Agnostic Clustering of Markovian Sequences

Ran El-Yaniv **Shai Fine** **Naftali Tishby** *
Institute of Computer Science and Center for Neural Computation
The Hebrew University
Jerusalem 91904, Israel
E-mail: {ranni,fshai,tishby}@cs.huji.ac.il
Category: Algorithms.

Abstract

Classification of finite sequences without explicit knowledge of their statistical origins is of great importance to various applications, such as text categorization and biological sequence analysis. We propose a new information theoretic algorithm for this problem based on two important ingredients. The first, motivated by the "two sample problem", provides an (almost) optimal model independent criterion for comparing two statistical sources. The second ingredient is a novel idea for agnostic "statistical merging" of sequences in an asymptotically optimal way.

In this paper we introduce the algorithm and illustrate its application for synthesis of text sources and for hierarchical classification of short text segments in 12 European languages.

1 Introduction

The problem of un-supervised classification of sequences of discrete symbols is encountered in many important applications of machine learning and data mining. Common applications include text categorization, speech recognition, biological sequence modeling, and analysis of neural spike trains. The problem is normally addressed by either asserting a statistical model for the possible sequences, such as hidden Markov models, or by introducing a similarity measure between sequences, based on additional a-priori knowledge. Both of these approaches make certain assumptions on the origin and diversity of the sequences which are not always justified.

*Corresponding author. Category: Algorithms. Preferred media: Poster.

The emergence of universal (i.e. distribution independent), compression techniques in information theory suggests alternative classification methods which make minimal assumptions on the nature of the sequences. As such, these techniques, when properly designed, are more robust and can be much more appropriate for real world data.

In this paper we propose an algorithm for agnostic, universal classification of sequences, when the only underlying assumption is that the sequences can be approximated by some finite order Markov sources. There are two ingredients to our algorithm. The first is a statistical similarity measure of sources and the second is a sequence merging algorithm which optimally approximates the possible common source of the two sequences. Our similarity measure is motivated by the well known "two sample problem"[Leh59]. A fundamental question in statistics is estimating the probability that two given samples originated from the same distribution. In the i.i.d. (Bernoulli) case this problem was thoroughly investigated and the optimal statistical test is given by the sum of the empirical *relative entropies* between the two samples and the *most likely common source*. We argue that this measure can be extended for arbitrary order Markov sources and discuss some of its analytical properties. It turns out to be a bounded symmetric measure that is most suitable for hierarchical statistical modeling and classification. The second ingredient of our algorithm is a novel and simple technique for "statistical merging" of sequences such that the produced sequence is asymptotically a sample of the most likely common source of the two sequences, under some technical conditions.

The similarity measure and the statistical merging algorithm are naturally combined into a clustering algorithm for sequences, which we demonstrate by application to hierarchical classification of short (1500 characters) text segments in 12 European languages. A more complete analysis of this algorithm, with further applications, will be given elsewhere.

2 Measuring similarity of statistical sources

The most popular information theoretic discrimination measure for probability distributions over a discrete alphabet Σ is the *Relative Entropy* or *Kullback-Leibler Divergence* [Kul59], defined as:

$$D_{KL}\left(p||q\right) = \sum_{\sigma \in \Sigma} p\left(\sigma\right) \log \frac{p\left(\sigma\right)}{q\left(\sigma\right)} \ . \tag{1}$$

Whereas the KL-divergence is clearly the natural information theoretic measure of distribution dissimilarity, it has some serious theoretical and practical drawbacks when applied to small finite samples. It is a non-symmetric and unbounded measure and thus can be used only when the probability distribution q is absolutely continuous with respect to p (i.e. $q = 0 \Rightarrow p = 0$). The more serious drawback of the KL-divergence is its high sensitivity to low probability events under q, which makes it difficult to estimate using small samples, without further assumptions on the distribution q. The "empirical" KL-divergence can thus be a very unreliable estimate of the true divergence. Moreover, this measure is inappropriate for the objective of sources' identification based on small sample realization.

There is a need for a measure that is not that sensitive to small sample fluctuations and that attaches similar importance to different subsets of the data. Essentially, $D_{KL}\left[p||q\right]$ measures the asymptotic code inefficiency when assuming that the sample's generating distribution is q where it is in fact p. Hence, KL-divergence is more adequate as a penalty for using the incorrect distribution in

coding or in maximum likelihood parameter estimation. The symmetric divergence, i.e. $D(p,q) = D_{KL}[p||q] + D_{KL}[q||p]$, proposed already by Jeffreys [Jef46], suffers from similar sensitivity and convergence problems, as can be easily verified. Posing a regularity condition, by constraining the probabilities to be ε-bounded from zero, for some $\varepsilon > 0$, is a common practice that, while reducing the sensitivity problem, introduces an uncontrolled bias.

2.1 The "two sample problem"

The following fundamental problem is the key to our new similarity measure. [Leh59]

Given two independent samples x and y, drawn from unknown sources, what is the probability that x and y are drawn from the same statistical source?

It follows from the method of types [CK81] that the probability that there exists a distribution $\hat{P}$ such that two samples x and y were independently drawn according to $\hat{P}$ is

$$\sum_P P_0(P) \cdot \Pr(x|P) \cdot \Pr(y|P) = \sum_P P_0(P) \cdot 2^{-(N_x D_{KL}[p_x||P] + N_y D_{KL}[p_y||P])},$$

where P_0 is a prior over all candidate distributions, p_x and p_y are empirical (sample) distributions, and N_x and N_y are the lengths of x and y respectively. For large samples this sum is dominated by the maximal term, which suggests the following definition of the λ-mixture of two stochastic sources.

Definition 1 *The* λ-mixture distribution, *M, of the two probability distributions P and Q is defined as:*

$$M_\lambda = \arg\min_{M'} \lambda D_{KL}(P||M') + (1-\lambda) D_{KL}(Q||M') \ .$$

$0 \le \lambda \le 1$ *is called the* mixture ratio *or the* prior.

Due to the convexity of the KL-divergence this minimum is unique and the λ-mixture is well defined and is given by: $M_\lambda = \lambda P + (1-\lambda) Q$

The new similarity measure between distributions naturally follows, as the minimal value of the above sum. Namely,

Definition 2 *The λ-similarity $\mathcal{D}_\lambda(p,q)$ between two distributions, $p(x)$ and $q(x)$, is*

$$\mathcal{D}_\lambda(p,q) = \lambda D_{KL}\left(p||\hat{P}_\lambda\right) + (1-\lambda) D_{KL}\left(q||\hat{P}_\lambda\right) \quad 0 \le \lambda \le 1 \ , \tag{2}$$

where $\hat{P}_\lambda$ is the λ-mixture of p and q.

The mixture ratio λ is determined by the the two sample sizes, assuming that they reflect the prior "distance" of each sample to the common source. This defines our discrimination measure, given any two samples. It should then be compared to a threshold, reflecting a natural principle: Discriminate samples only when there is sufficient probability that they came from different sources.

The two sample problem can now be resolved by first estimating the most likely common source of the two samples,

$$\hat{P} = \arg\min_P N_x D_{KL}(p_x||P) + N_y D_{KL}(p_y||P) = \frac{N_x p_x + N_y p_y}{N_x + N_y} \ ,$$

then comparing the probability that the samples were indeed generated by $\hat{P}$ to a confidence threshold $\delta > 0$. That is,

$$P_0(\hat{P}) \cdot 2^{-N\mathcal{D}_\lambda(p_x,p_y)} > 1-\delta,$$

with $\lambda = \frac{N_x}{N}$.

The above notion of the mutual, most likely, common source can be naturally extended to n samples.

2.2 Properties of the $\mathcal{D}_\lambda$ similarity measure

Let p and q be two probability distributions over a sample space Ω. The function $\mathcal{D}_\lambda(p,q)$ satisfies the following useful properties, which are not difficult to prove [EFT97, CT91]. Whenever $\lambda = \frac{1}{2}$, i.e. the two samples are of the same size, we use the shorthand notation $\mathcal{D}_{\frac{1}{2}} = \mathcal{D}$.

1. $\mathcal{D}_\lambda$ is non-negative, symmetric, but not a metric.
2. $\mathcal{D}_\lambda(p,q) = 0 \Longleftrightarrow p = q$ (i.e. $p_i = q_i \quad \forall i \in \Omega$)
   $\mathcal{D}_\lambda(p,q) = 1 \Longleftrightarrow p \cap q = \emptyset$ (i.e. $p_i \cdot q_i = 0 \quad \forall i \in \Omega$)
3. For any distribution w over Ω,
$$D_{KL}(p||w) + D_{KL}(q||w) = 2\mathcal{D}_\lambda(p,q) + 2D_{KL}\left(\frac{p+q}{2}||w\right)$$
4. $\mathcal{D}_\lambda(p,q)$ is convex in both p and q.
5. $\mathcal{D}_\lambda(p,q)$ is related to the entropies of $\hat{P} = \lambda p + (1-\lambda)q$, p and q, via
$$\mathcal{D}_\lambda(p,q) = H(\lambda p + (1-\lambda)q) - (\lambda H(p) + (1-\lambda)H(q))$$
where $H(p) = -\sum p_i \log p_i$.
6. Relations to $\mathcal{L}_1$ norm.
$$\frac{1}{4\ln 2}\|p-q\|_1^2 \leq \mathcal{D}_\lambda(p,q) \leq \mathcal{H}(\lambda)\cdot\|p-q\|_1 \quad \lambda \in [0,1] \quad .$$
Consequently,
$$\frac{1}{2\ln 2}\|p-q\|_1^2 \leq \mathcal{D}_\lambda(p,q) \leq \|p-q\|_1 \quad ,$$
where $\mathcal{H}(\lambda)$ is the binary entropy of $(\lambda, 1-\lambda)$ and $\|p\|_1 = \sum_i |p_i|$.

The next propositions hold for any finite order $< K$ ergodic Markov source over the finite alphabet Σ with $\Omega = \Sigma^*$. The proofs will be given elsewhere[EFT97].

Theorem 1 *If x and y are two samples of size N independently drawn according to the same distribution, then*

$$\mathcal{D}(P_x||Q_y) \to 0$$

as the sample size $N \to \infty$, with probability 1.

Theorem 2 *Given $\varepsilon > 0$ and $1 > \delta > 0$, to solve the "two-samples problem" to an accuracy at least ε with confidence not smaller then $1-\delta$, the sample size, N must be of order*

$$N > O\left(\frac{1}{\varepsilon}\log\frac{1}{\delta} + \frac{\left|\Sigma^K\right|}{\varepsilon}\log\frac{1}{\varepsilon}\right) \quad .$$

3 Mutual Sources of Markov Sequences

In this section we first generalize the notion of the mutual statistical source to finite order Markov probability measures. We identify the mutual source of Markovian sequences and show how to construct a typical random sample of the source.

More precisely, let x and y be two sequences, generated by the Markov processes with distributions P and Q, respectively. We present a novel algorithm for the generation of a typical sequence of an approximation to the most likely mutual source of x and y. The algorithm is agnostic with respect to the parameters of true sources P and Q and the computation of the sequence is done directly from the sequences data x and y.

There are several possible applications of this sequence merging algorithm. Here we demonstrate two possible applications, the source merging problem and bottom-up sequence-clustering. These algorithms are different from most existing approaches because they directly employ the sequenced data, similar to universal compression, without any modeling assumptions. Further analysis and applications of the algorithms will be presented elsewhere [EFT97].

Let Σ be a finite alphabet and let P and Q denote two ergodic Markov processes over Σ of orders K_P and K_Q, respectively. By Definition 1, the *λ-mixture source* S_λ of P and Q is $S_\lambda = \arg\min_{S'} D_{KL}(P||S') + (1-\lambda)D_{KL}(Q||S')$, where $D_{KL}(P||S) = \limsup_{n\to\infty} \limsup \frac{1}{n} \sum_{x\in\Sigma^n} P(x) \log \frac{P(x)}{S(x)}$. The following theorem identifies the mutual source of P and Q.

Theorem 3 *The unique λ-mixture mutual source S_λ of P and Q, of order $K = \max\{K_P, K_Q\}$, is given by the following conditional distribution. For each $s \in \Sigma^K, a \in \Sigma$,*

$$S_\lambda(a|s) = \frac{\lambda P(s)}{\lambda P(s) + (1-\lambda)Q(s)} P(a|s) + \frac{(1-\lambda)Q(s)}{\lambda P(s) + (1-\lambda)Q(s)} Q(a|s)$$

This can be naturally generalized to n sources with priors $\lambda_1, \ldots, \lambda_n$.

3.1 The "source merging" algorithm

Figure 1 describe a randomized algorithm that generates, based on the given sequences x and y, an approximate typical sequence z of the most likely mutual source, without any knowledge of P and Q.

Notice that the algorithm is completely unparameterized, it does not even need to know the alphabets of the sources, which may differ. The algorithm is naturally generalized to any number n of sequences.

3.2 Application 1: Generation and Synthesis

An immediate application of the source merging algorithm is for generation of typical sequences of the mutual source from some given data sequences, *without any access to the mutual source itself or to an explicit model of the source.* In the case that the underlying sources of x and y are different the algorithm "synthesizes" the most likely mutual source.

To illustrate this point consider the sequence in Figure 3.2. This sequence was randomly generated, character by character, from two natural excerpts: a 47,655-character string from Dickens' Tale of Two Cities, and a 59,097-character string from Twain's The King and the Pauper.

Initialization:

- $z\,[0]$ = choose a symbol from x with probability λ or y with probability $1-\lambda$
- $i = 0$

Loop:

Repeat until the approximation error is lower then a prescribe threshold

- r = choose x with probability λ or y with probability $1-\lambda$
- z_r = maximum length suffix of z appearing somewhere in r
- $r\,(z_r)$ = choose one of the occurrences of z_r in r (with uniform probability)
- $z\,[i+1]$ = the symbol appearing immediately after $r\,(z_r)$ at r
- $i = i+1$

End Repeat

Figure 1: The most-likely mutual source algorithm

```
    Do your way to her breast, and sent a treason's sword- and not empty.
"I am particularly and when the stepped of his own commits place.  No; yes,
of course, and he passed behind that by turns ascended upon him, and my bone
to touch it, less to say:  'Remove thought, every one!  Guards!  In miness?"
The books third time.  There was but pastened her unave misg his ruined head
than they had known to keep his saw whether think" The feet our grace he
called offer information?

                                                          [Twickens, 1997]
```

Figure 2: An excerpt from random text generated from Dickens and Twain sample sequences.

3.3 Application 2: Agnostic sequence clustering

The mutual source algorithm, combined with the new similarity measure, provides the means to cluster, or hierarchically classify, data sequences. Using known estimation methods of the KL-divergence [ZM93, BE97] we can estimate $\mathcal{D}_\lambda$ between any two sequences and then employ any clustering method to generate a classification tree. Notice that unlike many other string-matching techniques, no explicit assumption or model is made for the sequences, nor any "external metric" is used.

To illustrate the power of our new sequence clustering method we give the result of a rudimentary linguistic experiment using a greedy bottom-up (conglomerative) clustering of short excerpts from twelve languages. For each of the twelve languages we took two 1500-character strings (from different sources) and applied the bottom-up greedy clustering on this input. The clustering result, is presented in Figure 3, which does not show the entire bottom level. This entirely un-supervised algorithm clearly revels the familiar philologic tree of these European languages, as well as can be hoped for from such short segments.

Acknowledgments

We thank Ran Bachrach for helpful discussions. We also thank Sageev Oore for many useful comments.

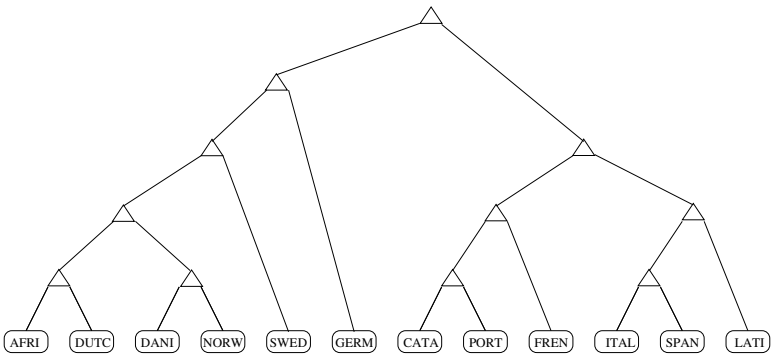

Figure 3: Bottom-up greedy clustering of strings from twelve languages: Afrikaans, Dutch, Danish, Norwegian, Swedish, German, Catalan, Portuguese, French, Italian, Spanish and Latin

References

[BE97] R. Bachrach and R. El-Yaniv, An Improved Measure of Relative Entropy Between Individual Sequences, unpublished manuscript.

[CK81] I. Csiszár and J. Körner. Information Theory: Coding Theorems for Discrete Memoryless Systems, Academic Press, New-York 1981.

[CT91] T. M. Cover and J. A. Thomas. Elements of Information Theory, John Wiley & Sons, New-York 1991.

[EFT97] R. El-Yaniv, S. Fine and N. Tishby. Averaging Markovian Sources, in preparations, 1997.

[Hoe65] W. Hoeffding. Asymptotically optimal tests for Multinomial distributions, The Annal of Mathematical Statistics, Vol 36, pp 369-401, 1965.

[Jef46] H. Jeffreys. An invariant form for the prior probability in estimation problems. Proc. Roy. Soc. (London) Ser. A, 186, pp 453-461, 1946.

[Kul59] S. Kullback. Information Theory and Statistics, John Wiley & Sons, New-York 1959.

[Leh59] E. L. Lehmann. Testing Statistical Hypotheses, John Wiley & Sons, New-York 1959..

[ZM93] J. Ziv and N. Merhav, A Measure of Relative Entropy Between Individual Sequences with Application to Universal Classification, IEEE Trans. Info. Theory, Vol. 39, No. 4, 1993.

4 Appendix: The clustering algorithm

A sketch of the sequence clustering algorithm is presented at Figure 4.

Input: S array of sequences
Output: T the clustering tree of the S
Initialization:

- P = construct array of all different pairs of sequences
- **For every** pair P_i in P define
  1. $P_i \rightarrow S_1$ and $P_i \rightarrow S_2$ are respectively the ingredient sequences of the pair
  2. $P_i \rightarrow S_{ms} = Mutal\_Source\_Algorithm\,(P_i \rightarrow S_1, P_i \rightarrow S_2)$
  3. $P_i \rightarrow \mathcal{D} = \mathcal{D}\,(P_i \rightarrow S_1, P_i \rightarrow S_2)$
- Insert all pairs in P into a heap H according to their distances $P_i \rightarrow \mathcal{D}$

Loop:

- **While** H is not empty do
  - **If** H contain only one pair P_{last}
  - **Then**
    - make P_{last} the root of T
  - **Else**
    - get P_l the pair with the lowest distance $\mathcal{D}$
    - $S_{new} = P_l \rightarrow S_{ms}$
    - delete from H all pairs which contain either $P_l \rightarrow S_1$ or $P_l \rightarrow S_2$
    - make $P_l \rightarrow S_1$ and $P_l \rightarrow S_2$ children of S_{new} in T (S_{new} has no parent yet)
    - delete $P_l \rightarrow S_1$ and $P_l \rightarrow S_2$ from S
    - **For every** sequence S_i in S do
      - construct new pair based on S_i and S_{new} and insert to H
    - insert S_{new} to S
- **End While**

Figure 4: The sequence clustering algorithm