	2.8967
Wave	1.5129	1.3623	1.6383	2.5909	2.0694	2.9293
Sea ice	1.5975	1.8291	2.3396	3.3016	2.9839	3.7429
Tsunami	1.3511	1.0429	1.185	2.0559	1.5868	2.1587

Table 3 shows the evaluation results for our Maritime Distress Safety Information dataset, utilizing the compression ratio as the metric for assessment. We chose adaptive arithmetic compression coder (Acc), x3, xz, Brotli, PPM, and PPMd as reference algorithms for our comparative experiments. Acc is a versatile lossless compression algorithm, while x3 is an efficient dictionary-based compression algorithm, and xz, Brotli, and PPMd are algorithms known for their effectiveness on smaller public datasets. The results in Table 3 demonstrate that the BPPMd algorithm achieved excellent compression ratios across all six datasets in the experiment. It outperformed other algorithms significantly on datasets including waves, sea ice, storms, tsunamis, and meteorology, with performance slightly

lower than Brotli only on the navigation warning dataset. The proposed algorithm exhibits superior compression performance compared to benchmark algorithms.

We performed significance tests on various algorithms across different datasets, conducting three tests for each dataset. Table 4 presents the unbiased standard deviation (S) and standard error (SE) of compression rates for various algorithms across these datasets. Figure 8 provides a visual comparison of the compression rates and standard errors of different algorithms, where asterisks (*) indicate statistical significance compared to other algorithms.

Table 4. The standard deviation and standard error of compression rates in the Maritime Distress Safety Information dataset.

	Acc		x3		xz		Brotli		PPMd		BPPMd	
	S	SE	S	SE	S	SE	S	SE	S	SE	S	SE
Weather	0.0142	0.0002	0.0528	0.0008	0.0496	0.0008	0.0803	0.0012	0.039	0.0006	0.0395	0.0006
Navigation	0.0387	0.0005	0.0322	0.0022	0.1834	0.0024	0.088	0.0011	0.0869	0.0011	0.1068	0.0014
Storm	0.0348	0.0026	0.0654	0.0021	0.0796	0.0025	0.0617	0.002	0.0546	0.0017	0.0377	0.0012
Wave	0.0228	0.0006	0.0836	0.0022	0.0995	0.0026	0.1295	0.0033	0.0652	0.0017	0.0531	0.0014
Sea ice	0.0055	0.0004	0.0322	0.0022	0.0299	0.0021	0.0165	0.0011	0.0234	0.0016	0.017	0.0012
Tsunami	0.0348	0.0026	0.1199	0.0089	0.1105	0.0082	0.0499	0.0037	0.0519	0.0039	0.1401	0.0104

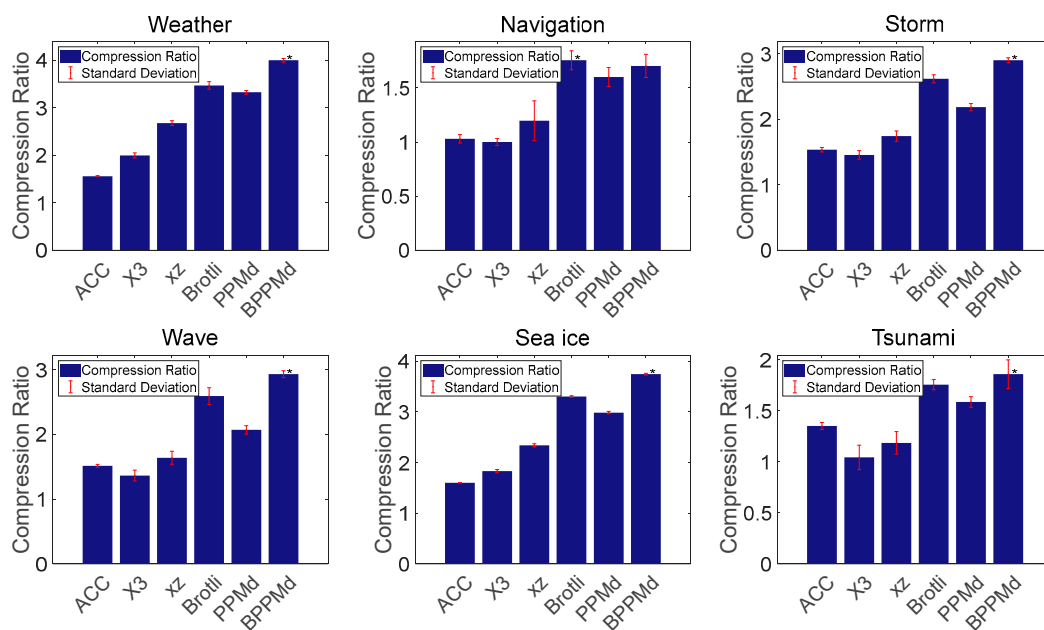


Figure 8. Comparison chart of standard errors in compression rates across different algorithms. The asterisk (*) denotes statistical significance compared to other algorithms.

By analyzing the average compression time (t_c) and decompression time (t_d) in Table 5, we can observe that the compression and decompression times of the BPPMd algorithm are higher than those of the comparison algorithms. When comparing the compression ratios of different datasets in Table 3, it can be seen that BPPMd sacrifices compression time to improve compression performance, but the overall compression time remains within an acceptable range.

The compression results obtained using our proposed encoding scheme are showcased in Table 6. Bar charts comparing the average compression ratios of diverse algorithms with and without ME on different datasets are shown in Figure 9. The experiments indicate that the average compression ratio of all algorithms improves after encoding. The BPPMd algorithm achieved the highest compression ratio on all datasets except for the navigation warning dataset, demonstrating excellent compression performance. In summary, our

proposed algorithm has demonstrated significant effectiveness in compressing maritime distress safety information datasets.

Table 5. The average compression time and average decompression time for the Maritime Distress Safety Information dataset.

	Acc		x3		xz		Brotli		PPMd		BPPMd	
	t_c	t_d	t_c	t_d	t_c	t_d	t_c	t_d	t_c	t_d	t_c	t_d
Weather	0.0218	0.0225	0.0447	0.0192	0.109	0.1031	0.0122	0.0121	0.0168	0.016	0.2061	0.1523
Navigation	0.0238	0.0193	0.0223	0.0156	0.1056	0.1031	0.0107	0.0107	0.0162	0.0155	0.1874	0.0913
Storm	0.0229	0.0202	0.0365	0.0178	0.1084	0.0304	0.0115	0.0114	0.0167	0.0154	0.2183	0.1248
Wave	0.0194	0.0196	0.031	0.0166	0.1067	0.1026	0.0112	0.0112	0.0166	0.0155	0.2156	0.1221
Sea ice	0.0209	0.0209	0.0431	0.0194	0.1087	0.1026	0.012	0.012	0.0186	0.0167	0.2133	0.1393
Tsunami	0.0182	0.0186	0.0231	0.0158	0.1049	0.1033	0.0109	0.0109	0.0164	0.0154	0.2065	0.1352

Table 6. Results of experiments for Maritime Distress Safety Information dataset with MSE. Best results in bold.

Dataset	Acc	x3	xz	Brotli	PPMd	BPPMd
Weather	1.6835	2.1602	2.7743	3.5885	3.3271	4.0202
Navigation	1.3901	1.0596	1.1988	1.8624	1.6006	1.744
Storm	1.8835	1.7697	2.0491	3.0819	2.1825	3.3594
Wave	1.6763	1.5064	1.7489	3.0319	2.0694	3.0516
Sea ice	1.8207	2.1153	2.6728	3.8417	2.9839	4.3965
Tsunami	1.4012	1.0826	1.1945	1.9879	1.5868	2.9677

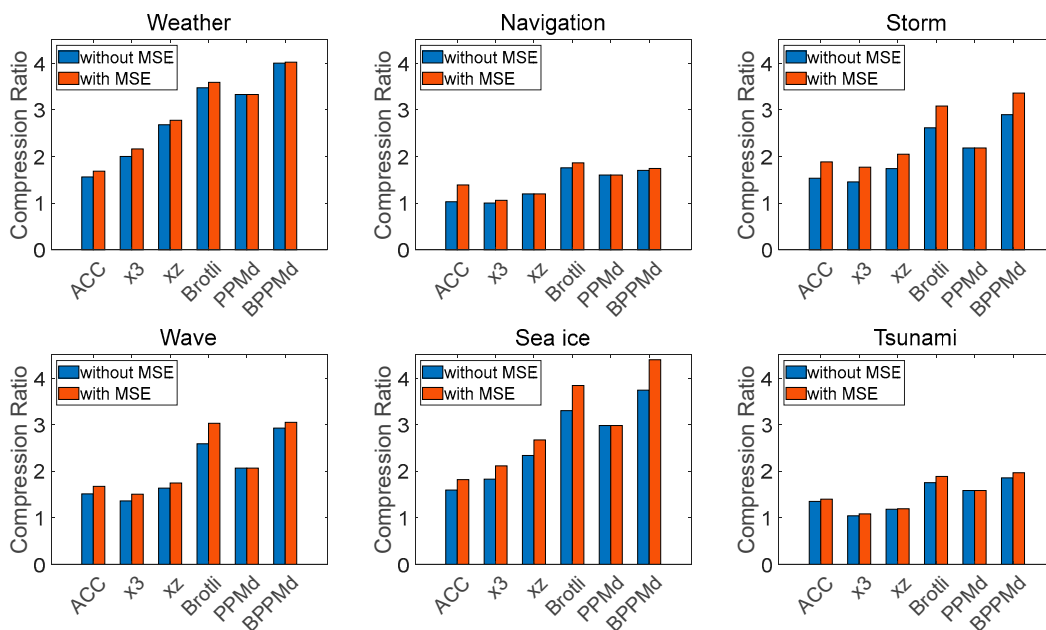


Figure 9. The comparison chart of compression ratios across different datasets, with and without ME.

5. Conclusions and Discussion

In this paper, we have made significant progress in the field of maritime safety information transmission and compression. Firstly, we systematically constructed the maritime distress safety information dataset and augmented it using large models to enhance the robustness of our experiments, providing strong support for research in maritime safety information. Secondly, we proposed and thoroughly evaluated the BPPMd algorithm, and validated its effectiveness by utilizing the maritime safety information coding algorithm to further compress it. The main objectives of the maritime safety information coding

algorithm and the BPPMd algorithm are to address the issues of low transmission rates, excessive redundancy, and low success rates when using SMC to transmit long warning messages. In the maritime safety information coding algorithm, we employed a pattern-matching approach for information processing. In the BPPMd algorithm, we introduced a byte encoding module that predicts based on the current input bit information and historical information, effectively reducing the byte space occupancy of textual data.

Finally, our experimental results demonstrate that the proposed algorithm exhibits notable compression performance, making it particularly well suited for compressing maritime safety information. By applying this algorithm to compress maritime safety information transmitted via the BeiDou SMC, we have successfully enhanced compression substantially reduced information redundancy during message transmission. This has ultimately led to improved overall transmission efficiency of maritime safety information.

Given the textual characteristics of the maritime distress safety information dataset, the algorithms are highly suitable for scenarios with limited text information, such as costly information transmission for ship search and rescue information exchange and satellite communication. Further research in these areas holds promising opportunities for exploring new domains and avenues. Future research directions will focus on further improving the predictive performance and efficiency of predictors and enhancing the compression performance of adaptive models.

Author Contributions: Conceptualization, Y.G. and J.H.; methodology, Y.G. and G.Z.; validation, H.L. and Q.J.; investigation, Y.G.; resources, H.L. and J.H.; writing—original draft preparation, Y.G.; writing—review and editing, J.H. and Q.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work is fully supported by the National Key Technologies R&D Program of China (Grant No. 2021YFB3901503).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are openly available in Calgary Corpus compression corpus at <http://www.data-compression.info/Corpora/CalgaryCorpus/>. Accessed on 2 April 2024.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, M.; Chai, H. Real-time marine PPP-B2b/SINS integrated navigation based on BDS-3. *Meas. Sci. Technol.* **2023**, *34*, 105113. [[CrossRef](#)]
2. Ji, S.; Sun, Z.; Weng, D.; Chen, W.; Wang, Z.; He, K. High-precision Ocean navigation with single set of BeiDou short-message device. *J. Geod.* **2019**, *93*, 1589–1602. [[CrossRef](#)]
3. Zhang, P.; Tu, R.; Zhang, R.; Gao, Y.; Cai, H. Combining GPS, BeiDou, and Galileo satellite systems for time and frequency transfer based on carrier phase observations. *Remote Sens.* **2018**, *10*, 324. [[CrossRef](#)]
4. He, K.; Weng, D.; Ji, S.; Wang, Z.; Chen, W.; Lu, Y. Ocean real-time precise point positioning with the BeiDou short-message service. *Remote Sens.* **2020**, *12*, 4167. [[CrossRef](#)]
5. Li, G.; Guo, S.; Lv, J.; Zhao, K.; He, Z. Introduction to global short message communication service of BeiDou-3 navigation satellite system. *Adv. Space Res.* **2021**, *67*, 1701–1708. [[CrossRef](#)]
6. He, X.; He, L. Beidou Short Message Communication Encryption Scheme with Improved SM4 Algorithm. In Proceedings of the 2023 4th International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT), Nanjing, China, 16–18 June 2023; pp. 461–466.
7. Li, G.; Yu, X.; Lu, W. Space-earth integrated high-precision positioning system based on 5G and Beidou navigation satellite system. In Proceedings of the 2022 7th Asia Conference on Power and Electrical Engineering (ACPEE), Hangzhou, China, 15–17 April 2022; pp. 649–653.
8. Wang, M.; Yang, W.; Xu, L.; Lv, X.; Chen, Y.; Wu, Q.; Liu, B. Covert wireless communication on beidou short message communication. In Proceedings of the China Satellite Navigation Conference, Beijing, China, 22–25 May 2022; Springer Nature: Singapore, 2022; pp. 310–320.

9. Li, G.; Li, D.; Xiong, Y.; Zhong, X.; Shi, J.; Zhang, H.; Song, D.; Yang, F.; Kang, Z.; Wu, X.; et al. Dynamic valuation of the provisioning services of marine fisheries ecosystem based on BeiDou VMS data: A case study of TACs project for *Acetes chinensis* in the Yellow Sea. *Ocean Coast. Manag.* **2023**, *243*, 106773. [[CrossRef](#)]
10. Cheng, J.; Liu, W.; Zhang, X.; Wang, F.; Li, Z.; Tang, C.; Pan, J.; Chang, Z. On-board validation of BDS-3 autonomous navigation using inter-satellite link observations. *J. Geod.* **2023**, *97*, 71. [[CrossRef](#)]
11. Chunfang, W.; Yongtao, C.; Chunlai, L.; Kejian, J. Technology and implementation of warning information distribution based on Beidou satellite. *J. Appl. Meteorol. Sci.* **2014**, *25*, 375–384.
12. Yong, Z.; Suting, C.; Yan, Z. The data transmission and management system of automatic meteorological station based on Beidou satellite. *Electron. Technol. Appl.* **2014**, *40*, 21–23.
13. Li, B.; Zhang, Z.; Zang, N.; Wang, S. High-precision GNSS ocean positioning with BeiDou short-message communication. *J. Geod.* **2019**, *93*, 125–139. [[CrossRef](#)]
14. Wang, W.; Chi, T.; Wu, Q.; Cheng, W.; Deng, Z.; Zhang, F.; Lv, C.; Song, L. On Beidou's short message service-based data transmission solution. *J. Comput. Theor. Nanosci.* **2015**, *12*, 2556–2565. [[CrossRef](#)]
15. Wang, J.; Yang, H.; Wang, Y. Research on information compression method based on Beidou short message. In Proceedings of the 3rd International Conference on Intelligent Information Processing, Guilin, China, 19–20 May 2018; pp. 5–10.
16. David, S. *Data Compression: The Complete Reference*; Springer: Berlin/Heidelberg, Germany, 2004.
17. Ziv, J.; Lempel, A. A universal algorithm for sequential data compression. *IEEE Trans. Inf. Theory* **1977**, *23*, 337–343. [[CrossRef](#)]
18. Leavline, E.J.; Singh, D. Hardware implementation of LZMA data compression algorithm. *Int. J. Appl. Inf. Syst. (IJAIS)* **2013**, *5*, 51–56.
19. Moffat, A. Huffman coding. *ACM Comput. Surv. (CSUR)* **2019**, *52*, 1–35. [[CrossRef](#)]
20. Witten, I.H.; Neal, R.M.; Cleary, J.G. Arithmetic coding for data compression. *Commun. ACM* **1987**, *30*, 520–540. [[CrossRef](#)]
21. Cleary, J.; Witten, I. Data compression using adaptive coding and partial string matching. *IEEE Trans. Commun.* **1984**, *32*, 396–402. [[CrossRef](#)]
22. Yang, F.; Li, G.; Zhang, J.; Sun, Z.; Zhang, R.; Zhao, L. Ocean decimeter-level real-time BDS precise point positioning based on short message communication. *GPS Solut.* **2024**, *28*, 39. [[CrossRef](#)]
23. Guo, S.; Li, G.; Zheng, J.; Ren, Q.; Wu, Y.; Shen, G.; Yue, H. Integrated navigation and communication service for LEO satellites based on BDS-3 global short message communication. *IEEE Access* **2023**, *11*, 6623–6631. [[CrossRef](#)]
24. Standardization Administration of China. Short-Range Weather Forecast. 2020. Available online: <https://openstd.samr.gov.cn/bz/gk/gb/newGbInfo?hcno=4741FC6129DC79A7428953C93DF0E7E2> (accessed on 1 April 2018).
25. Standardization Administration of China. The Standard Format of Navigational Warnings in People's Republic of China. 2017. Available online: <https://openstd.samr.gov.cn/bz/gk/gb/newGbInfo?hcno=FC518743F822BDA7A17E6D7463CD7DDD> (accessed on 1 June 2021).
26. Standardization Administration of China. The Issue of Marine Forecasts and Warnings—Part 1: The Issue of Storm Surge Warnings. 2020. Available online: <https://openstd.samr.gov.cn/bz/gk/gb/newGbInfo?hcno=7269707020AB3D3BD3020C7DCDBEE34B> (accessed on 1 October 2017).
27. Standardization Administration of China. The Issue of Marine Forecasts and Warnings—Part 2: The Issue of Wave Forecasts and Warnings. 2020. Available online: <https://openstd.samr.gov.cn/bz/gk/gb/newGbInfo?hcno=EE59EBA318CBC45A89C309C20A495D6D> (accessed on 1 October 2017).
28. Standardization Administration of China. The Issue of Marine Forecasts and Warnings—Part 3: The Issue of Sea Ice Forecasts and Warnings. 2017. Available online: <https://openstd.samr.gov.cn/bz/gk/gb/newGbInfo?hcno=58C1DBF73F5289C9F8E42EA31A4B7ECD> (accessed on 1 October 2017).
29. Standardization Administration of China. Grades of Tsunami. 2020. Available online: <https://openstd.samr.gov.cn/bz/gk/gb/newGbInfo?hcno=2F31F2C114C93BB22E337BEBD542A07E> (accessed on 1 June 2021).
30. Li, X.; Guo, R.; Chen, J.; Liu, S.; Chang, Z.; Xin, J.; Guo, J.; Tian, Y. New Orbit Determination Method for GEO Satellites Based on BeiDou Short-Message Communication Ranging. *Remote Sens.* **2022**, *14*, 4602. [[CrossRef](#)]
31. Han, Z.; Liang, M.; Wu, Y.; Ma, Y.; Li, X. Research on error correction of state data transmission system of moving carrier based on Beidou short message. In Proceedings of the 2020 5th International Conference on Electromechanical Control Technology and Transportation (ICECTT), Nanchang, China, 15–17 May 2020; pp. 184–187.
32. Zhang, C.; Zeng, J. An Attention-Averaging-Based Compression Algorithm for Real-Time Transmission of Ship Data via Beidou Navigation System. *J. Mar. Sci. Eng.* **2024**, *12*, 300. [[CrossRef](#)]
33. Sherkat, E.; Farhoodi, M.; Yari, A. A new approach for multi-pattern string matching in large text corpora. In Proceedings of the 7th International Symposium on Telecommunications (IST'2014), Tehran, Iran, 9–11 September 2014; pp. 72–77.
34. Zhou, Y.; Guanqi, D. Research of a Pattern Matching Algorithm Based on Statistical Eigenvalues. In Proceedings of the 2018 5th International Conference on Information Science and Control Engineering (ICISCE), Zhengzhou, Henan, 20–22 July 2018; pp. 431–435.
35. Zhou, Y.; Ding, G. Research of a pattern matching algorithm based on threshold and word frequency. In Proceedings of the 2018 IEEE international conference on computer and communication engineering technology (CCET), Beijing, China, 18–20 August 2018; pp. 320–324.

36. Hilal, T.A.; Hilal, H.A. Arabic text lossless compression by characters encoding. *Procedia Comput. Sci.* **2019**, *155*, 618–623. [[CrossRef](#)]
37. Avrunin, R.M.; Klein, S.T.; Shapira, D. Combining forward compression with PPM. *SN Comput. Sci.* **2022**, *3*, 239. [[CrossRef](#)]
38. Liao, S.Y.; Devadas, S.; Keutzer, K. Code density optimization for embedded DSP processors using data compression techniques. In Proceedings of the Sixteenth Conference on Advanced Research in VLSI, Chapel Hill, NC, USA, 27–29 March 1995; pp. 272–285.
39. Liu, W.; Chang, Z.; Teahan, W.J. PPM Compression-based Method for English-Chinese Bilingual Sentence Alignment. In Proceedings of the 2nd international Conference on Statistical Language and Speech Processing (SLSP 2014), Grenoble, France, 14–16 October 2014.
40. Moffat, A. Implementing the PPM data compression scheme. *IEEE Trans. Commun.* **1990**, *38*, 1917–1921. [[CrossRef](#)]
41. Cleary, J.G.; Teahan, W.J. Unbounded length contexts for PPM. *Comput. J.* **1997**, *40*, 67–75. [[CrossRef](#)]
42. Goyal, M.; Tatwawadi, K.; Chandak, S.; Ochoa, I. DZip: Improved general-purpose loss less compression based on novel neural network modeling. In Proceedings of the 2021 Data Compression Conference (DCC), Snowbird, UT, USA, 23–26 March 2021; pp. 153–162.
43. Capotondi, A.; Rusci, M.; Fariselli, M.; Benini, L. CMix-NN: Mixed low-precision CNN library for memory-constrained edge devices. *IEEE Trans. Circuits Syst. II Express Briefs* **2020**, *67*, 871–875. [[CrossRef](#)]
44. Goyal, M.; Tatwawadi, K.; Chandak, S.; Ochoa, I. Deepzip: Lossless data compression using recurrent neural networks. *arXiv* **2018**, arXiv:1811.08162.
45. Bellard, F. *NNCP v2: Lossless Data Compression with Transformer*; Technical Report; Amarisoft: Paris, France, 2021.
46. Hu, J.; Hong, Y.; Jin, Q.; Zhao, G.; Lu, H. An Efficient Dual-Stage Compression Model for Maritime Safety Information Based on BeiDou Short-Message Communication. *J. Mar. Sci. Eng.* **2023**, *11*, 1521. [[CrossRef](#)]
47. Zhu, M.; Gao, Y.; Hu, L.; Hu, J. Hierarchical Multi-label Classification Method for Maritime Distress Safety Information. In Proceedings of the 2024 4th International Conference on Neural Networks, Information and Communication (NNICE), Guangzhou, China, 19–21 January 2024; pp. 794–798.
48. Kanuga, P. New shift table algorithm for multiple variable length string pattern matching. In Proceedings of the 2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015], Nagercoil, India, 19–20 March 2015; pp. 1–5.
49. Siau, N.Z. A Teachable Semi-Automatic Web Information Extraction System Based on Evolved Regular Expression Patterns. Ph.D. Thesis, Loughborough University, London, UK, 2014.
50. LZ4 Development Team. LZ4: Fast Compression Algorithm. *LZ4 Official Website*. 2011. Available online: <https://lz4.github.io/lz4/> (accessed on 2 April 2024).
51. GNU Project. Gzip. Available online: <https://www.gnu.org/software/gzip/> (accessed on 2 April 2024).
52. Tukaani. XZ Utils. Available online: <https://tukaani.org/xz/> (accessed on 2 April 2024).
53. Facebook, Inc. Zstandard (zstd) Compression Algorithm. Available online: <https://facebook.github.io/zstd/> (accessed on 2 April 2024).
54. Google. Brotli Compression Format. Available online: <https://github.com/google/brotli> (accessed on 2 April 2024).
55. Barina, D.; Klima, O. x3: Lossless Data Compressor. In Proceedings of the 2022 Data Compression Conference (DCC), Snowbird, UT, USA, 22–25 March 2022; p. 441.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.