

AI Search Can Find the Meaning. But It Can Still Miss the Evidence.

The Checkmate Problem and Why the Best Search Uses More Than One Method

By John Tredennick, J.D., and Dr. William Webber



First published by EDRM and JD Supra, September 8, 2026

Editor's Note: This is the second article in a four-part series exploring AI Search and AI-Integrated Discovery. The full series [can be found here](#).

Artificial intelligence has fundamentally changed what lawyers can ask of search. For decades, searching a large document collection meant trying to predict the words that might appear in the evidence. If counsel wanted to find discussions about a delayed project, a defective product, an undisclosed risk, or an effort to conceal bad news, they first had to imagine how the people involved might have described those things. Search strategy therefore began with vocabulary: the right words, synonyms, Boolean combinations, proximity connectors, names, dates and exclusions.

Semantic search changes that equation because it looks beyond literal words to the meaning expressed in a passage. A lawyer can search for documents showing that management knew a project was in trouble before telling a client, that employees were worried about a potential product failure, or that someone was trying to keep information from regulators. The relevant documents do not have to contain the words used in the query. They need only express the same underlying concept. That ability represents one of the most important advances in legal search in decades.

But semantic search has an important limitation. Sometimes the most important evidence in a case is not a concept at all. It may be a project name, an acronym, a code, a distinctive phrase, or some other arbitrary identifier that carries enormous factual significance but very little semantic meaning. That is where semantic search can struggle, and it is what led us to what we call the Checkmate Problem.



The Checkmate Problem

Project Checkmate was not a hypothetical label. It was a 2006 collaboration between IBM and The Scripps Research Institute involving pandemic-virus research. In a supplementary test on the public Jeb Bush email collection, we gave the search system only the query “Project Checkmate.” We supplied no description of the project’s subject matter. That made the test intentionally difficult for semantic search and unusually favorable to literal matching, which was exactly the point: we wanted to see what happens when the words are the signal and their ordinary meaning is not.

For a keyword search, the task was straightforward. Search for Project Checkmate and rank documents containing that rare phrase highly. The search engine did not need to know what the project was or what the words meant in context; it simply needed to reward the literal language.

Semantic search approached the same query differently. The word checkmate has an ordinary meaning associated with chess, winning and defeat. Project is a generic term for an undertaking. Neither tells an embedding model that the phrase refers to scientific collaboration, influenza mutations, vaccine research, government funding or any of the other subjects discussed in the responsive material. The label is arbitrary, and that arbitrariness deprives semantic search of the conceptual signal it normally uses to find related passages.

The result was stark. Keyword search returned 34 responsive chunks among the first 50 results. Semantic search returned none. Keyword search found every one of the 25 chunks containing the exact phrase Project Checkmate within its first 50 results and also surfaced nine additional responsive chunks from context. Semantic search returned 50 results, but none was judged responsive to the project.

FIGURE 1. THE CHECKMATE RESULTS

Same query. Very different results.

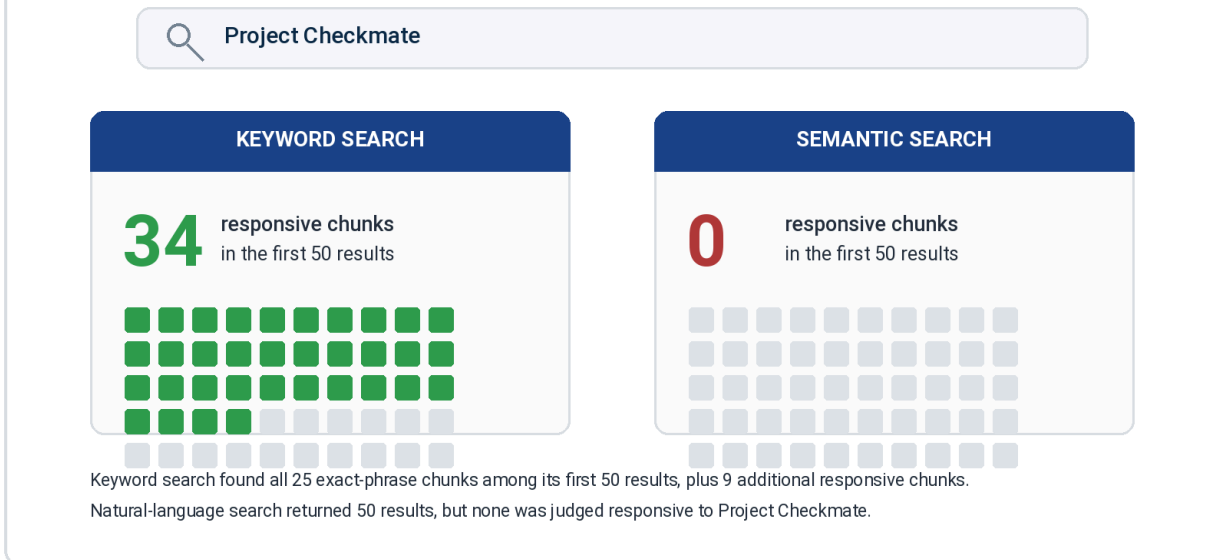


FIGURE 1. THE CHECKMATE RESULTS. Keyword search found 34 responsive chunks in the first 50 results; semantic search found none. The test was deliberately designed around an opaque code name, where exact language carries the signal.

AI Did What We Asked It to Do

At first glance, the Checkmate result seems surprising. Semantic search is powered by modern AI and can recognize relationships that traditional keyword searching cannot. How could it miss something a simple text search finds immediately?

The answer is that semantic search did not fail in the conventional sense. It did what it was designed to do: search for conceptual similarity. The problem was that the thing we were looking for was not really a concept. It was an identifier. In that setting, literal language was the stronger signal.

AI can understand meaning remarkably well. But sometimes the most important evidence has nothing to do with meaning.

When Meaning Is the Wrong Signal

The Checkmate example illustrates a broader point. Semantic search is often described as superior because it can move beyond exact words, but that is only true when meaning is the relevant signal. In many legal investigations, the exact language itself is what matters.



Project names are an obvious example, but the same problem arises with product codes, account numbers, abbreviations, transaction identifiers, email addresses, internal nicknames, unusual names, distinctive contract language and other terms whose importance comes from their use rather than their dictionary meaning. If an employee tells investigators that “Orion” was the internal name for a sensitive initiative, a semantic model may have little basis for associating the word Orion with the substance of that initiative. A literal search for Orion, however, may immediately uncover the relevant communications.

That does not make keyword search more intelligent than semantic search. It simply means the two methods solve different retrieval problems. Keyword search is particularly effective when the language itself carries the signal. Semantic search is particularly effective when the underlying idea matters more than the words used to express it. And that distinction becomes even clearer when we reverse the problem.

Now Search for the Idea

Suppose the lawyers do not know that Project Checkmate exists. Instead, they are trying to answer a broader factual question: Did management know the project was going to miss its deadline before telling the client?

This is where keyword searching becomes much harder. What words should counsel use? Employees might have written that “we are not going to make the date,” that “this is slipping badly,” that “we need another month,” or that “there is no way this gets finished by June.” Someone might have said, “Don’t tell them yet,” while another wrote, “Let’s keep this internal until we have a plan.” Each statement could be relevant to the same underlying issue, yet none necessarily uses the vocabulary in the lawyer’s question.

Experienced search professionals have long developed sophisticated techniques to deal with this problem. They expand terms, construct Boolean expressions, use proximity searches, apply stemming and wildcards, and iteratively test results. Those techniques remain valuable, but they cannot eliminate the basic limitation: keyword searching requires the searcher to anticipate the language that might appear in the documents.

Semantic search changes the problem from one of vocabulary prediction to one of conceptual description. Instead of asking, “What words might the authors have used?”, the lawyer can ask, “What evidence am I trying to find?” The system can then look for passages that express that idea even when the vocabulary is different.

That is a profound change. In a collection containing millions of emails, chats, reports and attachments, the same idea may be expressed in hundreds of ways. People may be direct, cautious, euphemistic, sarcastic or deliberately vague. They may use terminology the legal team has never encountered. Semantic search gives investigators a way to bridge the gap between the language of the question and the language of the evidence.

Different Methods, Different Blind Spots

The central lesson is that keyword search and semantic search should not be viewed simply as an old technology and its AI-powered replacement. They are different retrieval methods built around different signals, and each has failure modes the other can help address.

Keyword search is strongest when the wording itself matters. If counsel knows the project was called Checkmate, the right move is to search for Checkmate. If an investigation turns on a particular contract clause, account number, nickname or code, literal matching may be exactly what is needed. Semantic search is strongest when the meaning matters more than the exact words, such as when counsel is looking for evidence that managers knew a forecast was unrealistic, that employees were worried about safety, or that someone was trying to conceal a problem.

This is why the familiar debate over whether AI search will replace keyword search is too simplistic. The better question is not which method wins. The better question is why lawyers should be required to choose only one.

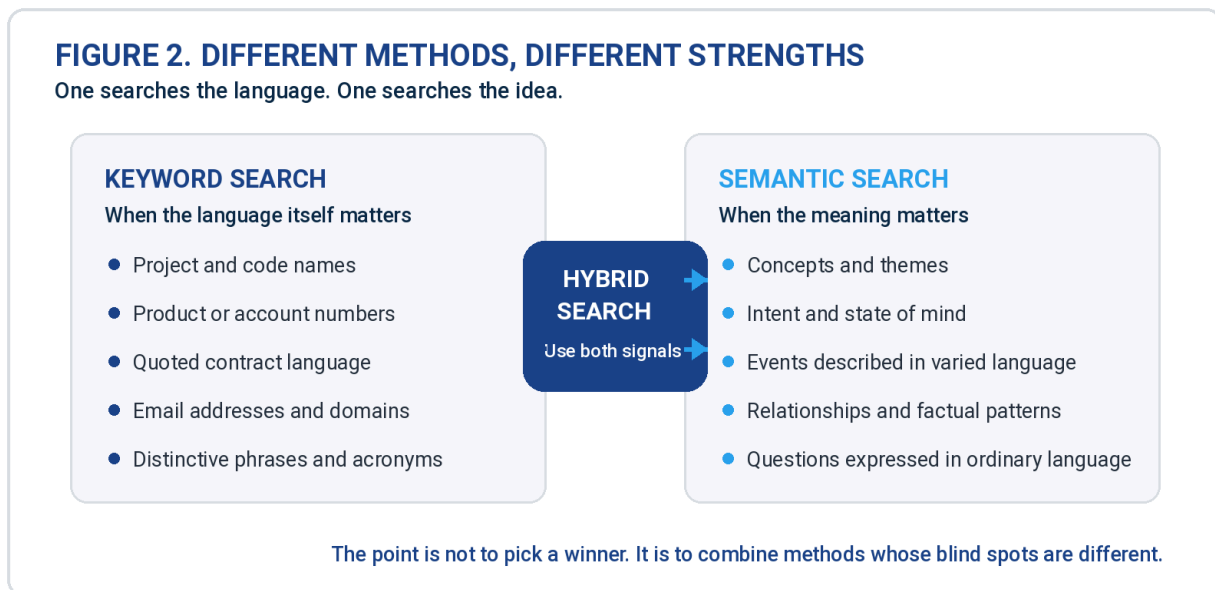


FIGURE 2. DIFFERENT METHODS, DIFFERENT STRENGTHS. Keyword search is strongest when exact language is the signal. Semantic search is strongest when meaning must bridge different vocabulary. Their blind spots are complementary.

***The goal is no longer to determine which retrieval method is best.
The goal is to combine methods that fail differently.***



Why Make the Lawyer Choose in Advance?

Most search systems still treat lexical and semantic searching as separate activities. The user selects a keyword search for one problem, a semantic search for another, and switches between them as the investigation develops. That sounds reasonable until we consider what lawyers actually know at the beginning of a matter.

Often, they know very little. They may not know the important project names, the internal vocabulary, the key players, the language employees used to describe the conduct, or even which factual questions will ultimately prove most important. Those are the things the investigation is supposed to uncover. Requiring the lawyer to choose the correct retrieval method before discovering them puts the burden in the wrong place.

A better architecture can use more than one retrieval method at the same time. That is the core idea behind Hybrid Search. Rather than forcing counsel to decide whether a question should be treated as lexical or semantic, the system can search both ways and combine the results.

Hybrid Search as Retrieval Insurance

Consider a lawyer asking: What did senior management know about the delays before the client was informed? A hybrid system can send that inquiry through multiple retrieval mechanisms. One can look for literal language and terms contained in the documents. Another can search for passages that are semantically related to the underlying concept. The system can then merge and rank the resulting candidates so that the strongest evidence rises toward the top.

We think of this as a form of retrieval insurance. The investigator is no longer forced to bet the search on one theory of how the evidence will present itself. If the critical evidence turns on an exact term, lexical retrieval can capture it. If it turns on an idea expressed in unexpected language, semantic retrieval can find it. If a document scores strongly under both approaches, that convergence can make it even more interesting.

The important shift is conceptual. The objective is no longer to determine which retrieval method is best in the abstract. The objective is to combine methods that fail differently. That approach is more consistent with the realities of legal investigation, where counsel rarely knows in advance whether the decisive clue will be a name, a phrase, a concept or something that only becomes meaningful after other evidence is found.

FIGURE 3. HYBRID SEARCH AS RETRIEVAL INSURANCE

The same question travels through more than one retrieval path.



FIGURE 3. HYBRID SEARCH AS RETRIEVAL INSURANCE. A single investigative question can be searched lexically and semantically at the same time, preserving both exact-language and conceptual signals in one ranked result set.

Let Lawyers Investigate

There is also a practical reason to move toward this model. Lawyers should not have to become information-retrieval engineers in order to take advantage of modern search. They must still exercise judgment, frame legal and factual questions, test hypotheses, evaluate evidence and decide what matters. But they should not have to determine whether a particular inquiry is better suited to lexical matching, BM25, vector similarity or another retrieval algorithm.

The natural role for the lawyer is to ask the investigative question: Who knew? What happened? When did they know it? Who disagreed? What changed after the regulator became involved? Which documents contradict the witness? The natural role for the technology is to use the available retrieval methods to locate the best evidence for answering those questions.

That division of labor becomes increasingly important as AI makes search more conversational. The real promise of AI search is not simply that it produces a better result list. It is that it reduces how much the lawyer must know before beginning the search.

A Third Signal: Learning From the Evidence

There is one more retrieval method that adds another dimension to this story. Continuous active learning, or CAL, works differently from both keyword and semantic search. Rather than beginning with particular words or a conceptual query, CAL learns from examples of documents that lawyers have already identified as relevant.



As reviewers identify useful documents, CAL uses those judgments to rank other documents that are likely to be relevant. In practical terms, that gives us three distinct retrieval signals. Keywords find language. Semantic search finds meaning. CAL learns from examples.

Each sees the collection differently, and each can uncover evidence the others may miss. The third article in this series will explore CAL in greater depth and then bring all three methods together to examine why Hybrid Search can outperform any single approach working on its own.

Search Should Match the Way Investigations Actually Work

Legal investigations do not proceed in a straight line. A lawyer starts with a question, and the first documents reveal new terminology. That terminology identifies new people. Those people lead to new events. The events generate new questions. A project name that meant nothing on day one may become central on day ten, while a phrase nobody thought to search may eventually explain an entire sequence of events.

Search technology should support that iterative process rather than force lawyers into a rigid methodology. The Checkmate Problem matters because it reminds us that relevance can reveal itself through different signals. Sometimes the signal is the exact word. Sometimes it is the underlying meaning. Sometimes it emerges only after investigators have seen enough evidence to recognize what matters.

AI has given lawyers an extraordinary new way to search documents by meaning. That is a major step forward, but it does not require us to discard the methods that already work. Keyword search remains exceptionally good at finding literal language, while semantic search can uncover conceptually related evidence that keyword queries may never reach. They are better understood as complementary views of the same collection.

That is the larger lesson of Project Checkmate. AI search can find the meaning, but it can still miss the evidence. The answer is not to retreat from AI. It is to give the search system more than one way to look.

Next in the Series

In the third and final article, we will examine continuous active learning and the third retrieval signal it adds to the process: learning from the documents lawyers themselves identify as relevant. We will then bring keyword search, semantic search and CAL together to show why Hybrid Search can outperform any single method working alone.

Research Note

The Project Checkmate example was a supplementary code-name test conducted on the public Jeb Bush email collection. Search and assessment were performed at the document-chunk



level. The query was simply “Project Checkmate.” The collection contained 25 chunks with the exact phrase. Keyword search found all 25 in the first 50 results and nine additional responsive chunks; natural-language search returned no responsive chunks in its first 50 results.

The Checkmate test was deliberately selected to examine a case in which literal language should dominate. It was separate from the broader twelve-topic study comparing keyword-oriented and conceptual searches. The study measures ranking quality at specified depths; it is not a claim of universal performance across collections, models or implementations.

About the Authors

John Tredennick

CEO & FOUNDER, MERLIN SEARCH TECHNOLOGIES

John Tredennick is CEO and founder of Merlin Search Technologies, a company pioneering AI-powered document intelligence for legal professionals. He spent the first 20 years of his career as a trial lawyer and senior litigation partner at Holland & Hart LLP, where he became one of the first CTOs of a major law firm. In 2000, he founded Catalyst Repository Systems, an international ediscovery technology company acquired by OpenText in 2019. He has authored or edited eight books and dozens of articles on legal technology and AI, has spoken on five continents, and served as Chair of the ABA's Law Practice Management Section. He has been recognized by The American Lawyer as one of the six top ediscovery pioneers.

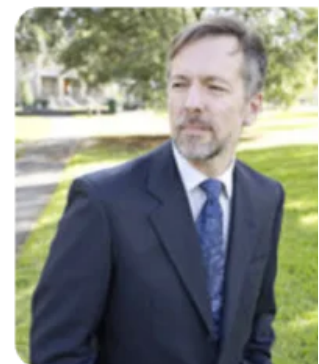
jt@merlin.tech



Dr. William Webber

CHIEF DATA SCIENTIST, MERLIN SEARCH TECHNOLOGIES

Dr. William Webber is Chief Data Scientist of Merlin Search Technologies. He completed his PhD in measurement in information retrieval evaluation at the University of Melbourne under Professors Alistair Moffat and Justin Zobel, and his post-doctoral research at the E-Discovery Lab of the University of Maryland under Professor Doug Oard. With more than 30 peer-reviewed publications in information retrieval, statistical evaluation and machine learning, he is a leading authority on AI





and statistical measurement for information retrieval and ediscovery. At Merlin he leads the data science behind the Alchemy™ platform, including its multi-LLM architecture and the validation methods that keep AI-generated answers sourced and measurable.

wwebber@merlin.tech