Pasteur's Quadrant, Computational Linguistics, LSA, Education

Thomas K Landauer
Knowledge Analysis Technologies. 4940 Pearl East Circle, Boulder, CO 80301
landauer@psych.colorado.edu

Abstract

This paper argues that computational cognitive psychology and computational linguistics have much to offer the science of language by adopting the research strategy that Donald Stokes called Pasteur's quadrant--starting and testing success with important real world problems--and that education offers an ideal venue. Some putative examples from applications of Latent Semantic Analysis (LSA) are presented, as well as some detail on how LSA works, what it is and is not, and what it does and doesn't do. For example, LSA is used successfully in automatic essay grading with content coverage feedback, computing optimal sequences of study materials, and partially automating metadata tagging, but is insufficient for scoring mathematical and short textual answers, for revealing reasons. It is explained that LSA is not construable as measuring co-occurrence, but rather measure the similarity of words in their effect on passage meaning,

1 Credits.

The research reported here has been supported by NSF, the Army Research Institute, the Air Force Office of Scientific Research, the Office of Naval Research, and the Institute of Educational Sciences. Many people contributed to the research including, but by no means limited to Susan Dumais, Peter Foltz, George Furnas, Walter Kintsch, Darrell Laham, Karen Lochbaum, Bob Rehder, and Lynn Streeter,

2 Introduction

In my outsider's opinion—I'm not a linguist and this is my first ACL meeting—this workshop marks an important turn in the study of language. Here is why I think so.

Donald Stokes, in *Pasteur's Quadrant* (1997), argues that the standard view that science progresses from pure to applied research to engineering implementations is often wrong. This doctrine was the brainchild of Vannevar Bush, who was Roosevelt's science advisor during war II. It has, of course, since been enshrined in the DoD's 6.1,2,3 funding structure, and modeled in the national research institutes and large industrial laboratories such as Bell Labs, IBM and Microsoft. Stokes shows that while this trajectory is sometimes followed, often with dramatic success, over the whole course of scientific advance it has been the exception rather than the rule, and for good reasons. Stokes summarized his view of the real relations in a two by two table much like the one in the figure, in which I have made a few minor additions and modifications.

Pure research	Pasteur's quadrant
(random walk research)	Pragmatic engineering

Table 1. Donald Stokes' (1997) illustration of his conception of science, slightly modified.

The upper left quadrant is "pure" research, driven by a desire to understand nature, its problems chosen by what natural phenomena are most pervasive, mysterious or intuitively interesting. Particle physics is its standard bearer. The lower right quadrant is empirical engineering, incremental cut and try, each improvement based on lessons learned from the successes and failures of previous attempts. Internal combustion engines are a type case.

The upper right quadrant, Pasteur's, is research driven by the desire to solve practical problems, for Pasteur preventing the spoilage of vinegar, beer, wine and milk, and conquering diseases in silkworms, sheep, chickens, cattle and humans. Such problems inspire and set concrete goals for research. To solve them it is often necessary to delve into empirical facts and first causes. The quadrant also offers an important way to evaluate scientific success; because failure proves a lack of full understanding.

Stokes doesn't name the lower left quadrant, but it might be dubbed "random walk" science. It resembles theological scholasticism, where the next problem is chosen by flaws in the answer to the last. In my field, cognitive psychology, it is exemplified by 100 years of experiment, thousands of papers, and dozens of quantitative models about how people remember lists of words.

Of course, these activities bleed into one another and sometimes evince the Bush progression. Even list learning has produced basic principles that can be used effectively in education and the treatment of dementia. Nonetheless, the argument is that efforts in Pasteur's quadrant, because they avoid the dangers of excessive-abstraction, simplification and irrelevance, are the most productive, both of scientific advance and of practical value.

I believe that the Pasteur attitude is especially important in psychology, because identifying problems that are critical for understand the human mind is anything but easy. Human minds do many unique and currently unexplainable things. Their first-cause mechanisms are hidden deeply in the intricate connections of billions neurons and billions of experiences. Better keys to the secrets of the mind are needed than hunches of the kind that have motivated list-learning research. To be surer that what we study is actually relevant to the real topic of interest we need to try to solve problems at the level of normal, representative mental functions. Although there are other good candidates, such as automobile driving and economic decision making, education is particularly apt. This is partly because cognitive psychology already knows quite a lot about learning, but more importantly because education is the primary venue in which society intentionally focuses on making a cognitive function happen well, and where success and failure can tell us what we do and do not know, and do so with some guarantee that the knowing is important to understanding the target phenomena.

It seems to me that computational linguistics is in much the same position. Much traditional linguistics has concerned itself with descriptions of abstract properties of language whose actual role in the quotidian human use of language is not often studied, and, therefore, whose promise to explain how language is acquired and works for its users is sometimes hard to evaluate. Computational linguistics itself appears to have been devoted mostly to the upper left and lower right quadrants; on one hand it has spent much of its effort automating or supporting traditional linguistic analyses such as parsing, part-of-speech tagging and semantic role classification. On the other hand, it has developed practical tools, such as dictionaries, ontologies and n-gram language models for doing practical language engineering tasks, such as speech-to-text conversion and machine translation. There has been relatively little effort to use the successes and failures of computer automations to guide, illuminate, or test models of how human language works.

This workshop, represents an important step northeast in Stokes' map. Not only is education accomplished primarily through the use of language, it is also a critical source of advanced abilities to use language-- reading, writing, and thinking--and is the primary medium by which the fruits of education are made useful. Thus trying to improve education is just the kind of thing that the Pasteur approach exploits, compelling reasons to understand, a laboratory for exploration, and strong, broad, relevant tests of success. Putting this argument starkly, it is too easy to treat language as an isolated abstract system and ignore its functional role in human life, and it is too easy to treat education as a humanity, where abstract philosophical arguments, ethical principles or historical precedent guide practice. Attempts to enhance the role of language in education through computation, which makes exquisitely specific what we are doing, should lead to new understanding of the nature of language--and vice versa.

Now for a few words on my own work, and some ways in which it has, at least in part, followed the Pasteur path, plus a few words on how computational linguistics in education might make use of some of its outcomes. This will be take the form of a review of Latent Semantic Analysis (LSA): its origins and history, its computationally simulated mental mechanisms, its applications in education, and some implications it may have for understanding how the mind does language. I'll briefly describe where LSA came from, how it works, what it does and doesn't do, some educational applications in which what it does is useful, some things that limit its usefulness and beg for better basic science, and some nitty-gritty on how and how not to apply it.

3 The History and Nature of LSA

In the early eighties the management of Bell Telephone Laboratories, where I was working, asked me to form a group to find out why secretaries in the legal department were having trouble using UNIX, an obvious godsend, and fix them. This led to trying to find out why customers sometimes couldn't find what they wanted in the Yellow Pages, why service representatives didn't always give correct charges even thought they were plainly stated in well indexed manuals, and why the new online databases for parts and circuits required so much training and yielded only small gains in speed and accuracy, if any.

We undertook a series of lab experiments whose details are skippable. What we discovered was this. In every case the words that people wanted to use, to give orders to computers, or to look things up, rarely matched the words the computer understood or the manuals were indexed by. Roughly, but almost always, the data could be summarized as: ask 100 people by what one word something should be called

and you will get 30 different answers. The distribution is such that it takes five words to cover half the answers. We called this the problem of "verbal disagreement" (Furnas et al., 1987).

Our first solution was brute force; find all the words people would use for what we called an "information object" and index by all of them, which we called "unlimited aliasing" (what do you think the chances are that anyone else would have named them that way?). Later, largely led by George Furnas (1985), we invented some ways to semi-automate that process by what he called "adaptive indexing", having the computer ask people if the words they had used unsuccessfully should be added as pointers to things they eventually found. Of course, we also worried about the problem of ambiguity, now known as "the Google problem", that almost every word has several very different meanings that will lead you astray. At least under some circumstances that was fixable by giving more context in the response, one version of which is Furnas' "fisheye view", to guide navigation. (Adaptive indexing also greatly reduces the ambiguity problem because the pointers are one way--from what people said to the one thing they actually told the systems they meant.)

So what had we done here? We'd used the practical problem to lead to empirically exploration of how people actually used words in daily life (although computers were not as much of daily life then as now, and some of their persisting problems may be due to our failure to get our solutions widely adopted. Here I am, still trying.) The surprising extent and universality of verbal disagreement could be viewed as a baby step in language science, at least as we construed language science.

But just pinning down the nature of the problem in the statistics of actual pragmatic word usage (we called the new field "statistical semantics", which didn't catch on), was only a start. Clearly the problems that computers were having understanding what people meant is special to computers. People understand each other much better. (People also have trouble, although less, with queries of one or two isolated words, but they are very good at using baseline statistics of what people mean by a word (which is, of course, Google's stock in trade, using an indirect version of adaptive indexing), and they appear to use context when available in a much more efficient manner (although this still needs research in the style of statistical semantics.)

What was needed was a way to mimic what people do so well--understand all the meanings of all the words they know, and know just how much and how any word is related to any other. It is perfectly obvious that people learn the meanings of the words in their language, only slightly less so that they must do so primarily from experiencing the words in context and from how they are used in combination to produce emergent meanings. With these facts and clues in mind, the next step was to find computational techniques to do something similar, and see if it improved a computer's understanding. (An apology is in order for idiosyncratic use of the words "meaning", "understanding", and "semantics". They are used here in special senses that differ from myriad usages in linguistics and philosophy, and may offend some readers. Because detailed definitions and circumlocutions would be burdensome and of little value, let us leave it to context.)

The best method we hit upon was what is now called Latent Semantic Analysis, LSA (or, in information retrieval, Latent Semantic Indexing, LSI.) Because there have been some misinterpretations in the literature it may be useful to give a conceptual explanation of how LSA works. It assumes that the meaning of a passage (in practice typically a paragraph) can be approximated by the sum of the meanings of its words. That makes a large print corpus a huge system of simultaneous linear equations. To solve such systems we used the matrix algebraic technique of Singular Value Decomposition (SVD), the general method behind factor analysis and principal components analysis. Applied to a corpus of text, the result is a vector standing for every word in the corpus, with any passage represented by the vector sum of its word vectors. (At first we could only do that with rather small corpora, but with improved algorithms and hardware, size is no longer a barrier.)

The first applications of LSA were to information retrieval, which we conceived of as a problem in the psychology of meaning, how to measure the similarity of meaning to a human of a query and a document given pervasive verbal disagreement. The method was to compute the similarity of corresponding vectors, typically by their cosine (of their angle in a very high dimensional "semantic space".) The result was that, everything else equal (e.g. tokenizing, term-weighting, etc.), LSI gave about 20% better precision-for-recall results, largely because it could rightly judge meaning similarity despite differences in literal word use. It also does any language, and cross language retrieval handily because its numerical vectors don't care whether the "words" are Chinese characters or Arabic script. If the training corpus contains a moderate number of known good translations, and is processed correctly, it does pretty well with no other help.

Along the way we discovered that choosing the right number of dimensions—the number of (independent) elements composing each vector--was critical, three hundred to five hundred being strongly

optimal. One way of describing the value of reducing the number of dimensions well below the number of word types or passages is that it forces the system to induce relations between every word and every other rather than keeping track of the full pattern of empirical occurrences of each, as standard vector retrieval methods do.

Because we like to think we are trying to model human minds as well as solve practical problems, we have also tested LSA on a variety of human tasks. For word meaning an early test was to give it a standardized multiple-choice vocabulary tests (it chooses the word with the most similar meaning by computing which has the highest cosine). Trained on text of similar volume and context to what an American high school senior has read, it does well on the Test of English as a Foreign Language (TOEFL), equaling successful non-native applicants to U.S. Colleges. It also mimics the astounding ten words per day vocabulary growth of middle school children as measured by multiple choice tests. To evaluate its representations of passage meaning, perhaps the most interesting and quantitative tests have been through its use in scoring the conceptual content of expository essays. In actual essay scoring systems we use a suite of analytic tools that includes other things. However, for the present purpose we need to consider how well LSA does when used alone. In doing this, LSA is used to predict the score a human grader would give a new essay on the basis of its similarity to other essays on the same topic that have previously been humanly scored. The LSA-based score predicts very nearly as well as does that of a second independent human reader. Several other evidences of passage-passage success will be described later.

The astute reader will be puzzled by how this could happen, given the very strong simplification of LSA's additivity assumption, by which word order within passages is completely ignored. We will return to this matter, and to more on essay grading later.

Before going on, a few more common misinterpretations of LSA need dealing with. First, LSA is not a measure of co-occurrence, at least as co-occurrence is usually conceived. For LSA a passage meaning is the combination of its word meanings. This does not imply that the words in a passage have the same meaning; indeed that would not be very useful. Empirically, over a typical large corpus, the correlation between the cosine between a random pair of words and the number of passages in which they both occurred is +.35, while the correlation with how often they occur separately, which by the usual interpretation should make them dissimilar, is +.30. By the same token--unlike n-gram language models--LSA estimates the probability that one word will follow another only indirectly and very weakly. (Although, surprisingly, LSA similarities have recently been shown to account for much of what goes on in recalling word lists in order, but not by conditional probability effects (Howard and Kahana, 2001)). More correct interpretations are that LSA reflects the degree to which two words could substitute for one another in similar contexts, that they tend to appear in similar (but not necessarily identical) contexts, and, most precisely, that they have the same effects on passage meanings.

Now what about the fact that LSA ignores word order and thus all syntactically conveyed grammatical effects on sentential meaning? First, it needs emphasis that LSA is very good at measuring the similarity of two words or two passages, sometimes good on sentence to sentence similarity and sometimes not, and least good on word to sentence, or word-to-passage meanings. A good and bad feature of its word-to-word function is that it merges all contextual effects (different senses) of a word into a frequency-weighted average. LSA, as a theory of psychological meaning, proposes that a word is represented as a single central meaning that is modified by context (see Kintsch (2002) for how this could play out in predication and metaphor). The reason it does well on passage-to-passage is that passages are redundant and complex, and that local syntactic effects tend to average out. (This is true for humans too—e.g. they ignore misplaced nots) LSA should be used with all of this in mind.

However, still, you might say, LSA's lack of understanding of prediction, attachment, binding, and constituent structure, thus of representation of logical propositions--all traditional foci of linguistic semantics and computational linguistics-- must surely weaken if not cripple it. Weaken surely, but by how much? Here is one "ballpark" estimate. A typical college educated adult understands around 100,000 word forms, an average sentence contains around 20 tokens. There are thus $100{,}000^{20}$ possible combinations of words in a sentence, therefore a maximum of $\log_2 100{,}000^{20} = 332$ bits of information in word choice alone. There are $20! = 2.4 \times 10^{18}$ possible orders of 20 words for and additional maximum of 61 bits from syntactic effects. Of the possible information in a sentence, then, the part that bag-of-words LSA can use is $332/(61+ 332) = 84\%$.

A substantial amount of human meaning is missing from LSA, but a much larger component is apparently captured. It turns out that, judiciously applied, this component can be quite useful. Moreover, applying it can help pin down the roles of what's missing and not and thus advance our understanding of the nature language as used. Some

successful and less so applications to education are described next, along with some implications, as well as some radical conjectures.

4 Applications of LSA in Education

First, a few more words on the use of LSA in information retrieval (IR) (and relevant to some educational applications described later) and essay scoring. What LSA captures in IR is the degree to which two documents are about the same thing, independent of what equivalent wording may be used. Thus it is useful for finding documents that talk about something, even though it misses details--sometimes important ones--about what was said about the matter. What kind of computation might achieve a representation of the rest?

To achieve a high degree of validity in representing word meaning, LSA uses only information on how words are used, it does not need to assume or identify more primitive semantic features. A possible hint from its success may be that the meaning of groups of words in their order may also rely entirely on how they relate to other groups of words in their orders. (Unpublished work of the psychologist Simon Dennis is pushing in this direction with very interesting results.) Could it be possible that word strings themselves actually are the deepest, most fundamental representation of verbal meaning, not some more abstract underlying primitive entities or structures?

In essay grading, LSA information turns out to be almost, but not quite enough. In practice we add a number of primarily statistical measures, for example n-gram model estimates of how well the words have been ordered relative to standard English statistics. The remarkable thing is that even without any explicit extraction or representation of the logic or propositions in the essays, the methods usually produce slightly more reliable scores than do humans. Is it possible that merely the joint choosing of a set of words and a normative order for arranging them (including nonlinear interactions) suffices to convey all that's needed, without needing any deeper level of representation? Clearly, this is very doubtful, but perhaps worth thinking about?

LSA's text analysis and matching capability, originally devised for IR, has found several fairly direct applications in education. One automatically measures the overlap between the content of courses by the text in their exams--agreeing well with teacher judgments on samples. This is used to help rationalize curricula. Another relates the content of job tasks, training materials, and work histories, all by placing their verbal descriptions in the same semantic space, and uses the results to assign people to jobs and just-in-time compensatory training. A new application automatically matches test items and learning materials to state achievement standards, with high agreement to human experts. Another automatically finds best-sentence summaries and categories as an aid for meta-data tagging of learning objects. A kind of inversion of the LSA representation automatically generates candidate keywords.

The closest relative to essay grading is LSA's role in the *Summary Street* program. In this application students read 4-10 page educational documents, then write 100-200 word summaries. Using LSA, the system tells the student about how well the summary covers each section of the document, how coherent it is--by measuring the similarity of successive sentences--and marks redundant and irrelevant sentences. (Interestingly, experiments have shown that students learn more from text that is coherent, but not excessively so, and LSA can be used to determine the right degree, although no working application has yet been built around the capability.)

Another version of the *Summary Street* and essay analysis technology is a web based tool that scores short essays written to summarize or discuss the content of chapters of college textbooks, providing feedback on what sections to re-read to improve coverage.

A somewhat different manner of extending LSA's text analytic properties lies behind another group of applications. Suppose that a student reads a document about the human heart, then wants to choose another to read that will best advance her knowledge. Experiments have shown that the greatest learning will occur if the next reading introduces neither too little nor too much new knowledge. We call this the Goldilocks principle. By LSA analysis of how all of a set of materials on a topic are related to one another it is possible to accurately place them on a continuum of conceptual sophistication and automatically choose optimum steps. For a large electronic maintenance model currently under development, the technique is being generalized to provide *optimum paths to knowledge*© in which users choose a starting place and a target procedure they want to know, and the system picks a sequence of sections to read that is intended to introduce the needed information in an effective and efficient order for understanding. Combined with fisheye views, adaptive indexing, meaning-based LSA search, embedded LSA-based constructed response assessments, and other guidance features the system is a sort of midway, automatically constructed, intelligent tutor.

Still another application combines aspects of the search and essay evaluation techniques to act as a kind of automated mentor for a collaborative learning environment. Its most interesting capabilities are monitoring and continuously assessing the content of the individual and the total group contributions, connecting individuals with others who are have made comments about similar things, posting alerts when the discussion wanders, both on request and autonomously reaching out to repositories for materials relevant to a discussion, and measuring the formation of consensus. In one small experiment, the system's automatic evaluation of individual content contributions over a semester had a correlation of .9 with independent ratings by participating instructors.

Still more applications are just entering the research stage. One set is stimulated by the widely perceived inadequacy of multiple choice testing; students need to be able to think of answers, not just choose someone else's. The goal is to replace, for example, missing word multiple choice vocabulary tests with ones in which the student supplies the word and the system evaluates how well it fits.

That's enough for successes. What about failures and limitations, what they teach, and where they point research? First, it is true that many laboratory tasks can reveal shortcomings and errors in LSA. Incorrect measures of similarity occur especially for sentences to sentence comparisons in which syntax has strong effects, where broader contextual information or pragmatic intent is involved, and where word meanings have strong relations to perceptual sources to which LSA training has had no access. In some of these cases, it is reasonable to suppose that the basic theoretical foundation is sound but the training data is not sufficient. In other cases it is fairly obvious that more fundamental limitations are at fault, such as the lack of a purely computation process by which to contextually disambiguate the phenomena traditionally described as multiple word senses.

But what about the lessons from trying to solve educational problems promised earlier? There are two glaring examples. One is scoring answers to math problems, or mathematical answers to problems in physics and chemistry (never mind questions requiring drawings or diagrams), something we are frequently asked to do. Syntactic expressions with abstract symbols, where order is critical to logic and bindings are arbitrary, are simply beyond the powers of LSA. How to get them into a fully computational model, one that does not use human help in the form, for example, of manually constructed rules that natural humans could not know, preferably one in which the system learns the capability from the same interaction with the world that humans do, is the challenge to computational cognitive psychology and linguistics that forcefully presents itself, and whose solution could not help but require important new scientific knowledge about language.

A second educational soft spot for LSA is its weakness on sentences. It would almost certainly be better to be able to treat the meaning of an essay as the combination of the meaning of its sentences and the propositional information that order, both within and between sentences, helps to convey. Moreover, simply scoring short answers, another frequent request is problematic. The usual LSA-based methods are not useless, but they fall significantly short of human reliabilities. There seem to be two issues involved. One is again the necessity of accounting for syntax, especially for negation, quantification, and binding. "The price of cloth will go up and the cost of plastics down" is not handled by LSA. The other is that short answer questions often require very specific responses in which some words must be literal entities and others admit of synonyms, circumlocutions and ambiguity. No one has found a way to match humans with without adding what we consider ad hoc methods, rules and triggers devised and coded by people who know the answer. What we want is a fully computational method that might be a possible model of how natural human minds represent knowledge and turn it into an answer of a few words or sentences that can be reliably evaluated by a human who has also learned the needed knowledge in a computationally realistic way. Finding one is another strong challenge whose successful attack would almost have to reveal new scientific truth.

Finally, it is worth noting that LSA has up to very recently relied exclusively on SVD for its central engine. There are certainly other possibilities for doing the same job, and perhaps for doing it better, and for doing more. For example, several new matrix decomposition methods (that's what LSA is) have recently been devised that have interesting new properties, such as more interpretable representations. Other new approaches use entirely different computations, for example the model of Simon Dennis mentioned earlier relies on string-edit theory, computing what operations it takes to change one sentence into another. There is no room, and as yet no results to warrant review of these here, but it is clear that the exploration of innovative computational models of language, ones that, like LSA, are quite different in spirit from linguistic tradition, is being pushed by a desire to solve practical problems, featuring especially ones in education, and that the effort has not nearly reached its limits.

References

Simon Dennis. Unpublished. A memory-based Theory of verbal cognition.

G. W. Furnas. 1985. Experience with an adaptive indexing scheme. In *Proceedings of CHI'85,* ACM, New York: 16-23.

G. W. Furnas, T. K. Landauer, L. M. Gomez, qnd S. T. Dumais. 1987. The vocabulary problem in human system communication. *Communications of the ACM,* 30(11): 964-971.

W. Kintsch. 2001. Predication. *Cognitive Science,* **25**: 173-202

T. K. Landauer. 2002. On the computational basis of learning and cognition: Arguments from LSA. In N. Ross (Ed.), *The Psychology of Learning and Motivation,* **41**: 43-84.

D. E. Stokes. 1997. *Pasteur's Quadrant: Basic Science and Technological Innovation*, Brookings Institution Press, Washington, DC