Bitogen-Based Representation and Cryptographic Preprocessing: A Conceptual Framework

Abstract

We propose a symbolic abstraction layer called a bitogen: a predefined bit-pattern acting as a stable symbolic unit for binary data. Bitogens extend dictionary-based and token-based modeling from text into arbitrary binary domains. We formalize bitogens, describe their potential for entropy reduction in compression systems, and outline a hybrid cryptographic construction where bitogen sequences undergo key-dependent permutations prior to classical AEAD encryption. We argue that bitogens may provide a new perspective on data structure modeling and open several research directions at the intersection of representation learning, coding theory, and cryptography.

1. Introduction

Modern data processing relies heavily on bit-level or byte-level representations. While effective, these representations do not expose high-level structure present in many binary domains. Inspired by linguistic subword tokenization and biological coding systems (e.g., codons in DNA), we introduce the concept of a bitogen: a symbolic, stable mapping from predefined bit-patterns to higher-level units. Bitogens enable modeling data at an intermediate granularity between single bits and application-level symbols.

2. Formal Model of Bitogens

A global bitogen map D is defined as a mapping from indices i to bit-patterns pattern_i, where patterns may be of variable length. The map is constant, public, and analogous to a coding table. Tokenization proceeds by longest-match parsing of raw bitstreams into sequences of bitogen indices. This yields a symbolic representation suitable for statistical modeling, compression, or cryptographic preprocessing.

3. Applications in Compression

Bitogens allow complex recurring structures to be modeled as single symbols. When combined with arithmetic/ANS entropy coding, the predicted distribution over bitogens can yield sub-bit average code lengths. Learned bitogen dictionaries, derived from large corpora of binary data, may outperform hand■crafted patterns or byte-level baselines. This aligns with recent progress in learned compressors.

4. Applications in Cryptography

We outline a hybrid bitogen-based encryption pipeline:

1. Bitogenize plaintext bits into symbolic indices.

2. Apply a key-derived permutation π to bitogen indices.

3. Optionally apply a stream-based transformation (XOR, rotations).

4. Encrypt the resulting sequence using AES■GCM or ChaCha20■Poly1305.

This provides structural obfuscation while retaining strong security guarantees through AEAD.

5. Security Considerations

The bitogen layer alone is not intended as a cryptographic primitive. However, as a preprocessing step prior to AEAD, it introduces symbolic diffusion and may hinder structural analysis of encrypted binary formats. Care must be taken to avoid leaking information through length changes or deterministic parsing.

6. Related Work

Bitogen concepts relate to several established research areas:

- Dictionary-based compression (LZ, LZW) and grammar-based compression, where repeated patterns are replaced by higher-level symbols.

- Tokenization and subword models (BPE, unigram LM) widely used in NLP and recently extended to binary-level tokenization for learned data compressors.

- Learned lossless compression (e.g., neural autoregressive models, latent modeling), where symbolic representations reduce entropy before arithmetic coding.

- Word-based and structure-based encryption schemes in specialized domains (e.g., linguistic encryption), though typically limited to text and not general binary data.

- Format-transforming encryption and format-preserving transformations, which similarly preprocess data before cryptographic operations.

No existing work combines a global binary token dictionary with key-dependent symbolic permutations and classical AEAD, giving this approach novelty as a research direction.

7. Proposed Research Plan

We propose a systematic evaluation along four axes:

7.1. Dictionary Learning

Train bitogen dictionaries using AI-driven clustering on large binary corpora (network packets, executables, images, logs). Compare entropy before and after bitogenization.

7.2. Compression Experiments

Implement bitogen→ANS/arithmetic pipelines. Compare with zstd, brotli, CMIX, and neural compressors. Metrics: entropy, throughput, dictionary size, generalization across domains.

7.3. Cryptographic Evaluation

Implement the hybrid bitogen cipher. Verify that AEAD security remains intact. Measure structural leakage, resistance to traffic analysis, and symbol frequency uniformity.

7.4. Theoretical Analysis

Study prefix-free constraints, optimality conditions for longest-match parsing, and expected entropy bounds. Analyze the permutation space and its relation to symbolic diffusion.

8. Conclusion

Bitogens introduce a structured symbolic abstraction over binary data. They unify ideas from dictionary coding, representation learning, and preprocessing for cryptography. As both a conceptual tool and a practical transformation, bitogens warrant further exploration across compression research, data modeling, and secure preprocessing.