6

Evaluating Self-Explanations in iSTART: Word Matching, Latent Semantic Analysis, and Topic Models

Chutima Boonthum, Irwin B. Levinstein, and Danielle S. McNamara

6.1 Introduction

iSTART (Interactive Strategy Trainer for Active Reading and Thinking) is a web-based, automated tutor designed to help students become better readers via multimedia technologies. It provides young adolescent to college-aged students with a program of self-explanation and reading strategy training [19] called Self-Explanation Reading Training, or SERT [17, 21, 24, 25]. The reading strategies include (a) comprehension monitoring, being aware of one's understanding of the text; (b) paraphrasing, or restating the text in different words; (c) elaboration, using prior knowledge or experiences to understand the text (i.e., domain-specific knowledge-based inferences) or common sense, using logic to understand the text (i.e., domain-general knowledge based inferences); (d) predictions, predicting what the text will say next; and (e) bridging, understanding the relation between separate sentences of the text. The overall process is called "self-explanation" because the reader is encouraged to explain difficult text to him- or herself. iSTART consists of three modules: Introduction, Demonstration, and Practice. In the last module, students practice using reading strategies by typing self-explanations of sentences. The system evaluates each self-explanation and then provides appropriate feedback to the student. If the explanation is irrelevant or too short, the student is required to add more information. Otherwise, the feedback is based on the level of overall quality.

The computational challenge here is to provide appropriate feedback to the students concerning their self-explanations. To do so requires capturing some sense of both the meaning and quality of the self-explanation. Interpreting text is critical for intelligent tutoring systems, such as iSTART, that are designed to interact meaningfully with, and adapt to, the users' input. iSTART was initially proposed as using Latent Semantic Analysis (LSA; [13]) to capture the meanings of texts and to assess the students' self-explanation; however, while the LSA algorithms were being built, iSTART used simple word matching algorithms. In the course of integrating the LSA algorithms, we found that a combination of word-matching and LSA provided better results than either separately [18].

Our goal in evaluating the adequacy of the algorithms has been to imitate experts' judgments of the quality of the self-explanations. The current evaluation system predicts the score that a human gives on a 4-point scale, where 0 represents an

evaluation of the explanation as irrelevant or too short; 1, minimally acceptable; 2, better but including primarily the local textual context; and 3, oriented to a more global comprehension. Depending on the text, population, and LSA space used, our results have ranged from 55 to 70 percent agreement with expert evaluations using that scale. We are currently attempting to improve the effectiveness of our algorithms by incorporating Topic Models (TM) either in place of or in conjunction with LSA and by using more than one LSA space from different genres (science, narrative, and general TASA corpus). We present some of the results of these efforts in this chapter.

Our algorithms are constrained by two major requirements, speedy response times and speedy introduction of new texts. Since the trainer operates in real time, the server that calculates the evaluation must respond in 4 to 5 seconds. Furthermore the algorithms must not require any significant preparation of new texts, a requirement precisely contrary to our plans when the project began. In order to accommodate the needs of the teachers whose classes use iSTART, the trainer must be able to use texts that the teachers wish their students to use for practice within a day or two. This time limit precludes us from significantly marking up the text or gathering related texts to incorporate into an LSA corpus.

In addition to the overall 4-point quality score, we are attempting to expand our evaluation to include an assessment of the presence of various reading strategies in the student's explanation so that we can generate more specific feedback. If the system were able to detect whether the explanation uses paraphrasing, bridging, or elaboration we could provide more detailed feedback to the students, as well as an individualized curriculum based on a more complete model of the student. For example, if the system were able to assess that the student only paraphrased sentences while self-explaining, and never used strategies such as making bridging inferences or knowledge-based elaborations, then the student could be provided additional training to generate more inference-based explanations.

This chapter describes how we employ word matching, LSA, and TM in the iSTART feedback systems and the performance of these techniques in producing both overall quality and reading strategy scores.

6.2 iSTART: Feedback Systems

iSTART was intended from the outset to employ LSA to determine appropriate feedback. The initial goal was to develop one or more benchmarks for each of the SERT strategies relative to each of the sentences in the practice texts and to use LSA to measure the similarity of a trainee's explanation to each of the benchmarks. A benchmark is simply a collection of words, in this case, words chosen to represent each of the strategies (e.g., words that represent the current sentence, words that represent a bridge to a prior sentence). However, while work toward this goal was progressing, we also developed a preliminary "word-based" (WB) system to provide feedback in our first version of iSTART [19] so that we could provide a complete curriculum for use in experimental situations. The second version of iSTART has integrated both LSA and WB in the evaluation process; however, the system still provides only overall quality feedback. Our current investigations aim to provide feedback based on identifying specific reading strategies.

6.2.1 Word Matching Feedback Systems

Word matching is a very simple and intuitive way to estimate the nature of a self-explanation. In the first version of iSTART, several hand-coded components were built for each practice text. For example, for each sentence in the text, the "important words" were identified by a human expert and a length criterion for the explanation was manually estimated. Important words were generally content words that were deemed important to the meaning of the sentence and could include words not found in the sentence. For each important word, an association list of synonyms and related terms was created by examining dictionaries and existing protocols as well as by human judgments of what words were likely to occur in a self-explanation of the sentence. In the sentence "All thunderstorms have a similar life history," for example, important words are *thunderstorm, similar, life*, and *history*. An association list for *thunderstorm* would include *storms, moisture, lightning, thunder, cold, tstorm, t-storm, rain, temperature, rainstorms*, and *electric-storm*. In essence, the attempt was made to imitate LSA.

A trainee's explanation was analyzed by matching the words in the explanation against the words in the target sentence and words in the corresponding association lists. This was accomplished in two ways: (1) Literal word matching and (2) Soundex matching.

Literal word matching - Words are compared character by character and if there is a match of the first 75% of the characters in a word in the target sentence (or its association list) then we call this a literal match. This also includes removing suffix -s, -d, -ed, -ing, and -ion at the end of each words. For example, if the trainee's self-explanation contains 'thunderstom' (even with the misspelling), it still counts as a literal match with words in the target sentence since the first nine characters are exactly the same. On the other hand, if it contains 'thunder,' it will not get a match with the target sentence, but rather with a word on the association list.

Soundex matching - This algorithm compensates for misspellings by mapping similar characters to the same soundex symbol [1, 5]. Words are transformed to their soundex code by retaining the first character, dropping the vowels, and then converting other characters into soundex symbols. If the same symbol occurs more than once consecutively, only one occurrence is retained. For example, 'thunderstorm' will be transformed to 't8693698'; 'communication' to 'c8368.' Note that the later example was originally transformed to 'c888368' and two 8s were dropped ('m' and 'n' are both mapped to '8'). If the trainee's self-explanation contains 'thonderstorm' or 'tonderstorm,' both will be matched with 'thunderstorm' and this is called a soundex match. An exact soundex match is required for short words (i.e., those with fewer than six alpha-characters) due to the high number of false alarms when soundex is used. For longer words, a match on the first four soundex symbols suffices. We are considering replacing this rough and ready approach with a spell-checker.

A formula based on the length of the sentence, the length of the explanation, the length criterion mentioned below, the number of matches to the important words, and the number of matches to the association lists produces a rating of 0 (inadequate), 1 (barely adequate), 2 (good), or 3 (very good) for the explanation. The rating of 0 or inadequate is based on a series of filtering criteria that assesses whether the explanation is too short, too similar to the original sentence, or irrelevant. *Length*

is assessed by a ratio of the number of words in the explanation to the number in the target sentence, taking into consideration the length criterion. For example, if the length of the sentence is 10 words and the length priority is 1, then the required length of the self-explanation would be 10 words. If the length of the sentence is 30 words and the length priority is 0.5, then the self-explanation would require a minimum of 15 words. *Relevance* is assessed from the number of matches to important words in the sentence and words in the association lists. *Similarity* is assessed in terms of a ratio of the sentence and explanation lengths and the number of matching important words. If the explanation is close in length to the sentence, with a high percentage of word overlap, the explanation would be deemed too similar to the target sentence. If the explanation failed any of these three criteria (Length, Relevance, and Similarity), the trainee would be given feedback corresponding to the problem and encouraged to revise the self-explanation.

Once the explanation passes the above criteria, then it is evaluated in terms of its overall quality. The three levels of quality that guide feedback to the trainee are based on two factors: 1) the number of words in the explanation that match either the important words or association-list words of the target sentence compared to the number of important words in the sentence and 2) the length of the explanation in comparison with the length of the target sentence. This algorithm will be referred as *WB-ASSO*, which stands for *word-based with association list*.

This first version of iSTART (word-based system) required a great deal of human effort per text, because of the need to identify important words and, especially, to create an association list for each important word. However, because we envisioned a scaled-up system rapidly adaptable to many texts, we needed a system that required relatively little manual effort per text. Therefore, WB-ASSO was replaced. Instead of lists of important and associated words we simply used content words (nouns, verbs, adjectives, adverbs) taken literally from the sentence and the entire text. This algorithm is referred to as *WB-TT*, which stands for *word-based with total text*. The content words were identified using algorithms from Coh-Metrix, an automated tool that yields various measures of cohesion, readability, other characteristics of language [9, 20]. The iSTART system then compares the words in the self-explanation to the content words from the current sentence, prior sentences, and subsequent sentences in the target text, and does a word-based match (both literal and soundex) to determine the number of content words in the self-explanation from each source in the text. While WB-ASSO is based on a richer corpus of words than WB-TT, the replacement was successful because the latter was intended for use together with LSA which incorporates the richness of a corpus of hundreds of documents. In contrast, WB-ASSO was used on its own.

Some hand-coding remained in WB-TT because the length criterion for an explanation was calculated based on the average length of explanations of that sentence collected from a separate pool of participants and on the importance of the sentence according to a manual analysis of the text. Besides being relatively subjective, this process was time consuming because it required an expert in discourse analysis as well as the collection of self-explanation protocols. Consequently, the hand-coded length criterion was replaced with one that could be determined automatically from the number of words and content words in the target sentence (we called this *word-based with total text and automated criteria*, or *WB2-TT*). The change from WB-TT to WB2-TT affected only the screening process of the length and similarity criteria. Its lower-bound and upper-bound lengths are entirely based on the target sentence's

length. The overall quality of each self-explanation (1, 2, or 3) is still computed with the same formula used in WB-TT.

6.2.2 Latent Semantic Analysis (LSA) Feedback Systems

Latent Semantic Analysis (LSA; [13, 14]) uses statistical computations to extract and represent the meaning of words. Meanings are represented in terms of their similarity to other words in a large corpus of documents. LSA begins by finding the frequency of terms used and the number of co-occurrences in each document throughout the corpus and then uses a powerful mathematical transformation to find deeper meanings and relations among words. When measuring the similarity between text-objects, LSA's accuracy improves with the size of the objects. Hence, LSA provides the most benefit in finding similarity between two documents. The method, unfortunately, does not take into account word order; hence, very short documents may not be able to receive the full benefit of LSA.

To construct an LSA corpus matrix, a collection of documents are selected. A document may be a sentence, a paragraph, or larger unit of text. A term-document-frequency (TDF) matrix X is created for those terms that appear in two or more documents. The row entities correspond to the words or terms (hence the W) and the column entities correspond to the documents (hence the D). The matrix is then analyzed using Singular Value Decomposition (SVD; [26]), that is the TDF matrix X is decomposed into the product of three other matrices: (1) vectors of derived orthogonal factor values of the original row entities W, (2) vectors of derived orthogonal factor values of the original column entities D, and (3) scaling values (which is a diagonal matrix) S. The product of these three matrices is the original TDF matrix.

$$\{X\} = \{W\}\{S\}\{D\} \tag{6.1}$$

The dimension (d) of $\{S\}$ significantly affects the effectiveness of the LSA space for any particular application. There is no definite formula for finding an optimal number of dimensions; the dimensionality can be determined by sampling the results of using the matrix $\{W\}\{S\}$ to determine the similarity of previously-evaluated document pairs for different dimensionalities of $\{S\}$. The optimal size is usually in the range of 300-400 dimensions.

The similarity of terms is computed by taking the cosine of the corresponding term vectors. A term vector is the row entity of that term in the matrix W. In iSTART, the documents are sentences from texts and trainees' explanations of those sentences. These documents consist of terms, which are represented by term vectors; hence, the document can be represented as a document vector which is computed as the sum of the term vectors of its terms:

$$D_i = \sum_{t=1}^{n} T_{ti} \tag{6.2}$$

where D_i is the vector for the ith document D, T_{ti} is the term vector for the term t in D_i, and n is number of terms in D. The similarity between two documents (i.e., the cosine between the two document vectors) is computed as

$$Sim(D1, D2) = \frac{\sum_{i=1}^{d}(D1_i \times D2_i)}{\sum_{i=1}^{d}(D1_i)^2 \times \sum_{i=1}^{d}(D2_i)^2} \qquad (6.3)$$

Since the first versions of iSTART were intended to improve students' comprehension of science texts, the LSA space was derived from a collection of science texts [11]. This corpus consists of 7,765 documents containing 13,502 terms that were used in two or more documents. By the time the first version of the LSA-based system was created (referred to as *LSA1*), the original goal of identifying particular strategies in an explanation had been replaced with the less ambitious one of rating the explanation as belonging one of three levels [22]. The highest level of explanation, called "*global-focused*," integrates the sentence material in a deep understanding of the text. A "*local-focused*" explanation explores the sentence in the context of its immediate predecessors. Finally, a "*sentence-focused*" explanation goes little beyond paraphrasing. To assess the level of an explanation, it is compared to four benchmarks or bags of words. The rating is based on formulae that use weighted sums of the four LSA cosines between the explanation and each of the four benchmarks.

The four benchmarks include: 1) the words in the title of the passage ("title"), 2) the words in the sentence ("current sentence"), 3) words that appear in prior sentences in the text that are causally related to the sentence ("prior text"), and 4) words that did not appear in the text but were used by two or more subjects who explained the sentence during experiments ("world knowledge"). While the title and current sentence benchmarks are created automatically, the prior-text benchmark depends on a causal analysis of the conceptual structure of the text, relating each sentence to previous sentences. This analysis requires both time and expertise. Furthermore, the world-knowledge benchmark requires the collection of numerous explanations of each text to be used. To evaluate the explanation of a sentence, the explanation is compared to each benchmark, using the similarity function mentioned above. The result is called a cosine value between the self-explanation (SE) and the benchmark. For example, *Sim(SE, Title)* is called the *title LSA cosine*. Discriminant Analysis was used to construct the formulae that categorized the overall quality as being a level 1, 2, or 3 [23]. A score is calculated for each of the levels using these formulae. The highest of the three scores determines the predicted level of the explanation. For example, the overall quality score of the explanation is a 1 if the level-1 score is higher than both the level-2 and level-3 scores.

Further investigation showed that the LSA1 cosines and the factors used in the WB-ASSO approach could be combined in a discriminant analysis that resulted in better predictions of the values assigned to explanations by human experts. However, the combined approach was less than satisfactory. Like WB-ASSO, LSA1 was not suitable for an iSTART program that would be readily adaptable to new practice texts. Therefore, we experimented with formulae that would simplify the data gathering requirements to develop LSA2. Instead of the four benchmarks mentioned above, we discarded the world knowledge benchmark entirely and replaced the benchmark based on causal analysis of prior-text with one that simply consisted of the words in the previous two sentences. We could do this because the texts were taken from science textbooks whose argumentation tends to be highly linear argumentation in science texts; consequently the two immediately prior sentences

worked well as stand-ins for the set of causally related sentences. It should be noted that this approach may not succeed so well with other genres, such as narrative or history texts.

We tested several systems that combined the use of word-matching and LSA2 and the best one is LSA2/WB2-TT. In these combinatory systems, we combine a weighted sum of the factors used in the fully automated word-based systems and LSA2. These combinations allowed us to examine the benefits of using the world knowledge benchmark (in LSA1) when LSA was combined with a fully automated word-based system and we found that world knowledge benchmark could be dropped. Hence, only three benchmarks are used for LSA-based factors: 1) the words in the title of the passage, 2) the words in the sentence, and 3) the words in the two immediately prior sentences. From the word-based values we include 4) the number of content words matched in the target sentence, 5) the number of content words matched in the prior sentences, 6) the number of content words matched in the subsequent sentences, and 7) the number of content words that were not matched in 4, 5, or 6. One further adjustment was made because we noticed that the LSA approach alone was better at predicting higher values correctly, while the word-based approach was better at predicting lower values. Consequently, if the formulae of the combined system predicted a score of 2 or 3, that value is used. However, if the system predicted a 1, a formula from the word-based system is applied. Finally, level 0 was assigned to explanations that had negligible cosine matches with all three LSA benchmarks.

6.2.3 Topic Models (TM) Feedback System

The Topic Models approach (TM; [10, 27]) applies a probabilistic model in finding a relationship between terms and documents in terms of topics. A document is conceived of as having been generated probabilistically from a number of topics and each topic consists of number of terms, each given a probability of selection if that topic is used. By using a TM matrix, we can estimate the probability that a certain topic was used in the creation of a given document. If two documents are similar, the estimates of the topics they probably contain should be similar. TM is very similar to LSA, except that a term-document frequency matrix is factored into two matrices instead of three.

$$\{X_{normalized}\} = \{W\}\{D\} \tag{6.4}$$

The dimension of matrix $\{W\}$ is W x T, where W is the number of words in the corpus and T is number of topics. The number of topics varies, more or less, with the size of corpus; for example, a corpus of 8,000 documents may require only 50 topics while a corpus of 40,000 documents could require about 300 topics. We use the TM Toolbox [28] to generate the $\{W\}$ or TM matrix, using the same science corpus as we used for the LSA matrix. In this construction, the matrix $\{X\}$ is for all terms in the corpus, not just those appearing in two different documents. Although matrix $\{X\}$ is supposed to be normalized, the TM toolbox takes care of this normalization and outputs for each topic, the topic probability, and a list of terms in this topic along with their probabilities in descending order (shown in Table 6.1). This output is easily transformed into the term-topic-probability matrix.

Table 6.1. Results from Topic Models Toolbox: science corpus, 50 topics, seed 1, 500 iteration, default alpha and beta.

TOPIC 2 0.0201963151	TOPIC 38 0.0214418635
earth 0.1373291184	light 0.1238061875
sun 0.0883152826	red 0.0339683946
solar 0.0454833721	color 0.0307797075
atmosphere 0.0418036547	white 0.0262046347
moon 0.0362104843	green 0.0230159476
surface 0.0181062747	radiation 0.0230159476
planet 0.0166343877	wavelengths 0.0230159476
center 0.0148681234	blue 0.0184408748
bodies 0.0147209347	dark 0.0178863206
tides 0.0139849912	visible 0.0170544891
planets 0.0133962364	spectrum 0.0151135492
gravitational 0.0125131042	absorbed 0.0149749106
system 0.0111884060	colors 0.0148362720
appear 0.0110412173	rays 0.0116475849
mass 0.0100108964	eyes 0.0108157535
core 0.0083918207	yellow 0.0105384764
space 0.0083918207	absorption 0.0102611992
times 0.0079502547	eye 0.0095680064
orbit 0.0073614999	pigment 0.0092907293
...	...

To measure the similarity between documents based on TM, the Kullback Liebler distance (KL-distance: [27]) between two documents is recommended, rather than the cosine (which, nevertheless, can be used). A document can be represented by a set of probabilities that this document could contain topic i using the following

$$D_t = \sum_{i=1}^{n} T_{it} \qquad (6.5)$$

where D_t is the probability of topic t in the document D, T_{it} is the probability of topic t of the term i in the document D, and n is number of terms appearing in the document D. The KL-distance between two documents (the similarity) is computed as follows:

$$KL(D1, D2) = \frac{1}{2} \sum_{t=1}^{T} D1_t log_2(D1_t/D2_t) + \frac{1}{2} \sum_{t=1}^{T} D2_t log_2(D2_t/D1_t) \qquad (6.6)$$

Constructing a TM matrix involves making choices regarding a number of factors, such as the number of topics, the seed for random number generation, alpha, beta, and the number of iterations. We have explored these factors and constructed a number of TM matrices in an effort to optimize the resulting matrix; however, for this preliminary evaluation, we use a TM matrix of 50 topics and a seed of 1.

The first TM-based system we tried was simply used in place of the LSA-based factors in the combined-system. The three benchmarks are still the same but sim-

ilarity is computed in two ways: (1) using cosines — comparing the explanation and the benchmark using the cosine formula (Referred as TM1) and (2) using KL distances — comparing the explanation and the benchmark using the KL distance (Referred as TM2). As before, formulae are constructed using Discriminant Analysis in order to categorize the quality of explanation as Levels 1, 2, or 3.

6.2.4 Metacognitive Statements

The feedback systems include a metacognitive filter that searches the trainees' self-explanations for patterns indicating a description of the trainee's mental state such as "now I see ..." or "I don't understand this at all." While the main purpose of the filter is to enable the system to respond to such non-explanatory content more appropriately, we also used the same filter to remove "noise" such as "What this sentence is saying is ..." from the explanation before further processing. We have examined the effectiveness of the systems with and without the filter and found that they all perform slightly better with than without it. Thus, the systems in this chapter all include the metacognitive filter.

The metacognitive filter also benefits the feedback system. When a metacognitive pattern is recognized, its category is noted. If the self-explanation contains only a metacognitive statement, the system will respond to a metacognitive category such as *understanding, not-understanding, confirmation, prediction,* or *boredom* instead of responding irrelevantly. Regular expressions are used to define multiple patterns for each metacognitive category. If any pattern is matched in the self-explanation, words matching the pattern are removed before evaluation. Examples of regular expression are shown below:

```
NOTUNDERSTAND :i(?:.?m|\W+am)(?:\W+\w+)?\W+\W+(?:(?:not
                (?:\W+\w+)?\W+(?:sure|certain|clear))|
                un(?:sure|certain|clear))
UNDERSTAND :now\W+i\W+(?:know|knew|underst(?:an|oo)d|
                remember(?:ed)?|recall(?:ed)?|recogniz(?:ed)?|get|
                got|see)
CONF :(?:so\W+)?i\W+(?:was|got\W+it)\W+(?:right|correct)
```

The first pattern will include "I'm not sure," "I am uncertain"; second pattern includes "Now I understand," "Now I remembered"; and the last pattern includes "So, I was right." We originally constructed over 60 patterns. These were reduced to 45 by running them on a large corpus of explanations and eliminating those that failed to match and adding those that were missed.

6.3 iSTART: Evaluation of Feedback Systems

Two experiments were used to evaluate the performance of various systems of algorithms that vary as a function of approach (word-based, LSA, combination of word-based and LSA, and combination of word-based TM). In Experiment 1, we

compare all eight systems in terms of the overall quality score by applying each system to a database of self-explanation protocols produced by college students. The protocols had been evaluated by a human expert on overall quality. In Experiment 2, we investigated two systems using a database of explanations produced by middle-school students. These protocols were scored to identify particular reading strategies.

6.3.1 Experiment 1

Self-Explanations. The self-explanations were collected from college students who were provided with SERT training and then tested with two texts, Thunderstorm and Coal. Both texts consisted of 20 sentences. The Thunderstorm text was self-explained by 36 students and the Coal text was self-explained by 38 students. The self-explanations were coded by an expert according to the following 4-point scale: 0 = vague or irrelevant; 1 = sentence-focused (restatement or paraphrase of the sentence); 2 = local-focused (includes concepts from immediately previous sentences); 3 = global-focused (using prior knowledge).

The coding system was intended to reveal the extent to which the participant elaborated the current sentence. Sentence-focused explanations do not provide any new information beyond the current sentence. Local-focused explanations might include an elaboration of a concept mentioned in the current or immediately prior sentence, but there is no attempt to link the current sentence to the theme of the text. Self-explanations that linked the sentence to the theme of the text with world knowledge were coded as "global-focused." Global-focused explanations tend to use multiple reading strategies, and indicate the most active level of processing.

Results. Each of the eight systems produces an evaluation comparable to the human ratings on a 4-point scale. Hence, we calculated the correlations and percent agreement between the human and system evaluations (see Table 6.2). Additionally, d primes (d′s) were computed for each strategy level as a measure of how well the system could discriminate among the different levels of strategy use. The d′s were computed from hit and false-alarm rates. A hit would occur if the system assigned the same self-explanation to a category (e.g., global-focused) as the human judges. A false-alarm would occur if the system assigned the self-explanation to a category (e.g., global-focused) that was different from the human judges (i.e., it was not a global-focused strategy). d′s are highest when hits are high and false-alarms are low. In this context, d′s refer to the correspondence between the human and system in standard deviation units. A d′ of 0 indicates chance performance, whereas greater d′s indicate greater correspondence.

One thing to note in Table 6.3 is that there is general improvement according to all of the measures going from left to right. As might be expected, the systems with LSA fared far better than those without LSA, and the combined systems were the most successful. The word-based systems tended to perform worse as the evaluation level increased (from 0 to 3), but performed relatively well at identifying poor self-explanations and paraphrases. All of the systems, however, identified the sentence-focused (i.e., 2's) explanations less successfully. However, the d′s for the sentence focused explanations approach 1.0 when LSA is incorporated, particularly when LSA is combined with the word-based algorithms.

Apart from better performance with LSA than without, the performance is also more stable with LSA. Whereas the word-based systems did not perform equally

Table 6.2. Measures of agreement for the Thunderstorm and Coal texts between the eight system evaluations and the human ratings of the self-explanations in Experiment 1.

Thunderstorm Text	WB-ASSO	WB-TT	WB2-TT	LSA1	LSA2	LSA2/WB2-TT	TM1	TM2
Correlation	0.47	0.52	0.43	0.60	0.61	0.64	0.56	0.58
% Agreement	48%	50%	27%	55%	57%	62%	59%	60%
d' of 0's	2.21	2.26	0.97	2.13	2.19	2.21	1.49	2.37
d' of 1's	0.84	0.79	0.66	1.32	1.44	1.45	1.27	1.39
d' of 2's	0.23	0.36	-0.43	0.47	0.59	0.85	0.74	0.70
d' of 3's	1.38	1.52	1.41	1.46	1.48	1.65	1.51	1.41
Avg d'	1.17	1.23	0.65	1.34	1.43	1.54	1.25	1.23

Coal Text	WB-ASSO	WB-TT	WB2-TT	LSA1	LSA2	LSA2/WB2-TT	TM1	TM2
Correlation	0.51	0.47	0.41	0.66	0.67	0.71	0.63	0.61
% Agreement	41%	41%	29%	56%	57%	64%	61%	61%
d' of 0's	4.67	4.73	1.65	2.52	2.99	2.93	2.46	2.05
d' of 1's	1.06	0.89	0.96	1.21	1.29	1.50	1.38	1.52
d' of 2's	0.09	0.13	-0.37	0.45	0.49	0.94	0.74	0.61
d' of 3's	-0.16	1.15	1.28	1.59	1.59	1.79	1.60	1.50
Avg d'	1.42	1.73	0.88	1.44	1.59	1.79	1.54	1.42

well on the Thunderstorm and Coal texts, there is a high-level of agreement for the LSA-based formulas (i.e., the results are virtually identical in the two tables). This indicates that if we were to apply the word-based formulas to yet another text, we have less assurance of finding the same performance, whereas the LSA-based formulas are more likely to replicate across texts.

Figure 6.1.a provides a closer look at the data for the combined, automated system, LSA2/WB2-TT and Figure 6.1.b for the TM2 system. As the d's indicated, both systems' performance is quite good for explanations that were given human ratings of 0, 1, or 3. Thus, the system successfully identifies poor explanations, paraphrases, and very good explanations. It is less successful for identifying explanations that consist of paraphrases in addition to some information from the previous sentence or from world knowledge. As one might expect, some are classified as paraphrases and some as global by the system. Although not perfect, we consider this result a success because so few were misclassified as poor explanations.

6.3.2 Experiment 2

Self-Explanations. The self-explanations were collected from 45 middle-school students (entering 8th and 9th grades) who were provided with iSTART training and then tested with two texts, Thunderstorm and Coal. The texts were shortened versions of the texts used in Experiment 1, consisting of 13 and 12 sentences, respectively. This chapter presents only the data from the Coal text.
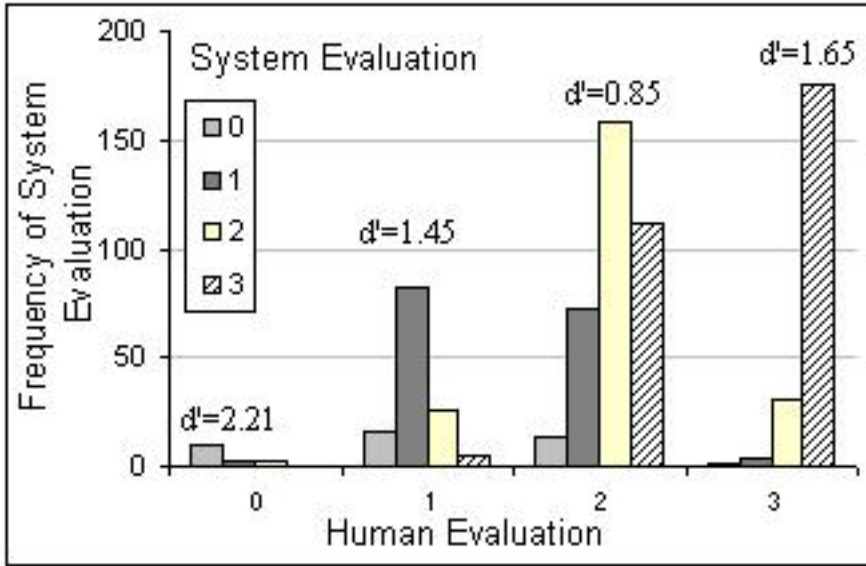
a) LSA2/WB2-TT — LSA with Word-based
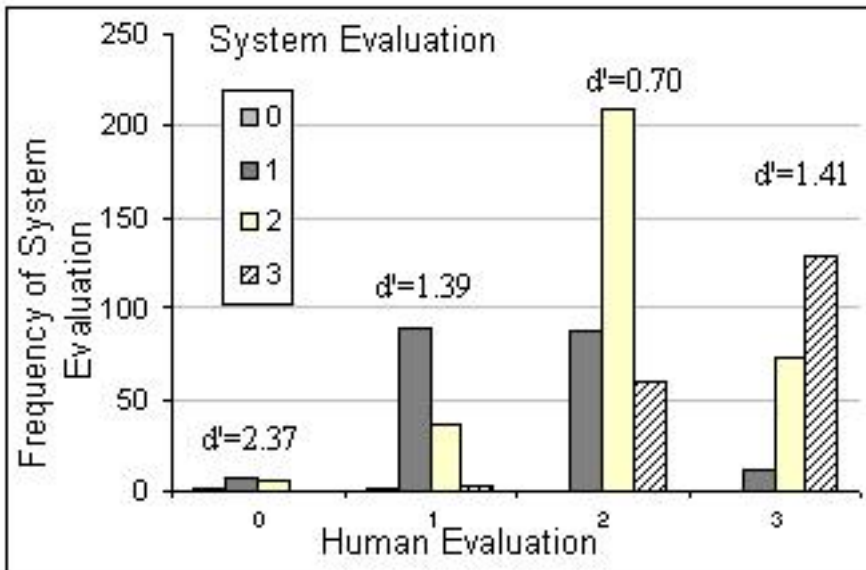
b) TM2 — Topic Models with KL distance

Fig. 6.1. Correspondence between human evaluations of the self-explanations and the combined system (LSA2/WB2-TT and TM2) for Thunderstorm text. Explanations were evaluated by humans as vague or irrelevant (0), sentence-focused (1), local-focused (2), or global (3).

The self-explanations from this text were categorized as paraphrases, irrelevant elaborations, text-based elaborations, or knowledge-based elaborations. Paraphrases did not go beyond the meaning of the target sentence. Irrelevant elaborations may have been related to the sentence superficially or tangentially, but were not related to the overall meaning of the text and did not add to the meaning of the text. Text-based elaborations included bridging inferences that made links to information presented in the text prior to the sentence. Knowledge-based elaborations included the use of prior knowledge to add meaning to the sentence. This latter category is analogous to, but not the same as, the global-focused category in Experiment 1.

Results. In contrast to the human coding system used in Experiment 1, the coding system applied to this data was not intended to map directly onto the iSTART evaluation systems. In this case, the codes are categorical and do not necessarily translate to a 0-3 quality range. One important goal is to be able to assess (or discriminate) the use of reading strategies and improve the system's ability to appropriately respond to the student. This is measured in terms of percent agreement with human judgments of each reading strategy shown in Table 6.3.

Table 6.3. Percent agreement to expert ratings of the self-explanations to the Coal text for the LSA2/WB2-TT and TM2 combined systems for each reading strategy in Experiment 2.

Reading Strategy	LSA2/WB2-TT	TM2
Paraphrase Only	69.9	65.8
Irrelevant Elaboration Only	71.6	76.0
Current Sentence Elaboration Only	71.9	71.2
Knowledge-Based Elaboration Only	94.6	90.3
Paraphrase + Irrelevant Elaboration	79.7	76.6
Paraphrase + Current Sentence Elaboration	68.2	67.3
Paraphrase + Knowledge-Based Elaboration	84.6	81.2

The results show that both systems perform very well, with an average of 77% for the LSA2/WB2-TT system and 75% for the TM2 system. This approaches our criteria of 85% agreement between trained experts who score the self-explanations. The automated systems could be thought of as 'moderately trained scorers.' These results thus show that either of these systems would guide appropriate feedback to the student user.

The score for each strategy score (shown in Table 6.3) can be coded either 0=present or 1=present. With the current coding scheme, only one strategy (out of seven) will be given a value of 1. We are currently redefining the coding scheme so that each reading strategy will have its own scores. For example, if the explanation contains both paraphrase and current sentence elaboration, with the current coding scheme, "Paraphrase + Current Sentence Elaboration" will be coded as a 1. On the other hand, with the new coding scheme, we will have at least 3 variables: (1) "Paraphrase" will be coded as a 1 for *present*, (2) "Elaboration" coded as a 1 for *present*, and (3) "Source of Elaboration" coded as a 2 for *current sentence elaboration*.

6.4 Discussion

The purpose of this chapter has been to investigate the ability of topic model algorithms to identify the quality of explanations as well as specific reading strategies in comparison to word-based and LSA-based algorithms. We found in Experiment 1 that TM systems performed comparably to the combined systems, though not quite as well. In Experiment 2, we found that the TM models performed nearly as well as the combined system in identifying specific strategies. These results thus broaden the scope of NLP models that can be applied to problems such as ours — providing real-time feedback in a tutoring environment. Indeed, the performance of both systems in Experiment 2 was highly encouraging. These results indicate that future versions of iSTART will be able to provide specific feedback about reading comprehension strategy use with relatively high confidence.

Our future work with the TM systems will be to attempt to combine the TM algorithms with the LSA and word-based algorithms. To venture toward that goal, we need to first identify the strengths of the TM algorithms so that the combined algorithm capitalizes on the strengths of the TM — much as we did when we created the combined word-based and LSA-based system. This will require that we analyze a greater variety of protocols, including self-explanations from a greater variety of texts and text genres. We are in the process of completing that work.

These NLP theories and their effectiveness have played important roles in the development of iSTART. For iSTART to effectively teach reading strategies, it must be able to deliver valid feedback on the quality of the self-explanations that a student types during practice. In order to deliver feedback, the system must understand, at least to some extent, what a student is saying in his or her self-explanation. Of course, automating natural language understanding has been extremely challenging, especially for non-restrictive content domains like self-explaining a text in which a student might say one of any number of things. Algorithms such as LSA opened up a horizon of possibilities to systems such as iSTART — in essence LSA provided a 'simple' algorithm that allowed tutoring systems to provide appropriate feedback to students (see [14]). The results presented in this chapter show that the topic model similarly offers a wealth of possibilities in natural language processing.

6.5 Acknowledgments

References

1. Birtwisle,   M.   (2002)   The   Soundex   Algorithm.   Retrieved   from: http://www.comp.leeds.ac.uk/matthewb/ar32/basic_soundex.htm
2. Bransford, J., Brown, A., & Cocking, R., Eds. (2000). How people learn: Brain, mind, experience, and school. Washington, D.C.: National Academy Press. Online at: http://www.nap.edu/html/howpeople1/

3. Chi, M. T. H., De Leeuw, N., Chiu, M., & LaVancher, C. (1994). Eliciting self-explanations improves understanding. Cognitive Science, 18, 439-477.
4. Chi, M. T. H., Bassok, M., Lewis, M. W., Reimann, R., & Glaser, R. (1989). Self-explanation: How students study and use examples in learning to solve problems. Cognitive Science, 13, 145-182.
5. Christian. P. (1998) Soundex — can it be improved? Computers in Genealogy, 6 (5)
6. Graesser, A. C., Penumatsa, P., Ventura, M., Cai, Z., & Hu, X. (2005). Using LSA in AutoTutor: Learning through mixed-initiative dialogue in natural language. In T. Landauer, D.S., McNamara, S. Dennis, & W. Kintsch (Eds.), LSA: A Road to Meaning. Mahwah, NJ: Erlbaum.
7. Graesser, A. C., Hu, X., & McNamara, D. S. (2005). Computerized learning environments that incorporate research in discourse psychology, cognitive science, and computational linguistics. In A. F. Healy (Ed.), Experimental Cognitive Psychology and its Applications: Festschrift in Honor of Lyle Bourne, Walter Kintsch, and Thomas Landauer. Washington, D.C.: American Psychological Association.
8. Graesser, A. C., Hu, X., & Person, N. (2001). Teaching with the help of talking heads. In T. Okamoto, R. Hartley, Kinshuk, J. P. Klus (Eds.), Proceedings IEEE International Conference on Advanced Learning Technology: Issues, Achievements and Challenges (460-461).
9. Graesser, A. C., McNamara, D. S., Louwerse, M. M., & Cai, Z. (2004). Coh-Metrix: Analysis of text on cohesion and language. Behavior Research Methods, Instruments, and Computers, 36, 193-202.
10. Griffiths, T., & Steyvers, M. (2004). Finding Scientific Topics. Proceedings of the National Academy of Science, 101 (suppl. 1), 5228-5235.
11. Kurby, C.A., Wiemer-Hastings, K., Ganduri, N., Magliano, J.P., Millis, K.K., & McNamaar, D.S. (2003). Computerizing Reading Training: Evaluation of a latent semantic analysis space for science text. Behavior Research Methods, Instruments, and Computers, 35, 244-250.
12. Kintsch, E., Caccamise, D., Dooley, S., Franzke, M., & Johnson, N. (2005). Summary street: LSA-based software for comprehension and writing. In T. Landauer, D.S., McNamara, S. Dennis, & W. Kintsch (Eds), LSA: A Road to Meaning. Mahwah, NJ: Erlbaum.
13. Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to Latent Semantic Analysis. Discourse Processes, 25, 259-284.
14. Landauer, T. K., McNamara, D. S., Dennis, S., & W. Kintsch. (2005) LSA: A Road to Meaning, Mahwah, NJ: Erlbaum.
15. Louwerse, M. M., Graesser, A. C., Olney, A., & the Tutoring Research Group. (2002). Good computational manners: Mixed-initiative dialog in conversational agents. In C. Miller (Ed.), Etiquette for Human-Computer Work, Papers from the 2002 Fall Symposium, Technical Report FS-02-02, 71-76.
16. Magliano, J. P., Todaro, S., Millis, K. K., Wiemer-Hastings, K., Kim, H. J., & McNamara, D. S. (2004). Changes in reading strategies as a function of reading training: A comparison of live and computerized training. Submitted for publication.
17. McNamara, D. S. (2004). SERT: Self-explanation reading training. Discourse Processes, 38, 1-30.
18. McNamara, D. S., Boonthum, C., Levinstein, I. B., & Millis, K. K. (2005) Using LSA and word-based measures to assess self-explanations in iSTART. In

T. Landauer, D.S. McNamara, S. Dennis, & W. Kintsch (Eds.), LSA: A Road to Meaning, Mahwah, NJ: Erlbaum.

19. McNamara, D. S., Levinstein, I. B., & Boonthum, C. (2004). iSTART: Interactive strategy training for active reading and thinking. Behavior Research Methods, Instruments, & Computers, 36, 222-233.

20. McNamara, D. S., Louwerse, M. M., & Graesser, A. C. (2002). Coh-Metrix: Automated cohesion and coherence scores to predict text readability and facilitate comprehension. Technical report, Institute for Intelligent Systems, University of Memphis, Memphis, TN.

21. McNamara, D. S., & Scott, J. L. (1999). Training reading strategies. In M. Hahn & S. C. Stoness (Eds.), Proceedings of the Twenty-first Annual Meeting of the Cognitive Science Society (pp. 387-392). Hillsdale, NJ: Erlbaum.

22. Millis, K. K., Kim, H. J., Todaro, S. Magliano, J. P., Wiemer-Hastings, K., & McNamara, D. S. (2004). Identifying reading strategies using latent semantic analysis: Comparing semantic benchmarks. Behavior Research Methods, Instruments, & Computers, 36, 213-221.

23. Millis, K. K., Magliano, J. P., Wiemer-Hastings, K., Todaro, S., & McNamara, D. S. (2005). Assessing comprehension with Latent Semantic Analysis. In T. Landauer, D.S. McNamara, S. Dennis, & W. Kintsch (Eds.), LSA: A Road to Meaning, Mahwah, NJ: Erlbaum.

24. O'Reilly, T., Best, R., & McNamara, D. S. (2004). Self-Explanation reading training: Effects for low-knowledge readers. In K. Forbus, D. Gentner, T. Regier (Eds.), Proceedings of the Twenty-sixth Annual Meeting of the Cognitive Science Society (pp. 1053-1058). Mahwah, NJ: Erlbaum.

25. O'Reilly, T., Sinclair, G. P., & McNamara, D. S. (2004). Reading strategy training: Automated verses live. In K. Forbus, D. Gentner, T. Regier (Eds.), Proceedings of the Twenty-sixth Annual Meeting of the Cognitive Science Society (pp. 1059-1064). Mahwah, NJ: Erlbaum.

26. Press, W.M., Flannery, B.P., Teukolsky, S.A., & Vetterling, W.T. (1986). Numerical recipes: The art of scientific computing. New York, NY: Cambridge University Press.

27. Steyvers, M., & Griffiths, T. (2005) Probabilistic topic models. In T. Landauer, D.S. McNamara, S. Dennis, & W. Kintsch (Eds.), LSA: A Road to Meaning, Mahwah, NJ: Erlbaum.

28. Steyvers, M., & Griffiths, T. (2005) Matlab Topic Modeling Toolbox 1.3. Retrieved from http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm

29. Streeter, L., Lochbaum, K., Psotka, J., & LaVoie, N. (2005). Automated tools for collaborative learning environments. In T. Landauer, D.S., McNamara, S. Dennis, & W. Kintsch (Eds.), LSA: A Road to Meaning. Mahwah, NJ: Erlbaum.