slide 4

Complete Codes

eg. Suppose the code sequence 0110 is ~~never~~ is not possible. (ie, will never be transmitted).

Then, after transmitting 011, the next symbol must be 1. Both the encoder & decoder know this. So, there is no need to transmit it, so our code is not optimal.

In general, ~~incom~~ a code is incomplete if some code sequence is not possible.

Incomplete codes are not optimal.

Slide 5

In a black-and-white image, suppose 13 of the first 106 pixels are black. Then, for the 107^{th} symbol, we could use

$$P(\text{Black}) = 13/106$$

$$\& \quad P(\text{white}) = 1 - P(\text{Black}) = 1 - \frac{13}{106} = \frac{93}{106}$$

slide 5

Problem: How do we get started?

Initially, all counts are 0.

So, P(Black) = 0/0

In general, 0 probabilities are a problem:

eg. ~~#~~ ① optimal codeword length $= \log(1/p)$

$= \log(1/0)$

$= \infty$

slide 5

② In Arithmetic coding, interval width = probability. However, intervals of width 0 cannot be subdivided.

In fact, the while loops in the Arithmetic coder & decoder will not terminate if a probability is 0.

slide 5

Simple solution: add 1 to every symbol count. ie, pretend we have seen each symbol at least once before we begin, so that initially, all symbols are equally likely.

~~eg. With source alphabet {a, b, c, d} the initial probabilities are~~

~~$Pr(a) = Pr(b) = Pr(c) = Pr(d) = \frac{1}{4}$~~

eg. Initially, $P(\text{Black}) = P(\text{White}) = \frac{1}{2}$

Slide 5

Likewise, if 13 of the first 106 pixels are black, then for the 107^{th} pixel, use

$$P(\text{Black}) = \frac{13+1}{106+2} = \frac{14}{108}$$

$$P(\text{White}) = \frac{93+1}{106+2} = \frac{94}{108}$$

Note: $13 + 93 = 106$

$$\frac{14}{108} + \frac{94}{108} = \frac{108}{108} = 1$$

Note: Arithmetic coding can easily be adapted to use these (changing) probabilities.

slide 5

Adding 1 to the symbol counts

ie, ~~eg~~ For a binary alphabet of two symbols, a + b, let n_a be the number of a's seen so far, + let n_b be the number of b's seen so far.

Use $pr(a) = \dfrac{n_a+1}{(n_a+1)+(n_b+1)} = \dfrac{n_a+1}{n_a+n_b+2}$

+ $pr(b) = \dfrac{n_b+1}{n_a+n_b+2}$

slide 5

This approach says that before any symbols are seen, (ie, when $n_a = n_b = 0$), then a & b are equally likely, (ie, $P_r(a) = P_r(b) = 1/2$).

This says that we know nothing about the upcoming message.

As more & more symbols are seen we learn more about the message content, & our initial 50:50 bias dissappears, since

$$\frac{n_a + 1}{n_a + n_b + 2} \approx \frac{n_a}{n_a + n_b} \quad \text{For large } n_a .$$

Slide 5

Exchangability

Note that with this approach, symbol probabilities depend only on symbol counts, not on symbol order.

eg. For each of the following source sequences,

abbaa, bbaaa, baaab

the probability that the next symbol is a is $(3+1)/(5+2) = 4/7$.

Context

Slide 11

In English, ~~th~~ qu is <u>much</u> more likely than uq.

ie $P(qu) \neq P(uq)$

& q & u are <u>not</u> independent.

In particular, the probability of u depends on what precedes it.

In this case, ~~a bett~~

$P(XY) = P(X) \cdot P(Y)$ is a poor model.

A much better model is

$P(XY) = P(X) \cdot P(Y|X)$

Slide 12

Markov Sources

Originally, we considered sources in which symbols are independent.

eg. $P(x_1 x_2 x_3) = P(x_1) \cdot P(x_2) \cdot P(x_3)$

Thus, the probability of each symbol does not depend on the other symbols in the sequence.

We also looked briefly at Laplace models, in which the probability of a symbol depends on the counts of all earlier symbols, but not their order.

Slide 12

Now, we shall consider Markov sources, in which the probability of a symbol depends on ~~"Recent" Symb~~ nearby symbols, including (possibly) their order.

In particular, in a $K^{\underline{th}}$ order Markov source, the probability of a symbol may depend on the preceding K symbols.

Slide 12

eg, In a second-order Markov source,

$$P(x_1 x_2 x_3 x_4 x_5 \cdots x_n)$$

$$= P(x_1) \cdot P(x_2 | x_1) \cdot P(x_3 | x_1 x_2) \cdot P(x_4 | x_2 x_3) \cdot P(x_5 | x_3 x_4) \cdots P(x_n | x_{n-2} x_{n-1})$$

ie, the probability of each symbol depends only on the preceding two symbols.

slide 12

We can rewrite this as

$$P(a_{i_1} a_{i_2} \cdots a_{i_n})$$
$$= P(a_{i_1}) \cdot P(a_{i_2} | a_{i_1}) \cdot M(i_1, i_2, i_3)$$
$$\cdot M(i_2, i_3, i_4)$$
$$\cdots$$
$$M(i_{n-2}, i_{n-1}, i_n)$$

where $M(i,j,k) = P(a_k | a_i a_j)$

Note that M can be stored as

~~a table of ~~

an array of size I^3.

Here, the ~~alpha~~ source alphabet is

$a_1, a_2 \cdots a_I$

Example

slide 12

Suppose the source alphabet is $\{a, b, c\}$. Then, M is a table of the following form:

X Y	$P(a \mid XY)$	$P(b \mid XY)$	$P(c \mid XY)$
a a	.1	.6	.3
a b	.9	.01	.09
a c		...	
b a			
b b			
b c			
c a			
c b			
c c			

Note: Each row sums to 1.

ie, Each row is a (conditional) probability table.

Arithmetic Coding

slide 12

Arithmetic coding can ~~easily~~ be adapted to use $P(X_i \mid X_{i-2} X_{i-1})$ to subdivide intervals, instead of using $P(X_i)$.

ii, Arithmetic coding does not rely on <u>independent</u> probabilities. It only requires <u>multiplicative</u> probabilities (as in Markov sources).

slide 13

Frequency Counts

XY	F(XYa)	F(XYb)	F(XYc)
a a	10	60	30
a b	180	2	18
a c			
b a		...	
b b			
b c			
c a			
c b			
c c			

~~use $P(b|aa) = F(aab)/[F(aaa)+F(aab)+F(aac)]$~~

~~$= 60/(10+60+30)$~~

~~$= .6$~~

~~$P(c|ab) = 18/(180+2+18)$~~ $= .09$

slide 13

Arithmetic coding will use the following probability estimates:

$$P(b|aa) = \frac{F(aab)}{F(aaa) + F(aab) + F(aac)}$$

$$= \frac{60}{10 + 60 + 30}$$

$$= 0.6$$

$$P(c|ab) = \frac{F(abc)}{F(aba) + F(abb) + F(abc)}$$

$$= \frac{18}{180 + 2 + 18}$$

$$= 0.09$$

etc.