

# Investigating Graphs in Textual Case-Based Reasoning\*

C. M. Cunningham, R. Weber, J. M. Proctor, C. Fowler, M. Murphy

College of Information Science & Technology, Drexel University, Philadelphia, PA 19104  
{cmc38, rw37, jp338, clf29, mpm37}@drexel.edu

**Abstract.** Textual case-based reasoning (TCBR) provides the ability to reason with domain-specific knowledge when experiences exist in text. Ideally, we would like to find an inexpensive way to automatically, efficiently, and accurately represent textual documents as cases. One of the challenges, however, is that current automated methods that manipulate text are not always useful because they are either expensive (based on natural language processing) or they do not take into account word order and negation (based on statistics) when interpreting textual sources. Recently, Schenker et al. [1] introduced an algorithm to convert textual documents into graphs that conserves and conveys the order and structure of the source text in the graph representation. Unfortunately, the resulting graphs cannot be used as cases because they do not take domain knowledge into consideration. Thus, the goal of this study is to investigate the potential benefit, if any, of this new algorithm to TCBR. For this purpose, we conducted an experiment to evaluate variations of the algorithm for TCBR. We discuss the potential contribution of this algorithm to existing TCBR approaches.

## 1 Introduction

Textual case-based reasoning (TCBR) extracts cases from textual documents whenever knowledge is contained in texts. There are extremely critical tasks and domains where tasks could be automated if text presented recognizable patterns and clear structure. Some examples of relevant domains include help desks [2], customer support [3], intelligent tutoring [4] and law [5]. In the legal domain alone, reasoning from text provides the ability to, for example: predict the outcome of legal cases [6]; construct legal argumentation [7][8], perform jurisprudence research [9], interpret and apply the facts of one case to a new case [8][10], and sentencing [11][12].

In fact, finding legal precedents is central to how the legal system in the US operates. Given the potential issues with acting upon incomplete information (e.g. poorly constructed arguments, misinterpretation and application of the law, erroneous decisions), it would be desirable if the methods used for jurisprudence research had high recall and precision. Recall is the ratio of useful documents that are retrieved to the total number of useful documents that exist [13]. Precision is a ratio of the number of useful documents that are retrieved to the total number of documents that are retrieved [13]. The most widely used technique for finding similar documents is Information Retrieval (IR), which is based on term frequency and measured in terms of recall and precision. IR in the legal domain is not adequate because term

\*In Funk, Peter; González Calero, Pedro A. (Eds.), *Advanced in Case-Based Reasoning (Lecture Notes in Artificial Intelligence, Vol. 3155)*: Springer-Verlag.

frequencies do not take into account domain-specific knowledge, therefore they only recall approximately 25% of relevant documents [14].

Unlike either IR or clustering methods [15], case-based reasoning (CBR) replicates reasoning by analogy to retrieve relevant cases based upon domain-specific knowledge [2][16]. CBR determines similarities between a current event and a past event similar to the manner in which people reason by using analogies. Furthermore, when using domain-specific knowledge to retrieve useful cases, one would expect that recall and precision would improve [9]. One of the challenges in TCBR, however, is finding an automated method to manipulate textual knowledge that takes into consideration order and negation [17] when interpreting the text.

Interestingly, recent developments in Graph Theory relate to text representation [1]. Graphs are mathematical representations that consist of vertexes (nodes) and edges (arcs), which offer a number of advantages over traditional feature vector approaches [18] - the most significant is the ability to create rich representations of cases [19]. Furthermore, unlike vector representations, the structure and word order of the original document can be retained. By definition, graph structures apply to representations that capture relationships between any two elements, as well as allowing an unlimited number of elements to be added or deleted at will [20]. This flexibility of the representation allows CBR cases to capture previously unforeseen information without the need to reconfigure the case base.

When the graphs are unlabeled or their labels are not fixed, the only applicable similarity methods are ones that search for identical subgraphs. This is the well-known subgraph isomorphism problem, which is NP-complete [21]. For this reason and for the nature of CBR similarity, we target case graphs that have fixed labels. This search is polynomial [1]. Additionally, because the fixed labels embed meaning, the similarity assessment is domain-specific. Although being polynomial, graph representations do have a significant computational cost. Fortunately, there are a number of methods and techniques aimed at reducing this problem [18].

There are promising developments in Graph Theory related not only to reduced complexity but also to text representation. In Schenker et al. [1], the authors proposed an algorithm to automatically convert textual documents (i.e. web pages) into graphs. Additionally, they have also demonstrated how to cluster the resulting graphs by using a variation of the  $k$ -means algorithm and by using the maximum common subgraph to measure similarity [22].

Given the successful use of graphs to represent web documents [1], the purpose of this paper is to examine the benefits of the algorithm presented in [1] for representing textual documents as case graphs in TCBR. Section 2 presents two algorithms to convert textual documents: the one proposed by Schenker et al. [1], to convert textual documents into graphs, henceforth referred to as Text-to-Graph (TtG); and our proposed variant that converts textual documents into case graphs, henceforth referred to as Text-to-Case-Graph (TtCG). Section 3 presents the experimental study we conducted to compare these algorithms with a feature vector CBR prototype and a human expert. We then discuss the potential impact of our findings on related work in Section 4. Finally, the conclusion and potential future work are presented in Section 5.

## 2 Graphs in Textual Case-Based Reasoning

Graphs are data structures that allow the easy implementation of algorithms. Therefore, it would be desirable to have textual content represented in graphs. The challenge is to determine a method for the conversion that preserves meaning while keeping graphs at a manageable size. If the goal is to compare graphs by searching for isomorphic subgraphs, this search is NP-complete. However, if the purpose is to assess distance by comparing graphs with fixed labels, then this search is polynomial [1]. In this section, we present the original algorithm presented in [1], TtG; and our proposed variant, TtCG, that represents a first attempt to convert textual documents into case graphs.

### 2.1 From Textual Documents to Graphs

In Schenker et al. [1], the authors introduced an algorithm to automatically convert textual web documents into graphs, Text-to-Graph (TtG). In the TtG approach, the unique words (excluding stop words) that appear in the web document are mapped to vertexes on the graph. Each vertex is then labeled with the unique word that it represents. The directed edges on the graph are drawn from the vertex that represents one word to the vertex that represented the word that immediately follows the first word. The edges are then labeled with the structural section in which the two words appeared. The TtG approach has several implied benefits to textual case-based reasoning. First, the structure and word order of the original document would be retained. Additionally, the TtG approach would reduce the amount of time required by knowledge engineers to encode representation of the textual sources.

Although the TtG approach does retain the word order and structure of the original text, it does not take into account negation. Furthermore, according to Aha [23], CBR is richer when it considers the relative importance of features; however, the TtG approach neither identifies features nor their relative importance. The creators of TtG [1] used a clustering algorithm to group similar textual documents together. It does not indicate the commonalities between the documents within a cluster. The ability to identify features and their relevance on a graph would mean that textual sources could automatically be converted to cases for CBR without the added expense of the time that it would take a knowledge engineer to manually represent a text as a case.

### 2.2 From Textual Documents to Case Graphs

We propose a variant of the TtG, which aims at converting textual documents into *case graphs*. Case graphs are representation formalisms that use graphs to represent situated experiences. Given that the essence of case-based reasoning is similarity, case graphs must be amenable to have their similarity assessed against other case graphs in conformity with the CBR hypotheses. Therefore, similarity is not a domain independent process, but one whose main goal is to replicate domain-specific similarity. For these reasons, our first attempt to create an algorithm to create case graphs from unrestricted data makes use of a list of potential domain-dependent

indexes, which we call *signifiers*. Signifiers can be single words or expressions that we can guarantee play a role in the description of the situated experience. In cases that describe personal injury claims, for example, the occurrence of the term chiropractor is a predictive index. Consequently, our algorithm differs from the one introduced in Schenker et al. [1] in that it preserves the signifiers independent of the level of their occurrence in the source text.

The use of the signifiers allows for the use of traditional graph distance techniques to be used for case-based reasoning. Without the signifiers, we could not use graph distance techniques because they are not suitable to replicate similarity assessment.

### 2.3 Graph Distance Algorithms

Several graph distance techniques rely on finding the maximum common subgraph (MCS) [22]. The maximum common subgraph of two graphs is the set of all linked nodes that the two have in common.

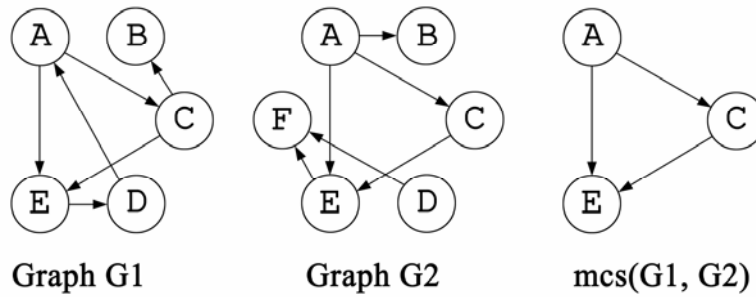


Figure 1. MCS example

In Figure 1, the nodes of the graphs are labeled A, B, C, etc.; these would be words in the graph representation of a document. The arrows indicate word order in the original text. For instance, the document represented by graph G1 in the figure has at least two occurrences of word A, one of which is followed by word C and the other by word E. Note that words B and D appear in both graphs, but they are connected differently, so are not part of the MCS.

Collectively, distance techniques that use MCS are called MCS-based techniques. In [22], the authors also refer to one particular distance formula as *MCS*. In order to distinguish MCS-based techniques from this formula, we refer to the formula as BLG (Bunke Largest Graph). We use BLG [25] and WGU [26], which require finding the maximum common subgraph.

BLG distance is determined by dividing the size (number of vertexes plus number of edges, denoted by  $|\dots|$  in the equations below) of the maximum common subgraph by the size of the larger of the two graphs being compared, and then subtracting the quotient from 1 as shown in Equation 1.

$$d_{\text{BLG}}(G_1, G_2) = 1 - \frac{|\text{mcs}(G_1, G_2)|}{\max(|G_1|, |G_2|)} \quad (1)$$

Unlike the BLG distance, WGU (Wallis Graph Union) distance is not sensitive to graphs of disparate sizes. The WGU distance is determined by dividing the size of the maximum common subgraph by the sum of the sizes of the two graphs being considered minus the size of the MCS (so those nodes are not counted twice), and then subtracting the quotient from 1, as shown in Equation 2.

$$d_{\text{WGU}}(G_1, G_2) = 1 - \frac{|\text{mcs}(G_1, G_2)|}{|G_1| + |G_2| - |\text{mcs}(G_1, G_2)|} \quad (2)$$

The range for both the BLG and WGU distances is from 0.0 (identical) to 1.0 (MCS is null – the graphs have no nodes in common). For example, referring to the graphs in Figure 1 above,  $|G_1| = 5 \text{ vertexes} + 6 \text{ edges} = 11$ ; similarly,  $|G_2| = 12$ , and  $|\text{mcs}(G_1, G_2)| = 6$ . The BLG distance between  $G_1$  and  $G_2 = 1 - (6/12) = 0.5$ , but the WGU distance is  $1 - (6/17) = 0.647$ .

### 3 Experimental Study

Table 1. Summary of approaches used in the experiment

	Domain Expert (DE)	Feature Vector CBR	TtG	TtCG
Source Text	Claim summary documents	Claim summary documents	Claim summary documents	Claim summary documents
Representation Method	DE	DE chose features and identified their values	TtG algorithm to automatically convert texts into graphs	TtCG algorithm to automatically convert texts into graphs
Representation Formalism	Mind of DE	Feature vectors	Graphs	Graphs
Similarity Assessment	DE judgment	Inferred and weighted nearest neighbor	MCS-based distance algorithms	MCS-based distance algorithms

Our hypothesis is that using an algorithm to convert textual documents into graphs is beneficial to textual case-based reasoning. We tested our hypothesis using precision and recall for four different approaches to manipulate text and retrieve relevant documents: domain expert, feature vector CBR, the TtG algorithm and the TtCG approach. The domain expert's assessment was the baseline for the analysis.

This section describes the methods that were used to manipulate and represent the textual documents as well as the techniques that were used to assess the similarity between documents; the dataset; and how our chosen metrics, precision and recall, were computed. Subsections 3.6 and 3.7 present the results and the discussion, respectively. Table 1 summarizes the methodologies.

### 3.1 Domain Expert Method

**Text Manipulation/Representation.** The domain expert was asked to read a collection of claim summary documents in order to identify the similar documents in the collection. In this case, there was not a formal representation of the documents.

**Similarity Assessment.** Based upon experience, the domain expert manually assessed the similarity between the claim summary documents. This method represents the baseline for subsequent analysis.

### 3.2 Feature Vector CBR

Table 2. Feature values in case 1

Question	Values
Was the incident reported?	yes
How soon was the incident reported?	same day
How old is plaintiff?	76
How many personnel injury lawsuits has the plaintiff filed before this complaint?	0
Was the plaintiff employed on the day of the incident?	no
Does the plaintiff have a criminal history that includes crimes of falsehood?	no
Did plaintiff have surgery as a result of the (alleged) incident?	no
How much did the plaintiff spend on medical bills?	4595
Does the plaintiff have pre-existing injuries in the same area as alleged in the current lawsuit?	yes
Is there a loss of consortium or per quod claim?	yes
Was plaintiff treated exclusively by a chiropractor?	no
Is there a permanent loss claim?	no
Is the injury claimed soft-tissue in nature?	yes
What is the plaintiff's annual income?	not available
How many days of work did the plaintiff miss due to the incident?	not available
Are there fact witnesses (other than plaintiff)?	yes
Case number	1
Case file	case1

**Text Manipulation/Representation.** The domain expert reviewed the claim summary documents in order to identify the features that should be used to build the case base. Knowledge engineers then used the features that the expert identified in order to represent the documents as cases within the case base. An example of the identified features and their values for case 1 is shown in Table 2. Additionally, the knowledge engineers worked very closely with the domain expert in order to assign weights to each feature in order to capture the relative importance of each feature. This was a very challenging effort.

**Similarity Assessment.** For the feature vector CBR, the similarity step was designed in a trial-and-error effort. We started by using the feedback feature weighting algorithm gradient descent, but the individual similarities between different values changed based on a variety of reasons. For example, a permanent injury is a

predictive index only when the plaintiff is below a certain age. Consequently, we had to use a number of rules to assign weights whenever conditions changed, and we were limited by the shell we used.

### 3.3 Textual Documents to Graphs

**Text Manipulation/Representation.** The TtG method was used to convert the claim summary documents into graphs. It was not possible to use the algorithm in its exact original form because the claim summary documents did not consistently have three structural sections that were common across all documents. We modified the TtG method of representing textual documents as graphs for the claim summary documents by defining two *sections*—titles and text—instead of TtG’s three (titles, text, and hypertext links), but kept other steps as similar as possible (see Section 2). Stop words were culled from the document, then the remaining ordered list of words was stemmed using Porter’s algorithm [24]. Each unique term in the resulting list was added as a vertex in the graph representation, with its occurrence count as an attribute of the vertex. Directed edges were created between the vertexes representing words that were adjacent in the document, where adjacency crosses stop words but not numerals or *breaking* punctuation (period, question mark, exclamation point, colon, semicolon, parentheses, brackets, and single and double quotation marks). The edges were labeled with the section (title or text) in which the adjacent vertexes appeared. As with the vertexes, the count of adjacent occurrences was an attribute. Finally, the graphs were pruned to only include vertexes which occurred with a minimum frequency (specified at run-time).

**Similarity Assessment.** We used the techniques described in Subsection 2.3 to compute the distance between the resulting graphs.

### 3.4 Textual Documents to Case Graphs

**Text Manipulation/Representation.** As a further test of the potential benefit of a graph-based representation to TCBR, we modified the TtG method above to enhance the graphs using some domain knowledge. Based on the feature list provided by the expert, we identified ten *signifier* words, which represent expressions that are meaningful in the domain and thus may indicate the similarity between documents. These signifier words were never removed from the graph regardless of the frequency with which they actually occurred in the document. Dates and other numbers (usually monetary values) were also considered important to the expert, so we included month-names in the list of feature signifiers, and modified the methods that prepare the word list not to exclude numbers. This TtCG method was a first step towards adapting the graph-based work to textual CBR. We have introduced one aspect only as a preliminary amendment. Further adjustments remain for future work.

**Similarity Assessment.** The methods for the similarity assessment for the TtCG approach were identical to the methods used with the TtG approach. It should be noted, however, that unlike the case for the TtG representation, the features were also

taken into consideration when computing the maximum common subgraph for the TtCG representation.

### 3.5 Precision and Recall

The precision was computed by dividing the number of useful (i.e. relevant) documents by the total number of documents that were retrieved. We did not, therefore, explore the ordering of the retrieved documents. The recall was computed by dividing the number of useful documents by the total number of relevant documents in the collection. The average precision and average recall were then computed by taking the averages of individual precision and recall values. It should be noted that the precision and recall for cases 9, 10 and 23 were not included when computing the averages because the domain expert stated that there were no similar documents in the collection for those specific cases.

### 3.6 Dataset

The data consisted of twenty-six claim summary documents from a law firm handling insurance cases. Insurance companies create these documents for insurance claims where there are legal questions or where the claimant has retained legal counsel. Cases are usually loosely related because law firms tend to specialize in the types of cases they handle. However, we do not know about the specifics of the dataset except for what the methods used in the study revealed. The number of words range from 942 (case 27) to 9192 (case 17) with a mean of 3886.8, a median of 3898 (interpolated), and a standard deviation of 2151.8. Cases are numbered consecutively except for 8 and 19, which were missing. Finally, we used the same dataset for each of the methodologies discussed in the previous subsection.

### 3.7 Results

With respect to our hypothesis, our preliminary finding is that the use of an algorithm to convert textual documents into graphs is potentially beneficial to textual case-based reasoning. Table 3 shows the resulting precision and recall for the different approaches in our study. When comparing the different methods in our study to the baseline, we concluded that the TtG method alone can reach levels comparable to the alternative approaches tested. Besides, the performance of the TtCG method suggests that graph-based approaches can be tailored to domain specific tasks, potentially becoming significant to TCBR.

Table 3. Average observed precision and recall

	DE	Feature Vector CBR	TtG		TtCG	
			BLG	WGU	BLG	WGU
Precision	100%	16.3%	21.1%	21.4%	21.6%	21.6%
Recall	100%	33.3%	42.0%	44.2%	42.8%	46.7%

For reference, we compared the precision and recall values in Table 3 with the average values that could be obtained by random selection of the same number of similar documents for each cell in Table 5. These average probability values for precision and recall are presented in Table 4.

Table 4. Average probabilities for precision and recall

	DE	Feature Vector CBR	TtG		TtCG	
			BLG	WGU	BLG	WGU
Precision	n/a	.082	.089	.08.9	.08.9	.08.9
Recall	n/a	.237	.136	.130	.148	.153

The generally low values are an indication of the sparseness of the original dataset. On average, the domain expert selected 2.2 claim summary documents as being similar to any given document. When the observed values in Table 3 are compared with the random probabilities in Table 4 using paired-samples *t* tests, the scores for the feature vector CBR are not statistically different at  $p < 0.05$ , but the scores for all graph methods are. This disparity is primarily because the feature vector CBR selected more similar documents than the graph methods - its baseline probabilities indicated higher recall and lower precision than the graph methods.

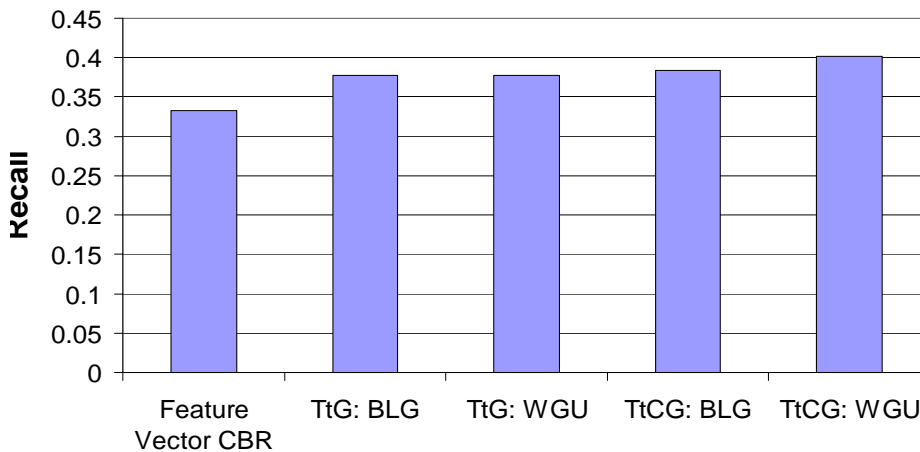


Figure 2: Average recall obtained by the methods

For the TtG method, the BLG and WGU distance methods produced very similar results; for  $N=235$  pairs, Pearson's  $r=0.967$ ,  $p<0.01$ . For the TtCG approach, the correlation between BLG and WGU distances was also very similar ( $r=0.970$ ,  $p<0.01$ ). We tested BLG and WGU measures between TtG and TtCG, and found  $r=0.979$  and  $r=0.982$ , respectively with  $p<0.01$  in both cases. Our results confirmed that the WGU distance technique is more accurate than the BLG technique when the sizes of the graphs vary widely [26].

Given the relevance of recall to the legal domain, Figure 2 compares the recall rates across the methods in our study. Although there is not a substantial difference

among the graph-based methods, they performed noticeably better than the feature vector CBR.

Table 5. Similar documents found by different methods

#	Domain Expert	Feature Vector CBR	TiG Method		TiCG Method	
			BLG	WGU	BLG	MGU
1	5,17	3,4,5,7,9	3,4,5	3,4,5	3,4,5	3,4,5
2	3,4	14,15,16,17,18,20	3,4,7,9,20	3,4,7,9,20	4,7,9,20	1,3,4,7,9,20
3	2,4	1,4,5,6,7,9	1,4,5,12	1,2,5,12	1,4,5,12	1,2,5,7,12
4	2,3	1,3,5,6,7,9,22	1,5	1	1,5	1
5	1,17	1,3,4,6,7,9,22	1,3,4,10	1,3,4,9	1,3,4,10	1,3,4,9
6	13	3,4,5,7,9,26	1,2,5,7,9	1,3,5,7,9	1,2,5,7,9,26	1,5,7,9,21,26
7	24,25,26,28	1,3,4,5,6,9,10,17,22	1,2,4,12,28	1,4,11,12,28	1,2,4,28	1,4,11,12,28
9	None	1,3,4,5,6,7,10,22	2,5,6,16	2,4,5,6	2,6,16,26	2,4,5,6,16,21,26
10	None	7,9	1,3,4,5	1,4,5,7,9	1,3,4,5	1,4,5,7,9
11	12,21,27	12	7,13,21,28	7,13,21,28	7,13,21,27,28	7,13,21,27,28
12	11,21,27	11	1,3,4,5,7	1,3,4,7	1,3,4,5,7	1,3,7,28
13	6	26	11,21	11,21	11,21	11,21,26,27
14	18	2,15,16,17,18,20,23,27	18,25	18,25	18,25	18,25
15	22	2,14,16,17,18,20,23,26,27	1,5,23,24	5,23,24	1,5,23,24	5,23,24,26
16	20	2,14,15,17,18,20,23,24,27	18,20,25	18,20,25	9,18,20,25	14,18,20,25
17	1,5	2,7,14,15,16,18,20,23,26,27	1,4,5,12	1,4,5,12,23	1,4,5,12,15	1,4,5,12,15,23
18	14	2,14,15,16,17,20,23,24,27	14,25	14,25	14,25	14,25
20	16	2,14,15,16,17,18,23,24,27	2,16,24	2,16,25	2,16,24	2,16,25
21	11,12,27	18	13,25	13,16,25,26	11,13,25,26	13,16,25,26
22	15	4,5,7,9	2,7,9,20	9,21	2,7,9,20,23	9,20,23
23	None	14,15,16,17,18,20,24,26,27	5,15,24,26	5,15,26	5,9,15,24,26	5,9,15,26
24	7,25,26,28	16,18,20,23,26,27	15,20	2,15,16	1,4,5,15,20	2,15,26
25	7,24,26,28	None	14,16,18,21	14,16,18	14,16,18	14,16,18
26	7,24,25,28	6,13,15,17,18,23,24,27,28	6,9,16,21,23	6,9,21,23	6,9,16,21,23	9,16,21,23,27
27	11,12,21	14,15,16,17,18,20,23,24,26	14,18,21	14,18,21	14,18,21	14,18,21,26
28	7,24,25,26	15,17,23,26	2,7,12	7,11,12	2,7	7,11,12

Table 5 describes the results of the similarity assessments from the study. The table lists case numbers that were considered to be similar to the target case. In the columns designated for the graph-based methods, the table includes results from both of the distance measures that were used in the study.

The metrics in our study were precision and recall, which represent measures of retrieval accuracy. Ideally, we would like to find a way to automatically and accurately represent textual documents as cases. Therefore, we should also consider the potential reduction in the knowledge engineering requirements that a graph-based approach would facilitate. We did not measure the knowledge engineering effort because of its subjectivity. However, this is another implied characteristic that substantiates the potential benefit of this approach to TCBR.

### **3.8 Discussion**

As expected, the TtCG method yielded an improvement over the original TtG method. The improvement, however, was only a slight improvement. We believe that the reason that the improvement was not more pronounced was because our preliminary adjustment to the TtG approach did not account for the relative importance of the features. Additionally, a major drawback of the signifier list was that it did not account for the range of synonyms and other semantic constructs, which an expert can interpret but a word-by-word analysis may not detect. This is an area for future research.

The results suggest other peculiarities of the legal domain. While feature vector systems are commonly used in a variety of tasks, when trying to use this representation to model the similarities between claim summary documents, we faced several difficulties. In part, the problems stemmed from the limitations of using a shell, but more significant were the number of exceptions learned in the knowledge elicitation sessions. For example, the features designated for annual income and the number of days the plaintiff was out of work become irrelevant when the plaintiff is not employed. In the legal domain, for example, this could mean finding one additional jurisprudence that may change the outcome of a legal case.

## **4 Impact of Graph-Based Method on Related Work**

In TCBR, the representation of the text source is key because it is used as the basis for computing the similarity between cases, which ultimately determines which cases are retrieved. As such, the primary focus within TCBR has been on identifying features that can be used to index the cases. Since the first TCBR workshop [27], progress has been made to add domain-specific thesaurus [28][29], assign indexing concepts to texts [29], add linguistic knowledge in order to deal with negation [5] and use latent semantic analysis to extract semantic similarity of words and phrases [30] in order to build more meaningful representations.

Building CBR systems from textual knowledge has involved very expensive and manual efforts [7], basic text retrieval using information retrieval (IR) techniques

[31], Information Extraction (IE) techniques [16], Natural Language Processing (NLP) techniques and Machine Learning techniques [32]. Although fast and easy to use, the disadvantage of using the IR approach is that feature vectors do not take into consideration the word order of the text, the structure of the text, negation, the semantic meaning of words and phrases. Instead, IR is a domain-independent approach that is based merely on statistics. IE, on the other hand, involves building templates that can be used for meaningful pattern matching [9]. However, developing the extraction rules that are used in IE is a very labor-intensive and expensive task that requires large training data or domain knowledge. Furthermore, pattern-matching techniques are only good for semi-structured texts that have a limited number of phrases [5]. NLP is a technique that parses the text based upon grammar. Unfortunately, textual documents, particularly technical documents, do not always contain grammatically correct sentences [28].

A graph-based method could overcome some of the issues with the previously discussed approaches. The use of graphs in TCBR in the legal domain is not new. For example, Branting [33] used graphs to determine case precedents. A graph-based approach, however, could contribute to the work of Gupta and Aha [3] by providing the ability to automatically identify unknown attributes (i.e. feature value pairs). Additionally, the graph-based approach could contribute to the work of Brüninghaus and Ashley [32][29][5] by eliminating words that are not a part of the factors used to build relationships between features in order to reduce the required knowledge engineering efforts. Furthermore, unlike NLP techniques, which are computationally too inefficient for processing large amounts of data [34], both the graph-based algorithm [22] and, in principle, the adapted algorithm are computationally efficient. NLP also requires a complete dictionary of terms a priori, which is not often practical. The adapted algorithm, on the other hand, can use a partial list that identifies the relationships between features in computing the similarity between cases. One of the immediate benefits of the TtG algorithm is that, unlike Weber's approach [35], the TtCG algorithm can be used with unstructured text. This represents a clear advantage over template mining techniques and a potential contribution of the TtG algorithm to TCBR.

## 5 Conclusions and Future Work

In this study, we examined one step towards developing an algorithm to convert textual documents into case graphs. This is a first step in the investigation of the potential usefulness of graphs for textual case-based reasoning. We compared the precision and recall rates that were obtained using different methods. Specifically, our proposed variant, the TtCG algorithm, yielded better precision and recall rates when compared to the results of both the TtG algorithm and the feature vector CBR.

Though not explicitly measured, the incorporation of any automated approach to the TCBR process impacts its cost because it reduces its required engineering effort. Given the expected reduction in engineering requirements in conjunction with a potential improvement in the levels of accuracy, we conclude that there is sufficient motivation for continuing to study graph-based approaches to textual CBR.

Furthermore, our proposed approach does not require source text to be structured. This point is particularly important in the legal domain because the structure of the legal documents varies from one jurisdiction to another, and even between courts within the same jurisdiction. While the preliminary results seem promising, there are, however, further additional adjustments that should be made in future work.

We have also learned from investigating the use of graphs in TCBR that graph distance techniques from Graph Theory are not suitable for assessing similarity between case graphs. This is because they are not designed to incorporate domain specific aspects that guide similarity assessment, e.g. representing varying relative importance.

We used case graphs with fixed labels in order to facilitate similarity assessment by using domain specific information. Using fixed labels has the additional benefit of reducing complexity given that distance algorithms applied to graphs with fixed labels are polynomial and not NP-complete [1].

Although our TtCG method shows some potential to represent textual documents as case graphs, our method does not address negation. Negation, however, is important in the legal domain as well as in other domains such as medicine. Moreover, in order to conform to the CBR hypotheses, it is desirable to incorporate the relative importance of indexes on the graph. Furthermore, graphs have the powerful ability to represent concepts that are described in relationships. Therefore, it would be useful to capture the domain-specific relationships that exist between features and represent them in graphs. All of these abilities would further capture the richness of a domain in the representation, which would potentially improve the recall, and represent an inexpensive means towards automatically, accurately and efficiently converting textual documents into case graphs. Therefore, we intend to incorporate negation, relationships between features, the relative importance of each feature and domain-specific rules in a future study.

## **Acknowledgements**

We would like to thank Drs. Abe Kandel and Adam Schenker for allowing us to use their algorithm in this study. We also would like to thank Dr. Bunke, Dr. Kandel, and Dr. Last for inciting the use of their methods for case-based reasoning. Thanks to Zane Reynolds for his help with his implementation and helpful comments. We are also indebted to Attorney Lisa Green for contributing her domain expertise to this study. We would also like to express our gratitude to Daniella Goral for assisting in the cleansing of the documents. Dr. Rosina Weber is supported in part by the National Institute for Systems Test and Productivity at USF under the USA Space and Naval Warfare Systems Command grant no. N00039-02-C-3244, for 2130 032 L0, 2002.

## References

- [1] Schenker, A., Last, M., Bunke, H., Kandel, A.: Clustering of Web Documents using a Graph Model. In: Antonacopoulos, A. and Hu, J. (eds.): *Web Document Analysis: Challenges and Opportunities* (2003) 1-16
- [2] Lenz, M.: *Defining Knowledge Layers for Textual Case-Based Reasoning*. In: B. Smyth, P. Cunningham (eds.): *Advances in Case-Based Reasoning, Lecture Notes in Artificial Intelligence*, Vol. 1488. Springer-Verlag, Berlin Heidelberg New York (1998) 298-309
- [3] Gupta, K. M. and Aha, D. W.: Towards Acquiring Case Indexing Taxonomies from Text. In: Barr, V. and Markov, Z. (eds.): *Proceedings of the Seventeenth Annual Conference of the International Florida Artificial Intelligence Research Society*. AAAI Press, Menlo Park, CA (2004) 172 – 177
- [4] Ashley, K. and Alevan, V.: Toward an Intelligent Tutoring System for Teaching Law Students. In: *Proceedings of International Conf. on AI and Law*. ACM Press, New York (1991) 42-52
- [5] Brüninghaus, S. and Ashley, K. D.: The Role of Information Extraction for Textual CBR. In: Aha, D.W. and Watson, I. (eds.): *Case-Based Reasoning Research and Development, Lecture Notes in Artificial Intelligence*, Vol. 2080. Springer-Verlag, Berlin Heidelberg New York (2001) 74-89
- [6] Brüninghaus, S., and Ashley, K. D.: Combining Model-Based and Case-Based Reasoning for Predicting the Outcomes of Legal Cases. Bridge, D., Ashley, K. D. (eds.): *Case-Based Reasoning Research and Development. Lecture Notes in Artificial Intelligence*, Vol. 2689. Springer-Verlag, Berlin Heidelberg New York (2003) 65-79
- [7] Ashley, K. D.: *Modeling Legal Argument: reasoning with cases and hypotheticals*. A Bradford book. The MIT Press, Cambridge, Massachusetts (1990)
- [8] Rissland, E.L., Ashley, K.D., Loui, R.P.: AI and Law: A fruitful synergy. *Artificial Intelligence* 150 (2003) 1 – 15
- [9] Weber, R.: *Intelligent Jurisprudence Research*. Doctoral dissertation, Department of Production Engineering, Federal University of Santa Catarina, Brazil. (1998) Available online: <http://www.pages.drexel.edu/~rw37/dissertation.zip>
- [10] Ashley, K. D. and Rissland, E. L.: Law, Learning and Representation. *Artificial Intelligence* 150 (2003) 17-58
- [11] Bain, W.: (1986). *Case-based reasoning: A computer model of subjective assessment*. Ph. D. dissertation, Department of Computer Science, Yale University.
- [12] Bain, W.: Judge. In: *Inside Case-Based Reasoning*. Riesbeck, C. K. and Schank, R.C. (eds.): Erlbaum, Northvale, NJ (1989)
- [13] Salton, G.: *Dynamic Information and Library Processing*. Prentice-Hall, Inc. Englewood Cliffs, New Jersey (1975)
- [14] Blair, D. and Maron, M.: Full-text information retrieval: further analysis and clarification. 1990. *Information Processing and Management* 26, 3 (1990) 437-447
- [15] Ghosh-Roy, R., Habiballah, I.O., Stonham, T.J. and Irving, M.R.: On-line legal aid: Markov chain model for efficient retrieval of legal documents. *Image and Vision Computing* 16 (1998) 941-946
- [16] Weber, R., Martins, A., and Barcia, R.: On legal texts and cases. In: Lenz, M. and Ashley, K. (eds.): *Textual Case-Based Reasoning: Papers from the AAAI-98 Workshop (Technical Report WS-98-12)*. AAAI Press, Menlo Park, CA (1998) 40-50
- [17] Ashley, K.: Progress in Text-Based Case-Based Reasoning. Invited Talk at the Third International Conference on Case-Based Reasoning. Seon, Germany. (1999)
- [18] Bunke, H.: Recent Developments in Graph Matching. In: *Proceedings of the 15th International Conference on Pattern Recognition*, Vol. 2. IEEE Computer Society Press, Los Alamitos, CA (2000) 117 – 124

- [19] Gebhardt, F., Vob, A., Grather, W. & Schmidt-Belz, B.: Reasoning with complex cases. Kluwer Academic, Boston, MA (1997)
- [20] Sanders, K., Kettler, B., Hendler, J.: The case for graph-structured representations. In: Leake, D. and Plaza, E. (eds.): Case-Based Reasoning Research and Development, Lecture Notes in Artificial Intelligence, Vol. 1266. Springer-Verlag, Berlin Heidelberg New York (1997) Available online: <http://citeseer.nj.nec.com/sanders97case.html>
- [21] Garey, M. R. and Johnson, D.S.: Computers and Intractability. W. H. Freeman and Company, New York (1979)
- [22] Schenker, A., Last, M., Bunke, H., Kandel, A.: Comparison of Distance Measures for Graph-based Clustering of Documents. In: Hancock, E. and Vento, M. (eds.): Lecture Notes in Computer Science, Vol. 2726 (2003) 202–213
- [23] Aha, D. W.: Feature weighting for lazy learning algorithms. In: H. Liu and H. Motoda (eds.): Feature Extraction, Construction and Selection: A Data Mining Perspective. Kluwer, Norwell, MA, (1998) 13-32
- [24] Porter, M.: An Algorithm for Suffix Stripping. Program 14, 3 (1980) 130–137. Available online: <http://www.tartarus.org/~martin/PorterStemmer/> (retrieved 25 Feb 2004)
- [25] Bunke, H. and Shearer, K.: A Graph Distance Metric Based on the Maximal Common Subgraph. *Pattern Recognition Letters* 19 (1998) 255–259
- [26] Wallis, W.D., Shoubridge P., Kraetz, M., Ray, D.: Graph Distances using Graph Union. *Pattern Recognition Letters* 22 (2001) 701–704.
- [27] Lenz, M. and Ashley, K. (eds.): Textual Case-Based Reasoning: Papers from the AAAI-98 Workshop (Technical Report WS-98-12). AAAI Press, Menlo Park, CA (1998)
- [28] Lenz, M.: Managing the Knowledge Contained in Technical Documents. In: Proceedings of the Second International Conference on Practical Aspects of Knowledge Management (PAKM98). October 29-30, Basel, Switzerland. (1998)
- [29] Brüninghaus, S. and Ashley, K. D.: Bootstrapping Case Base Development with Annotated Case Summaries. In: Althoff, K.D., Bergmann, R. and Branting, L.K. (eds.): *Case-Based Reasoning Research and Applications*. Lecture Notes in Computer Science, Vol. 1650. Springer-Verlag, Berlin Heidelberg New York (1999) 59-73
- [30] Foltz, P. W., Laham, D., and Landauer, T. K.: Automated Essay Scoring: Applications to Educational Technology. In: Collis, B. and Oliver, R. (eds.): Proceedings of EdMedia. (1999)
- [31] Leake, D., and Wilson, D.: Combining CBR with Interactive Knowledge Acquisition, Manipulation and Reuse. In: Althoff, K.D., Bergmann, R. and Branting, L.K. (eds.): Case-Based Reasoning Research and Applications. Lecture Notes in Computer Science, Vol. 1650. Springer-Verlag, Berlin Heidelberg New York (1999) 203-217
- [32] Brüninghaus, S. and Ashley, K. D.: How Machine Learning Can Be Beneficial for Textual Case-Based Reasoning. In: Proceedings of the AAAI-98/ICML-98 Workshop on Learning for Text Categorization (AAAI Technical Report WS-98-05) (1998) 71-74
- [33] Branting, L.K.: A reduction-graph model of precedent in legal analysis. *Artificial Intelligence* 150 (2003) 59-95
- [34] Lenz, M. and Glintschert, A.: On Texts, Cases, and Concepts. In: Proceedings of XPS-99: Knowledge-Based Systems. Lecture Notes in Computer Science, Vol. 1570. Springer-Verlag, Berlin Heidelberg New York (1999) 148-156
- [35] Weber, R.: Intelligent Jurisprudence Research: a new concept. In: Bing, J., Jones, A. J. I., and Gordon, T. F. (eds.): Proceedings of the Seventh International Conference on Artificial Intelligence and Law. ACM Press, New York (1999) 164-172