CHAPTER 3

A Tour of Technology-Assisted Review

Maura R. Grossman and Gordon V. Cormack

Introduction

Since the publication of our 2011 article, *Technology-Assisted Review in E-Discovery Can Be More Effective and More Efficient Than Exhaustive Manual Review*, in the Richmond Journal of Law and Technology (JOLT article),[1] the term "technology-assisted review" (TAR)—along with related terms such as "computer-assisted review" (CAR) and "predictive coding" (PC)—has been used extensively in writings, presentations, and legal documents to advance or assail a plethora of tools and methods aimed at decreasing the cost and burden associated with document review in electronic discovery (e-discovery). The conflation of diverse tools and methods under a single label has resulted in confusion in the marketplace, and inapt generalizations regarding the effectiveness (or ineffectiveness) and proper use of such tools and methods. In this chapter, we outline and illustrate—using a running example—the principal distinctions among the various tools and methods that we consider to be variants of TAR, and differentiate them from other tools and methods, such as concept search, clustering, visualization, and social network analysis, which we do not consider to be TAR. While e-discovery tools fall on a continuum in terms of their utility for search, analysis, and review, we describe certain components that, in our opinion, are essential for a tool to be considered "technology-assisted review."

Distinguishing Among Search, Analysis, and Review

Search, analysis, and review are different tasks with different objectives. The objective of search is to find enough documents to satisfy an information need, such as the answer to a question or support for a proposition. The objective of analysis is to gain understanding from available data, so as to support decision-making. The objective of review is to classify all documents in a collection as meeting—or not meeting—certain criteria, such as responsive to a request for production, subject to attorney–client privilege or work–product protection, or available for disposal. Following information retrieval practice, in this chapter, we refer to documents meeting the review criteria as "relevant," and those not meeting the criteria as "nonrelevant," although in the legal domain, this distinction is sometimes referred to as "responsive" and "nonresponsive."

While there is some overlap between the tools and methods used for search and analysis, and those used for review, we apply the term "technology-assisted review" or "TAR" only to tools and methods used specifically for review.

Almost everyone is familiar with Web search engines like Google, Bing, and Yahoo!, and the ease

with which they may be used to find a few relevant documents in response to an information need. Search engines, however, are much less amenable to the task of finding all, or nearly all, relevant documents, as required for review.

Analytic tools such as clustering, visualization, and social network analysis may yield insights leading to the discovery of relevant documents, but, like search engines, they are not in and of themselves well-suited to review.

Search, analytic, and review tools are often combined in commercial document review platforms to provide a suite of options to deal with the large collections of data that are commonly encountered in e-discovery in legal and regulatory investigations or proceedings. The components of these offerings are not always susceptible to easy labels because, in reality, tools used for search, analysis, and review can fall on a continuum, and many e-discovery approaches rely on ad-hoc, hybrid methods that move in iterative cycles among search, analysis, and review.

Defining Technology-Assisted Review and Distinguishing Rule-Based from Supervised Machine-Learning Approaches to TAR

Methods used for technology-assisted review may be broadly classified into those employing rule bases and those employing supervised machine learning. A rule base consists of a large set of rules, composed by one or more subject-matter and rules-construction experts, to separate documents meeting the criteria for relevance, from those that do not. These rules may take the form of complex Boolean queries—specifying combinations of words and their order or proximity—that indicate relevance or nonrelevance, exceptions to those rules, exceptions to the exceptions, and so on, until nearly all documents in the collection are correctly classified. Supervised machine learning, on the other hand, infers, from exemplar documents, characteristics that indicate relevance or nonrelevance, and uses the presence or absence of those features to predict the relevance or nonrelevance of other documents.

Our 2011 JOLT article compared the effectiveness of a rule-based TAR method (H5) and a supervised machine-learning TAR method (Waterloo) to that of human review, concluding that both of these TAR methods were able to equal or exceed the effectiveness of human review—as measured by recall, precision, and F_1—with a small fraction of the human review effort. Since its publication, a number of service providers have cited the JOLT article as support for the efficacy of their own tools or methods, under names such as "technology-assisted review," "computer-assisted review," "assisted review," "language-based analytics," and "predictive coding," among others. Many of these approaches, however, bear little resemblance to the two methods studied in our JOLT article.

It is important to note that there is no standards-setting or regulatory body controlling the application of these terms to various tools and methods, and therefore, there is no guarantee that a tool labeled "TAR," or "predictive coding," or "language-based analytics" employs any particular method, or that the method it does employ—whether aptly named or not—is effective for review. Nevertheless, because a common vocabulary is essential to rational discourse, to this end, in 2013, we published a glossary of terms in Federal Courts Law Review (the Glossary),[2] which we will augment in this chapter, to taxonify current methods used for technology-assisted review. (This glossary also appears as the appendix of this book.) The Glossary defines technology-assisted review (TAR) as:

> A process for Prioritizing or Coding a Collection of Documents using a computerized system that harnesses human judgments of one or more Subject Matter Expert(s) on a smaller set of Documents and then extrapolates those judgments to the remaining Document Collection. Some TAR methods use Machine Learning Algorithms to distinguish Relevant from Non-Relevant Documents, based on Training Examples Coded as Relevant or Non-Relevant by the Subject Matter Experts(s), while other TAR methods derive systematic Rules that emulate the expert(s)' decision-making process. TAR processes generally incorporate Statistical Models and/or Sampling techniques to guide the process and to measure overall system effectiveness.

We intended this definition to exclude search and analytic tools per se because they do not extrapolate relevance judgments to the collection. Certainly, one could use a search tool for review, by deeming all nonretrieved documents to be nonrelevant. Similarly, one could use a clustering tool for review by deeming all members of each cluster, as a group, to be either relevant or nonrelevant. However, neither approach has been shown to be effective for review; thus, to label an underlying search engine or clustering method as a "TAR" tool would, in our view, weaken the definition of TAR so as to render it meaningless. We suggest that it would be more fruitful to label methods as "TAR" only if they rank or make affirmative predictions about the relevance or nonrelevance of documents in the collection based on human judgments that are applied to a subset of the collection. Rule-based and supervised machine-learning methods meet this definition because they generate rules or mathematical models that discriminate between relevant and nonrelevant documents.

The Path from Keyword Search to TAR

When one tries to delineate the precise boundary between what is and is not TAR, it becomes apparent that this distinction is not entirely clearcut; rather, methods fall on a continuum from those geared more toward search and analysis, to those geared more toward review. Accordingly, it seemed to us that the best way to explain TAR would be to begin with keyword search and to walk through the process of adding additional technological components that start to convert a search method into technology-assisted review. We thought it might also be useful to demonstrate these different components using a common example to illustrate their relative effectiveness, or ineffectiveness, for review.

A Running Example: Fantasy Football

To illustrate the evolution of methods from simple keyword search to TAR, we use as a running example Topic 207 from the TREC 2009 Legal Track, which was reprised in the TREC 2010 Legal Track, and used again, as a practice example, in the TREC 2015 Total Recall Track.[3] Topic 207 requires the identification of,

> [a]ll documents or communications that describe, discuss, refer to, report on, or relate to fantasy football, gambling on football, and related activities, including but not limited to, football teams, football players, football games, football statistics, and football performance[,]

from a set of documents collected from Enron Corporation by the Federal Energy Regulatory Commission. The particular collections used for TREC 2009 and TREC 2010 are no longer publicly

available; for this example, we used the collection supplied to participants with the baseline model implementation (BMI) for the TREC 2015 Total Recall Track.[4] This collection contains 723,386 documents, of which (unbe-knownst to our hypothetical searcher) 7,798, or about 1.1 percent, are relevant to Topic 207.

Basic Keywords or Search Terms

To identify documents concerning fantasy football, our hypothetical searcher might simply begin by selecting all documents containing the word "football"—a task that is readily accomplished using the search feature in most document review platforms. Most likely this search would yield a substantial number of documents concerning actual football teams, players, games, and statistics, but it would also be likely to identify documents with metaphorical, humorous, and other uses beyond those specified by Topic 207. In our example, again unbe-knownst to our searcher, the term "football" selects 4,319 documents from the corpus, of which 3,480 are relevant, and 839 are not. In other words, a search for "football" has 44.6 percent recall (where "recall" is defined as the fraction of all relevant documents in the collection identified by the search; in this case 3,480/7,798 = 44.6 percent), and 80.6 percent precision (where "precision" is defined as the fraction of the documents identified by the search that are relevant; in this case 3,480/4,319 = 80.6 percent).

Without additional testing or inquiry, our searcher would have no way of knowing what level of recall and precision her search had achieved, and therefore no tool other than her intuition to assess the adequacy of her search.[5] Human intuition concerning the adequacy of keyword search, however, has been shown to be woefully inadequate. In a seminal 1985 study,[6] Blair and Maron asked skilled paralegals, supervised by lawyers, to use search terms to find documents responsive to each of 51 requests. Although the lawyers and paralegals believed they had achieved an adequate result—which they defined to mean having achieved recall of at least 75 percent—they had, in fact, achieved recall of only 20 percent, on average. No more recent study that we are aware of has shown human intuition about search terms to have improved since then.

Control Sets

The first step in the transition from search terms to TAR would involve the addition of a principled mechanism to ensure that a substantial majority of the relevant documents were identified by the process. To this end, many processes rely on statistical sampling to estimate how many relevant documents there are to be found in the collection, how many relevant documents have been identified by a particular search or review strategy, and thus, the recall and precision of that strategy. A common (but not universal) approach is to use a random sample taken at the outset of the process, known as a "control set," to estimate these quantities, and thereafter to guide the process, or to validate its effectiveness.

Each document in the control set is reviewed and coded as relevant or not by a subject-matter expert (SME). The fraction of relevant documents in the control set provides an estimate of the fraction of relevant documents in the collection, with some margin of error. This proportion is referred to as "prevalence."[7] Provided that the control set is independent of the search effort—that is, no information about the search is used in choosing and coding the control set, and no information about the content or coding of the control set is used in devising the search strategy—the control set

may be used to yield valid estimates of the recall and precision of the search or review process.[8]

For our running example, we chose a control set consisting of 3,000 random documents. We found that 31 of the 3,000 documents (1.03 percent) were relevant to Topic 207, and therefore assume that approximately 1.03 percent of the collection (7,473 documents) is likely to be relevant. This estimate is reasonably close to the true value of 1.08 percent (7,798 documents). Furthermore, 17 of the 20 "hits" for the term "football" in the control set are relevant, yielding a precision estimate of 85.0 percent, and a recall estimate of 64.5 percent. While the recall estimate is considerably higher than the true value of 44.6 percent, it is still within the margin of error one might expect for a sample of this size.

Combining Search Terms

A single search term is rarely sufficient to capture substantially all relevant documents in a collection. For Topic 207, the term "football" identified 44.6 percent of the responsive documents; the problem remains of how to identify the remaining 55.4 percent. Assuming that it is determined—either by intuition, statistics, or otherwise—that it is necessary to find more relevant documents, an obvious choice would be to add additional search terms; a less obvious choice is *which* search terms to use. Table 3.1 contains 120 possible search terms our searcher might incorporate into her keyword search. Which would you pick?

If we knew, for example, that the term "NFL" selected 1,666 documents, of which 1,542 were relevant, we might be inclined to include this term. But 1,261 of these documents were already selected by the term "football," so the net effect is that "NFL" identifies only 281 new responsive documents, at the expense of 106 new nonresponsive ones. Knowing this, should our searcher still include the term "NFL"?

TABLE 3.1 120 Candidate Search Terms Related to TREC Topic 207 (Fantasy Football)

football*	game*	pick*	bass	bowl	nfl	agent	week	sport*	miami
superbowl	qb	wager*	ticket*	defens*	ffl	big	sooner*	fan	giant*
cheatsheet*	lenhart	bid*	tease	arnold	watch	team*	rb	bronco*	lsu
season	lamphier	oct	go	ut	texas	dallas	cuilla	bet	bets
trade	trades	tenn	league	espn	texan*	send	pool	raider*	point*
play*	will	tennessee	sportsbook	quarterback	murrel	minn	horn	commission*	boy
taylor	roster	phillip	phily	mike	mail	hiemstra	eric	want	super
harry	dsc	college	win	wins	tb	faulk	dean	brown	yard*
reagan	ram	rams	osu	mnf	mack	longhorn	kansas	coach	brother
brad	block*	sunday	schroeder	schafer	san	saint	ryder*	playoff	place
oklahoma	niner*	nick	michael	margaux	helsinki	height	george	gary	garcia
draft	denver	dawson	constance	colorado	carolina	burg	bledso	aggie	aggies

* Refers to a root extender, which permits the search to identify alternate forms of a root term; for example, a search for "teach*" would identify documents containing the terms "teach," "teaches," "teaching," "teacher," or "teachers."

As more search terms are used, it becomes increasingly difficult to choose additional terms that select many new, relevant documents, and few nonrelevant ones. There are approximately 10^{40} (i.e., 1 followed by 40 zeros) possible combinations of the terms in Table 3.1. The chance of finding the best combination of them, through intuition alone, is remote.

We could use our control set for this purpose, but once we use the control set to aid our search strategy, it is no longer independent, and therefore ceases to be a valid statistical sample for estimating recall and precision for validation purposes. When the selection of search terms is done with the benefit of feedback derived from the control set, terms may be selected that correctly identify the relevant documents in the control set, but not necessarily in the rest of the collection. This phenomenon is known as overfitting.[9]

When a set of documents is coded and used to guide the selection of keywords—or any other aspect of the search or review process— such a set is more properly referred to as a "training set." The systematic use of a training set is a key factor that distinguishes TAR from search.

For the purpose of illustrating a carefully crafted set of keywords, we selected the terms in Table 3.1, and ordered them by their estimated effectiveness (from left to right, by row), using an algorithm and a training set.[10] While we have no empirical evidence to support our claim, we posit that a human reviewer relying solely on her intuition would be hard-pressed to come up with a better list; we therefore use this list to model the best possible result that could be achieved with reasonable effort. But, even given this list of terms, the question still remains: How many of the "best possible" terms should be used for the search?

As we have seen, the term "football," alone, yields 44.6 percent recall and 80.6 percent precision. The terms "football" and "game," together, yield 67.0 percent recall and 28.9 percent precision. In other words, it is necessary to review 18,092 documents—13,773 more than the 4,319 selected by "football" alone—to raise recall from 44.6 percent to 67.0 percent. In the extreme, the 120 search terms combined select 570,466 documents, or about 80 percent of the collection, achieving 99.7 percent recall, but only 1.4 percent precision.

Table 3.2 illustrates the recall-precision trade-off that ensues from using various combinations of

search terms. The first column (Search Terms) shows increasingly large sets of the "best possible" search terms selected from Table 3.1. The second column (Hits) shows the number of documents selected by the set of search terms, representing the amount of effort that would be required to review all documents selected by the search terms. The third and fourth columns (Recall and Precision, respectively) show the recall and precision of the search.

TABLE 3.2 Hit Counts, Recall, and Precision for Various Combinations of High-Quality Search Terms

Search Terms	Hits	Recall	Precision
football	4,319	44.6%	80.6%
football, game	18,092	67.0%	28.9%
football, game, pick	35,444	73.8%	16.2%
football, game, pick, bass*	42,288	82.1%	15.2%
football, game, pick, bass, bowl	42,737	84.3%	15.4%
football, game, pick, bass, bowl, NFL	42,860	85.1%	15.5%
football, game, pick, bass, bowl, NFL, agent	66,194	85.7%	10.1%
* "Bass" refers to the surname of the employee who oversaw the fantasy football league at Enron Corporation.			

It is easy to see that using more search terms increases recall at the expense of increased effort (and, correspondingly, decreased precision). No matter how carefully search terms are selected, high recall can be achieved only at the expense of low precision, which translates to greater review effort.[11] It is therefore unlikely that a human, selecting search terms using her intuition alone, could guess a set of search terms that would yield higher recall and higher precision than any of the searches proposed in Table 3.2.

Boolean and Proximity Operators

The search tools incorporated in most document review platforms allow a searcher to combine search terms using Boolean operators, such as "AND," "OR," and "BUT NOT," as well as proximity operators, such as "FBY [followed by] *n*" or "WITHIN *n*." In this chapter, we refer to a combination of one or more search terms using Boolean operators as a "Boolean expression," and a combination of one or more search terms using Boolean and proximity operators as an "extended Boolean expression."

The simplest Boolean expression is a single search term that selects all documents containing it, but Boolean expressions can consist of more than one search term. Two Boolean expressions, E_1 and E_2, may be combined as follows:

- "E_1 OR E_2" selects all documents that contain *either* "E_1" *or* "E_2";
- "E_1 AND E_2" selects all documents that contain *both* "E_1" *and* "E_2";
- "E_1 BUT NOT E_2" selects all documents that contain "E_1," *but only if they do not contain* "E_2."

Boolean expressions may be extended by including pairs of search terms, T_1 and T_2, combined as follows:

- "T_1 FBY *n* T_2" selects all documents containing the term "T_1" followed by the term "T_2," separated by no more than *n* words;
- "T_1 WITHIN *n* T_2" selects all documents containing both terms "T_1" and "T_2," in either order, separated by no more than *n* words.

In the following discussion, we refer to Boolean and extended Boolean expressions collectively as "Boolean expressions," or, when used for searching, as "Boolean queries." The use of Boolean queries to select documents is referred to as "Boolean search."

Boolean queries allow searchers to particularize the documents selected by search terms. For example, if the term "game" selects a large number of documents about video games, as opposed to football, our searcher might use the Boolean query "game BUT NOT video." Constructing a Boolean query to achieve high recall and high precision is a formidable task that requires testing and iteration. A set of Boolean queries that is systematically constructed with the objective of selecting all and only the relevant documents is a form of rule base, discussed later in this chapter.

Relevance Ranking

As defined in our Glossary, relevance ranking is "[a] search method in which the results are ranked from the most likely to the least likely to be Relevant to an Information Need. . . ."[12] A familiar example of relevance ranking is Web search: A user types a few keywords into a search engine such as Google, Bing, or Yahoo! and receives, as a result, the top-ranked documents from millions of possible hits. These top-ranked documents are those that the search engine's relevance-ranking algorithm determines are most likely to contain the information that the user is seeking. Some (but not all) commercial document review platforms include relevance ranking, but, for reasons that are not entirely clear to us, relevance ranking is seldom used in e-discovery.

Using relevance ranking with a broad set of search terms (such as those in Table 3.1) is an effective alternative to the use of a narrow, carefully selected set of keywords alone (such as those in Table 3.2). To show this, we fed the 120 search terms listed in Table 3.1 to a well-known relevance-ranking algorithm (i.e., BM25, as implemented by the Wumpus Search software[13]) and achieved results—shown in Table 3.3—which are nearly as good as those achieved by our carefully crafted search terms using an automated method.[14] Table 3.3 shows the recall and precision that would be achieved if various numbers of the top-ranked documents were selected from the collection. The first column (# Top-Ranked Documents) reflects review effort, and is directly comparable to the second column (Hits) of Table 3.2. We can see, in our relevance-ranking example, that selecting the top-ranked 18,092 documents achieves 65.3 percent recall and 28.2 percent precision, whereas, in our keyword-search example, selecting 18,092 documents using the search terms "football" and "game" achieves 67.0 percent recall and 28.9 percent precision.

When using relevance ranking, the key decision that needs to be made—beyond identifying a broad set of search terms—is how many of the top-ranked documents should be selected so as to achieve the desired trade-off between recall and precision (i.e., completeness versus review effort). Although the precise recall and precision measures shown in Table 3.3 can be known only with clairvoyance, they

could be estimated using a control set.

TABLE 3.3 Relevance Ranking for 120 Search Terms: Recall and Precision at Various Cut-Off Points

# Top-Ranked Documents	Recall	Precision
4,319	41.4%	74.8%
18,092	65.3%	28.2%
35,444	75.1%	16.5%
42,860	77.7%	14.2%
66,194	83.7%	9.9%

Relevance ranking is a fully automated technique that performs about as well as the best combination of keywords that we were able to construct using a combination of technology, knowledge of the collection, and hindsight, and almost certainly better than any combination of keywords whose selection was based solely on human intuition. Nonetheless, the use of relevance ranking remains relatively uncommon in the e-discovery community. Some of the resistance to its use may be due to the fact that the mechanism behind relevance ranking is not as easily understood as the mechanism behind keyword search, a property sometimes referred to as "transparency." It is important to note, however, that familiarity with the nuts and bolts of a method does not confer any information about the *effectiveness* of that method for its intended purpose. The same sense of familiarity may have led the lawyers and paralegals who participated in the Blair and Maron study to believe they had achieved 75 percent recall when they had achieved only 20 percent.

Similarity Search

Moving further down the path from search tools to TAR, there are a number of methods that seek to identify documents that are similar to a given document, or to one of a given set of documents, for some definition of "similar." These tools are often informally referred to as "find [me] more like this." In essence, the document or documents of interest assume the same role as search terms in a keyword search, and the challenge of choosing a good set of documents—often referred to within the context of similarity search as a "seed set"—parallels the challenge of finding a good set of search terms. There is a common misconception that knowledge of the seed set confers information about the effectiveness of the similarity search, much as there is an unfounded belief that knowledge of the search terms conveys information about the effectiveness of the keyword search. In both cases, the searcher does not know what she does not know, and transparency regarding the search mechanism does not change that. Replacing the unprincipled choice of search terms with the unprincipled choice of seed documents is no less a game of Go Fish.[15]

Many notions of "similar" are employed in similarity search tools. Similarity may be expressed in terms of a similarity measure (e.g., X percent) that indicates the degree of similarity between two documents. Alternatively, similarity may be expressed as the "distance" between two documents, where a larger distance means less similarity. Similarity search identifies any document that is more similar (or nearer) to a given document than some threshold value. Most similarity search tools allow

this threshold to be adjusted, in much the same way that the cut-off for relevance ranking can be adjusted to vary the trade-off between recall and precision.

The most widely used similarity measures include cosine similarity and, more generally, the vector-space model (VSM), which are based on the words that appear in the documents (typically after stemming, stop-word elimination, and tf-idf (term frequency-inverse document frequency) weighting[16]).

Table 3.4 shows the results of a "find similar" search using cosine similarity and tf-idf weighting, using the 31 relevant football-related documents found in our control set as a seed set. It shows the recall and precision achieved when documents similar to one of the 31 seed documents are selected for various threshold values. When compared to Tables 3.2 and 3.3, Table 3.4 shows, for this particular example and this particular similarity measure, somewhat inferior results to those achieved by the well-crafted search terms or relevance ranking. In this example, for a review effort of 18,092 documents, cosine similarity achieves 49.1 percent recall and 21.2 percent precision—substantially inferior to the recall and precision results shown for the 18,092 Hits and # Top-Ranked Documents in Tables 3.2 and 3.3, respectively.[17]

TABLE 3.4 Similarity Search Using 31 Control-Set Seed Documents: Recall and Precision at Various Cut-Off Points

# Top-Ranked Documents	Recall	Precision
4,319	30.2%	54.5%
18,092	49.1%	21.2%
35,444	61.8%	13.6%
42,860	65.8%	11.9%
66,194	74.2%	8.7%
95,545	80.0%	6.5%

Classifiers

According to our Glossary, a classifier is "[a]n Algorithm that Labels items as to whether or not they have a particular property. . . ."[18] Within the context of e-discovery, classifiers are algorithms that yield a determination of relevant or nonrelevant. Classifiers may be constructed either manually or automatically.

A rule base is a manually (but systematically) constructed set of rules that is subsequently applied automatically to rank or classify documents as to their relevance. Thus, a rule base is simply a manually constructed classifier. In contrast to a rule base, a learned classifier is constructed automatically by a supervised machine-learning algorithm. The systematic construction of classifiers from training documents is the key factor that distinguishes TAR from search and analysis.

Whether manually constructed or learned, there are many different kinds of classifiers. A Boolean query could be used to form a simple classifier by deeming the "hits" to be relevant and the "misses" to be nonrelevant, or vice versa. Similarly, a classifier could be derived from a relevance ranking by

deeming the top (or bottom) *k* results to be relevant, and the rest to be nonrelevant, for some cut-off value, *k*. A classifier might also be derived from a similarity search by deeming all documents with similarity *s* or greater to a document in the seed set to be relevant, and the rest to be nonrelevant, or vice versa.

A particularly simple but effective type of classifier is a linear classifier, which is simply a score for each term (or more generally, "feature") that may occur in a document, with positive scores indicating relevance, and negative scores indicating nonrelevance. The score of the document is simply the sum of the scores for the terms or features it contains, and the document is classified as relevant if its score exceeds some threshold, *c*, and nonrelevant if its score is less than *c*. Linear classifiers may be constructed by hand, in which case they are rule bases, or by supervised machine-learning algorithms, such as support vector machines (SVMs), logistic regression (LR), or naïve Bayes (NB).[19] Some of these classifiers are transparent, in that the mechanics of their operation—but not their effectiveness— is exposed; most are opaque, either because of their complexity, or because they rely on sophisticated pattern matching, parsing, or inference algorithms.

Rule-Based Methods for TAR

The construction of an effective rule-based TAR system is an expert-intensive process, relying on one or more domain experts (also known as subject-matter experts, or SMEs), as well as one or more rules-construction expert(s) (also known as knowledge engineers, and typically persons with expertise in linguistics, statistics, and/or computer science). While it may be easy to understand operationally how simple rule bases—notably flowcharts and decision trees—work, it is not easy to predict their effectiveness at discriminating between relevant and nonrelevant documents.

The rule-based approach determined to be effective in our JOLT article relied on extensive collaboration among a team of subject-matter experts, linguists, statisticians, and document reviewers, to construct and validate a complex set of rules in the form of Boolean expressions. The rules-construction experts formulated a series of Boolean queries, and reviewed examples of hits and misses for each query. Where the hits were found to contain too many nonrelevant documents, supplemental queries were constructed to remove these false-positive documents; where the misses were found to contain too many relevant documents, supplemental queries were constructed to identify the false-negative documents. This process was continued until the overall set of queries yielded acceptable recall and precision, as estimated using a control set.

There is some debate in the legal field as to whether Boolean search constitutes TAR. As previously illustrated, a classifier can consist of Boolean expressions. This is not to say that any use of a Boolean query is TAR—it may constitute TAR if the method of construction is systematic and is based on the iterative examination of exemplar documents using a control set. In any event, the label "TAR" does not, in itself, imply that the classifier is effective.

A number of do-it-yourself linguistic tools are available in the market for rule-base construction. One simple method relies on the examination of the vocabulary contained in the document collection —an exhaustive index of the words it contains, with stop-words eliminated— and asks the searcher to identify those words that appear most relevant to the subject matter of interest, and those that appear unlikely to be relevant. Another asks the searcher to highlight terms or phrases of interest. Unless followed up by an iterative query-refinement process, and a method to gauge progress by

examining its effect on the documents in the collection, such approaches do not fit our definition of TAR.

Other do-it-yourself rule bases employ different kinds of classifiers, such as patterns or grammars. At the time of this writing, we are unaware of any scientific study demonstrating the effectiveness of any of these methods.

Similarity Search for TAR

Some of the most widely adopted TAR tools on the market rely on similarity search. Although commonly associated with supervised machine-learning methods, TAR methods based on similarity search closely resemble rule-based methods. The seed set represents a set of rules that determine which documents to deem relevant (due to their similarity to a relevant seed document) and which documents to deem nonrelevant (due to their dissimilarity to all relevant seed documents and/or their similarity to a nonrelevant seed document). Some TAR systems based on similarity search construct only a partial classifier, which is unable to classify certain gray-area documents as either relevant or nonrelevant.

As with untested search terms in Boolean rule-base construction methods, an initial seed set chosen intuitively (or at random) is unlikely to yield an effective classifier. It is necessary to apply the classifier to example documents, and, where the reviewer disagrees with the classifier's result, amend the seed set, either by adding new documents to the seed set, or by deleting or changing the coding of the seed document(s) responsible for the incorrect classification. This iterative process of sampling and revision continues until the classifier yields sufficient recall and precision.

While the mechanism of these similarity search tools is easily understood, as we have previously stated, the documents in the seed set—just like a list of search terms or Boolean queries—offer little insight as to the classifier's effectiveness.

Supervised Machine Learning for TAR

A number of TAR tools on the market—often referred to as "predictive coding" in e-discovery circles—employ an approach known in the information retrieval community as supervised learning, in which a machine-learning algorithm infers how to distinguish between relevant and nonrelevant documents, based on exemplar documents, each of which has been labeled by a subject-matter expert as relevant or nonrelevant. The entire set of exemplar documents is properly known as the "training set," although it is often incorrectly referred to as the "seed set." Typically, the learning method identifies combinations of features from the documents that distinguish relevance from nonrelevance; in many systems, the features consist of the words contained in the documents; in other systems, the features may include word fragments, phrases, concept clusters, punctuation, emoticons, or metadata.

Many supervised learning algorithms have been proposed. Among the most effective are support vector machines, logistic regression, bagging, boosting, random forests, and k-nearest neighbor (k-NN).[20] Common, but generally less effective, methods include naïve Bayes (colloquially known as "Bayesian"), nearest neighbor (NN, or 1-NN, which is essentially similarity search), and Rocchio's vector-space method.[21]

It is important to distinguish *supervised* learning methods from *unsupervised* learning methods.

Unsupervised methods do not use a training set, or any other human input as to what constitutes relevance. Such methods, which automatically group documents or document features without human oversight, include clustering, latent semantic indexing or analysis (LSI or LSA), and probabilistic latent semantic indexing or analysis (PLSI or PLSA).[22] These methods are not TAR tools in their own right, but may be used as a component of similarity search, or to derive features (e.g., concept clusters) for use in a supervised machine-learning method.

We have recently taxonified the various supervised machine-learning TAR protocols in use today into three major categories, referred to as "simple passive learning" (SPL), "simple active learning" (SAL), and "continuous active learning" (CAL).[23] SPL represents the most basic application of supervised machine learning. In SPL, all of the training documents are selected either at random, by the human operator of the TAR system, or through a combination of both methods. SAL extends SPL to allow the learning algorithm to identify exemplar documents to be reviewed, coded, and added to the training set. CAL repurposes the role of the classifier in supervised learning, abandoning the objective of creating, once and for all, the best possible classifier, in favor of constructing a series of disposable classifiers—each with the sole purpose of identifying more relevant documents for review—and continuing to construct classifiers— each trained on all documents reviewed to date—until substantially all relevant documents in the collection have been reviewed. These core protocols are compared and contrasted in the following sections and are set forth in Table 3.7.

Active Versus Passive Machine-Learning Methods

In taxonifying supervised machine-learning protocols for TAR, it is important to distinguish between the roles of teacher and learner. The *teacher* is the human operator of the TAR system, while the *learner* is the machine-learning algorithm. The term "passive learning" means that the learner (i.e., the algorithm) is passive in the selection of training examples; it plays no part in that selection and uses for training only those exemplar documents selected by the teacher, or through random selection. By contrast, the term "active learning" means that the learner (i.e., the algorithm) actively selects some—usually most— of the documents from which it will learn. These documents are coded by the teacher and then fed back to the algorithm for further training. In other words, *passive versus active learning is assessed from the perspective of the learner*—the algorithm—and its level of involvement in selecting the documents it will be given for training purposes; the algorithm is either an active or a passive participant in the selection process.

Passive learning must be distinguished from passive teaching. In passive teaching, the teacher plays no role in selecting the training examples for the algorithm; instead they are selected either at random, by the learner, or through a combination of both methods. By contrast, the term "active teaching" means that the teacher selects some—or most—of the documents used for training using her judgment, for example, by selecting training examples identified during witness interviews, through ad-hoc search methods, from prior productions, and even through the creation of "synthetic documents" that would be of interest, if they were to be found in the collection.

One principal factor distinguishing different TAR protocols is how the training documents are chosen: by the teacher, by the learner, at random, or through some combination of these approaches. While some controversy persists as to which method of selecting training examples is preferable, the question is amenable to empirical evaluation.[24]

The argument for exclusively randomly selected training examples appears to derive from several sources:

- The fact that much theoretical and empirical research on supervised machine learning assumes that the training set is a random sample of the collection;
- The conviction—possibly owing to experience with similarity search—that the training documents *must* represent all the different kinds of potentially relevant documents in the collection, and the mistaken assumption that random sampling is likely to achieve this;
- The erroneous belief—derived from the confusion between control and training sets—that using a random training set of a particular size will guarantee a particular level of classifier effectiveness; and,
- Distrust of reliance on the teacher's skill, knowledge, and goodwill in choosing appropriate training examples, and fear that any "bias" in selection will be transferred to, if not amplified by, the learner.

The argument for active teaching (whether or not augmented by random training and/or active learning) derives from the fact that the teacher is familiar with the subject matter, and therefore best positioned to find a set of representative documents from which the learner can deduce the best classifier, so that no obvious category of responsive document will be overlooked. Current research suggests, that such subject-matter expertise may not be necessary for certain active learning methods, for which a single relevant document may be sufficient to kick-start the TAR process.[25]

The argument for active learning (whether or not augmented by random training and/or active teaching) derives from the fact that the learning algorithm is best positioned to identify the documents from which it would learn most, akin to a small child repeatedly pointing and asking, "Is that a relevant document?" "What about this one?" "And, that one?"

Simple Versus Continuous Machine-Learning Methods

The second principal factor distinguishing supervised machine-learning protocols is: When should training stop? For simple passive learning and simple active learning, the answer is, "When the classifier is good enough," for some definition of "good enough," which is typically measured in terms of the recall, precision, or F_1 of the classifier, and often referred to as the "stabilization" point. When the stabilization point has been reached, and it is determined that further training will not improve the classifier, the training phase ceases and the review phase begins. The learned classifier does not continue to improve as more documents are reviewed.

For continuous active learning, the answer to the question, "When should training stop?" is, "When enough relevant documents have been found," for some definition of "enough relevant documents," typically measured by a precipitous drop-off in precision, and subsequently, by calculating recall at the end of the review process. For CAL, there is no distinction between the training and review phases. The learned classifier continues to improve as more and more documents are reviewed.

With all TAR protocols, the definition of "enough"—whether measured by a "good-enough" classifier or by identifying "enough" relevant documents—is based on proportionality considerations: How much could the result be improved, in terms of the value of the relevant information in any new

documents found, relative to the additional review effort required to find that information? Over and above imprecision in quantifying the value of the information, and the challenge of measuring it with sufficient accuracy, such a determination requires clairvoyance to determine the future consequences of stopping or continuing training. This is currently one of the biggest challenges for all TAR protocols.

While a standard definition of "enough" has yet to be established, some commentary has argued—and some case law has accepted—a statistical recall estimate of at least 75 percent, with a margin of error of ±5 percent, and 95 percent confidence level, as indicating that "enough relevant documents" have been identified.

Core Supervised Machine-Learning Protocols Used for TAR

The machine-learning approach found to be effective in our JOLT article used a combination of active teaching, active learning, and continuous learning. Initial examples for review and training were the top-ranked hits (based on relevance ranking) of hundreds of simple keyword searches, while subsequent examples were those given the highest score by a logistic regression algorithm, or found through supplemental keyword searches. Review and training continued until an insubstantial fraction of the top-scoring documents that had not yet been reviewed were relevant; in other words, the precision of the top-ranked documents was low, suggesting that there were few relevant documents left to be found.

The protocol just described is an example of continuous active learning. The essential characteristic of CAL is that, after the initial seed set, which may be judgmentally selected by the human teacher, throughout the remainder of the review, it is the learning algorithm that repeatedly suggests additional documents for review. These documents are then coded by the teacher and used for training the learning algorithm. Typically, the selected documents are those that the learning algorithm is most confident are likely to be relevant, hence the name "relevance feedback." Review and training continue in iterative cycles until substantially all relevant documents have been identified. Documents never identified by the process are presumed to be non-relevant, following appropriate sampling and other validation procedures. The use of keyword searches to find documents both at the outset and throughout the TAR process (i.e., active teaching), though beneficial, is not an essential aspect of the particular CAL implementation studied in our JOLT article; nor is the use of logistic regression for the learning algorithm, and certain other design choices.

The results of applying CAL to identify the relevant documents in our fantasy football example are shown in Table 3.5. This CAL implementation[26] uses a single synthetic seed document, consisting only of the request for production for Topic 207, set forth earlier in this chapter; all other training examples were selected by the learner. A comparison of Tables 3.2, 3.3, and 3.4 with Table 3.5 shows that, for any given level of review effort, CAL achieves substantially higher recall and higher precision than the carefully crafted search terms, relevance ranking using these search terms, and similarity search using as a seed set the 31 relevant documents found in the control set. When 18,092 documents are reviewed, CAL achieves 95.5 percent recall and 41.2 percent precision, as compared to 67.0 percent recall and 28.9 percent precision achieved by the best of the other methods previously described in this chapter.

TABLE 3.5 Recall as a Function of Review Effort for the CAL Implementation Distributed to the TREC 2015 Total Recall Track Participants

# Documents Reviewed	Recall	Precision
3,210	40.0%	97.2%
4,079	50.0%	95.6%
5,153	60.0%	90.8%
6,186	70.0%	88.3%
7,580	80.0%	82.3%
10,344	90.0%	67.9%
18,092	95.5%	41.2%

At the time of this writing, CAL has yet to be widely adopted within the e-discovery community, although its use is gradually increasing. Most commercial TAR offerings still employ either simple passive learning or simple active learning.

The defining characteristic of SPL is that the learning algorithm plays no role in selecting the training examples. Examples are selected either at random, through active teaching (i.e., judgmental selection by the teacher), or through a combination of both. Once reviewed and coded, the examples are used to train the learning algorithm. The effectiveness of the training is evaluated with respect to a sample of documents in the collection (typically, the control set), and, if the effectiveness is found to be satisfactory (a point commonly referred to as stabilization), training ceases and the resulting classifier is applied to the entire collection to identify a "review set" of presumed-relevant documents. Documents not in the review set are presumed to be non-relevant after sampling or other validation processes.

Table 3.6 shows the recall and precision achieved for the fantasy football example for SPL using, as a training set, the same 3,000 randomly selected documents that we previously used as a control set, and for identifying seed documents for the similarity search described earlier in the chapter. The SPL review effort (Total, shown in the third column) has two components: first, review of the training set (Training, shown in the first column) and then, the review set (Top Ranked, shown in the second column). For a total review effort of 18,092 documents, SPL achieves 73.2 percent recall and 37.8 percent precision, substantially better than the previously described search methods, but worse than CAL.

TABLE 3.6 Review Effort, Precision, and Recall for SPL Using 3,000 Randomly Selected Training Documents

# Documents Reviewed			Recall	Precision
Training	Top-Ranked	Total	Top-Ranked Only	
3,000	210	3,210	2.6%	100%
3,000	1,079	4,079	13.7%	98.8%
3,000	2,153	5,153	26.3%	95.1%
3,000	3,186	6,186	35.6%	87.1%
3,000	4,580	7,580	44.8%	76.3%
3,000	7,344	10,344	57.4%	69.0%
3,000	15,092	18,092	73.2%	37.8%
3,000	33,444	36,444	84.6%	19.7%
3,000	39,288	42,288	86.3%	17.1%

SPL achieves high recall with less review effort than a carefully crafted keyword search, keyword-based relevance ranking, and similarity search, but requires considerably greater effort than CAL to achieve the same recall. Notably, very few responsive documents are found during the initial effort to review the 3,000 training documents. This initially unproductive training effort may serve as a disincentive to the use of SPL.

SAL falls somewhere between CAL and SPL. Like CAL, SAL employs an active learning strategy in which, after using some examples supplied either by the teacher or through random selection, the learner then selects the remaining training examples. However, the SAL learner typically employs an "uncertainty sampling" approach, selecting documents about which it is least certain, and from which it can therefore learn the most. Like SPL, SAL employs separate training and review phases, where the classifier is fixed at the end of the training phase. The net effect is that SAL shares with SPL a generally unproductive training phase (in the sense that a low proportion of relevant documents are initially identified) and the need to determine when the classifier is "good enough" (i.e., a stabilization point). When this determination is accurately made, SAL can achieve similar recall and precision to CAL.

TABLE 3.7 Comparing CAL, SAL, and SPL Protocols for TAR

CAL Protocol	SAL Protocol	SPL Protocol
STEP 1: Choose a seed set using judgmental sampling (e.g., attorney search, known relevant documents, synthetic documents, etc.). **STEP 2:** Review and code the documents in the seed set. **STEP 3:** Use the machine-learning algorithm to suggest the next most-likely responsive documents for review (i.e., relevance feedback). **STEP 4:** Review and code the newly suggested documents and add them to the training set. **STEP 5:** Repeat steps 3 and 4 above until substantially all relevant documents have been reviewed (i.e., precision drops off precipitously). **STEP 6:** Validate the TAR process.	**STEP 1**: Create a random control set. **STEP 2:** Review and code the documents in the control set. **STEP 3:** Choose a seed set using random sampling, judgmental sampling (e.g., attorney search, known relevant documents, synthetic documents, etc.), or a combination of both. **STEP 4:** Review and code the documents in the seed set. **STEP 5:** Use the machine-learning algorithm to suggest those documents from which the algorithm will learn the most (i.e., uncertainty sampling). **STEP 6:** Review and code the newly suggested documents and add them to the training set. **STEP 7:** Repeat steps 5 and 6 above until training is deemed to be sufficient and "stabilization" occurs. **STEP 8:** Run the machine-learning algorithm for the final time to categorize or rank all documents in the collection. **STEP 9:** Review the documents categorized as "relevant," or ranked above some "cut-off" score. **STEP 10:** Validate the TAR process.	**STEP 1**: Choose a seed set using random sampling, judgmental sampling (e.g., attorney search, known relevant documents, synthetic documents, etc.), or a combination of both. **STEP 2:** Review and code the documents in the seed set. **STEP 3:** Run the machine-learning algorithm and evaluate the effectiveness of training thus far. **STEP 4:** If the result is insufficient, repeat the steps above with an augmented training set. **STEP 5:** When training is deemed to be sufficient, run the machine-learning algorithm for the final time to categorize or rank all documents in the collection. **STEP 6:** Review the documents categorized as "relevant," or ranked above some "cut-off" score. **STEP 7:** Validate the TAR process.

Other Aspects of the Electronic Discovery Process That May Impact TAR

Any implementation of TAR will necessarily occasion certain choices regarding the identification, collection, and culling of the documents subject to search and review; staffing and other workflow decisions related to the specific tool and protocol to be applied; and the selection and implementation of quality-control and validation processes. While these activities are ancillary to TAR per se, they are important aspects of the review process, and can exert a significant impact on the quality and success of a TAR effort. Many of these important issues are addressed in other chapters of this book.

---

1. Maura R. Grossman & Gordon V. Cormack, *Technology-Assisted Review in E-Discovery Can Be More Effective and More Efficient Than Exhaustive Manual Review*, 17 RICH. J.L. & TECH. 11 (2011), http://jolt.richmond.edu/v17i3/article11.pdf.
2. Maura R. Grossman & Gordon V. Cormack, *The Grossman-Cormack Glossary of Technology-Assisted Review*, with Foreword by John M. Facciola, U.S. Magistrate Judge, 7 FED. COURTS. L. REV. 1, 32 (2013), http://www.fclr.org/articles/html/2010/grossman.pdf.
3. Bruce Hedin, Stephen Tomlinson, Jason R. Baron, & Douglas W. Oard, *Overview of the TREC* 2009 *Legal Track*, *in* NIST S PECIAL P UBLICATIONS: SP 500-278, THE EIGHTEENTH TEXT RETRIEVAL CONFERENCE (TREC 2009) PROCEEDINGS, 6 (2009), http://trec.nist.gov/pubs/trec18/papers/LEGAL09.OVERVIEW.pdf; Gordon V. Cormack, Maura R. Grossman, Bruce Hedin, & Douglas W. Oard, *Overview of the TREC* 2010 *Legal Track*, *in* NIST SPECIAL PUBLICATION: SP 500-294, THE NINETEENTH TEXT RETRIEVAL CONFERENCE (TREC 2010) PROCEEDINGS 2 (2010), http://trec.nist.gov/pubs/trec19/papers/LEGAL10.OVERVIEW.pdf; Adam Roegiest, Gordon V. Cormack, Maura R. Grossman, & Charles L. A. Clarke, TREC 2015 Total Recall Track, http://trec-total-recall.org. The Text REtrieval Conference (TREC) is an annual workshop sponsored by the National Institute of Standards and Technology (NIST), the purpose of which is to support research in the information retrieval community by providing the infrastructure necessary for large-scale evaluation of text-retrieval methodologies. The Total Recall Track is part of TREC 2015, *see* http://trec.nist.gov.
4. Roegiest et al., *supra* note 3.
5. Blogger Ralph Losey has compared this approach to the child's game of Go Fish. Ralph C. Losey, *Child's Game of "Go Fish" is a Poor Model for e-Discovery Search*, E-DISCOVERY TEAM BLOG (Oct. 4, 2009), http://e-discoveryteam.com/2009/10/04/childs-game-of-go-fish-is-a-poor-model-for-e-discovery-search/.
6. David C. Blair & M.E. Maron, *An Evaluation of Retrieval Effectiveness for a Full-Text Document-Retrieval System*, 28 COMMC'NS ACM 289 (1985).
7. Grossman & Cormack, *supra* note 2, at 26 ("Prevalence: The fraction of Documents in a Population that are Relevant to an Information Need. Also referred to as Richness or Yield.").
8. The validity of the repeated use of control sets for the purposes of statistical estimation is beyond the scope of this chapter, but is addressed in William Webber, Mossaab Bagdouri, David D. Lewis, & Douglas W. Oard, *Sequential Testing in Classifier Evaluation Yields Biased Estimates of Effectiveness, in* PROCEEDINGS OF THE 36TH INTERNATIONAL ACM SIGIR CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL (SIGIR '13), 933–36 (2013), http://dl.acm.org/citation.cfm?id=2484159.
9. *See* STEFAN BÜTTCHER, CHARLES L.A. CLARKE, & GORDON V. CORMACK, INFORMATION RETRIEVAL: IMPLEMENTING AND EVALUATING SEARCH ENGINES (MIT Press 2010), ch. 10, at 338.
10. We applied the TWFS method described in Section 4.2, at p. 4, of D. Sculley & Gordon V. Cormack, *Going Mini: Extreme Lightweight Spam Filters, in* PROCEEDINGS OF THE SIXTH CONFERENCE ON EMAIL AND ANTI-SPAM (CEAS '09) (2009), http://ceas.cc/2009/papers/ceas2009-paper-47.pdf.
11. *See* Eero Sormunen, *Extensions to the STAIRS Study—Empirical Evidence for the Hypothesized Ineffectiveness of Boolean Queries in Large Full-Text Databases*, 4 INFO. RETRIEVAL J. 257 (2001), http://www.sis.uta.fi/infim/julkaisut/fire/INRT87-JKwithFigs1.pdf.
12. Grossman & Cormack, *supra* note 2, at 28.
13. *See* STEFAN BÜTTCHER ET AL., *supra* note 9, ch. 5, at 138–41 & ch. 8, at 258–81.
14. *See* Sculley & Cormack, *supra* note 10.
15. Losey, *supra* note 5.
16. *See generally* STEFAN BÜTTCHER ET AL., *supra* note 9, ch. 3 & 4, at 84–136.
17. It is important to note that the results presented in Tables 3.2 through Table 3.6 are illustrative examples, only. While the results are typical of those we have observed, they do not show that any class of methods is superior to, or inferior to, any other class of methods in all circumstances. Such questions are best addressed through controlled scientific studies. *See, e. g.*, Cormack & Grossman *infra* note 23.
18. Grossman & Cormack, *supra* note 2, at 11.
19. *See generally* Büttcher et al., *supra* note 9, ch. 10 & 11, at 310–404.
20. *See generally id*.
21. *See generally id*.

22. *See* Grossman & Cormack, *supra* note 2, at 11, 22, 26.

23. Gordon V. Cormack & Maura R. Grossman, *Evaluation of Machine-Learning Protocols for Technology-Assisted Review in Electronic Discovery, in* PROCEEDINGS OF THE 37TH INTERNATIONAL ACM SIGIR CONFERENCE ON RESEARCH AND DEVELOPMENT IN INFORMATION RETRIEVAL (SIGIR '14), 153–62 (2014), http://dx.doi.org/10.1145/2600428.2609601.

24. *See generally id*.

25. Gordon V. Cormack & Maura R. Grossman, *Autonomy and Reliability of Continuous Active Learning for Technology-Assisted Review*, arXiv:1504.06868 (2015), http://arxiv.org/abs/1504.06868.

26. The particular implementation applied was the baseline model implementation (BMI) supplied to TREC 2015 Total Recall participants. *See* Roegiest et al., *supra* note 3.