# Data-Driven Lightweight Encoding Selection

**Hao Jiang, Aaron J. Elmore**

**Department of Computer Science**
**The University of Chicago**

This project explores an automated method to select efficient lightweight encoding schemes for column-based databases. Existing methods either rely on database administrators' experience or use simple rules to make selection. In practice, neither of these methods achieve optimal performance. We propose a method to solve the problem in a systematic data-driven way.

## Lightweight Encoding

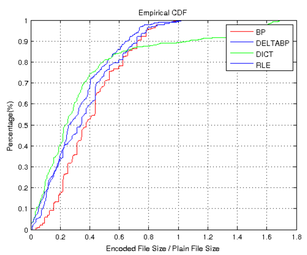
Popular Lightweight encoding schemes includes Run-Length Encoding, Dictionary Encoding, Bit-Packing and Delta Encoding. Comparing to popular compression techniques like gzip and snappy, lightweight encoding schemes have many advantanges such as Encoding Speed, Local Computation and Support to In-Site Query Execution.

Best encoding scheme given a data attributes is determined by many factors including data type and data nature, and there is no simple rule on how to choose it. The system we propose includes the following features:

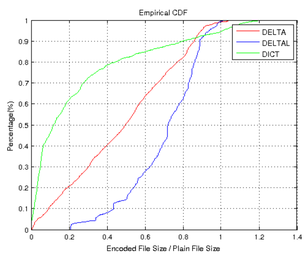
1. Dataset Analysis
2. Pattern Mining
3. Data-Driven Encoding Prediction

## Dataset Analysis

We have collected over 7000 columns from approximately 1200 datasets with a total size of 500G data. These datasets are all from real-world data sources and cover a rich collection of data types (integer, date, address, etc.), with diverse data distributions.



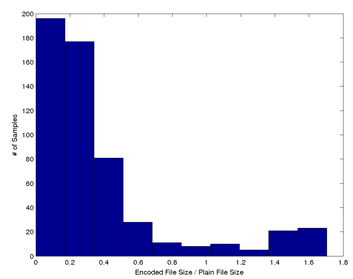

(a) Encoding Compression Ratio for Integer Columns



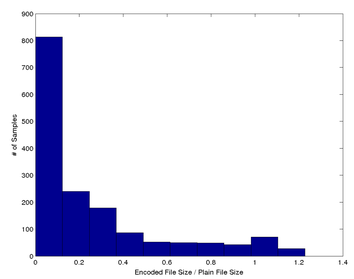

(b) Encoding Compression Ratio for String Columns

The figures above show the CDF of the compression ratio for different types of encodings. It can be seen there is no single best encoding for either data types.

Dictionary Encoding is the default encoding for many column storages (e.g, Apache Parquet and CarbonData). However, the figures below demonstrates that Dictionary Encoding performs sub-optimal for a considerable amount of data samples.




(a) Dictionary Encoding for Integer Columns




(b) Dictionary Encoding for String Columns

## Pattern Mining

Determining the proper data type is crucial to encoding task. For example, storing a U.S. phone number in string requires 10 bytes, while storing it in integer format takes no more than 5 bytes. The challenge is that most real-world datasets are semi-structured, in which only part of the data contains valid common structures. We implement multiple Pattern Mining techniques to automatically identify and extract these structures.

### Recognizing Pattern helps better encoding

HA32394
HB33421
HA423423
HB324234

String format need 7 Bytes

HA = 0
HB = 1
HA = 0
HB = 1

Categorical Data, 1 bit

32394
33421
23423
324234

Bit-packed Integer, 20 bits

In this example, our system observes a pattern <STR><NUM> from the data column and use this pattern to generate more efficient encoding schemes.

### Use NLP techniques to discover text patterns

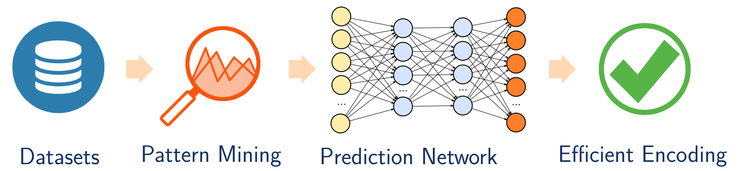
711-2880 Nulla St. Mankato Mississippi 96522
P.O. Box 283 8562 Fusce Rd. Frederick Nebraska 20620
606-3727 Ullamcorper. Street Roseville NH 11523
Ap 867-859 Sit Rd. Azusa New York 39531

| | | | | |
|---|---|---|---|---|
| 711-2880 Nulla | St. | Mankato | Mississippi | 96522 |
| P.O. Box 283 8562 Fusce | Rd. | Frederick | Nebraska | 20620 |
| 606-3727 Ullamcorper. | Street | Roseville | NH | 11523 |
| Ap 867-859 Sit | Rd. | Azusa | New York | 39531 |

We also use NLP techniques to look for similar words in text and identify more subtle patterns. The example above shows that our method is able to discover two similar word groups from a column storing address information. This allows us to encode these words more efficently, in addition, it enable us to use these words to split the text into pieces and do further pattern mining.

## Data Driven Encoding Prediction

we are building a data-driven encoding selector using neural network that is able to "predict" the best encoding scheme by looking at only a limited section of the entire dataset, e.g., the first 1% of the records.




Choosing core features that truly represents the relationship between data and label is crucial to the success of neural network training task. The features we choose including statistical information such as mean and variance of data length, entropy. We also use NLP technique to learn from column names. Finally, we also try to use LSTM RNN network to learn patterns from the sampled column.