Foundations of Computer Security

Lecture 30: Exploring Encodings

Dr. Bill Young
Department of Computer Sciences
University of Texas at Austin

Information Content and Bits

Information content is commonly measured in terms of *bits*. "Bit" has two connotations, *which are not the same*:

bit_1: a binary digit (discrete);

bit_2: a quantity of information (continuous).

The *information content* of a message is measured in bit_2s and the *capacity* of a channel in bit_2s per second (bps).

In general, the best way of transmitting (encoding) a series of messages is that way that minimizes the number of bit_2s required, on average.

Finding an Encoding

Four bits are adequate to encode 16 possible messages: $M_0, \ldots, M_{15}$:

Msg	code	Msg	code
M_0	0000	M_8	1000
M_1	0001	M_9	1001
M_2	0010	M_{10}	1010
M_3	0011	M_{11}	1011
M_4	0100	M_{12}	1100
M_5	0101	M_{13}	1101
M_6	0110	M_{14}	1110
M_7	0111	M_{15}	1111

Call this *the naïve encoding*.

- *Can we do better for one message? What would that mean?*
- *How about transmitting n messages, each of which is one of 16 possible values?*

Probability and Encoding

Suppose you need to send 1000 messages, each of which can be one of 16 possibilities. But *on average* 99.5% will be message 10.

Question: Does it still require $4 \times 1000 = 4000$ bits to send your 1000 messages?

Answer: It is possible to come up with an encoding that will do better on average than the naïve encoding.

Note, when we talk about sending 1000 messages, we've gone from talking about the information content of a message to talking about that of a *language*.

A Better Encoding

Use the following encoding:

Msg	code	Msg	code
M_0	10000	M_8	11000
M_1	10001	M_9	11001
M_2	10010	M_{10}	0
M_3	10011	M_{11}	11011
M_4	10100	M_{12}	11100
M_5	10101	M_{13}	11101
M_6	10110	M_{14}	11110
M_7	10111	M_{15}	11111

Given 1000 messages, on average 995 of them will be message 10, and 5 will be other messages. This encoding takes $995 + (5 \cdot 5) = 1020$ bits or 1.02 bits per message.

Some Observations

- Our encoding is pretty good, but can we do even better? Is there a limit to how well we can do?
- Computing the number of bits per message depends on knowing the *prior probabilities*—how often each message appears in an arbitrarily long sequence of messages.
- The "on average" part is important; some sequences would be less efficient under our encoding.
- We used the "naïve encoding" as our benchmark, but there are much worse encodings.
- Is it possible to find an *optimal* encoding? What would that mean?

Lessons

- "Bit" has two distinct meanings that are easily confused.
- For any language, one can find a naïve encoding that will work, but it's often possible to do better.
- "Doing better" means using fewer bits, on average, to transmit messages in the language.

Next lecture: Languages and Encoding