The Message Engine

by Tom Rutledge

Today, machines of all kinds are today sending and receiving messages with ever-increasing complexity, subtlety and depth. At the same time, the quality of human communication has hit an all-time low.

New media enable the exchange of ever more bytes, but the benefits are localized, held captive in friends' lists and subreddits. When the cliques collide, cacophony results.

The business world has its own version of the chaos, with specialization and complexity turning meetings between factions into a Tower of Babel. As a professional investor, I have had a front row seat. On a typical day, a parade of strangers bearing ideas comes through my door. They pitch. I scratch my head at the torrents of jargon and general disorder. Are they being intentionally vague, and evasive? Or are they just immersed in their own worlds and unaware of others'? It's hard to know.

This is bad. Repeated incompetent acts of communication have a serious human and economic cost. Good deals need to get done.

So, since machines have had an awfully good run in the 21st century, why don't we ask how they got so good at communication?

Say hello to Claude Shannon. Shannon was a 20th century genius who, among many achievements in science and engineering, is the undisputed founder of information theory: a body of scholarship and engineering that has something to say about almost every human activity.

In his 1948 paper "A Mathematical Theory of Communication," Shannon formalized communication in a way that allowed him to deploy some heavy-duty mathematics. He laid out the principles for a general communication system—what I will call a "message engine." Over the years, Shannon's work has helped engineers streamline and optimize phone calls, Netflix original series, Tinder swipes and Tweets.

As it turns out, Shannon is just the man to tune up the human message engine. By stripping communication to its essence, Shannon reacquaints us with its basic principles. Logic, brevity and clarity apply to machines as well as to people--and despite an apparent cultural chasm between the Technorati and the Literati, so do more allusive techniques like literary devices. In surprising ways, engineering concepts correspond elegantly to the more artistic and multimedia elements of communication. They all make the engine work.

Let's walk through the essential elements of Shannon's message engine, interpret what they mean for people, and identify what makes a message either work, or fail.

THE SIGNAL, CODER AND DECODER

You start with a message.

That message is turned into a signal that can be sent over a communication channel. Then the signal is turned back into a message. The transformations from message-to-signal and then signal-to-message are performed by a Coder and a Decoder, respectively. A familiar case is when the Coder turns English words into Morse Code, that digital string is sent over a telegraph wire, and then the Decoder turns the code back into words.

If any part of a message is going to reach the Destination, the Coder and Decoder have to overlap. A shared Coder/Decoder, or Codec, is the ideal. A Codec ensures that the encoding process is done and then undone in the same way.

In human terms, sender and receiver need some common vocabulary. Simple.

And yet, this is where most messages die. Codecs really only exist in the machine world. Human senders have special information to deliver. Human receivers understand different parts of messages. Despite this, senders routinely speak to audiences the way they speak to peers, taking for granted a shared Codec.

"Know your audience" is the oldest rule in your 8th grade English teacher's rulebook. But if it's not possible to know the audience well, the message-maximizing choice is a Coder likely to match up the audience's Decoders. A great choice is plain, concise English, a global *lingua franca* for the business world.

NOISE AND EQUIVOCATION

"Noise" sounds like, well, a sound--the crackly radio static obscuring FDR's Pearl Harbor speech. For Shannon, noise is the thing that causes the signal to change during transmission. Look at it that way, and Coder/Decoder difficulties can be considered noise. If I speak only Spanish and you speak only Farsi—or if I'm a programmer and you're a drama critic—my message may not succeed.

Success is not binary, however. You may know a little of my language, or recognize some word roots, or my facial expressions, or body language. When referring to the ambiguity in a signal, Shannon hijacks a non-technical term: equivocation. His math quantifies equivocation as sent information minus received information. The equivocation is what is lost or ambiguous upon receipt.

Can we use the ambiguity of the message to send more information? Yes. I can send you a brief message that blossoms into lots of information in your brain. But we are entering a semantic minefield. We're taking message risk. This requires a skilled communicator and the assistance of the receiver.

DATA COMPRESSION

Here's how machines compress data: the letter "e" 456 times in a row becomes the shorter phrase "456 e's."

People do it differently. Instead of "National Basketball Association," they say "NBA." This means the league, as well as a host of associations, experiences, memories, and facts. All that data—some of it shared by many, some of it unique to you--is "compressed" in the acronym.

Literary data compression techniques are numerous. Jargon was noise in the example above, but it can be data compression between two bond traders, or a pilot and a co-pilot. Examples, analogies, metaphor, simile, even pronouns--all invite receivers to invoke the wealth of information stored in their brains.

There is, however, one shared abstraction that towers above all others. Nothing compares to a story. Stories are supercharged fuel for the human message engine.

What's a story? Washington & Jefferson College's Jonathan Gottschall suggests that stories are algorithms based on three inputs: (a) character, (b) predicament, and (c) attempted extrication. Once your brain recognizes the form of a story, it forges relationships among the inputs, while filling in additional background and atmosphere.

The algorithm works with lightning efficiency on simple information. You know exactly what to do with (a) hero (b) damsel tied to train tracks (c) hero untying the ropes.

It also works with a rich, complex film like *The Godfather*. If you weren't continuously arranging the movie's gigabytes of data as a story, your brain would be drowning. The story archetype gives the information order and meaning, building a vessel for the flood of information still to come.

Whether warranted or not, good stories also have automatic credibility. They use the sequence, rhythm and contour found in the scientific method (question-hypothesis-prediction-testing-analysis) and in logical constructs like syllogisms (major premise-minor premise–conclusion). A good story feels true.

CHANNEL MAXIMIZATION

"Bandwidth" is the key technical term here, and even the barely tech-savvy know it. Is the wire big enough to carry the message, and is the message small enough to fit?

We may have thought that Shakespeare's "brevity is the soul of wit" was a mere aesthetic observation. Now we see its technical roots. Messages have clarity and power only if they are delivered intact. If there is insufficient bandwidth—be it human or electronic—the message arrives garbled.

A message that comes too fast is overwhelming. Think of the fast-talker, or the expert in an alien topic who won't allow questions or define terms. Maybe we get pieces of the message, or maybe the signal devolves into noise. Paradoxically, more information actually results in less meaning.

This is yet another argument for brevity and simplicity. Plain English, once again, is your friend.

However, there's a cheat.

Over the past several decades, machine bandwidth has become cheaper and more plentiful, seemingly by the hour. While humans' capacity for certain modes of communication is limited, we have five senses, powerful nervous systems, and very large brains. We can take in multi-media information at an incredible rate. In this particular way, human beings are a good match for machines.

Still, message risk is unavoidable. Taking large, semantically unwieldy bundles of information and make them rich messages requires great skill, rare gifts, and the right audience.

ERROR CONTROL

Shannon has a very simple approach for preventing message errors: Say it twice.

Your 8th grade English teacher's pal, the debate coach, had the same advice: "Tell them what you're going to tell them, tell them, then tell them what you told them."

Engineers have other tricks. Before you correct an error, you have to detect it. One method: add the digits up in a digital signal. If the sum is even, signify that by adding a 0-bit to the end. If it's odd, add a 1-bit to the end. The receiver can then compare last bit—the "parity bit"—to the sum of the message's other digits. If they don't match, there's an error.

We have parity bits in real life too, clues within the message about the accuracy of the message. Is the message logical, plausible, consistent? Do the messenger's reputation, skill and knowledge suggest reliability? The finely tuned human nervous system can often divine the truth from a presenter's body language.

DESTINATION

We're at the finish line now. The message is delivered. Did something good happen?

Probably not yet. But some of this may happen eventually:

- Since the process of communication resembles other technical processes, technical specialists will begin to understand and participate in communication strategy.

- Techies and their communication professionals will start speaking the same language about their acts of communication. They'll have a Codec.
- Techies and communicators will start having higher quality arguments.
- Communication professionals will have to start explaining why things work, not just insist that they will work.
- Techies will start listening to these explanations.
- Communicators' dogma will be exposed as dogma.
- Techies won't be able to dismiss good communication strategy as dogma.

None of the principles laid out here diminish the value of a great communicator's gifts. There is plenty of room for artistry and technique—albeit in the guise of data compression, channel maximization and error control.

There's no guarantee that this approach will transform all message engines into well-oiled machines. But maybe we'll get a few more deals done.

CASH RULES EVERYTHING AROUND ME

But wait. There may be more.

After Shannon published his seminal paper, scholars from almost every discipline looked for ways to adopt his model. Perhaps this was to be expected since, in some respects, every scholar traffics in "information." Shannon had left the door ajar for this kind of cross-pollination since, for one of the 1948 paper's constructs, he drew on the concept of entropy as defined in thermodynamics. But he offered only discouragement to interlopers. "The establishing of [new] applications is not a trivial matter of translating words to a new domain," Shannon said, "but rather the slow tedious process of hypothesis and experimental verification."

Shannon made one major exception. In 1956, John L. Kelly Jr.—another mid-century math genius entrenched in the world of giant computers, Bell Labs, MIT and card counting—wrote a paper linking information theory to gambling. Kelly noticed the math similarities between the problems of (a) deciding how much to bet on a given risk given some information about the risk, and (b) determining the amount of information that can be successfully transmitted on a noisy channel, that is, a channel with some equivocation.

Shannon really liked this. He advised Kelly on the paper and encouraged him to publish it. The paper laid out what is now known in finance theory as the Kelly Criterion, a rule for allocating capital to risky propositions, whether at the blackjack table, in the insurance world, or in the stock market.

So, how about the message market? Maybe we should buy a round trip ticket from Shannon to Kelly and back again to Shannon, and import some tricks from finance to the business of sending messages. Messages provide information, and equivocation.

Investments provide returns, and risk. And Kelly showed us how the same math works for both.

I can envision a world where a message is evaluated like an investment:

- For a given message and communication system, what is the chance that I will deliver no information or bad information? That is, what is the Message Risk?
- What is the amount of information can I expect to deliver? That is, what are my Message Returns?
- Am I delivering enough information given the amount of risk I'm taking? That is, what is the Message Sharpe Ratio?
- Is my message so good that I'm delivering more information than I would expect given my Message Risk? That is, is my Message Alpha positive?
- Would I be better off sending different messages to exploit their uncorrelated results? That is, can I optimize my Message Portfolio?

Very ambitious goals, and the route to get there is uncertain. In this essay, we have incorporated a lot of qualitative, hard-to-measure elements. We exited off the Shannon highway long ago, leaving a lot of his rigor behind.

However, communication technology is advancing very fast. Relevant work is likely underway in the natural language processing labs of the great universities, or R&D-heavy companies like Google and Facebook. The semantic structures of various communication modes are under intense scrutiny, and thinkers may have developed metrics to evaluate Message Risk and Return. Or perhaps cognitive scientists are developing means of monitoring human nervous systems, flashing messages' performance scores on a laboratory scoreboard.

But even if the big brains of technology cannot assign precise measures, we can still consider the tradeoffs. Not knowing the exact numbers doesn't stop people in finance from talking about risk and return. It's a language (or more accurately, I confess, a data-compressing class of jargon) that frames the question and hastens progress toward an answer.

-END-

