Hidden Markov Model Evaluation from First Principles

Russell Leidich
pkejjy@gmail.com

October 31, 2020

Abstract

Hidden Markov models (HMMs) are a class of generative stochastic process models which seek to explain, in the simplest possible terms subject to inherent structural constraints, a set of equally long sequences (time series) of observations. Given such a set, an HMM can be trivially constructed which will reproduce the set exactly. Such an approach, however, would amount to overfitting the data, yielding a model that fails to generalize to new observations of the same physical system under analysis. It's therefore important to consider the information cost (entropy) of describing the HMM itself – not just the entropy of reproducing the observations, which would be zero in the foregoing extreme case, but in general would be the negative log of the probability of such reproduction occurring by chance. The sum of these entropies would then be suitable for the purpose of ranking a set of candidate HMMs by their respective likelihoods of having *actually* generated the observations in the first place. To the author's knowledge, however, no approach has yet been derived for the purpose of measuring HMM entropy from first principles, which is the subject of this paper, notwithstanding the popular use of the Bayesian information criterion (BIC) for this purpose.

Keywords: HMM, model evaluation, Markov, BIC, Bayesian

1. HMM Anatomy

Assume we have a discrete probability distribution ("probability vector") with Z components, each of which giving the respective probability e_i of emitting whole number i, which is less than Z. We call this distribution an "emission vector" because each of its scalar components represents the probability that its whole-numbered index will be emitted, yielding an observable. By definition, because an emission vector is a (discrete) probability distribution, we have:

$$\sum_{i=0}^{Z-1} e_i \equiv 1$$

All HMMs can assume one of H possible "hidden states". Each hidden state is associated with an emission vector and a "transition vector" giving the respective probabilities t_i of transitioning from such a state to any other (including itself) immediately *after* emitting a whole number and immediately *prior to* emitting the next one. As with emission vectors, we have:

$$\sum_{i=0}^{H-1} t_i \equiv 1$$

for all H transition vectors. We can thus construct an H-by-H "transition matrix" containing all transition vectors and their constituent transition probabilities, wherein the rows are the transition vectors the corresponding indexes.

We can then create a Z-by-H "emission matrix" consisting of H rows of emission vectors, corresponding one-to-one to each transition vector in the transition matrix.

We now have all the components required to create an HMM with a single emission channel:

The number H of hidden states.
The number Z of emittable states.
The transition matrix.

The emission matrix.

2. The Entropy of Reversibly Encoding an HMM

It should now be straightforward to compute the cost of transmitting all of the above to a receiver who knows only that the received instructions are provided for the sake of constructing an HMM. (Of course, the cost of fully explaining the set of sequences must also include *their* cost – not just the cost of encoding the HMM itself – but we'll get to that later.)

To that end, we ignore the cost of transmitting *H* and *Z*, which is asymptotically negligible in the limit of large HMMs, and which cost tends to be similar among a group of candidates, and therefore tends to have negligible impact on model evaluation ranking. This leaves only the transition and emission matrices, insofar as the encoding of the HMM itself is concerned.

First of all, given that all vectors of both of these matrices are normalized, their information value can be computed from all but one scalar component. They can therefore be fully reconstructed from a (*Z*-1)-by-*H* matrix and an (*H*-1)-by-*H* matrix.

But what of the vectors themselves? They consist of probabilities, which in the general case are transcendental irrationals, and thus contain an infinite amount of information. In practice, they're likely to be approximated by fixed-size floating-point values ("floats"). Ignoring all the ways in which they might be compressed, floats could be said to have an entropy equivalent to their constituent number of bits – generally, 32 or 64.

However, if we were to evaluate HMM entropy in this manner, then all HMMs with the same *H* and *Z* would yield the same sum, which would be completely uninformative as to which are likelier than others to be the true model which best encodes the observed sequences. This would leave all the distinctions to the negative log of probability of

having generated the set of sequences, inevitably leading to overfitting.

At this point, we have to introduce one assumption: the receiver will assume, prior to receiving any information, that all *legal* transition and emission vectors are equally likely to be the correct. Under this assumption, the entropy of an HMM can then be straightforwardly approximated by quantizing its constituent emission and transition vectors.

3. Quantization of Probability Vectors

As explained above, a probability vector consisting of P components, p_i, can be faithfully represented with only (P-1) of them, which we arbitrarily take to be the *first* (P-1). Because all P components sum to one, we can categorize their successive partial sums, S_j, given by

$$S_j \equiv \sum_{i=0}^{J} p_i, 0 \leq j < P$$

into one of N intervals, each of size (1/N). We say that N is the "quantizer" because it defines the granularity (resolution) to which S_j is known. For example, if (N=7), then 0.390941 would map to interval 2 because (2/7) <= 0.390941 < (3/7). In this way, we can map real numbers to whole numbers. (In the rare corner case where we encounter a partial sum equal to one – other than the last one, which is *always* one – we map it to (N-1) rather than N.) The result of this mapping is a "frequency vector" giving the counts F_j of the number of partial sums which map to each respective whole number less than N. So if (N=7) and (P=15), then we might have:

No partial sums in interval zero.
2 partial sums in interval one.
No partial sums in interval 2.
One partial sum in interval 3.
6 partial sums in interval 4.

3 partial sums in interval 5.
2 partial sums in interval 6.

(This accounts for only 14 partial sums because the last one is always one and thus contributes no information.) Notice that we have only whole numbers now, so we can easily compute the entropy.

4. Entropy of Bucketized Partial Sums

In the foregoing example, we grouped (*P*-1) partial sums into *N* buckets, each of which identified with a whole numbered index. Absent any further assumptions, the entropy of encoding the probability matrices pursuant to such quantization is asymptotically equal to the log of the number of *possible* such configurations. So, how many are there?

First of all, it helps to think of the buckets in terms of dividers on a bookshelf, which for instance might divide a large number of books (partial sums) into a few contiguous sets by author (whole numbered index). Note that (*N*-1) dividers will serve the purpose of *N* buckets, as the former can delineate *N* intervals, the union of which being the unit interval. Note also that we might have multiple contiguous dividers which separate only 2 books, leaving some empty buckets in between.

Given *N* and *P* in this case, we have 6 dividers and 14 books. The number of unique such arrangements is then just the number of ways, *W*, that (6+14) bits can contain exactly 6 zeroes (or ones): ((6+14) nCr 6). In general, then, *P* partial sums can be partitioned into *N* buckets in exactly

$$W \equiv (N + P - 2)\, nCr\, (N - 1)$$

different ways. The entropy *E* due to encoding this number is just its natural log (if measured in nats) or its log to base 2 (if measured in bits):

$$E \equiv \sum_{i=1}^{N-1} [ln\,(N + P - 1 - i) - lni]$$

We can avoid the summation by using the following loggamma (logΓ) identity:

$$log\Gamma\,(N + 1) \equiv \sum_{i=1}^{N} lni$$

which implies that

$$E \equiv log\Gamma\,(N + P - 1) - log\Gamma N - log\Gamma P$$

(in units of nats) which is a rather elegant expression for the entropy of *P* partial sums derived from equally many individual probabilities, bucketized into N intervals.

This brings us to our second assumption: the quantizer *N* will be constant for all emission and transmission vectors. In the general case in which observations have statistical biases, posterior uncertainties among emission and transition parameters are clearly asymmetric on account of greater or lesser activation, but accounting for those distinctions would be a refinement beyond the approach presented in this paper, so we universally assume (1/N) parameter granularity.

Therefore, if we have *H* emission vectors containing *Z* components, and *H* transition vectors containing *H* components, then the total entropy E_M of the entire machine, at a resolution implied by dividing the unit interval into *N* equal segments, is approximated by:

$$E_M \approx H * [(log\Gamma\,(H + N - 1) - log\Gamma H - log\Gamma N) + (log\Gamma\,(Z + N - 1) - log\Gamma Z - log\Gamma N)]$$
$$E_M \approx H * [log\Gamma\,(H + N - 1) + log\Gamma\,(Z + N - 1) - log\Gamma H - log\Gamma Z - 2log\Gamma N]$$

While asymptotically accurate, technically, we have one more vector for which to account, which is the prior probability vector h_0 of hidden state probabilities. This "initial vector" contains H components which convey the same amount of information as each of the transition vectors, as it provides the respective probabilities of being in any given hidden state prior to the first emission. Accounting for this, we more accurately have:

$$E_M \equiv H * [log\Gamma(H+N-1) + log\Gamma(Z+N-1) - log\Gamma H - log\Gamma Z - 2log\Gamma N] + log\Gamma(H+N-1) - log\Gamma H - log\Gamma N$$

which is thus how we define E_M. (This is still a lower bound because it omits the cost of encoding H, Z, and N themselves, in addition some small compression overhead, but these omissions are negligible for the purposes of model selection.)

Accounting for multiple parallel emission channels is a straightforward extension of this formula and beyond the scope of this paper.

5. Entropy of Sets of Observed Sequences

Under an arithmetic encoding scheme, the entropy of encoding a symbol is asymptotically equal to the negative log of its probability of occurrence. This comes directly from the Shannon entropy, which is optimal under the assumption that successive symbols are selected independently.

Strictly speaking, arithmetic encoding could not be used to compress the sequences of wholes emitted from an HMM, simply because the former assumes a constant probability vector. Nevertheless, asymptotically, the logarithmic optimum size still applies because the sequences are arithmetically encodable simply by updating the probability vector prior to encoding each whole.

If multiple sequences have been provided, then the total entropy is simply the sum of the individual sequence entropies (which for sake of

brevity, we won't account for here). The entropy E_S of a single sequence, based on this dynamic form of arithmetic encoding, is then simply:

$$E_s \equiv -\sum_{i=0}^{Q-1} lnp_i(W_i)$$

where Q is the nonzero number of wholes in the sequence, W_i is the whole at index i, and $p_i(W_i)$ is its probability of its occurrence, having arrived at index i (comprehending all hidden state changes necessary to do so). But how might we obtain these p_i values?

p_0 has Z components p_{0k} – one for each emittable whole – which is the product of the probability h_{0j} of being initially in hidden state j, and emission probability e_{jk} of emitting k when in said state:

$$p_{0k} \equiv \sum_{j=0}^{H-1} h_{0j} e_{jk}$$

p_0 is thus determined. But what of p_1, p_2, etc.?

Given the first observation W_0, we can then compute the components h_{0j}' of the posterior hidden state probability vector (but prior to passing through the transition matrix):

$$h_{0j}' \equiv \frac{h_{0j} e_{jW0}}{\sum_{k=0}^{H-1} h_{0k} e_{kW0}}$$

which is simply to say that the components of h_0' equal the components of h_0 weighted by the respective probabilities of having emitted W_0.

We can then pass h_0' through the transition matrix to yield h_1, the components h_{1j} of which providing the probability of being in hidden state j immediately prior to emitting W_1 (the second observation).

$$h_{1j} \equiv \sum_{k=0}^{H-1} h_{0k}' t_{kj}$$

where t_{kj} is the probability of transitioning from hidden state k to j. Thus h_1 is now determined as well. But reapplying the same technique as above, we can then determine p_1 and, by induction, p_2, etc. This will then will allow us to determine E_S.

The total entropy of the machine and its observations, E_T, is then just (E_M+E_S). The least E_T would suggest to us the most credible of a set of candidate HMMs.