A Machine Learning Perspective on Predictive Coding with PAQ by Knoll & de Freitas

Presentation by Phil Trommer

December 10, 2019

Overview

Introduction to PAQ

What is PAQ8

- What is it?
- How does it work?
- What makes it so famous?

Introduction to PAQ

What is PAQ?

- A lossless, open-source compression algorithm
- Brings high perfomance at the cost of increased memory usage and time consumption
- Related to PPM, is envisioned as PPMs improvement

Introduction to PAQ

What is PAQ?

- A lossless, open-source compression algorithm
- Brings high perfomance at the cost of increased memory usage and time consumption
- Related to PPM, is envisioned as PPMs improvement

Matt Mahoney

- Born 1955
- Recieved Ph.D in computer science at Florida Tech in 2003
- Released PAQ1 on January 6, 2002

Introduction to PAQ

Principles of PAQ

- Modeling combined with adaptive arithmetic encoding
- Open to additions and improvements
- Improves perfomance of PPM by including several predictors (i.e. models of data)
- Combines the result of the predictors

Introduction to PAQ

Exemplary Predictors

The order-n context predictor

- Examines the last n bits and counts the 1's and 0's
- Estimates probability whether next bit is 1 or 0 like PPM

Exemplary Predictors

The order-n context predictor

- Examines the last n bits and counts the 1's and 0's
- Estimates probability whether next bit is 1 or 0 like PPM

The sparse context predictor

- Context consists of a specific amount of non-contiguous bytes before the current bit

Exemplary Predictors

The order-n context predictor

- Examines the last n bits and counts the 1's and 0's
- Estimates probability whether next bit is 1 or 0 like PPM

The sparse context predictor

- Context consists of a specific amount of non-contiguous bytes before the current bit

Whole word order-n context

- Context is the latest n whole words

PAQ & Predictors

- PAQ encoder looks at the beginning of input file for deciding which predictors are used
- Ways to combine predictions change through with the different versions
- Each predictor outputs a pair of bit counts (n_0, n_1)
- Counts of each predictor are weighted with context length
- Those counts get summed up

PAQ8L

PAQ8 - What's new?

- Predictors don't produce a pair of bit counts anymore
  $\hookrightarrow$ those counts get weighted and normalized into the interval $[0, 1] \subset \mathbb{R}$
- *paq8l* is a stable version of paq8, released by Matt Mahoney

PAQ8L

PAQ8 - What's new?

- Predictors don't produce a pair of bit counts anymore
  $\hookrightarrow$ those counts get weighted and normalized into the interval $[0, 1] \subset \mathbb{R}$
- *paq8l* is a stable version of paq8, released by Matt Mahoney

PAQ8L - Machine Learning Perspective

- paq8l is the version of PAQ used by *Byron Knoll & Nando de Freitas*
- They try to show the possibilities of PAQ beyond data compression

Architecture

Architecture of PAQ8

- Uses weighted combination of predictions from Large number of models
- Allows non-contiguous context matches
- paq8l uses **552** prediciton models
- Combines the output of them into a single one
  $\hookrightarrow$ Passes this through an *adaptive probability map* (APM) before using the arithmetic coder

Architecture

Architecture of PAQ8

- Uses weighted combination of predictions from Large number of models
- Allows non-contiguous context matches
- paq8l uses **552** prediciton models
- Combines the output of them into a single one
  $\hookrightarrow$ Passes this through an *adaptive probability map* (APM) before using the arithmetic coder

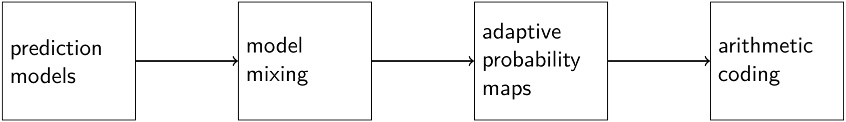

Figure: PAQ8 Architecture

Neural network

Neurons of a neural network

A neuron takes one or more inputs and gives an output.
Within the topic of machine learning, the neuron can be understood as a function.

Neural network

Neurons of a neural network

A neuron takes one or more inputs and gives an output.
Within the topic of machine learning, the neuron can be understood as a function.

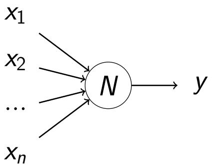

Figure: Neural network architecture

Neural network

Layer in neural network

A layer is a group of neurons.

Neural network

Layer in neural network

A layer is a group of neurons.

A neural network

Neural networks is defined by its layers:

- 1 input layer with n inputs
- 1 output layer with k outputs
- M layers between input and output layer (i.e. hidden layers)
- Layers can consist of different amounts of neurons

Neural network

Layer in neural network

A layer is a group of neurons.

A neural network

Neural networks is defined by its layers:

- 1 input layer with n inputs
- 1 output layer with k outputs
- M layers between input and output layer (i.e. hidden layers)
- Layers can consist of different amounts of neurons

General structure of neural network

Let it be an generic neural network with:

- $x_1, ...x_n$ inputs and $y_1, ..., y_k$ outputs
- There are M different layers between input and output

Neural network

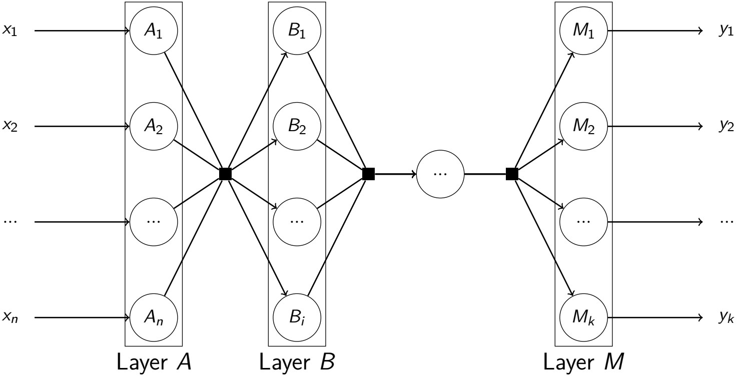

Figure: Neural network architecture

Model Mixer

Model Mixer of paq8l

- Resembles a neural network with one hidden layer
- Subtle differences from a standard neural network

Model Mixer

Model Mixer of paq8l

- Resembles a neural network with one hidden layer
- Subtle differences from a standard neural network

Differences between paq8l and neural networks

1. Weights for first and second layers are learned online and independently for all nodes:
   - Each node trained separately
   - reduces predictive cross-entropy error (unlike back propagation)
2. Hidden nodes are partitioned into seven sets

Model Mixer

Hidden Node Partitioning

- For every bit of data 1 node from each set
- Only edges of selected nodes are updated
- $552 \times 7 = 3,864$ weights updated per bit

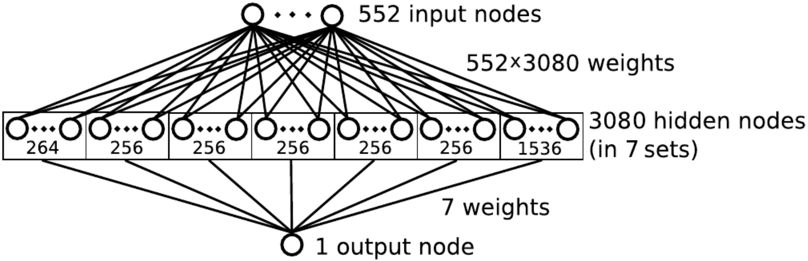

Figure: Model Mixer architecture (Graphic by Knoll & De Freitas)

Model Mixer

Node selection

- Sets 1,2,4 and 5 choose node based on single byte in context
- Set 6 chooses based on length of longest context matched
- Sets 3 and 7 use combination of several bytes
- Depending on the context, a specific node is selected

Model Mixer

Node selection

- Sets 1,2,4 and 5 choose node based on single byte in context
- Set 6 chooses based on length of longest context matched
- Sets 3 and 7 use combination of several bytes
- Depending on the context, a specific node is selected

Mixtures of Experts

- Technique published by *Jacobs et al. 1991*
- Used for neural network training
- Requires a gating network to select expert model

Adaptive Probability Maps

Definition APM

- Takes prediction from model mixer
- is an two dimensional table and low order context as input
- Outputs a new prediction on non-linear scale
- Table entries adjusted after each bit is coded

Classification

Classification as the basic principle

- Compression based classification discovered by researches (Marton et al., 2005)
- Standard procedures for compression based classification exists
- SMDL, AMDL & BCN

Classification

Classification as the basic principle

- Compression based classification discovered by researches (Marton et al., 2005)
- Standard procedures for compression based classification exists
- SMDL, AMDL & BCN

Classification procedures

- **SMDL** $\rightarrow$ uses differences between test dictionary and result dictionary
- **AMDL & BCN** $\rightarrow$ uses difference between compressed file sizes (training files & test file)

Applications for PAQ8

What applications?

PAQ8 is useful even beside compressing files.

- Adaptive Text Prediction
- Text categorization
- Shape recognition
- Lossy compression (i.e. JPEG)

Results are calculated by an module called *PAQclass*

Applications for PAQ8

What applications?

PAQ8 is useful even beside compressing files.

- Adaptive Text Prediction
- Text categorization
- Shape recognition
- Lossy compression (i.e. JPEG)

Results are calculated by an module called *PAQclass*

Adaptive Text Prediction

- PAQ8 can be used to find string x for some training string y
- Can be set to work in speech recognition and text prediction (for typing)

Methodology	Protocol	Percent correct
extended version of Naive Bayes (Rennie et al., 2003)	80-20 train-test split	86.2
SVM + error correcting output coding (Rennie, 2001)	80-20 train-test split	87.5
language modeling (Peng et al., 2004)	80-20 train-test split	89.23
AMDL using RAR compression (Marton et al., 2005)	80-20 train-test split	90.5
multiclass SVM + linear kernel (Weinberger and Saul, 2009)	70-30 train-test split	91.96
PAQclass	**80-20 train-test split**	**92.35**
multinomial Naive Bayes + TFIDF (Kibriya et al., 2005)	80-20 train-test split	93.65

Figure: Text categorization comparison (Graphic by Knoll & De Freitas)

Shape recognition

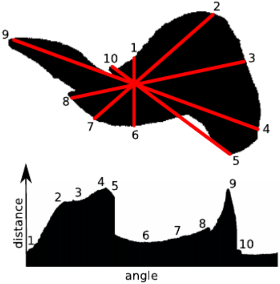

Figure: Shape recognition principle (Graphic by Knoll & De Freitas)

Shape recognition

Methodology	Protocol	Percent correct
1-NN + Levenshtein edit distance (Mollineda et al., 2002)	leave-one-out	≈ 67
1-NN + HMM-based distance (Bicego and Trudda, 2008)	leave-one-out	73.77
1-NN + mBm-based features (Bicego and Trudda, 2008)	leave-one-out	76.5
1-NN + approximated cyclic distance (Mollineda et al., 2002)	leave-one-out	≈ 78
1-NN + convert to time series (Wei et al., 2008)	leave-one-out	80.04
SVM + HMM-based entropic features (Perina et al., 2009)	leave-one-out	81.21
SVM + HMM-based nonlinear kernel (Carli et al., 2009)	50-50 train-test split	85.52
SVM + HMM-based Fisher kernel (Bicego et al., 2009)	50-50 train-test split	85.8
PAQclass + convert to time series	**leave-one-out**	**87.22**

Figure: Shape recognition comparison (Graphic by Knoll & De Freitas)

Lossy compression

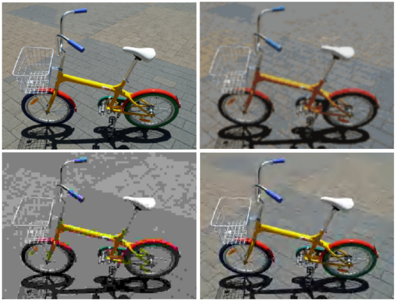

Figure: Picture compression comparison (Graphic by Knoll & De Freitas)

Upper-Left: uncompressed 700x525 pixel	Upper-Right: compressed by paq8 4083 bytes
Bottom-Left: JPEG 16783 bytes	Bottom-Right: JPEG2000 4097 bytes

References

- *A Machine Learning Perspective on Predictive Coding with PAQ*, Byron Knoll & Nando de Freitas, 2019/08/17
- https://stackoverflow.com/questions/41990250/what-is-cross-entropy
- http://mattmahoney.net/
- https://www.techopedia.com/definition/33264/hidden-layer-neural-networks
- *Handbook of Datacompression*, 5th Edition, Springer