Text-to-Text Similarity of Sentences

Vasile Rus[1], Mihai Lintean[1], Arthur C. Graesser[2], Danielle S. McNamara[2]
[1]*Department of Computer Science*
[2]*Department of Psychology*

Institute for Intelligent Systems
The University of Memphis
Memphis, TN 38152
USA

ABSTRACT

Assessing the semantic similarity between two texts is a central task in many applications, including summarization, intelligent tutoring systems, and software testing. Similarity of texts is typically explored at the level of word, sentence, paragraph, and document. The similarity can be defined quantitatively, e.g. in the form of a normalized value between 0 and 1, and qualitatively in the form of semantic relations such as elaboration, entailment, or paraphrase. In this chapter, we focus first on measuring quantitatively and then on detecting qualitatively sentence-level text-to-text semantic relations. A generic approach that relies on word-to-word similarity measures is presented as well as experiments and results obtained with various instantiations of the approach. In addition, we provide results of a study on the role of weighting in Latent Semantic Analysis, a statistical technique to assess similarity of texts. The results were obtained on two data sets: a standard data set on sentence-level paraphrase detection and a data set from an intelligent tutoring system.

INTRODUCTION

Computational approaches to language understanding can be classified into two major categories: *true-understanding* and *text-to-text similarity*. In true understanding, the goal is to map language statements onto a deep semantic representation that relate language constructs to world and domain knowledge. Current state-of-the-art approaches that fall into this true-understanding category offer adequate solutions only in very limited contexts, i.e. toy-domains, lacking scalability and thus having limited use in real world applications such as summarization or intelligent tutoring systems.

Text-to-text similarity approaches (T2T) to text semantic analysis avoid the hard task of true understanding by defining the meaning of a text based on its similarity to other texts, whose meaning is assumed to be known. Such methods are called benchmarking methods as they rely on a benchmark text, analyzed by experts, to indentify the meaning of new, unseen texts. We adopt in this chapter a T2T approach to semantic text analysis.

In particular, we focus on the task of quantifying how similar two texts are, and based on this, we then decide whether they are similar enough to be considered a paraphrase or not. An example of two texts, a textbase (T) and student paraphrase (SP; reproduced as typed by the student in iSTART, an intelligent tutoring system; McNamara, Levinstein, & Boonthum, 2004), is provided below (from the User Language Paraphrase Challenge; McCarthy & McNamara, 2008):

T: *During vigorous exercise, the heat generated by working muscles can increase total heat production in the body markedly.*

SP: *alot of excercise can make your body warmer.*

Human judges deemed the T and SP in this example to be similar, i.e. in a paraphrase relationship,

We present in this chapter two categories of approaches to the task of sentence-level paraphrase identification: knowledge-based and statistical-based. A generic approach that relies on knowledge-based

word-to-word similarity measures is discussed. In addition, we present a generic approach based on Latent Semantic Analysis (LSA; Landauer et al., 2007), a statistical technique to assess similarity of texts, which is used in combination with several weighting schemes to address the task of paraphrase identification. These approaches were tested on two data sets: the Microsoft Research Paraphrase corpus (MSRP; Dolan et al., 2004), a standard data set on sentence-level paraphrase detection, and a data set from the intelligent tutoring system iSTART (McCarthy & McNamara, 2008) .

BACKGROUND

In this section, we present background information related to word-level similarity measures, as they form the foundation of methods we propose.

There are two main groups of word-level similarity techniques: knowledge-based and statistical. In the knowledge-based category, the lexical database WordNet is used as a knowledge base (Miller, 1995). WordNet groups words with same meaning into synsets (synonymoys sets). Each synset defines a concept, i.e. a uniquely identified meaning. A word can belong to more than one synset in cases where the word is polysemous, i.e. it has many senses. WordNet contains only content words: nouns, verbs, adjectives, and adverbs. It should be noted that WordNet simply offers a glossary of possible senses for words. Identifying the exact meaning (out of many) of a word according to WordNet is equivalent to identifying the synset that best captures the meaning of the word given its context in a particular text fragment. That is, the meaning of the word is entailed by the company it keeps in a text fragment. The task of identifying the correct sense given the context is called word sense disambiguation, one of the most difficult tasks in natural language processing. Text-to-text similarity methods that rely on WordNet-based word-to-word similarity measures need word sense disambiguation in order to be used.

There are nearly a dozen WordNet-based similarity measures available (Pedersen et al., 2004). These measures are further divided into two groups: similarity measures and relatedness measures. The similarity measures are limited to within-category concepts (noun-noun or verb-verb) and usually they work only for nouns and verbs. The text relatedness measures on the other hand can be used to compute similarity among words belonging to different categories, e.g. between a noun and an adjective. The cross-category applicability is very important to us because, for instance, the semantic similarity between the adjective *warmer* in the student paraphrase and the noun *heat* in the textbase of the example given in the *Introduction* section can be computed with relatedness measures but not with similarity measures. Therefore, we focus in this chapter on the relatedness measures.

Word relatedness measures use lexico-semantic information in WordNet to decide semantic similarity between words. As already mentioned, WordNet groups words that have the same meaning, i.e. synonyms, into *synsets* (synonymous sets). For instance, the synset of *{affectionate, fond, lovesome, tender, warm}* corresponds to the concept of *(having or displaying warmth or affection),* which is the definition of the concept in WordNet. Each concept has attached to it a gloss, which contains its definition and several usage examples. Words can belong to more than one synset/concept in WordNet, in case they have more than one meaning. Concepts are linked via lexico-semantic relations such as *hypernymy (is-a), hyponymy (reverse is-a),* and *meronymy (part-of).* The nouns and verbs are organized into a hierarchy using the hypernymy relation. An example of a hypernymy relation is shown in Figure 1 between the concept of organ and that of body-part. The interpretation is that organ is a subconcept, or a specialization of body-part or simply "organ is a type of body-part". A snapshot of the WordNet hierarchy is shown in Figure 1.

In general, two concepts are semantically more related if they are closer to each other in the WordNet web of concepts. In Figure 1, the concept of *{muscle, musculus}* is more related to the concept of *{contractile organ, contractor}* than to *{body}*. The rationale is that the former two concepts are one *hypernymy* link away whereas the latter two are four links away (including one change of direction while following the *hypernymy* link between *{ body part}* and *{body}*). Various WordNet relatedness measures compute the distance in different ways. We experimented with several measures, described later.

Latent Semantic Analysis (LSA; Landauer et al., 2007) is a statistical approach to both representing the semantic of texts in the form of a vector representation and computing similarity between texts. It can be used to assess word-to-word, sentence-to-sentence, and paragraph-to-paragraph semantic similarities.

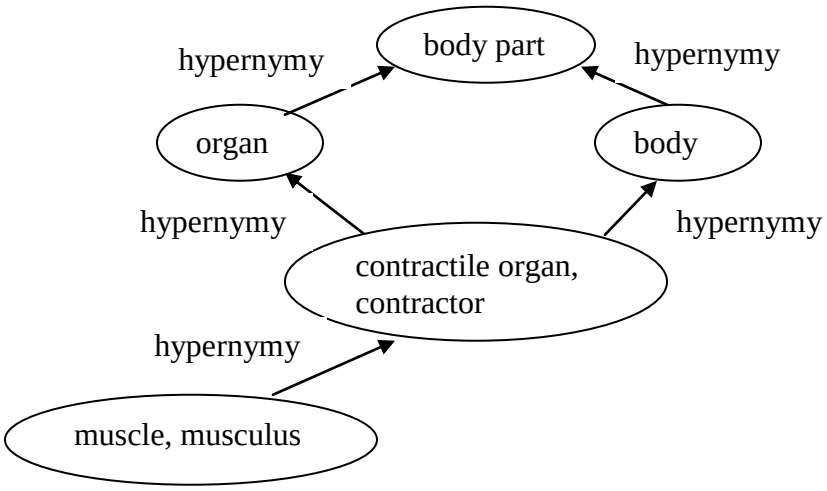

Figure 1. Snapshot of WordNet hierarchy.

LSA defines the meaning of words through a co-occurrence analysis of words in large collections of texts (called *documents*). The idea is that two words are related if they co-occur frequently in same contexts. The co-occurrence statistics are first stored in a term-by-document matrix in which each cell represents the weight of the term in the corresponding document. The weight can be binary, raw or normalized frequency, or any other weighting such as the well-established term-frequency/inverted-document-frequency weight (tf-idf) used in information retrieval. The term-by-document frequency is then passed as input to a mathematical procedure, Singular Value Decomposition (SVD), with the ultimate goal to find an approximation of the initial term-by-document matrix based on the largest *k* singular values obtained with SVD. Usually, k is somewhere between 300 and 500 resulting in a representation for individual words in the form of a 300-500-dimension vector. Each dimension in this representation is a so-called latent concept. The similarity of two words can be quantitatively characterized by computing the cosine of the two words' LSA vectors, i.e. the normalized dot-product of the vectors. If binary weighting is used then the dot-product is identical to computing a word overlap score, i.e. the number of common words. Normalization discounts the effect of document lengths on the value of the dot-product.

In a way, the meaning of a word is defined by the words it co-occurs with, i.e. the company it keeps. In contrast to WordNet, according to LSA each word has a unique meaning. Polysemy is not possible in LSA-based representations. A consequence of this latter fact is that there is no need for word sense disambiguation in text-to-text similarity approaches that rely on LSA-based representations. One advantage of the LSA representation is its scalability beyond individual words to sentences and paragraphs. The LSA representation of a sentence can be obtained by adding up the LSA vectors corresponding to the individual words in the sentence. Similarly, for paragraphs one can simply add the LSA vectors of the corresponding words. The sum of vectors can be weighted, meaning the vectors of some words in the sentence or paragraph can be given more or less importance. In McNamara, Cai, and Louwerse (2007), we evaluated variations of the LSA algorithm to examine whether the performance of LSA could be improved by varying two factors: emphasis on high versus low frequency words, and similarity strictness. In the first study, a variation of LSA in which the weight of rare words was emphasized enhanced the algorithm's ability to detect differences in relatedness between three types of sentence pairs. This variation also better detected differences

between high and low cohesion texts. By contrast, performance in terms of making fine grained distinctions between paraphrases was enhanced when the algorithm emphasized more frequent words. Overall, the study indicated that different algorithms may be more apt to detect differences in meaning depending on the level of analysis. Thus, different algorithms may be more or less appropriate and effective depending on the cognitive processes that are targeted in the particular study.

Another type of weighting has been explored by Hu and colleagues (2003) in which they considered dimensionality weighting based on the observation that the first dimension in LSA vectors is always larger than the remaining dimensions. To summarize, there are three types of weighting in LSA: pre-weighting in the term-by-document matrix, post-weighting when summing up the vectors of individual words to obtain LSA-based representations of longer texts, and dimensionality weighting that can be applied during cosine computation. Post-weighting is an important aspect of extending word-to-word similarity methods beyond the word-level in our text-to-text similarity methods presented later.

PARAPRASE: A TEXT-TO-TEXT SEMANTIC RELATION

A major issue in paraphrase identification is that the exact definition of a paraphrase at sentence level is yet to be discovered. A quick search with the query *What is a paraphrase?* on a major search engine reveals many definitions for the concept of paraphrase. Table 1 presents a small sample of such definitions (also, see McCarthy, Guess, & McNamara, 2009). From the table, we notice that the most common feature in all these definitions is different/own words. That is, a sentence is a paraphrase of another sentence if it conveys the same meaning using different words. While these definitions seem to be quite clear, when it comes to sentence-level paraphrases, i.e. among texts the size of a sentence, there seems to be some issues that apparently are in contradiction with the above definitions.

Source	**Definition (A paraphrase is...)**
Wikipedia	a restatement of a text or passage *using different words.*
WordNet	express the same message in *different words*; *rewording* for the purpose of clarification.
Purdue's Online Writing Lab	*your own rendition* of essential information and ideas expressed by someone else, presented in a new form.
Pearson's Glossary	to record someone else's words in the writer's *own words*.

Table 1: Definitions of paraphrases from various sources.

For sentential paraphrases, the feature of *different words* seems to be too restrictive, although not impossible. Instances in both the iSTART and MSRP corpora support this claim as the paraphrases in the these corpora tend to have many words in common (67.98% in MSRP versus 57.65% in iSTART; when lemmatizing the words the overlaps are 69.5% and 57.65%, respectively). While the high lexical overlap of the paraphrases in the MSR corpus can be explained by the protocol used to create the corpus - keywords were used to retrieve the same stories from different sources on the web, in general, we can argue that avoiding the high word overlap issue in sentential paraphrasing would be hard. Given an isolated sentence, it would be quite challenging to omit/replace some core concepts when trying to paraphrase it. Here is an example of a sentence, *Junya Tanase is a forex strategist at JP Morgan Chase.*, which would be hard to paraphrase with many new/different words due to the large number of named entities in it (this is from instance 962 in MSR corpus). In the iSTART corpus, the cause of high-lexical overlap may be students' modest background on the topic at hand, i.e. biology. If given a biology sentence and asked to paraphrase, students tend to re-use much of the words in the original sentence as their knowledge is limited. It is beyond the scope of this chapter to provide a final answer with respect to whether high lexical overlap should be acceptable or not in sentential paraphrases.

Another interesting aspect of sentential paraphrases is the fact that there seem to be two different ways to judge sentential paraphrases. On the one hand, two sentences are considered paraphrases of each other if and only if they are semantically equivalent, i.e. they both convey the same message with no additional information present in one of the sentences. An example of two sentences in a semantic equivalence is given below.

Text A: *York had no problem with MTA's insisting the decision to shift funds had been within its legal rights.*

Text B: *York had no problem with MTA's saying the decision to shift funds was within its powers.*

In this case, to detect whether two sentences are paraphrases of each other, we only need to find one concept which is present in one sentence but not in the other to make a non-paraphrase decision.

On the other hand, two sentences are in a paraphrase relation if they convey roughly the same message with some minor details being different. In this case, the paraphrase relation can be looked at as a bidirectional entailment relation (Rus et al., 2009). To exemplify such loose paraphrases, we show below a pair of sentences that has been tagged as a paraphrase in the MSR Paraphrase Corpus:

Text A: *Ricky Clemons' brief, troubled Missouri basketball career is over.*

Text B: *Missouri kicked Ricky Clemons off its team, ending his troubled career there.*

In this example, the first sentence specifies that the career of Mr. Clemons was brief, while the second sentence specifies the reason why Mr. Clemons' career is over. The MSRP corpus, one our experimental data set, contains both types of sentential paraphrases, i.e. precise and loose paraphrases. This characteristic of the MSRP corpus impacts the performance of general approaches to paraphrase identification, such as ours, that are not tailored towards judges' biases.

Solutions and Recommendations

As already mentioned, we focus in this chapter on text-to-text similarity approaches that rely on word-to-word similarity measures to compute similarity between two longer texts.

Knowledge-based Text-to-text Similarity. These approaches combine word-level similarities between pairs of words in the two texts. For instance, Rus and colleagues (in press; Rus & Graesser, 2006) used a lexical overlap component combined with syntactic overlap and negation components to compute a unidirectional subsumption score between two sentences T (Text) and H (Hypothesis). The subsumption score reflects how much a text is subsumed or contained by another. Equation 1 provides the overall subsumption score, which can be averaged both ways to compute a similarity score, as opposed to just the subsumption score, between the two texts as shown in Equation 2. The lexical component can be used by itself (given a weight of 1 with the syntactic component given a weight of 0) in which case the similarity between the two texts is just an extension of word-to-word similarity measures. The *match* function in Equation 1 can be any word-to-word similarity measure including simple word match, WordNet similarity measures, or LSA-based similarity.

$$entscore(T,H) = (\alpha \times \frac{\sum_{V_h \in H_v} \max_{V_t \in T_v} match(V_h, V_t)}{|V_h|} + \beta \times \frac{\sum_{E_h \in H_e} \max_{E_t \in T_e} match(E_h, E_t)}{|E_h|} + \gamma) \times (\frac{1 + (-1)^{\#neg\_rel}}{2})$$

Equation 1. Entailment score between a Text and a Hypothesis. Max represents the maximum function while match is a matching operation of words V_h and V_t, one in the Hypothesis and one in Text, respectively.

$$paraphase(T1, T2) = \frac{entscore(T1, T2) + entscore(T2, T1)}{2}$$

Equation 2. Paraphrase score between texts T1 and T2 based on an Entailment score. Entscore (T1, T2) is the entailment score between T1 and T2 obtained using Equation 1.

Such a similar approach has been proposed by Rus et al. (2009), who weighted each word by their importance using an inverted-document-frequency weight. They only used lexical similarities (no syntactic similarities) to assess the degree of semantic relatedness between two sentences. Lexical similarities between words were assessed using various WordNet-based word-to-word similarity functions. Equation 3 illustrates the weighted sum of word-level similarities. This score can be generalized as shown in Equation 4 where we use generic *weight* and *word-sim* functions. We will use this generic approach and instantiate it by replacing the generic weight and word-sim functions with specific weights and word-to-word similarity measures. If the weight for each word is set to 1 then the similarity score in Equation 3 is equivalent to the lexical component in Equation 1. Likewise, Mihalcea, Corley, and Strapparava (2006) used a similarity technique based on word-to-word similarity measures for identifying paraphrase relations between sentences in a pair. Fernando and Stevenson (2008) also relied on word-level similarities to compute similarities of sentences. Instead of taking a maximum similarity as we do, they take averages between a word in one text and all the words in the other. Taking the average gives credit to words in one sentence that are relatively similar but not very closely related to words in the other sentence.

$$score(T1, T2) = \frac{\sum_{v \in T1} idf(v) * \max_{w \in T2} WordNet - sim(v, w)}{\sum_{v \in T1} idf(v)}$$

Equation 3. Semantic similarity score between texts T1 and T2 using WordNet-based word-to-word similarity measures.

$$score(T1, T2) = \frac{\sum_{v \in T1} weight(v) * \max_{w \in T2} word - sim(v, w)}{\sum_{v \in T1} idf(v)}$$

Equation 4. Generic semantic similarity score between texts T1 and T2.

Text-to-Text Similarity based on Latent Semantic Analysis. The use of LSA to compute similarity of texts beyond word-level relies mainly on combining the vector representation of individual words. Specifically, the vector representation of a text containing two or more words is the weighted sum of the LSA vectors of the individual words. If we denote $weight_w$ as the weight of a word given by some scheme, local or global, then the vector of a text T (sentence or paragraph) is given by Equation 5. In Equation 5, *w* takes value from the set of unique words in a text T, i.e. from the set of word types of T. If a word type occurs several times in the document that will be captured by the local weight ($loc - weight$). $Glob - weight$ in Equation 2 represents the global weight associated with type *w*, as derived from a large corpus of documents, Wikipedia in our case.

$$V(T) = \sum_{w \in T} loc - weight_w * glob - weight_w V_w$$

Equation 5. Formula to generate an LSA-based vector for a text T based in the individual LSA vectors of the words it contains.

To find the LSA similarity score between two texts T1 and T2, i.e. LSA(T1, T2), we first represent each sentence as vectors in the LSA space, V (T1) and V (T2), and then compute the cosine between the two vectors as shown in Equation 1. Cosine is the normalized dot-product of the corresponding vectors.

There are several ways to compute local and global weights. For local weighting, the most common schemes are *binary*, *type frequency*, and *log-type frequency*. *Binary* weighting refers to the use of 1 if the word type occurs at least once in the document and 0 if it does not occur at all. *Type frequency* weight is defined as the number of times a word type appears in a text, sentence, or paragraph. *Log-type frequency* weight is defined as log(1 + *type frequency*). Dumais (1991) argued that type frequency gives too much weight/importance to very common, i.e. frequent, words. A frequent word such as *the* which does not carry much meaning will have a big impact although its entropy (described next) is low, which is counterintuitive. To diminish the frequency factor for such words, but not eliminate it entirely, the log-type weighting scheme was proposed.

As global weight, we started with a binary weight: 1 if the word exists in the text, 0 otherwise. The most commonly used global weight is entropy-based. It is defined as $1+\sum_j \frac{p_{ij} * \log_2 p_{ij}}{\log_2 n}$, where $p_{ij} = tf_{ij}/gf_i$, tf_{ij}= type of frequency of type i in document j, and gf_i= the total number of times that type i appears in the entire collection of n documents. We also used IDF (Inverted Document Frequency), as a global weight. IDF was derived from the English section of Wikipedia (details provided later)

Experiments and Results. We present experiments and results with the generic approach in Equation 3 as well as with the LSA-based approach in Equation 5.

Two sentence-level data sets were used in our experiments: the Microsoft Research Paraphrase Corpus (Dolan et al., 2004) and the intelligent tutoring systems ULPC/iSTART corpus (McCarthy & McNamara, 2008). We report results using the performance measures of accuracy and kappa. *Accuracy* is the percentage of correct predictions out of all predictions. *Kappa* coefficient measures the level of agreement between predicted categories and expert-assigned categories while also accounting for chance agreement. Results were obtained using 10-fold cross-validation, except for the MSR dataset, which contains an explicit test subset. In k-fold cross validation the available data is divided into k equal folds. Then, k trials are run, one for each fold. In each trial one fold is set aside for testing and the other (k − 1) are used for training. The average of the accuracies for the k trials is reported. When k = 10, we have 10-fold cross validation.

Given the need for word distributional information for our weighting schemes, i.e. inverted document frequency (idf), it is important to derive as accurate estimates of word statistics as possible. Accurate word statistics means being representative of overall word usage (by all people at all times). The accuracy of the estimates is largely influenced by the collection of texts from which the statistics are derived. Various collections have been used to derive word statistics. For instance, Corley and Mihalcea (2005) used the British National Corpus as a source for their IDF values. We chose Wikipedia instead because it encompasses texts related to both general knowledge and specialized domains and it has been edited by many individuals, thus capturing diversity of language expression across individuals. Furthermore, Wikipedia is one of the largest publicly available collections of English texts. Extracting IDF values and word statistics from very large collections of text, such as Wikipedia, is a non-trivial task. Due to space limitations we do not present the details of this step. We just mention that the number of distinct words chosen after many processing steps was *2,118,550*. We have collected distributional information for this set of words and used it in our experiments.

Sentence-level Similarity in iSTART

We focus in this section on evaluating student input in iSTART (Interactive Strategy Training for Active Reading and Thinking; McNamara et al., 2004), an ITS that provides students with reading strategy training. One of the modules in iSTART focuses on training students to paraphrase science sentences, called the *textbase* (T). Assessing the student paraphrases (SP) is a critical step in iSTART because this assessment detects possible student misunderstandings and provides the necessary corrective feedback.

The User Language Paraphrase Corpus (ULPC; McCarthy & McNamara, 2008) is a compiled data set of 1998 pairs of Textbase-SP in iSTART. There are 10 dimensions of analysis available in the

ULPC including elaboration, semantic completeness, entailment, lexical similarity, and paraphrase quality. It should be noted that some of these dimensions have meanings in the ULPC that need be specified as they are not obvious or differ from definitions used by others. In the ULPC, elaboration refers to student paraphrases regarding the theme of the textbase rather than a restatement of it. Semantic completeness refers to a SP having the same meaning as the textbase, regardless of word- or structural-overlap. This dimension is of most interest to us because it best matches our goal of detecting semantic similarities among texts. Paraphrase quality takes into account semantic-overlap, syntactical variation, and writing quality. Given these definitions, the semantic completeness dimension in ULPC is equivalent to the paraphrase evaluation in the MSRP corpus (Dolan et al., 2004).

An example of a textbase and student paraphrase in iSTART was provided in the *Introduction* section. We present next several instantiations of the generic approach to Text-to-Text similarity presented earlier for the problem of assessing student paraphrases in iSTART. The instantiations were obtained by replacing the WordNet-sim function in the formula in Equation 3 with several knowledge-based word-to-word similarity measures.

WordNet-based Similarity. As mentioned earlier, our goal is to assess various instantiations of the generic approach to text-to-text similarity shown in Equation 4. We first discuss how to instantiate it using WordNNet-based similarity measures. We used the following word relatedness measures (implemented in the WordNet::Similarity package; Pedersen, Patwardhan, & Michelizzi, 2004): HSO (Hirst & St-Onge, 1998), LESK (Banerjee & Pedersen, 2003), and VECTOR (Patwardhan, 2003). Given two WordNet concepts, these measures provide a real value indicating how semantically related the two concepts are.

The HSO measure is path based, i.e., it uses the relations between concepts, and assigns direction to relations in WordNet. For example, the *is-a* relation is upwards, while the *has-part* relation is horizontal. The LESK and VECTOR measures are gloss-based. That is, they use the text of the gloss as the source of meaning for the underlying concept. One challenge with the above word-to-word relatedness measures is that they cannot be directly applied to compute the similarity of larger texts such as sentences.

One challenge with using the WordNet-based similarity measures is that texts use words and not concepts, which are needed as input by these measures. To be able to use the measures we must map words to concepts in WordNet, i.e. we must do word sense disambiguation (WSD). It is beyond the scope of our investigation to fully solve the WSD problem, one of the hardest in the area of Natural Language Processing. Instead, we address the issue in two ways: (1) map words in the textbase T and SP onto the concept corresponding to their most frequent sense, which is sense #1 in WordNet, and (2) map words onto all the concepts corresponding to all the senses and take the maximum of the relatedness scores for each pair of senses. We label the former method as ONE (sense one), whereas the latter is labeled as ALL (all senses).

In our evaluation, we have explored a space of 3x2x2=12 solutions/instantiations as a result of combining three relatedness measures (HSO, LESK, and VECTOR), two word sense disambiguation methods (ONE and ALL), and the two weighting schemes (with and without IDF weighting). The labels of each instantiation have a specific meaning. For instance, the label ALL_IDF_LESK means the instantiation that uses ALL the senses of words to compute word-to-word relatedness, weights words using IDF values, and applies the LESK relatedness measure. If no IDF is mentioned in the name of a solution but rather a dash (-), e.g. ONE-LESK, it means no word weighting was used. In order to measure accuracy, we used the binary values for human judgments in the iSTART/ULPC corpus (1.00-3.49 = 0 [low]; 3.50-6.00 = 1 [high]).

Each method goes through a training phase. The training consists of finding a threshold value for a particular solution, e.g. ONE-LESK, above which a prediction is considered high, and low otherwise. These predictions are then compared with the binary human judgments in order to compute the accuracy.

An analysis of the results indicated that the LESK measure provides best results across weighting schemes and word sense disambiguation methods (accuracy=79.47%). Using IDF seems to help when

using the LESK measure while using the ALL disambiguation method does not seem to have a positive impact. That is, picking the first sense in WordNet as the disambiguation method is sufficient for the LESK method. Actually, LESK with no weighting and ALL disambiguation yields slightly worse results than with the ONE disambiguation method.

Regarding weighting, using IDF helps all the related measures regardless of the disambiguation method while the disambiguation method doesn't seem to make a difference. Actually, when no weighting is used the ALL method seems to hurt some the accuracy results.

For the more balanced dimension of paraphrase quality, kappa varies from 0.335 to 0.449(for ONE_IDF_LESK). LSA yields a kappa of 0.361 and the best kappa for an Entailer method is 0.434 (for R-Ent).

Latent Semantic Analysis.

We investigated the impact of several local and global weighting schemes for the ability of Latent Semantic Analysis' (LSA) to capture semantic similarity between two texts. We show results with two local and two global weighting schemes. For local weighting, we used binary weighting and raw term-frequency. For global weighting, we relied on binary and inverted document frequencies (idf) collected from the English Wikipedia. The results are shown in Table 2. Best overall accuracy (78.13) is obtained for a combination of raw frequency local weighting and idf global weighting.

global\local weighting	**Binary**		**Raw**	
	Accuracy	Kappa	Accuracy	Kappa
Binary	76.88	.368	76.88	.364
Idf	77.83	.409	78.13	.417

Table 2. Results obtained with various combinations of local and global weighting in LSA on the ULPCS/iSTART corpus.

The Microsoft Research Paraphrase Corpus

The Microsoft Research (MSR) Paraphrase Corpus (Dolan, Quirk, & Brockett 2004) is a standardized data set for paraphrase identification. Although it has limitations (see Zhang & Patrick, 2005; Lintean, in press), the MSR Paraphrase Corpus is the largest publicly available annotated paraphrase corpus. It has been frequently used in many recent studies that address the problem of paraphrase identification. The corpus consists of 5801 sentence pairs collected from newswire articles, 3900 of which were labeled as paraphrases by human annotators. The whole set is divided into a training subset (4076 sentences of which 2753, or 67%, are true paraphrases), and a test subset (1725 pairs of which 1147, or 66%, are true paraphrases).

An example of a paraphrase from the Microsoft Research Paraphrase (MSRP) corpus (Dolan et al., 2004) in which Text A is a paraphrase of Text B and vice versa is given below.

Text A: *York had no problem with MTA's insisting the decision to shift funds had been within its legal rights.*

Text B: *York had no problem with MTA's saying the decision to shift funds was within its powers.*

Two sentences can be judged as forming a paraphrase if they convey roughly the same message (minor details being different is acceptable). To exemplify this definition of a loose paraphrase, we show below a pair of sentences that has been tagged as paraphrase in the MSR Paraphrase Corpus:

Text A: *Ricky Clemons' brief, troubled Missouri basketball career is over.*

Text B: *Missouri kicked Ricky Clemons off its team, ending his troubled career there.*

In this example, the first sentence specifies that the career of Mr. Clemons was brief, while the second sentence specifies the reason why Mr. Clemons' career is over. The MSR Paraphrase corpus, our experimental data set, contains both types of sentential paraphrases, i.e. precise and loose paraphrases. This characteristic of the MSR corpus impacts the performance of general approaches, such as ours, to paraphrase identification that is not biased towards judging styles.

The obtained results are shown in Table 3. In terms of accuracy, the results seem to be very close to each other for the four combinations of local and global weighting schemes.

global\local weighting	**Binary**		**Raw frequency**	
	Accuracy	Kappa	Accuracy	Kappa
Binary	70.38	.247	70.55	.244
Idf	69.85	.231	69.74	.228

Table 3. Results obtained with various combinations of local and global weighting in LSA for the MSRP corpus.

FUTURE RESEARCH DIRECTIONS

There are many possible avenues for continuing the quest for the best way solve the problem of text-to-text similarity in general and of paraphrase identification in particular. First, we plan to integrate more types of knowledge in our basic approaches to text-to-text similarity, e.g. semantic information from semantic parsers. Second, we plan to explore further the impact of the various parameters of the proposed methods such as various weighting schemes as well as using averages of word-to-word measures instead of taking the maximum of the similarity scores between pairs of words, one in T1 and one from T2. Third, we plan to explore how well the knowledge-based measures work on the MSRP corpus.

CONCLUSION

We presented in this paper several methods to address the task of text-to-text similarity of texts the size of a sentence. The methods fall into the two major categories of general Natural Language Processing approaches: knowledge-based and statistical. A comparison between knowledge-based and statistical on the iSTART corpus gave a slight advantage to the knowledge-based methods as the best accuracy results with the knowledge-based methods was 79.47% versus 78.13% for best Latent Semantic Analyis method which used raw frequency for local weighting and inverted-document-frequency for global weighting. The advantage of the knowledge-based approaches is their interpretability, which means decisions can be justified based on the explicit paths and semantic relations in the knowledge-based that were used to compute the similarity between words. On the other hand, Latent Semantic Analysis has a scalability advantage. It would be interesting to see how a combination of these two approaches would perform, a possible item in our plans for the future.

REFERENCES[i]

Banerjee, S., and Pedersen, T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, 805-810.

Dolan, B.; Quirk, C.; and Brockett, C. 2004. Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources. In *Proceedings of COLING*, Geneva, Switzerland.

Hirst, G., and St-Onge, D. (1998). *Lexical chains as representations of context for the detection and correction of malapropisms*. In Fellbaum, C., ed., WordNet: An electronic lexical database. MIT Press.

Ibrahim, A.; Katz B.; and Lin, J. 2003. *Extracting Structural Paraphrases from Aligned Monolingual Corpora.* in Proceeding of the Second International Workshop on Paraphrasing, (ACL 2003).
Iordanskaja, L.; Kittredge, R.; and Polgere, A. 1991. *Natural Language Generation in Artificial Intelligence and Computational Linguistics.* Lexical selection and paraphrase in a meaning-text generation model, Kluwer Academic.
Landauer, T., McNamara, D.S., Dennis, S., and Kintsch, W. (Eds), Latent Semantic analysis: A road to meaning, 2007, Mahwah, NJ:Erlbaum.
Patwardhan, S. (2003). *Incorporating dictionary and corpus information into a context vector measure of semantic relatedness*. Master's thesis, Univ. of Minnesota, Duluth.Pedersen, T., Patwardhan, S., and Michelizzi, J. (2004). WordNet::Similarity - Measuring the Relatedness of Concepts, In the Proceedings of the Nineteenth National Conference on Artificial Intelligence (AAAI-04), pp. 1024-1025, July 25-29, 2004, San Jose, CA (Intelligent Systems Demonstration)
McCarthy, P.M. & McNamara, D.S. (2008). User-Language Paraphrase Corpus Challenge, online, 2008.
McNamara, D.S., Cai, Z., & Louwerse, M.M. (2007). Comparing latent and non-latent measures of cohesion. In T. Landauer, D.S. McNamara, S. Dennis, & W. Kintsch (Eds.), *Handbook of Latent Semantic Analysis* (pp. 379-400). Mahwah, NJ: Erlbaum.
McNamara, D.S., Levinstein, I., and Boonthum, C. (2004). iSTART: Interactive Strategy Trainer for Active Reading and Thinking, Behavioral Research Methods, Instruments, and Computers, 2004, 36 (2), 222-233.
Mihalcea, R., Corley, C., & Strapparava, C. Corpus-based and knowledge-based measures of text semantic similarity. In *Proc. of the 21st conference of American Association for Artificial Intelligence (AAAI-06)*, Boston, Massachusetts, July 16-20 2006.
Rus, V., Lintean, M., Shiva, S., & Marinov, D. (submitted). Automated Identification of Duplicate Defect Reports using Word Semantics. Submitted to the *Workshop on Mining Software Repositories*.
Rus, V., McCarthy, P. M., Graesser, A. C., & McNamara, D. S. (2009). Identification of Sentence-to-Sentence Relations Using a Textual Entailer. *The International Journal on Research in Language and Computation*.
Wan, S.; Dras, M.; Dale, R., and Paris, C. Using dependency-based features to take the 'para-farce' out of paraphrase. *2006 Australasian Language Technology Workshop* (ALTW 2006); University of Sydney. 2006: 131-138. ISBN: 1741081467.
S. Fernando and M. Stevenson. A semantic approach to paraphrase identification. In *Proceedings of the 11th Annual Research Colloquium of the UK Special-interest group for Computational Lingusitics*, Oxford, England, 2008.

KEY TERMS & DEFINITIONS

Latent Semantic Analysis: Statistical method for semantic representation of language constructs such as words, sentences, and paragraphs.

Paraphrase: Semantic relation between two texts having the same meaning. While the texts have the same meaning, it is widely accepted they should express it in different ways,

Word weighting: Technique to assign more importance to some words when composing the meaning of larger chunks of texts from the meaning of individual words.
Word-to-word Similarity Measures: Methods to quantify how semantically similar two words are.

Text-to-Text Similarity Measures: Methods to quantify how semantically similar two texts are. The two texts should contain two or more words, i.e. being phrases, sentences, paragraphs, or even documents.

Knowledge-based Similarity Measures: Similarity measures the rely on knowledge resources that explicitly encode semantic relations among words. WordNet is an example of such a resources that specifies semantic relations among the words.

Intelligent Tutoring Systems: Systems that mimic human tutors in their attempt to provide high-quality instruction to students on topics varying from biology to physics to financial literacy.

---

[i] References should relate **only** to the material you actually cited within your chapter (this is not a bibliography). References should be in APA style and listed in alphabetical order. Please do not include any abbreviations.