Asymptotics on the Lempel-Ziv 78 compression of Markov sources

Exploring analytic information theory : from Markov source sampling to combinatorial analysis proofs

Guillaume Duboc

Computer Science Department
Ecole Normale Supérieure de Lyon

7 août 2018

ENS DE LYON

Table of contents

ENS DE LYON

Table des matières

ENS DE LYON

Words or sequences, and memoryless sources

Definition : word or sequence or string

Given an alphabet $\mathcal{A}$, a ***word*** or ***sequence*** or ***string*** is an infinite sequence of random variables $X = (X_k)_{k\mathbb{N}^*}$, each X_k representing a symbol in $\mathcal{A}$.

Definition : Bernoulli or Memoryless source

A source of information is a ***Bernoulli*** or ***memoryless source*** when all the symbols of $\mathcal{A}$ occur independently with a fixed probability. The word can be seen as an *infinite sequence of Bernoulli trials*.

ENS DE LYON

Markov sources definition

Definition : Markov source

An information source is a ***Markov source*** when there is a ***Markov dependency*** between the consecutive symbols of a string.

Definition : order of a Markov source

Let $V = |\mathcal{A}|$. A ***Markov source*** is of ***order*** r when the dependency can be encoded in a transition matrix of size $V^r \times V$, with coefficients :

$$P(c|w) \qquad \forall\, (w, c) \in \mathcal{A}^r \times \mathcal{A}$$

Informally : *the probability that a symbols occurs depends on the previous r symbols.*

Table des matières

ENS DE LYON

Description of the LZ78 algorithm

Algorithm

Given a word w.

- Initialize an empty dictionary
- While it is possible :

  Find longest prefix of w that is not in the dictionary

  Add it to the dictionary, cut it from w

Elements description

The data representation is (dictionary_reference, symbol).

Remarks

The LZ78 algorithm builds a prefix tree from which the original word can be reconstructed.

Definition : number of phrases

After compressing a word w, the number of phrases in the dictinary is noted $M(w)$.
For words of size n, we write $M_n(w)$.

Code length

$$C(w) = \sum_{k=0}^{M(w)} \left(\lceil \log_2(k) \rceil + \lceil \log_2(\mathcal{A}) \rceil \right)$$

ENS DE LYON

Table des matières

ENS DE LYON

Definition : compression ratio

Let w a word, and $C(w)$ its *encoding* by a compression algorithm. The ***compression ratio*** of w is $\frac{|C(w)|}{|w|}$.

Main goals of compression algorithms

- Improving the compression ratio
- Fast compression/decompression speed in Mb/s

T

he tradeoff between these two goals is a sensitive research problem. Different compression standards :

- Google (Brotli, 2015)
- Facebook (Zstandard, 2016)

Optimal encoding

Entropy of a Markov source

Let π be a stationary distribution. The entropy of a Markov chain is

$$h = -\sum_{i=1}^{V} \pi \sum_{j=1}^{V} p_{ij} \log(p_{ij})$$

Optimality of LZ78

Considering words of length n.

$$\frac{|C(w)|}{|w|} - h \text{ goes to zero for } n \to +\infty$$

Table des matières

ENS DE LYON

Architecture - Données étudiées

- *Génération de graphismes* : **imageset/img-generation.py** et **make_video.sh** | ***Pygame***, module Python

  On génère un dataset complet, scindé en deux datasets : celui d'entraînement, $\mathcal{E}$, dont on tronque une partie des étiquettes, et celui de test, $\mathcal{T}$

- *Étiquetage de dataset* : **imageset/labelling.py**

  Dans le cas de l'extrait de documentaire, il a fallu étiqueter les images, et traiter les données pour qu'elles rentrent dans l'architecture *Python* déjà en place.

- *Chargement du dataset, mise en forme des données* : **make_dataset.sh**, **load_dataset.py**

- *Gestion des hyperparamètres :* **constants.py**

Pré-traitement du dataset réaliste

Pour le pré-traitement de la vidéo en vue de la reconnaissance d'objets : ***Inception-v3*** de Google.
Inception est un réseau de neurones entraîné pour extraire les caractéristiques visuelles d'une image.

Table des matières

ENS DE LYON

Architecture - Implémentation du réseau de neurones

Entraînement (et codage) du réseau de neurones convolutif $\mathcal{R}$:
fichier **training.py** | ***Keras*** et ***TensorFlow***
Ce fichier contient à la fois la structure du réseau de neurones, codé avec Keras, et la routine d'entraînement que l'on effectue sur un dataset pré-chargé.

ENS DE LYON

Remarques

- ***TensorFlow*** est un outil de bas niveau. N'ayant pas de connaissances avancées en réseaux de neurones - notamment dans leur implémentation, il m'a fallu, pour les utiliser, les considérer comme une boîte noire déjà assemblée, à laquelle je fournissais les entrées et les sorties, ce que permet de faire Keras.
- ***Keras*** est un *wrapper*, une librairie de fonctions de haut niveau simplifiant l'utilisation d'une librairie de plus bas niveau. Chaque commande de *Keras* génère une couche spécifique de réseau de neurone.

ENS DE LYON

Table des matières

ENS DE LYON

Architecture - Régression

Fonctions de régression : **compareReg.py, regressionConstants.py, regNeuralNetwork.py, regSigmoid.py, kMeans.py**
Ces modules contiennent les fonctions implémentant les régressions temporelles, spatiales et spatiotemporelles.

ENS DE LYON

Table des matières

ENS DE LYON

Architecture - Résultats

- *Phases de test* sur le dataset $\mathcal{T}$:
  testingLessSupervisionRegular.py,
  testingRandomSpatialReg.py,
  testingRandomTemporalReg.py,
  testingRegularTemporalReg.py

  Ce sont des scripts testant l'efficacité des différentes régressions possibles.
- *Graphes de sortie* : **plot-data.py, dataPlotting.py**

ENS DE LYON

Schéma global

Table des matières

ENS DE LYON

Vidéo

Une ***vidéo*** est un ensemble d'images $\{I_1 \dots I_n\}$.

Fonction de supervision

La ***fonction de supervision*** σ est une autre manière de voir les ***étiquettes*** d'un dataset. Le temps est la succession des images. On peut donc voir une image I_k comme un instant t. σ associe à chaque instant t l'étiquette σ_t correspondant aux objets visibles dans l'image figurant cet instant.

ENS DE LYON

Table des matières

ENS DE LYON

Continuité temporelle

On parle de ***continuité temporelle*** pour désigner l'ensemble des techniques visant à compléter les valeurs de σ ***sans utiliser*** V mais uniquement les valeurs de σ.

Solutions

Deux catégories :

- des *solutions analytiques* : on prolonge la supervision par des fonctions linéaires par exemple
- des *solutions obtenues par apprentissage* : en partant d ?une régression paramétrée (fonction logistique par exemple), on utilise un réseau pour apprendre les paramètres les plus susceptibles de coller à la fonction de base

Régression par des fonctions linéaires (vert) ou constantes par morceaux (orange)

$$X(1) = 0000000\ldots$$
$$X(2) = 1010101\ldots$$
$$X(3) = 1001101\ldots$$
$$X(4) = 0011001111\ldots$$

Table des matières

ENS DE LYON

Algorithmes de continuité spatiale

Continuité spatiale

On garde le même problème d ?une vidéo V, de ses images $\{I_1 \dots I_n\}$ et d ?une fonction de supervision σ partiellement définie. Cette fois, nous tenons compte de la ***continuité spatiale*** ie ***si deux images se ressemblent ? pour des critères à définir - on va pouvoir considérer que leurs supervisions sont les mêmes.***

ENS DE LYON

Solution immédiate : plus proche distances

Un première méthode consiste à utiliser la distance L_2 entre les images. Soit I_k une image *non supervisée*. On choisit $\operatorname{argmin} \{I_j \text{ supervisée} \mid d(I_k, I_j)\}$ où d est la distance L_2. On prend ainsi l ?image supervisée la plus proche.

Raffinement : plus proche distances par zones d'intérêt

L'algorithme *k*-means, implémenté en Python dans la bibliothèque ***ScikitLearn***, prend en entrée *n* observations et renvoie une partition de ces observations en *k* groupes. Chaque observation est associée au groupe de la moyenne duquel elle se rapproche le plus.

Choix de k

Soit n le nombre d ?images et p la proportion de supervisions possibles. Deux possibilités :

- faire systématiquement k-means pour $k = n \cdot p$ puis choisir de superviser une image dans chaque groupe.
- fixer k, et le limiter à environ 10.

Ainsi, k s'interprète comme le ***nombre de catégories visuelles immédiatement perceptibles par le cerveau humain.***

Table des matières

ENS DE LYON

Principe d'application

L ?idée est de considérer le cas *optimal* de la continuité temporelle, qui consiste à étiqueter de manière régulière de le temps les images afin de rendre moins fréquents les cas de disparition/apparition qui faussent la régression. Ce que l'on fait alors, c'est sélectionner pour chaque *cluster* les images supervisées de manière à ce qu ?elles soient réparties le plus régulièrement dans le temps.

Sommaire

- La reconnaissance d'images automatisée en conditions réelles : problème complet et complexe.
- L'idée de continuité spatiotemporelle permet d'aborder le problème sous différents angles intéressants.