



Technology: A simple guide to statistical document sorting technology

Forget the math--all you need is a basic understanding of the process

BY THOMAS LIDBURY, MICHAEL BOLAND

Many lawyers' eyes glaze over when we hear of Bayesian networks, concept clustering, predictive coding, suggestive coding, machine-assisted review, meaning-based coding, latent semantic analysis, probabilistic latent semantic analysis, Shannon's theory, the Markov blanket, de Finetti's theorem, latent Dirichlet allocation, Gibbs sampling and so on. Like Chevy Chase, most of us believed "there would be no math."

Can lawyers still hold to that view now that document collections are measured in gigabytes and terabytes, and sophisticated mathematical document sorting technology is going mainstream? Yes, we can. We only need a basic understanding of what the technology does so that we can know how to effectively use it in a defensible workflow.

While each e-discovery software developer and vendor is proud of its own secret sauce, the basic ingredient in advanced e-discovery technology is some version of an algorithm that identifies sub-groups of documents that focus on a concept unique to the rest of the dataset. The algorithms assign a differentiating weight to the words and phrases in the dataset to find correlations between documents.

The more ubiquitous a word or phrase is, the lower weight it will be assigned. Conversely, the assigned weight increases with rareness in the dataset. The algorithms also identify patterns and associations between words and phrases so as to group documents that refer solely to "terrorism" with others that refer solely to "suicide bombing," while not grouping documents about edible "apples" with those about "Apple" computers. Mathematicians may quibble with some of this "caveman lawyers" summary, but this is all that most of us will ever need to know about the math. As U.S. Magistrate Judge Andrew Peck has written and reiterated at conferences, the bench is no more eager than Chevy Chase to hear about the math; just the process.

Developers initially touted this technology for early case assessment purposes because the users hadn't devised workflows to use it for primary document culling. But users now have developed defensible workflows that have been in use for most of the past decade. These workflows can be divided into two categories:

1. The first approach is to sort the entire dataset into clusters before humans look at the documents, review the clusters (without reviewing each document) to separate those that do not promise to contain relevant documents from those that do, and review the documents only in the latter.

A real-life example will illustrate this approach. A document collection comprising email and loose electronic files pulled from scores of employees totaled approximately 8 million documents and an unknown number of pages. These documents clustered into about 14,000 concept folders. For a week, two associates reviewed the concept clusters and ruled out 90 percent as not likely to contain relevant documents. Contract lawyers received the remaining 800,000 documents for review (keeping document families and email strings together so that the same reviewer would see them at the same time).

To achieve greater efficiency than a purely linear—document-by-document, page-by-page—review, the team was instructed to leverage the modern features of the review tool, such as email threading, near-duplication and sorting by metadata fields such as title, sender and date. The per-file cost of the review turned out to be about 19 cents measured against the original 8 million files, and about \$1.90 measured against the 800,000 files. This represents a dramatic improvement over the traditional process.

2. A second approach is to use a sample set of documents, sorted into relevant and irrelevant groups, to teach the algorithm what sorts of documents are wanted and unwanted. The algorithm then finds a subset of the whole that it thinks are also relevant based on the sample set. The team reviews this subset for relevance, and the algorithm learns again from the growing sample of coded data.

Some vendors' processes perform these iterations in large steps, whereas others perform them in closer to real time to prioritize automated document assignments. With either process, at some point the percentage of relevance found in the review sets drops off to a point at which the lawyers can exercise a judgment to stop and to not review the remainder.

There is a common misconception that the algorithms in some of these systems are trained on a very small subset of the whole, and once the lawyers are satisfied that the algorithm is smart enough, the algorithm codes the rest of the dataset predictively. That is not how it works. Each document that is coded to be produced comes from the iterative review sets that humans reviewed. Many of these technologies do suggest to reviewers what the algorithm thinks the coding should be, but these are just suggestions that the reviewers accept or reject as they complete their assigned review sets.

This technology is very powerful and defensible when combined with the right process. Users have developed good workflows to leverage this technology to very efficiently cull relevant from irrelevant documents. As new and mysterious as it may sound, it has been in real-world use in some of the biggest litigation in the country for at least a decade. And you do not need a Ph.D. to use it in your next case.