Artificial Intelligence for Engineering Design, Analysis and Manufacturing (2016), *30*, 446–465.

doi:10.1017/S0890060416000408

Discourse analysis based segregation of relevant document segments for knowledge acquisition

N. MADHUSUDANAN, AMARESH CHAKRABARTI, AND B. GURUMOORTHY
Virtual Reality Laboratory, Centre for Product Design and Manufacturing, Indian Institute of Science, Bangalore, India

(RECEIVED October 1, 2015; ACCEPTED May 31, 2016)

Abstract

Documents are a useful source of expert knowledge in organizations and can be used to foresee, in an earlier stage of a product's life cycle, potential issues and solutions that might occur in later stages of its life cycle. In this research, these stages are, respectively, design and assembly. Even if these documents are available online, it is rather difficult for users to access the knowledge contained in these documents. It is therefore desirable to automatically extract the knowledge contained in these documents and store them in a computer accessible or manipulable form. This paper describes an approach for the first step in this acquisition process: automatically identifying segments of documents that are relevant to aircraft assembly, so that they can be further processed for acquiring expert knowledge. Such identification of relevant segments is necessary for avoiding processing of unrelated information that is costly and possibly distracting for domain relevance. The approach to extracting relevant segments has two steps. The first step is the identification of sentences that form a coherent segment of text, within which the topic does not shift. The second step is to classify segments that are within the topics of interest for knowledge acquisition, that is, aircraft assembly in this instance. These steps filter out segments that are unrelated, and therefore need not be processed for subsequent knowledge acquisition. The steps are implemented by understanding the contents of documents. Using methods of discourse analysis, in particular, discourse representation theory, a list of discourse entities is obtained. The difference in discourse entities between sentences is used to distinguish between segments. The list of discourse entities in a segment is compared against a domain ontology for classification. The implementation and results of validation on sample texts for these steps are described.

Keywords: Aircraft Assembly; Discourse Analysis; Discourse Representation Theory; Text Segmentation

1. INTRODUCTION

In different stages of a product's life cycle, knowledge is generated and, where possible, recorded. Such knowledge is useful when planning products (Marx et al., 1998). This knowledge can form the basis for design (computer-aided design models and process descriptions); it can also be used to avoid decisions that led to issues in the product's life cycle (as reflected in change requests or incident reports). In the latter case, expert knowledge from downstream stages of the product life cycle could be reused to diagnose and remedy issues in earlier stages. Some applications for such knowledge are in manufacturability (Venkatachalam et al., 1993), life cycle assessment (Park & Seo, 2003), and tool making (Xi et al., 2004). Systems for representing and applying this knowledge have long since existed (e.g., Liu et al., 1995; Pokojski, 2006; Hoque & Szecsi, 2007). The challenge, however, has remained in automating the acquisition of the knowledge (Feigenbaum, 2003), especially in design (Chandrasegaran et al., 2013), with many efforts currently under way (Mozina et al., 2008; Madhusudanan & Chakrabarti, 2014).

Knowledge acquired from resolving issues or problems in the product life cycle is either stored in the human expert's memory or is captured in formal documents that report on the issues/problems and their resolution. Formal document here implies documents that have been archived after rigorous scrutiny and reviews for veracity and correctness by multiple personnel of an organization, especially domain experts. Thus, formal documents represent the agreed general opinion, and can be treated as an authoritative and legitimate source of knowledge in an organization.

It is difficult and cumbersome for the designers to access the knowledge contained in these documents even if these documents were available online. This is because the contents in these documents are not structured as knowledge but as

Reprint requests to: N. Madhusudanan, Virtual Reality Laboratory, Centre for Product Design and Manufacturing, Indian Institute of Science, Bangalore 560 012, India. E-mail: madhu@cpdm.iisc.ernet.in

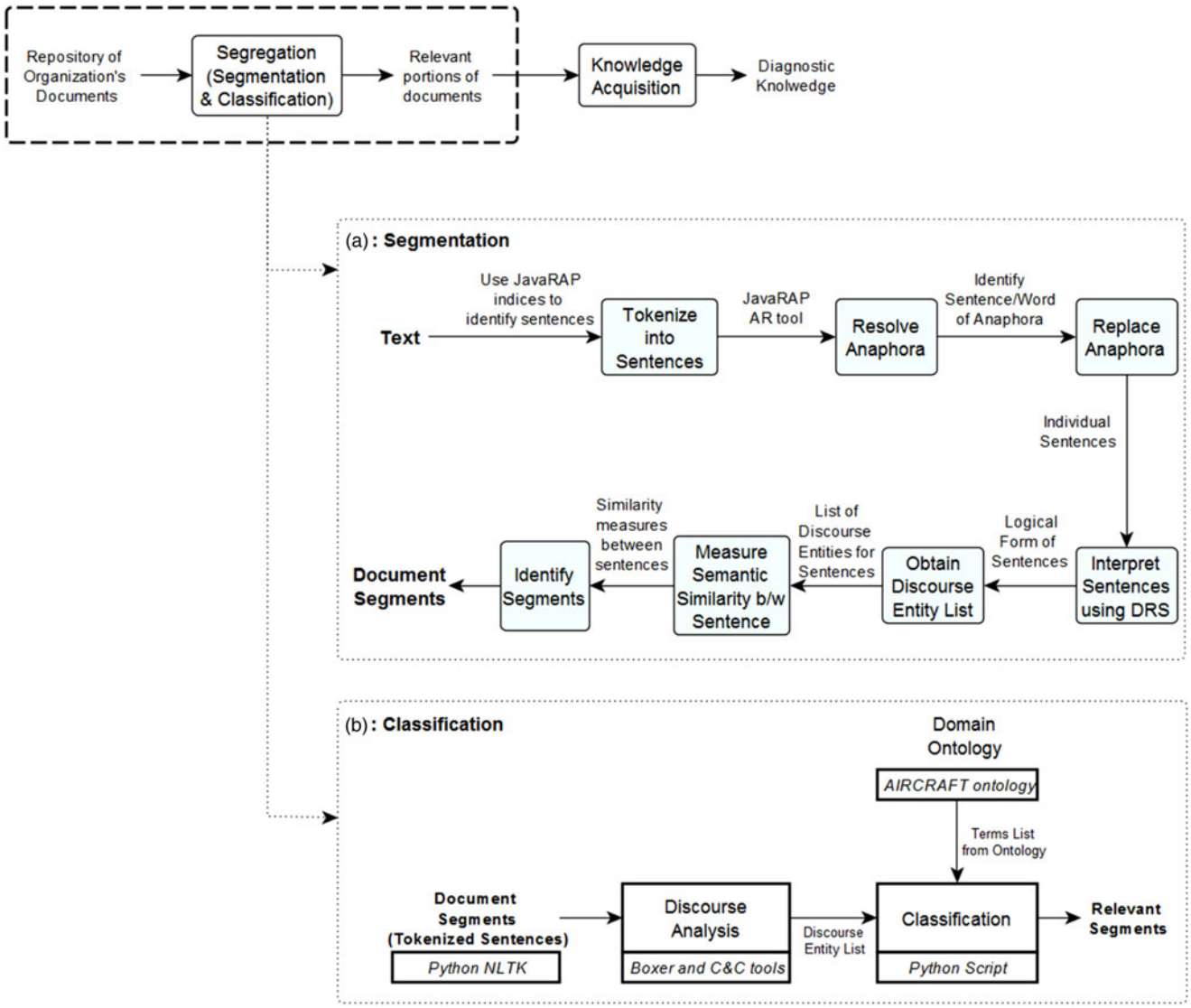

Fig. 1. Overview of the knowledge acquisition procedure from documents; (a) detailed implementation details of segmentation part and (b) the same for classification part.

case studies. The user therefore has to search, read, and interpret the information in the text to extract knowledge of relevance. In addition, it is not necessary that the relevant knowledge will be in documents from a single domain/department. Moreover, the knowledge available from documents may either be directly relevant or could complement the understanding of the domain knowledge (such as domain-specific synonyms of terms). The number of such documents is typically large especially in mature sectors such as automotive and aerospace. Although some commercial tools (e.g., IHS Goldfire, http://www.ihs.com/products/design/software-methods/goldfire/index.aspx) support effective manual search for knowledge, the large domain for search and the varied types of knowledge that need to be extracted, as explained above, make automated acquisition of knowledge from documents very desirable. Due to the difference in matching human understandable language to a machine-understandable form, it is a challenge to understand and extract knowledge from documents (Gruber, 1989).

The overall goal of our research is to develop a methodology for automated acquisition of knowledge from formal documents. Presently, the knowledge to be extracted/acquired will be applicable during planning of manual assembly of aircraft structures. The knowledge will be used to identify issues that may arise during assembly and suggest remedies. This paper proposes a process for automated knowledge acquisition (Fig. 1) with two broad steps: s*egregation of text*, which analyzes a given document to segregate its relevant portions for further analysis, and knowledge *acquisition,* which analyzes these relevant portions for acquiring knowledge in a form that can be directly applied for diagnosis. The focus of this paper is on the first step: segregation of relevant portions of a document (shown in a dotted rectangle at the top in Fig. 1).

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

1.1. Clarifying the problem

Before diagnostic knowledge can be acquired from documents, one must first identify whether a given document, or any sections of the document, belongs to the domain of interest. We call this "segregation" of relevant text. Identification of related documents/sections would help filter out texts that are not of interest for knowledge acquisition, avoiding wasteful processing of irrelevant portions during knowledge acquisition. Segregating text might imply different things to different researchers, as seen in literature. It can be interpreted as one of the natural language processing applications that are trending today. It is related to topic identification (Stein, 2004) and topic segmentation (Reynar, 1999). It can also be seen as a classification task (Nyberg, 2011; Wijewickrama, 2013) as the goal is to classify the text as either related or not related to the domain of interest.

Locating relevant knowledge documents from large collections of documents is itself an issue (Alavi & Leidner, 2001; Liu et al., 2006, 2008). The domain of *text classification* explores filtering of texts based on a user's information need (Goller et al., 2000) and is based on methods from the domain of pattern classification and machine learning (e.g., Liu et al., 2004). In this paper, the domain of interest is aircraft assembly, along with allied domains. For example, a document pertaining to issues in assembly of aft-fuselage is relevant, whereas a document about annual sales of a toy is not.

It may not always be useful to perform this classification only at a document level; sometimes only portions of a document may be related to assembly. For instance, in a document that contains feedback about workplace difficulties from an organization's employees, only the feedback from shop-floor employees would be of interest. Therefore, we concentrate only on sections of a document, rather than the entire document.

However, even before relevant chunks of a document can be identified, we have to recognize what the "chunks" themselves mean. Because for our purpose, the contents of a document need to finally understood (explained later in this section), identification of chunks cannot be only at the word level. Further, single sentences seldom make much sense. Hence, the chunks had to be collections of sentences. Such closely related sentences are referred to here as *coherent* chunks. Coherent chunks form continuous and meaningful parts of a discourse. Such *coherence*, which means that subsequent sentences of a discourse are related to one another through the topic of discussion, is sometimes also referred to as *cohesive* (e.g., in Foltz, 1998; Giora, 2003). However, a distinction is made between the two in Morris and Hirst (1991).

The chunks then have to be checked to see if the topic of discussion semantically relates to the domain of interest, namely, that of aircraft assembly. We call this second part *relevance*. This step is essentially a *classification* of the chunks into two types: related and unrelated.

The combination of coherence and relevance of a segment of text in a document indicates that the segment is of interest for knowledge acquisition in a given domain. The step of segregation (the focus of this paper) is therefore proposed to be done in two steps: segmentation (i.e., separating portions of a document based on coherence) and classification (i.e., separating portions of a document based on relevance to the domain of interest). Further processing would be then performed on the resulting segments to acquire required diagnostic knowledge. It is expected that knowledge acquisition will require an *understanding* of the documents' contents. By this, we mean that the objects and events being described, and the relations among them, should be possible to be interpreted in an automated manner. If this understanding can be captured within the segregation step itself, it is expected that it would integrate well into the next step of knowledge acquisition (Fig. 1).

To summarize, the objectives of this paper are

- to develop novel means to segment (identify coherent sections) and classify (identify relevant sections of) documents for knowledge acquisition and
- to implement and validate the method in the domain of aircraft assembly.

The rest of this paper is organized as follows. Section 2 reviews current literature in document segregation and classification to identify gaps. A method for segmentation and classification that addresses the identified gaps is then proposed in Section 3. The implementation and validation of the proposed method are then described Sections 4 and 5. The paper concludes with a discussion of the method in Section 6 and future work in Section 7.

2. REVIEW OF METHODS FROM LITERATURE

This section reviews existing methods for segregation of text for their suitability for knowledge acquisition in this work. Many methods have been proposed for segmenting data into meaningful chunks. As mentioned hitherto, machine learning-based methods are quite useful (Chen, 2010). However, such methods usually require large amounts of training data to be available *a priori*, with the data being manually labeled. Mathematical methods combined with semantics are available for text categorization as a standalone application (Chen, 2010). In addition, dedicated efforts have been made to link the referred entity to its counterpart in a knowledge base, based on the topic of relevance (Han & Sun, 2012). A large body of research focuses on activities for processing the meaning of entities, such as entity linking (using knowledge bases such as Wikipedia), disambiguation of entities and topic models (Kataria et al., 2011; Zhang et al., 2011; Han & Sun, 2012). Hence, it may be possible to model topics of discussion and identify changes of topics.

The collection of words in a document can be used to determine the topic of discussion in the document, this being termed in literature as a "bag of words" approach (Li et al., 2008). In a similar manner, one method uses word sequences as a means of classification (Li et al., 2008). Document-clus-

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

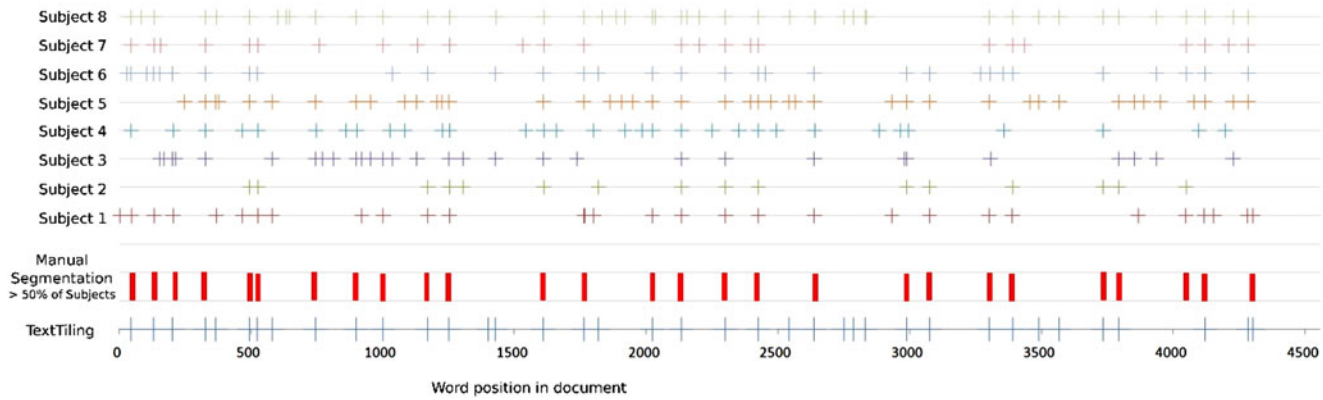

Fig. 2. Segment breaks indicated by TextTiling and human subjects.

tering is a popular application of techniques that can work without training data (Andrews & Fox, 2007), as opposed to classification methods that require trained data. There is existing literature about the use of phrases and their semantic relationships, as well as the use of ontology for clustering (Zheng, 2009). Clustering documents using a graph-based technique by detecting frequent subgraphs of related terms is another approach (Hossain & Angryk, 2007). Yet another approach uses sampling to discriminate segments of documents (Chen, 2010). In this approach, a probabilistic method called the generalized mallows model is used to model the topics of a text, which are used for segmentation. Methods in the area of automated coding (e.g., Mu et al., 2012) have also managed to achieve high accuracies in segmenting and identifying classes in conversations. However, in the methods of Mu et al. and others, topics are well specified and still depend on some human-labeled data and well-identified features, which are not available in the present problem. In the domain of product life cycle management systems, a method is used to model and elicit information about key relationships and stakeholders by analyzing E-mails (Loftus et al., 2009).

Another relevant approach is based on multiparagraph segmentation using a TextTiling algorithm (Hearst, 1994), which divides a given text into predetermined blocks of equal size, and then analyzes the semantic relatedness of words among these blocks to chunk them. Beeferman et al. (1999) also propose statistical models that combine topicality and cue-word features.

Passonneau and Litman (1997) describe a combination of pauses, cue phrases, and referential noun phrases to segment transcripts. Lexical and syntactic methods (e.g., Tofiloski et al., 2009) have been used to identify segments even within sentences. Thanh et al. (2004) describe segmentation into elementary discourse units using lexical cohesion, discourse cues, and syntactic information, as well as prescribe their own integrated approach. However, our work requires chunks of related sentences, sentences being the unit of processing. Within the domain of discourse analysis, segmented discourse representation theory has been used to model the discourse semantics and provide a discourse structure (Lascarides & Asher, 2008).

To summarize, a limitation of many of the above methods is either their dependence on training data or their unsuitability for the current purpose. Such large training corpus data related to aircraft assembly, unlike in other domains like natural sciences, are not yet available. The use of training data also requires large numbers of staff hours for labeling them. Another issue with most methods is that there is no intent on *understanding* the document content in these methods. We argue that such understanding is central to knowledge acquisition, as explained in Section 3.1.2.

2.1. Preliminary studies

Exploratory studies for identifying means for segregating text were first conducted. One way to classify documents or their parts is to list the words and their frequency in the text being considered. In a preliminary exercise, this approach was tested on some documents; the results of the classification were not always indicative of the content at the sentence level. As mentioned before, TextTiling is a well-known method for segmenting sections from a given text. It is freely available as an implementation in NTLK, a Python based Natural Language Tool Kit (Loper & Bird, 2002). This method was tested on a portion of a test document with 4303 words (http://www.oup.com/us/static/companion.websites/9780195157826/chapter19.pdf). The outcome was compared with outcomes from the document being given to human subjects (see Fig. 2). The horizontal axis represents the position of each sentence in the document. The red markings at the bottom indicate areas in the document where 50% or more of subjects saw a coherent segment, that is, without considerable change of topic. The bottom row shows segmentation provided by the TextTiling implementation. While using the algorithm, two parameters (block length and the block size) had to be adjusted to obtain a reasonable number of segments. The combination with maximum number of segments (39) was considered.

Some observations on the results are as follows:

- The two segmentations (using Tiling and using manual subjects) match in most of the locations. However, Text-

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

Tiling looks at paragraph breaks as a shift in focus, for example, for an itemization in the document, which was not perceived as a shift by all but one of the subjects. This, in contrast, was treated as four segments by TextTiling.
- When there were multiple segments in a paragraph, except at one point (that arose due to formatting issues in the input), tiling performed as expected. In addition, tiling's segments matched with three subjects on four instances, with two subjects on three instances, with one subject on four instances, and with no subjects on one instance. Hence, the performance for tiling was taken to be satisfactory in this case.

The following conclusions were derived from this study:

- TextTiling performed segmentation at the most prominent segment boundaries. However, it also performed segmentation at other boundaries not identified by the test subjects.
- It was difficult to adjust the block-length and block-size parameters for every document. They directly influence the number of segments that are produced as output.
- Segmentation is only the first step in filtering relevant portions from text. It was important to understand the content of a document and extract diagnostic knowledge from it. Even the next step, classification, cannot be handled using the output from tiling alone. Hence, extracting the semantic content was necessary for subsequent steps in the research.

An additional case for using understanding-based techniques was made by the fact that methods that analyze words and their meanings do not address the task of resolving pronouns and other anaphora. *Anaphora* are words that refer to other words that have been used previously in a text. For example, pronouns are used to avoid repeating words. Resolving anaphora is important because they implicitly contain references to other words (in the same sentence or in other sentences) and may not be captured and counted by such methods.

At this point, it is useful to consider a document as a discourse from the author to the reader. The discourse context is useful for identifying the things being talked about in a sentence. In a given discourse, the current context is defined by the entities that are being talked about, the activities that concern them, and the relations among these entities. The list of entities is called a discourse entity list (Allen, 2011). Based on this idea, we now explain our proposed method for segregation (i.e., segmentation and classification).

3. PROPOSED METHOD

3.1. Discourse analysis

3.1.1. Discourse

A discourse is a natural form of communication, and is useful to realize the semantic content of a natural language exchange. Literature on discourse analysis proposed various theories and approaches for understanding discourses (e.g., Grosz & Sidner, 1986). One theory is that a discourse has a hierarchical structure (Allen, 2011). Some of the ways in which a discourse can proceed are using interruptions, digressions, and itemizations. Boundaries between discourse segments can be marked using cue phrases (also called discourse markers; Fraser, 1999). Phrases like "*after that*" and "*on the other hand*" signal transitions between segments. As discourse analysis helps monitor the influence of previous sentences on a subsequent one (Allen, 2011), it may be practical to consider documents as discourses. In such a discourse, the authors intend to communicate with the reader of the document. However, the documents being considered in this research are technical in nature, and hence formal, and do not contain the same language as used in conversations. Hence, discourse markers (cue phrases) may not always be used in such documents. Referring to Section 2, we do not intend to study the detailed discourse. Rather, it is interesting to use methods of discourse representation to understand texts.

3.1.2. Discourse representation structure (DRS)

The theory used to understand discourses in this work is called discourse representation theory (Kamp & Reyle, 1993). This theory models discourses as a combination of entities and conditions. This information is represented in a structure called DRS (Blackburn & Bos, 2006). An example (drawn using Python natural language toolkit) is shown in Figure 3. A DRS has two components: entities and conditions. Discourse referents are those referring to entities in a DRS. They may contain pronouns, which are in turn resolved using an identity assignment. Discourse conditions are first-order logic conditions showing the relations between discourse entities. The conditions are predicates that convey the meaning of these sentences. They can also contain statements that convey the resolution of pronouns. These conditions may also contain other DRSs. Figure 3 shows a DRS for an example sentence. In the figure, *x1, x2*, and *x4* are top-level entities, where *x1* is a discourse referent to the entity *riveting*, and *n_riveting(x1)* is a condition that says *x1* refers to *riveting*. *x4* is a process, which is explained using the condition prop(*x2*,[*DRS*]), where [*DRS*] is the nesting of a DRS within another that explains the plates and pin using other conditions.

3.2. Assumptions

The following assumptions were made in our research:

1. A document is treated as a one-way discourse between the author and the reader.
2. The knowledge represented in documents are correct and valid.
3. Available semantic resources such as dictionaries and lexica are sufficient to cover the language used in technical documents.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

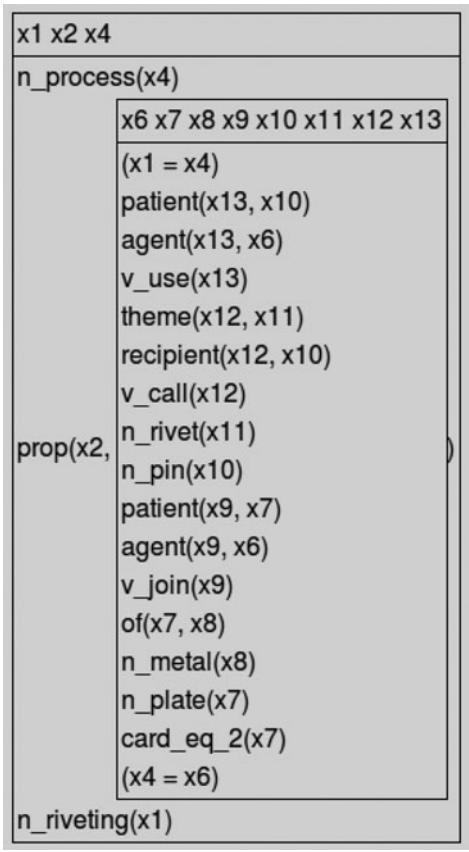

Fig. 3. An example of a discourse representation structure for the sentence "*Riveting is the process of joining two plates of metal using a pin called a rivet*."

3.3. Proposed method for segregation

Having established the need to use discourse analysis for our segregation exercise, we present below the steps of our proposed method:

1. Segmentation (shown in Fig. 1a):
   a. Tokenize the input text into sentences.
   b. In every sentence, resolve anaphora. They may be within a sentence or across sentences.
   c. Obtain a list of discourse entities, including those referred to by anaphora, to obtain a discourse entity list for each sentence.
   d. Segment the sentences that are both physically close to one another and share parts of their discourse entity lists within a threshold.
2. Classification (shown in Fig. 1b):
   a. Evaluate the entities in the discourse entity list of each segment to determine how many of them relate to the relevant domain. The basis for comparison are one or more ontologies.
   b. If the discourse entity list for a segment matches with the domain ontology by more than a certain threshold, then classify that segment as being related to aircraft assembly.

The next section describes the implementation and validation of the two steps in the segregation procedure.

4. IMPLEMENTATION AND VALIDATION FOR SEGMENTATION AND CLASSIFICATION

As described in Section 3.3, we separately consider segmentation and classification parts. In this section, individual implementation and results of validation are presented for each of these parts.

4.1. Implementation and validation of segmentation

4.1.1. Implementation

This section describes the details of implementation and validation of the segmentation part. The implementation of the segmentation part of the proposed method required the use and integration of various natural language understanding and processing tools. An overview of the specific procedure developed for implementation is described in Figure 1a.

Assigning sentence indices. The first step is to tokenize the text into individual sentences. One possibility is to use punctuation markers, such as a period. However, in the next step, we use an anaphora resolution tool called JavaRAP (Qiu, 2004, and JavaRAP website). JavaRAP uses sentence indices to refer to anaphora and their antecedents. Hence, it is meaningful to use sentence and word indices used by JavaRAP. The JavaRAP Sentence Splitter utility is used, and it assigns indices starting with zero. For example, in Figure 4, (1, 6) refers to the *seventh* word in the *second* sentence.

Anaphora resolution. After assigning indices to sentences, the next step is to resolve the anaphora in the input text, for which the JavaRAP tool is used. Although Boxer and C&C Tools is capable of anaphora resolution, we chose to perform anaphora resolution separately due to the weak ability of these tools in this respect (Bos, 2008). The text is supplied to JavaRAP, which outputs resolved anaphora as shown in Figure 4. The indices identified earlier are used to identify and connect anaphora and their referents.

Replacing anaphora. After resolving anaphora, we needed to replace anaphora with their discourse referents, so that the actual discourse entity is used while interpreting sentences. Otherwise, the correct entity would have to be substituted in the interpreted form. An example is shown below:

Original Sentence: "*Riveting is a complicated process. It involves many parts and tools*."
After replacing "it," the second sentence becomes "*Riveting involves many parts and tools*."

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

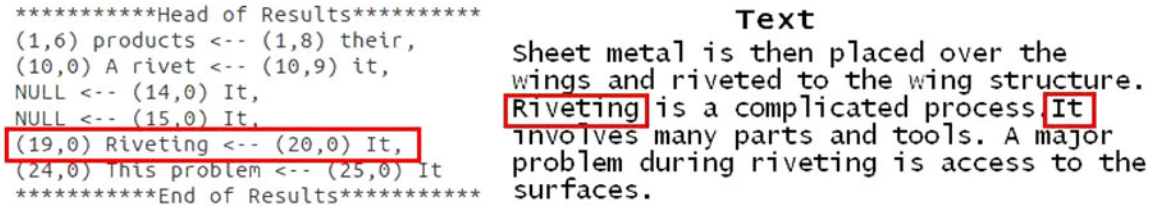

Fig. 4. Anaphora resolution output.

As one would expect, the challenge in replacing anaphora with their referents is that the correct form of the referent has to be replaced. Consider the sentence: "*Manufacturing is the process of realizing products from their design.*" After replacing anaphora, this becomes "*Manufacturing is the process of realizing products from products design*."

The modified sentence is almost fine, except for the missing apostrophe in the possessive noun. The correct form has to be captured using grammar rules. The bigger challenge is when the referents of an anaphora are phrases. Such complex forms are not yet handled in our implementation. This requires expansion of the procedure to interpret anaphora results.

Interpretation of sentences. The next step was to obtain a semantic interpretation of sentences to get their meaning. The input are individual sentences and the output are DRSs. The tool used for the interpretation task was the C&C and Boxer tool set (Curran et al., 2007). These tools have interfaces built into the Natural Language Tool-Kit (NLTK); the implementation was written in Python. An example interpretation is shown in Figure 5. It is in a different form than the "boxed one shown in Figure 6. It is in a more computer-understandable form, a recursive list. Similar to Figure 3, there are a list of discourse referents here, for example, *x1*, *x4*, and discourse conditions, such as *n_riveting(x1)* and *n_process(x4)*. The other conditions represents specific predicates in first-order logic; for example, *v_join* represents the action of joining, the predicate *of* connects two referents, and *prop* is a proposition composing another DRS.

Obtaining discourse entity list. It was mentioned that semantic interpretation gives us the list of discourse entities. From the DRS interpretation of sentences, it is possible to directly obtain the list of discourse referents for that sentence. In Figure 5, the discourse referents are the variables *x1*, *x2*, and *x4*. One can get the predicate for discourse entities by reading

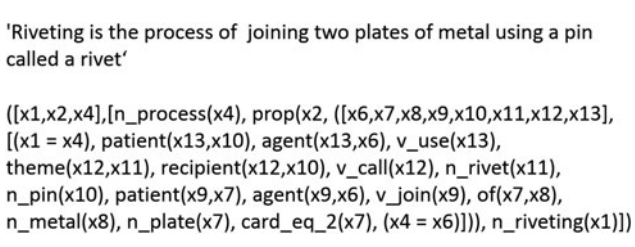

Fig. 5. An example of discourse representation structure interpretation of a sentence."

the discourse conditions for an entity, for example, n_xxxxx(), ne_nam_xxxx(), and so on. This is different from using only the part-of-speech (POS) tags, because the conditions convey more information, for example, relation of an object to other objects and/or events. An example of the list of discourse entities for a sentence is shown in Figure 6.

Measure semantic similarity between sentences. The next step of implementation was to measure the similarity between every two consecutive sentences. This measure was used to identify segments based on large jumps in meaning. We measured how similar or different two consecutive sentences are to or from each other. For this, we propose a measure based on the semantic similarity between words. Examples of similarity measures between individual words in WordNet (Miller, 1995) are Jiang-Conrath similarity, Lin Similarity, and path similarity (Mihalcea et al., 2006). Starting with such word similarity measures, we arrived at a similarity measure between two sets of words, the sets being discourse entity lists of adjacent sentences.

For example, consider the two lists

["*quantity*," "*part*," "*riveting*," "*tools*"] and
["*surfaces*," "*rivet*," "*problem*," "*access*"]

The similarity for all the words from the first list to the words in the second list can be averaged to get a single measure. Even then, at least three choices were available for indicating similarity of one word from the first list to the words of the second list. These were

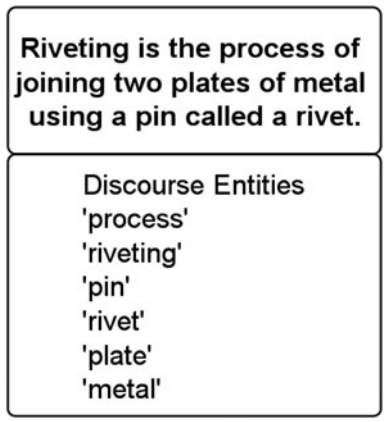

Fig. 6. An example of discourse entities in the sentence.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

1. *maximum* among similarities of a word from the first list to all words of the second list,
2. *minimum* among similarities of a word from the first list to all words of the second list, and
3. *average* of similarities of a word from the first list to all words of the second list.

Depending on which option is chosen from the above, the final averaging for the overall score between the two lists differs. In the first two cases, they would have to be averaged over the number of elements of the first set. In the third case, the average must be taken over the elements of both sets.

Table 1 shows how the above calculations for each word pair were carried out in the implementation. For each word pair such as "*quantity*"–"*surfaces*," we used the Lin similarity measure of these words.

For the above lists of words, the overall similarities between the lists are

- average max: 0.4401
- average min: 0.0
- average of averages: 0.0381

Identification of segments from values of similarity. Once the similarity between sentences was determined, segment boundaries were identified by tracking large changes in the similarity values. Literature does not provide any definitive strategy to identify such changes. The closest that could be used is the one by Hearst (1994), who used a method of segmentation based on change of slope and a cutoff value as threshold.

In this research, we propose a strategy for finding such segments based on a comparative study of a manual reading exercise versus our calculated values. This is discussed in the following section.

4.1.2. *Validation of segmentation part*

A comparative study was carried out to validate the segmentation implementation. The purpose was to check if the implementation would identify changes in topic as seen by human subjects.

A sample document was constructed for this purpose. It reflected the needs of the validation; it had sections with varying topics where some variations in topic were strong, while others were not so strong. The variation in topic was largely linear in nature (meaning the structure of discourse was not hierarchical). The document was mostly about assembly processes and riveting. Two completely unrelated sections (about "*running*" and "*employee salaries*") were deliberately inserted in the document. There were 31 sentences in total. We did not compare this result with TextTiling because it was not clear what values of the parameters used in TextTil-

Table 1. *Different similarity measures between words from two lists*

Word	*"surfaces"*	*"rivet"*	*"problem"*	*"access"*	Min	Max	Average
"quantity"	0.0000	0.0000	0.2754	0.0589	0.0000	0.2754	0.0836
"part"	0.0000	0.0000	0.1052	0.0802	0.0000	0.1052	0.0463
"riveting"	0.2297	1.0000	0.0000	0.0000	0.0000	1.0000	0.3074
"tools"	0.3799	0.3102	0.0000	0.0000	0.0000	0.3799	0.1725
Average					0.0000	0.4401	0.0381

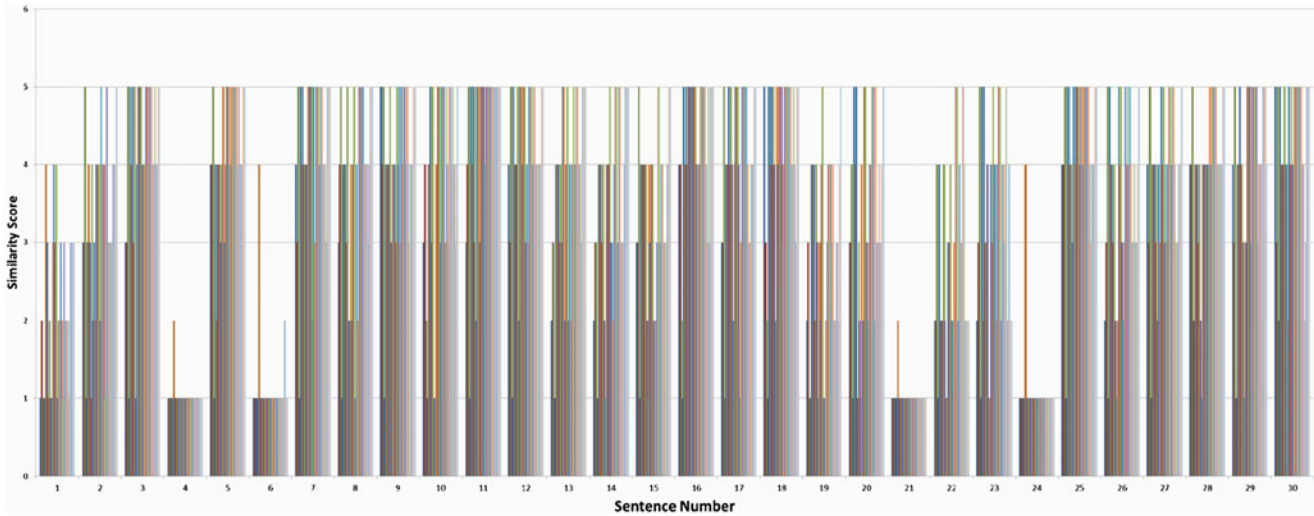

Fig. 7. Feedback obtained from 31 subjects for a total of 30 adjacent sentence-pair similarities.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

ing would serve as a standard for comparison. Hence, the comparative study with human subjects was carried out.

A total of 31 subjects were asked to give a score of how similar adjacent sentences were on a scale of 1 to 5 (1 = *most dissimilar* and 5 = *most similar*). Their feedback is shown in Figure 7. Using subjects' scores, we marked segments that were considered large changes in the topic of discussion. Here, large changes were due to (a) a drastic decrease in the similarity scores and (b) low or very low score of similarity.

Two gradations of the above scores were considered: a major change of a decrease of 3 or 4 for (a) and a value of 1 for (b) and a minor change of a decrease of 2 for (a) and a value of 2 for (b).

Table 2 shows a summary of how many subjects matched the above criteria. For a major cutoff, we considered 15 subjects for a value to be considered significant. The prominent segments are at Sentences 5, 7, 22, and 25. A minor segment can also be seen at 20, if we relax the cutoff to 10 subjects.

The implementation was tested on this text to calculate similarity scores between adjacent sentences. It was then used to calculate a single measure to indicate if the meaning (and the context that decides the topic) changes as the program reads through the text. To develop a score of similarity that would reflect a change in context, we tried various options including average-, min-, and max-based values of intersentence similarity.

Comparing the plots with the subjects' feedback, none of these individual measures corresponded well for most locations. We realized that this was because each of these measures behaved differently. To explain, consider the analogy of the two word lists as two clusters of points. The average of two words can be considered to represent the center of the clusters, and the min and max, respectively, would represent the two closest and farthest points.

It was decided to combine all three measures into a single measure, in the hope of using their individual characteristics. The proposed measure was calculated as follows:

- Calculate the change in average-, minimum-, and maximum-based scores of every sentence pair. Hence, we have $(n - 2)$ such difference values for n sentences and $(n - 1)$ sentence pairs.
- Normalize each of the difference values by dividing by the highest respective (absolute) value. The normalization is done so as to give equal weights to each of the scores.
- Sum up the normalized difference values.

The measure for a given sentence pair i was thus calculated as

$$\mathrm{SM}_i = \frac{\mathrm{Avg}_i - \mathrm{Avg}_{i-1}}{\max_i(\mathrm{Avg}_i - \mathrm{Avg}_{i-1})} + \frac{\mathrm{Max}_i - \mathrm{Max}_{i-1}}{\max_i(\mathrm{Max}_i - \mathrm{Max}_{i-1})} + \frac{\mathrm{Min}_i - \mathrm{Min}_{i-1}}{\max_i(\mathrm{Min}_i - \mathrm{Min}_{i-1})}.$$

The absolute values of intersentence similarity and the values of SM_i are plotted as shown in Figures 8 and 9, respectively. In Figure 9, if we consider a cutoff of –0.5 for the values of SM_i, we get five points as shown in the plot. Two are the points indicated by a majority of subjects (Points 3, 5 corresponding to Sentences 5, 7). Further, at a cutoff of –0.25, we can observe another major point marked by subjects (Point 20–Sentence 22). A minor point indicated by subjects (Point 18–Sentence 20) also has a negative value. However, we still needed to explain the other minimum points in the plot, which have not been correspondingly rated by subjects.

- Point 10 has a minimum value because the first of the two sentences and is a section heading having one word only. This single word ("*Riveting*") is responsible for small values of average and min values, while the max value is high.
- The other low values are Points 12, 14, 16, 17, 18, and 29.
  - Point 12 was also marked by eight subjects, and also the word "*sealed*" has not been considered by the program, it being a verb.
  - For Point 14, "*riveted*" was a verb and not considered for a comparison.

Table 2. *Numbers of subjects for various values and differences of intersentence similarity*

No.	Low Value (Dissimilar)	Very Low Value (Very Dissimilar)	Large Decrease	Very Large Decrease
1	9	10	—	—
2	3	3	0	0
3	0	2	0	0
4	1	30	4	25
5	1	1	0	0
6	1	29	3	26
7	1	1	0	0
8	4	2	2	1
9	0	1	0	0
10	1	4	2	3
11	1	1	0	0
12	1	1	2	0
13	4	1	3	0
14	3	2	0	1
15	4	1	0	0
16	1	1	0	1
17	1	2	0	2
18	2	1	0	1
19	7	5	10	2
20	4	3	0	0
21	1	30	10	14
22	14	3	0	0
23	7	3	0	0
24	0	30	5	16
25	0	1	0	1
26	4	2	3	2
27	1	1	0	0
28	2	1	0	1
29	1	2	2	0
30	3	0	1	0

Note: The shaded areas indicate where a majority of subjects indicated a drastic decrease or a low value of similarity.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

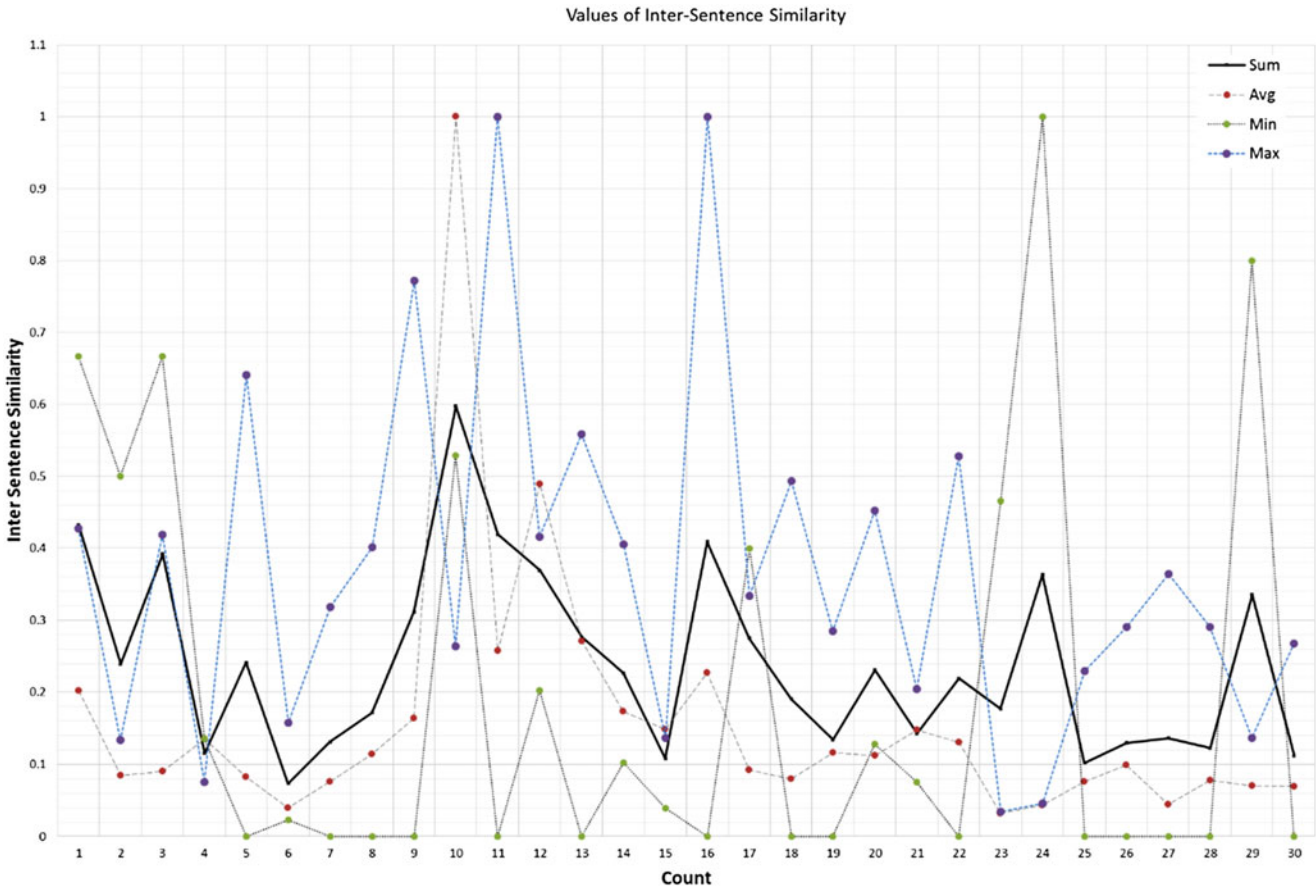

Fig. 8. Values of intersentence similarity between adjacent sentences.

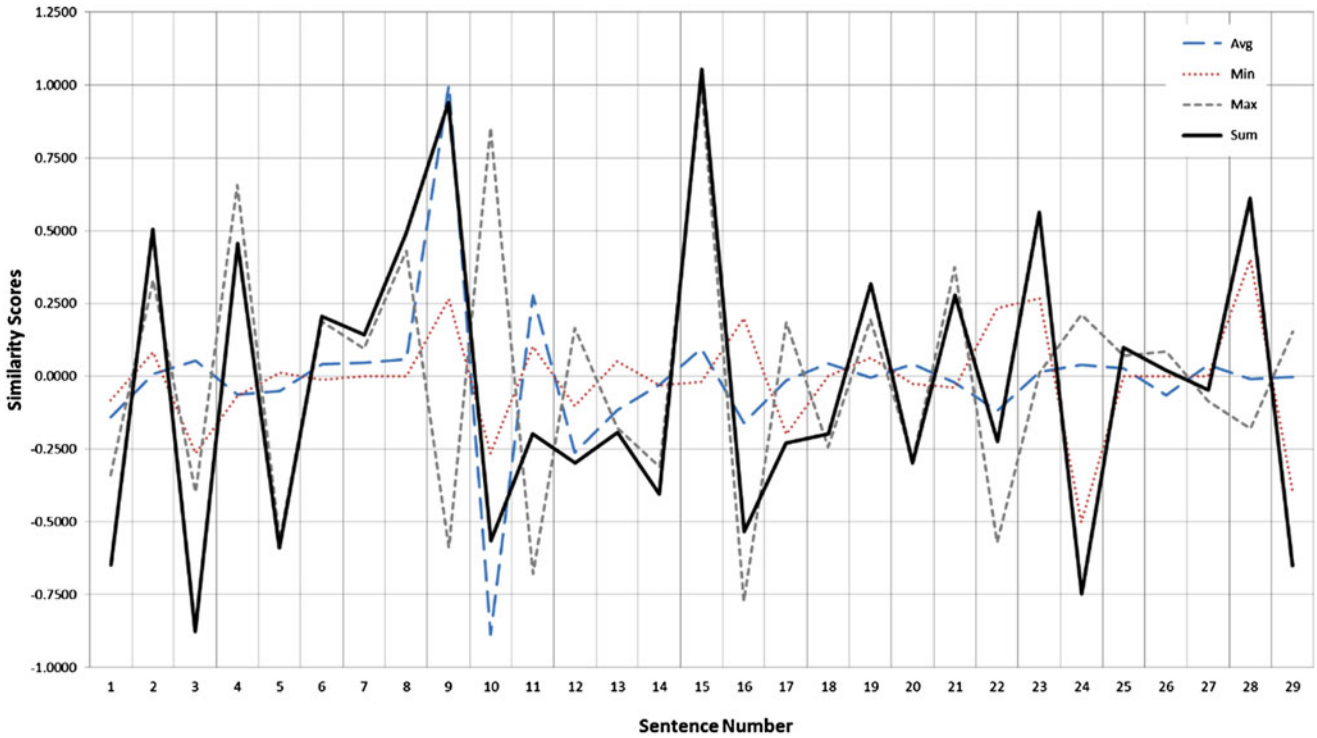

Fig. 9. Values of SM_i for between adjacent sentences.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

- Point 16 also has some issues, such as not considering "*rivet*," but considering words like "*such*" and "*thing*" (with no WordNet Entry being present for "*such*"). Hence, correct similarity values have not been calculated at this place between the sentences "*It is possible to rivet plates of large thickness, such as in bridges*" and "*It is also possible to rivet sheet-metal, as is the case in aircraft construction.*"
- Point 17 is between Sentences 18 and 19. The word "*rivet*" is in a verb form in Sentence 18, thus reducing the value of the Similarity measure with Sentence 19.
- Point 18 has been indicated as a minor point (by 12 subjects for individual scores and 12 subjects for difference).
- Similarly Point 24 has low scores. There were two reasons. First, anaphora resolution for "*them*" failed. Second, no suitable entries for the word "*salary*" were found. Together it has resulted in the entity "*salary*" being counted out twice for similarity measurement. This is also possibly the reason that Point 23 has a high value, because we assign a default of 1.0 to Min to start with.
- Point 29 corresponds to sentence pairs 29–30 and 30–31. The dip in value is interesting because, both 30 and 31 are related to 29 only (it is similar to an itemization). Hence, the value between Sentence 30 and Sentence 31 is not high, although subjects have marked it so.

From the results, it was observed that a large decrease in slope indicates the presence of a change of topic. Titles of sections, when included, created an anomaly. In addition, a combination of the difference of average, min, and max values was necessary to distinguish segments.

Some issues and possible improvements in the implementation were the following:

- Some words did not have an equivalent synset (synonym entry) in the WordNet lexicon. *Riveting* can be either a noun or a verb. We have currently chosen the closest noun of another form of the word using the *morphy* utility in NLTK WordNet.
- Verbs can also be counted for semantic similarity; for example, *manufacturing* is a verb only. Hence, we used its closest WordNet entry. However, an attempt to do so did not raise the similarity score very well, because there were some verbs that do not contribute to the score. For example, phrasal verbs like "*have*" were distractive, lowering the score where one would normally ignore it.
- A linear change in context was assumed, and this was not always the case.

4.2. Implementation and validation of classification

This section describes the implementation and validation of the classification part.

4.2.1. Implementation

The classification part of our proposed method was initially implemented on a sentence-by-sentence basis. The overall procedure for implementation (see Section 3.3 for the method) is shown in Figure 1b.

Tools used. For implementing the proposed method for classification, a combination of tools was used. These are

- *Boxer and C&C Tools*, to provide a representation in the form of DRSs.
- *Python Natural Language ToolKit*, to perform routine natural language processing tasks (e.g., tokenizing text)
- *Ontology related tool* (Protégé), to manually read an existing ontology file (in OWL format).
- *LaTeX*, to print classified sentences with different colors (see Fig. 10).

The Python script tokenized a given text into sentences. These sentences were then interpreted as DRSs, and for each the entities (indicated by Boxer) were extracted and listed as discourse entities. A list of aircraft-related terms was already obtained from an AIRCRAFT ontology (Ast et al., 2014) and was used as a reference. A screen grab of this ontology's hierarchy (as seen in Protégé) is shown in Figure 11. At this stage, this list is derived only from the class hierarchy of the ontology. It is always possible to expand this list from the object properties and the class descriptions.

4.2.2. Validation

After the implementation of the classification method, the next step was to validate the method on test data. It was not clear whether a gold standard exists for a task comprising discourse segmentation and classification. Thus, a document that was available in the public domain was used for testing; it was part of the Wikipedia article for Riveting (http://www.en.wikipedia.org/wiki/Rivet). The document was manually classified by the researchers and by test subjects who were master's and doctoral students and members of project staff at the university. The document consisted of a mix of sentences that did and did not relate to the aircraft domain. It had 177 sentences, from various domains including aircraft

where reliability and safety count : Semantic score = 0 [16]
A typical application for solid rivets can be found within the structural parts of aircraft : Semantic score = 1 [17]
Hundreds of thousands of solid rivets are used to assemble the frame of a modern aircraft : Semantic score = 1 [18]
Such rivets come with rounded (universal) or 100 countersunk heads : Semantic score = 0 [19]
Typical materials for aircraft rivets are aluminium alloys (2017, 2024, 2117, 7050, 5056, 55000, V-65), titanium, and nickel-based alloys (e.g., Monel) : Semantic score = 1 [20]
Some aluminum alloy rivets are too hard to buck and must be softened by annealing prior to being bucked : Semantic score = 0 [21]
"Ice box" aluminum alloy rivets harden with age, and must likewise

Fig. 10. Example of sentences classified using the implementation.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

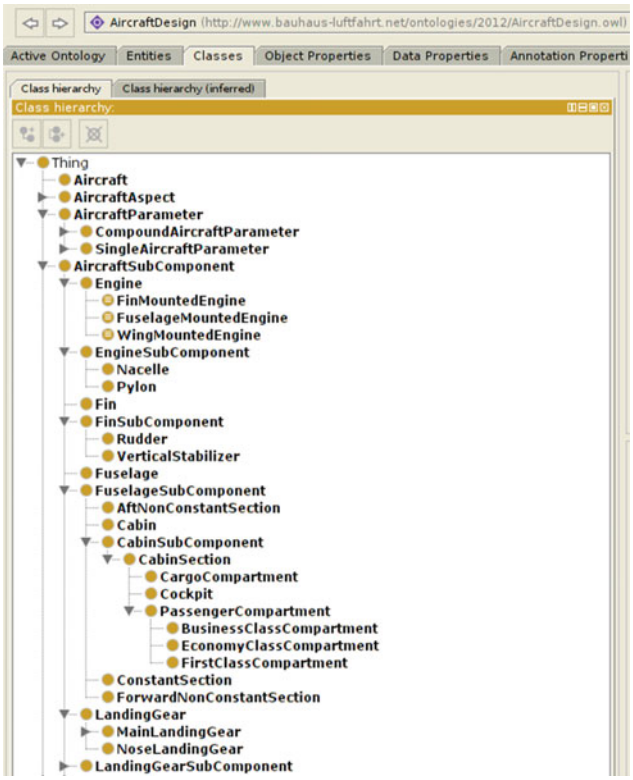

Fig. 11. A screen grab of the AIRCRAFT ontology as seen in Protégé.

domain. However, it did not contain any terms from the ontology other than the terms "aircraft" and "wing."

Figure 12 shows the responses of 15 subjects as to which sentences they thought were related to the aircraft domain, versus the classification made by the implemented method. The figure shows the relevant sentences along the abscissa, and each subject's identifier along the ordinate axis. Shown at the top is the classification performed using the implementation. There were clear clusters of sentences that were common among these classifications.

Observations. A few interesting observations were made during the preliminary validation. The background domain knowledge of the subject seemed to influence how each subject understood domain-related terms. Domain knowledge might have helped subjects with their inferences for terms such as "*aerodynamic drag*" for classification (and this was not present in the AIRCRAFT ontology). Although different subjects treated the text differently (e.g., some of them read some sentences together with other sentences), there were a set of sentences that were classified by almost all of the subjects.

Another observation was that proximity of sentences seemed to be a factor for some subjects to decide the relevance of sentence to the aircraft domain. For example, if two sentences that are separated by a sentence in between had been classified as related to aircraft domain by most subjects, some subjects marked the in-between sentence as also related to that domain. This indicates that they implicitly inferred the sentence to be related to these two sentences even though the sentence did not contain the terms they were looking for.

5. INTEGRATION OF SEGMENTATION AND CLASSIFICATION

This section describes the integration of segmentation and classification steps as detailed in Section 3.3. Although the implementation was not combined in a single program, they were used sequentially and validated together. Referring back to objectives in Section 1.1, the goal in this validation step was to check if relevant sections of a document were identified automatically by the implementation. This means that the identification of sections (i.e., segmentation), as well as judging if these are relevant (i.e., classification) must both be performed. To verify the results of the validation, we performed these activities, once again, with human subjects and compare.

5.1. Test document

In order to test the integrated method, a test document was needed that represented the objectives of the work, as well as the capabilities of our method. Certain considerations were made while preparing the document. For example, the variation in topic had to be linear. Too many anaphora in the text might affect the accuracy of the method, because anaphora resolution capabilities were currently limited, as seen in Section 4.1.2. The document also needed to have domain-related and unrelated sections. In addition, as the independent validation of the segmentation revealed in Section 4.1.2, the usage of related words within a segment is a factor in reading a segment together. The document must also explicitly convey itself, meaning that its interpretation should not depend on background knowledge of the reader. A test document was prepared using parts of real-life documents (e.g., about aircraft construction). It had 42 sentences. A sample of the document used, along with the subject's response, is shown in Figure 13.

5.2. Validation

For validation, we conducted a reading exercise with subjects in two steps. The subjects were master's students, doctoral students, and members of project staff at the university. They held at least a bachelor's degree in engineering. The requirement for each subject was that he/she must have had at least a minimum exposure to engineering terminology, and a reasonable grasp on the English language. Subjects' responses for segmentation and classification are presented in Table 3.

In the first step, we asked 30 subjects to read the document (individually). They were clearly instructed to mark segments that conveyed a shift in topic of discussion, and to assign a score of 1–3 for each shift (1 = *not so strong*, 3 = *very strong*). They were also asked to explain the reason for that marking.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

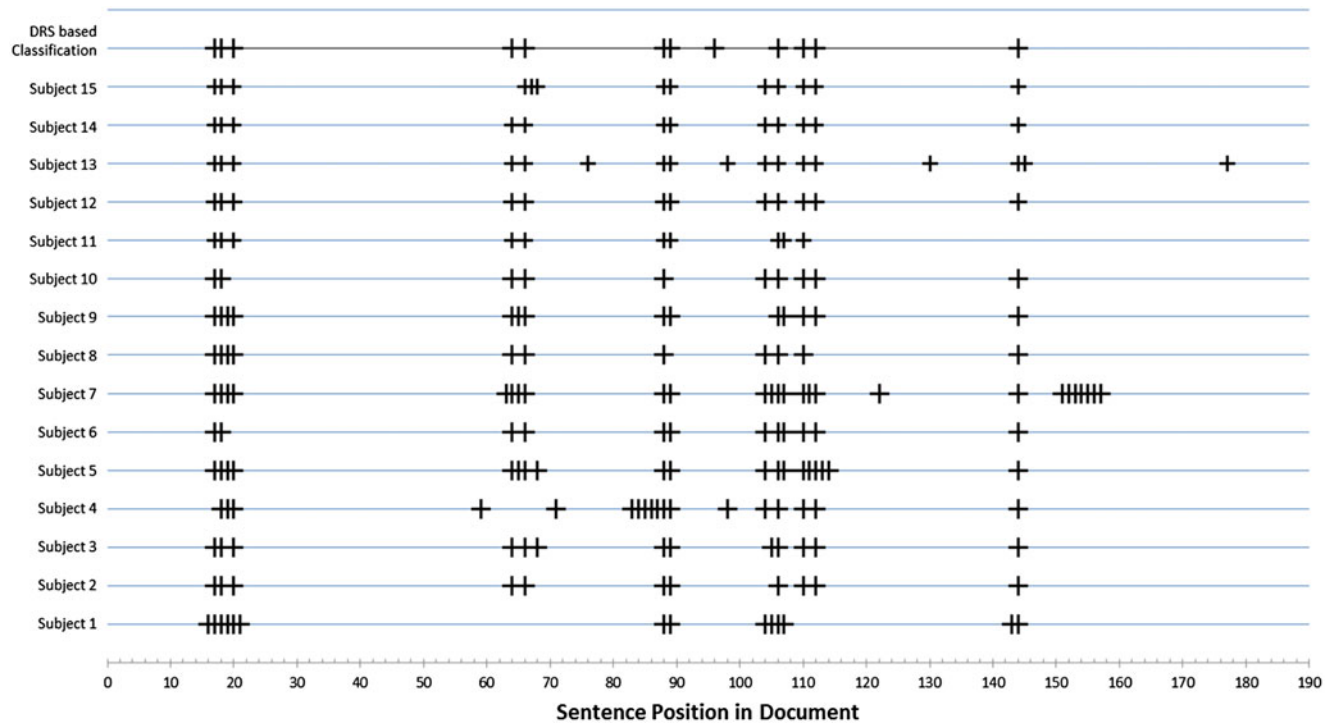

Fig. 12. Comparison of the performance of the implementation against those of subjects.

In the second step, the subjects were asked to mark the segments related to the aircraft domain. Domain relevance was limited to aircraft domain, as other domains like assembly needed more knowledge and further resources for their implementation (such as an assembly ontology). Subjects were asked to mark out the whole segment even if only a part of it was related. They were asked to give a score of how strong the segment relevance was, on a scale of 1–3 (1 = *not so strong*, 3 = very *strong*; the scale was kept from 1 to 3 only because, unlike the individual validation where it was 1 to 5, we do not have opposing qualities in marking, and only those with a strong value are marked anyways). Scores given by 30 subjects were consolidated and tabulated. We present the results of both steps in Table 3.

The Python program was then run to find out the segments and classify them. Similar to Section 5.2, the values of intersentence similarity and the difference between their adjacent values were plotted. Figures 14 and 15, respectively, show the absolute and similarity measure value plots of the intersentence similarity measures for the average, min, max, and sum variations.

Here we also use a strategy similar to that in Section 4.1.2 to get a combination of low absolute values. We refined the strategy used earlier, by adding a heuristic. As usual, we took a combination of

- Low values of average and max values: Here, cutoffs of 0.04 (major) and 0.03 (minor) for average and 0.3 (major) and 0.4 (minor) for max values were considered. Min values were not considered because there were many zero values.
- Low value of Normalized Sum: Here, a major cutoff of –1 and min cutoff of –0.5 were used.

Using the above values, the heuristic was that even if one out of the three above values were a major cutoff, it would be a major segment change. Otherwise, the segment change is deemed minor. Applying this heuristic, major segment starts were identified at Sentences 7, 12, 18, 25, 28, 38, and 42. In addition, minor segments starts were at 5 and 34. Comparing this with Table 3, the segments identified by subjects were 7, 12, 19, 25, and 38. However, no corresponding feedback was present by subjects for the min points seen in the implementation. Let us consider each of the points that do not match.

- For Sentence 5 (Point 3), a possible reason is low value of difference in Similarity Value, due to both incorrect semantic interpretation of Sentence 3 and the incorrect choice of synset entry for the word "*household*" in Sentence 4.
- In Sentence 34 (Point 32), one direct reason we see is the choice of incorrect synset for the word "*spar.*" This again, is due to the problem of ambiguity.
- Sentence 18 (Point 16) seems to have got a low value (and Point 17 a high value) due to the incorrect choice of synsets for the words "*process*" and "*weld.*" For "*weld*" this ambiguous choice resulted in a synset corresponding to a dye, rather than a joining of material. Similarly, for "*process*," the ambiguity results in it being understood as a mental process, rather than a manufacturing

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

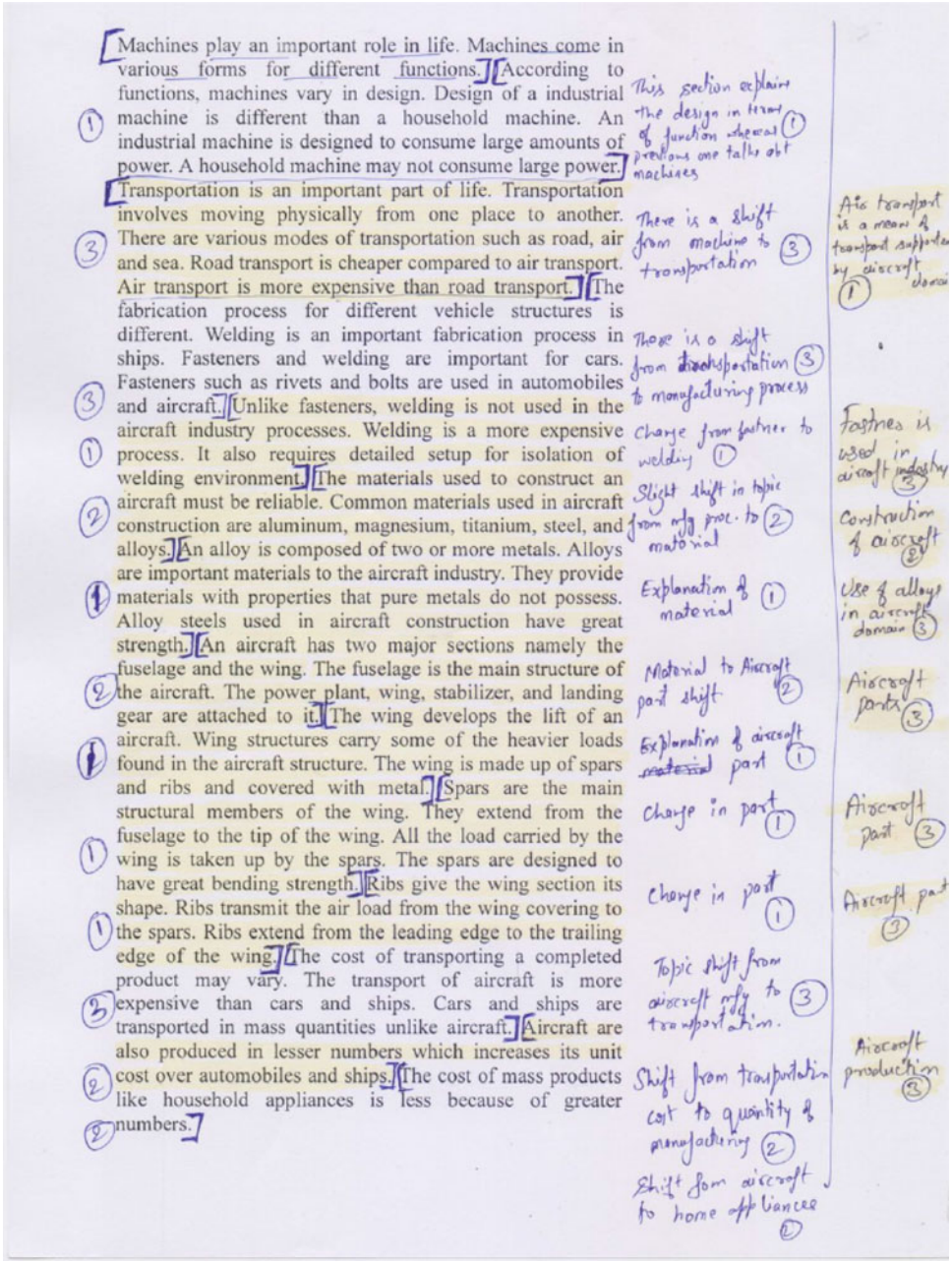

Fig. 13. A subject's response sheet for the reading exercise.

one. This was also verified on the online similarity measuring tool WS4J (http://ws4jdemo.appspot.com).

- In Sentence 28 (Point 26), one issue was that the phrase "*power plant*" was read as two words, and the word "*plant*" was interpreted as in a manufacturing plant, rather than an engine. Hence, ambiguity has again played a role in getting a low value of similarity.

Regarding the classification step, we looked at the relevant segments for the aircraft domain. Due to different segments being marked by different subjects, we looked at the aggregate numbers provided by subjects. Considering 20 Subjects as a major cutoff, and 15 as a minor cutoff, the relevant segments are marked in Table 4. The relevant semantic score as given by the subjects is also indicated.

Figure 16 shows the sentences indicated relevant by subjects, marked above the segments classified by the implementation. The bottom part of the figure shows the sentence index number. The two top rows of colored squares indicate segment boundaries and relevant sentences as given by subjects. (Red colors are major, and orange ones are minor). The two bottom rows indicate the same for the implementation.

Out of 33 sentences classified by a large number of subjects, 28 sentences were classified by the program. Two extra

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

Table 3. *Number of subjects for both segment boundaries and relevance of sentences*

Sentence No.	No. of Subjects			
	Relevance (Not Strong)	Relevance (Strong and Very Strong)	Segment Bound. (Not Strong)	Segment Bound. (Strong and Very Strong)
1	5	0	0	0
2	5	0	2	0
3	5	0	4	0
4	3	0	3	3
5	3	0	3	1
6	3	0	1	0
7	9	11	0	28
8	10	13	1	2
9	13	14	4	2
10	13	14	0	0
11	13	14	0	0
12	4	10	1	28
13	4	9	4	5
14	5	12	4	6
15	6	15	4	2
16	7	16	5	3
17	6	10	1	6
18	6	10	0	0
19	0	30	2	28
20	0	30	0	0
21	1	26	9	5
22	1	29	3	2
23	1	29	0	0
24	1	29	1	1
25	0	30	2	26
26	0	30	1	0
27	0	30	0	1
28	0	30	5	6
29	0	30	3	1
30	0	30	3	0
31	0	30	4	2
32	0	30	0	0
33	0	30	0	0
34	0	30	0	0
35	0	30	6	3
36	0	30	0	0
37	0	30	0	0
38	4	18	1	29
39	5	17	1	1
40	6	16	2	2
41	5	22	7	8
42	2	9	5	11

Note: The dark and light shaded areas show where a major and minor number of subjects indicated relevance and segment boundary, respectively.

sentences (12, 13) were included by the classification. They have not been counted in the subject's feedback because their segments were not all the same, and the counting (Table 3) was done on a per-sentence basis. Sentences 7 to 12 were classified as minor, relevant segments by subjects, but not by the implementation. It was found that although some parts of this segment did relate to the aircraft domain (such as talking about *air transport*), they did not contain any matching terms from the AIRCRAFT ontology. Because of this, the implementation did not identify them as relevant.

6. CONCLUSIONS

6.1. Contributions

This paper has identified the need, and proposed a method for segregating coherent and relevant pieces of text for knowledge acquisition from documents. This segregation method is the main contribution of this paper. The method supports understanding of a document, which would be useful for further processing of text necessary for knowledge acquisition.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

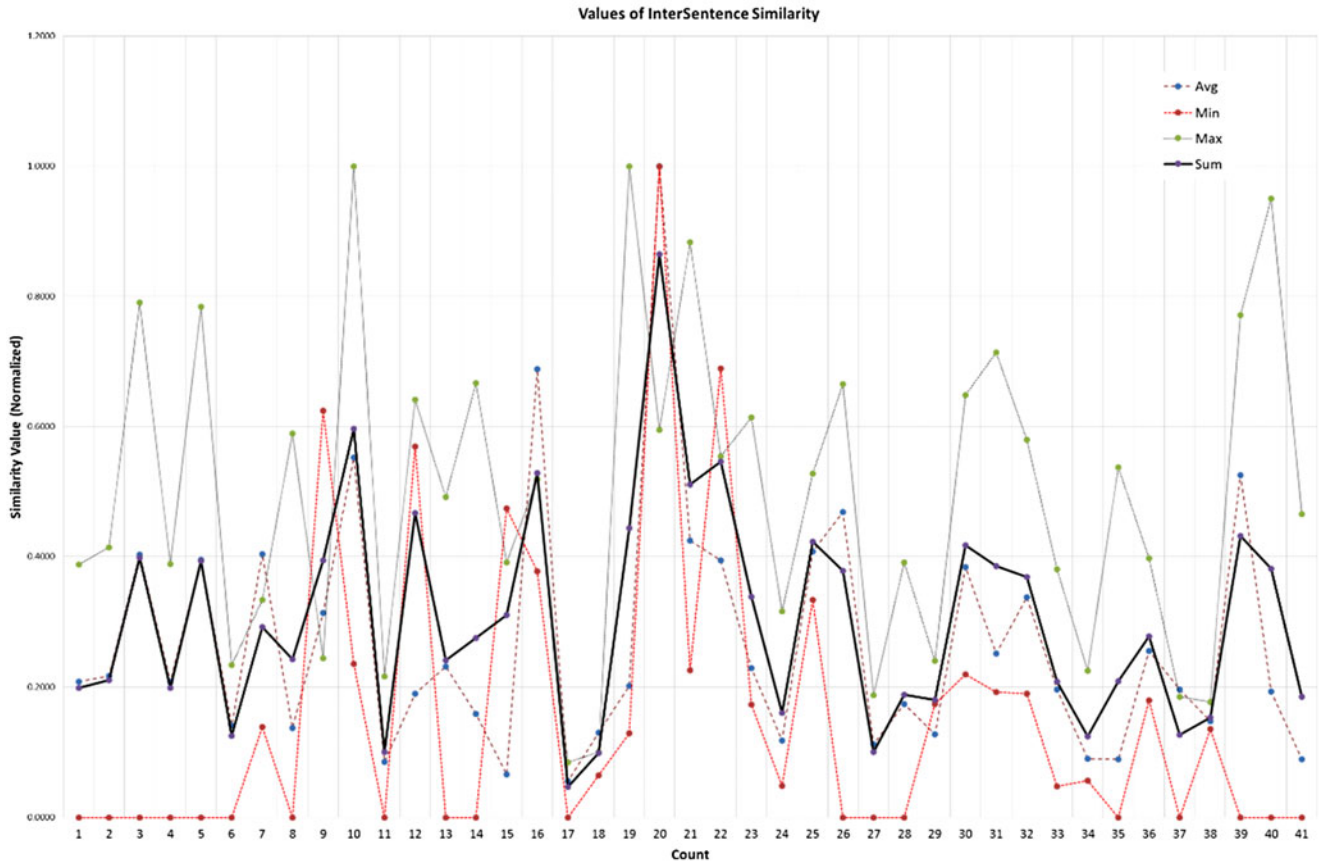

Fig. 14. Values of intersentence similarity.

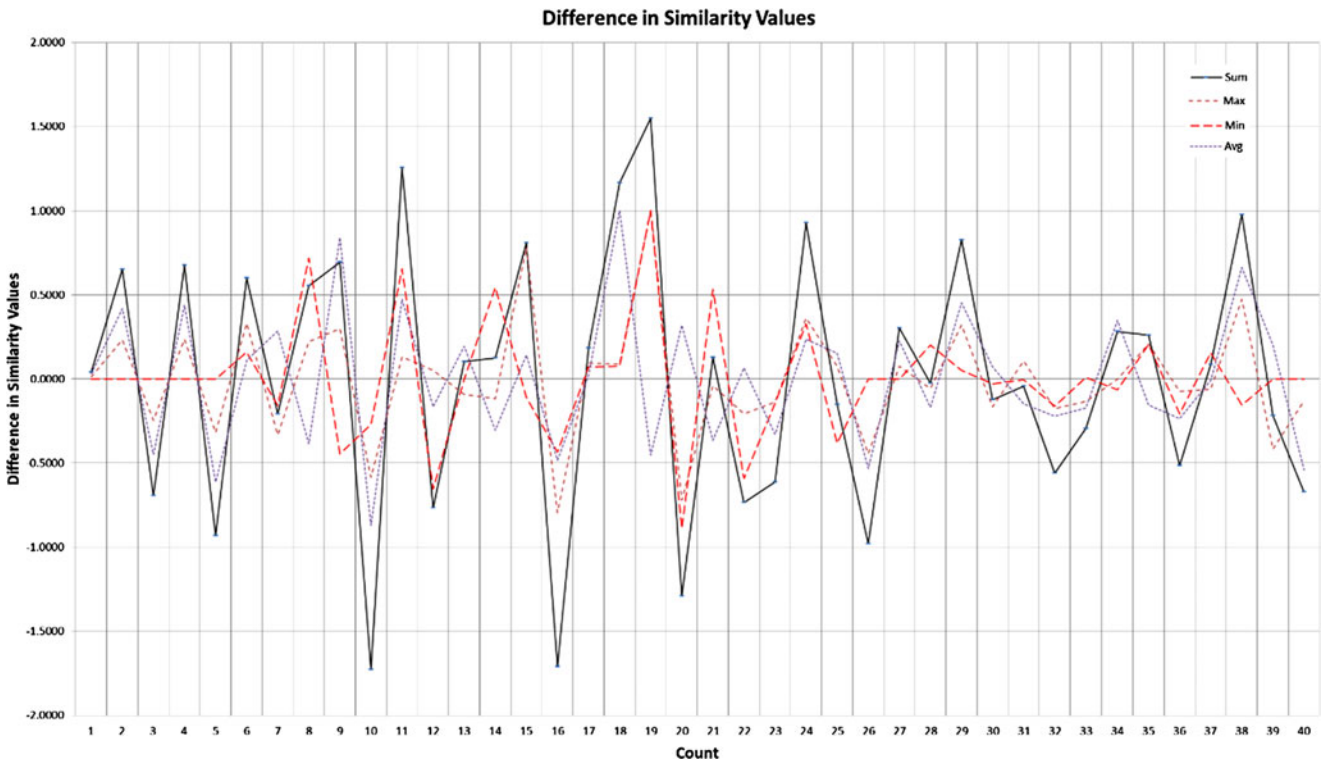

Fig. 15. Values of SM_i and max, avg, and min for adjacent sentences.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

Table 4. *Relevance scores assigned by the implementation for its segments*

Segment	No. of Entities in Segment	Score for Relevance
1–4	14	0
5–6	3	0
7–11	15	0
12–17	20	2
18–24	24	4
25–27	6	3
28–33	19	6
34–37	15	3
38–41	15	3

The proposed method is based on discourse analysis that is centered on understanding the semantic content of the sentences in a document. It has been implemented and validated in an iterative manner, separately in two parts (segmentation and classification), then refined, and finally evaluated in an integrated form. The method relies on entities found in a text and semantic relations between entities across sentences. Large variations in semantic relations, as quantified by proposed measures and specified threshold values, have been shown to indicate segments, between which changes in topics occur. Once segments are identified through use of a domain ontology, a means for classifying segments related to the aircraft domain is proposed.

The segment identification part has been validated using two documents, by comparing the results from the implementation with those provided by test subjects. These matched in 75% of the cases in the first instance, and in 80% of the cases in the second. However, the implementation also identified other segments that were not marked by subjects. Possible reasons for these have been explained in this paper.

The classification step has been validated independently and then in combination with the segmentation step. It has been found to be able to satisfactorily classify segments, showing a reasonable agreement with subjects' feedback of around 85% for a specified threshold of half of the subjects' consensus. This translates to a precision of 0.93 and a recall of 0.84 for the segregation. This step completes the process of segregation of segments from a document.

6.2. Discussion

Though we discussed the performance of the proposed method in the previous subsection, it is by no means perfect, due to the challenges that make processing of natural language texts difficult. Some of these are summarized below:

a. *Efficiency:* Because this implementation is heavily dependent on the usage of currently available practical tools each of which executes a part of the method, the overall efficiency of the method is eventually a function of the individual efficiencies of these tools. For example, when anaphora resolution for a sentence did not work as expected, it also affected the outcome of segmentation (Section 5.2). Efficiency is also affected in the semantic interpretation of text given the lack of coverage of domain-specific terms in general English lexicons.
b. *Ambiguity:* A difficulty faced by us in semantic processing was ambiguity of meaning. Although we attempted to solve the problem of not getting a synset (a WordNet entry) by getting the closest one, it still does not address the ambiguity problem. The correct sense in which a word is used in a sentence can be identified by the use of other natural language processing techniques such as word sense disambiguation. For example, word sense disambigulation has been used in an implementation of similarity measurement between sentences (http://www.codeproject.com/Articles/11835/WordNet-based-semantic-similarity-measurement). In addition, there could be ambiguity arising out of the different ways in which a single word could be used.
c. *Nonlinear change in context:* The current variation in context is detected in a linear fashion, that is, as a document is read from the beginning to the end. However, the variation in context may also manifest in other

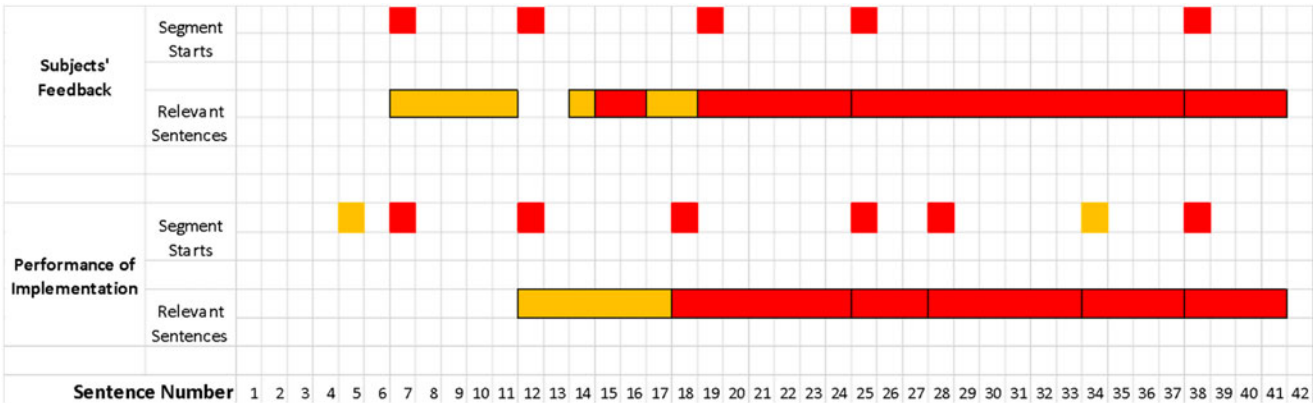

Fig. 16. A comparison of classified segments by subjects and the implementation.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

ways. For example, within a segment of larger context, there may be smaller segments with a different context. The current method would have to be modified to accommodate such cases.

d. *Phrases:* Recognition of phrasal terms (e.g., "*power plant*") is necessary to correctly identify the meaning of sentences containing these. This is more relevant in the purview of documents written in a technical language.

7. FUTURE WORK

The method proposed, implemented, and validated is a first version of an approach for segregation of relevant segments from text. There is plenty of ample scope for future work, both in increasing the efficiency of the implementation and in extending the method to address the complexities of real-life documents. Unlike the test documents, real-life documents would be more difficult to process due to factors such as background knowledge, noise, and use of more complicated technical language.

An immediate extension to the above segmentation method would be to use a moving window of sentences, and check for variations in meaning, compared to using only pairs of adjacent sentences. This might help in looking at sentences that are a few sentences apart, but are nonetheless related.

Another factor yet to be explored in depth is that of anaphoric links. Our understanding is that if two sentences are linked by one or more anaphora, they must be related to each other. This has to be factored into the similarity measure between sentences.

We have modeled the whole document as only a single linear piece of text. However, much of the information about the meaning of a part of a document can be derived by analyzing at the topic heading of the section to which the part belongs. Such a model of the document based on section and subsection headings (such as Liu et al., 2006) can identify the context of the sentences in that section.

The classification in the current work has been with respect to an AIRCRAFT ontology. We need a larger set of domains to cover aircraft assembly. This would require a combination of domains such as manufacturing, assembly, hydraulics, and workplace ergonomics. Hence, the reference ontology set would have to be enlarged with ontologies from these domains. A challenge would be in finding open ontologies in these domains that are often proprietary to organizations. We have already faced difficulty in locating such large, open ontologies in these domains.

The similarity measure discussed in Section 4.1.2 was a basic one, because it gave equal weightage to the difference in average, min, and max values of intersentence similarity. This measure could be refined further with variable weights. The inter-sentence similarity strategy could also be more complex, such as using matching similarity instead of average, min, or max.

For presenting the comparison of results of the program and the subjects' feedback, we have merely presented the results side by side, and given a percentage of matching cases. Mathematical measures such as the popular P_k measure and the WindowDiff measure for segmentation (Pevzner & Hearst, 2002) could be used in the future to report the results in terms of a single number.

An important factor in the reading of documents is the background knowledge of the reader. We are yet to investigate if there are means of incorporating such background knowledge for understanding texts.

After having identified the relevant segments, the goal of this research would be in identification of presence of issues and their causes in text. This is the core purpose in the larger perspective of the research presented in this paper, for which the work presented prepares a document.

ACKNOWLEDGMENTS

The authors are grateful to the creators of different tools used here, such as Johan Bos and Curran and Clark for Boxer and C&C tools, Long Qiu for JavaRAP, and Dan Garrette for the NLTK implementation of Boxer interface. Without these tools being available in an Open manner, it would have been impossible to create this implementation in its current form. We also profusely thank the test subjects from the Centre for Product Design and Manufacturing, Indian Institute of Science, Bangalore. The research work reported here is part of a project carried out with the Boeing Company under SID Project PC 36030.

REFERENCES

Alavi, M., & Leidner, D.E. (2001). Review: knowledge management and knowledge management systems: conceptual foundations and research issues. *MIS Quarterly 25(1)*, 107–136.

Allen, J. (2011). *Natural Language Understanding*, 2nd ed. New York: Pearson.

Andrews, N.O., & Fox, E.A. (2007). *Recent developments in document clustering*. Technical Report TR-07-35. Blacksburg, VA: Virginia Tech, Computer Science.

Ast, M., Glas, M., Roehm, T., & Luftfahrt, V.B. (2014). *Creating an Ontology for Aircraft Design*. Bonn: Deutsche Gesellschaft für Luft-und Raumfahrt-Lilienthal-Oberth eV.

Beeferman, D., Berger, A., & Lafferty, J. (1999). Statistical models for text segmentation. *Machine Learning 34(1–3)*, 177–210.

Blackburn, P., & Bos, J. (2006). *Working With Discourse Representation Theory: An Advanced Course in Computational Semantics*. Accessed at http://ling.uni-konstanz.de/pages/home/butt/main/material/bb-drt.pdf

Bos, J., (2008). Wide-coverage semantic analysis with boxer. *Proc. 2008 Conf. Semantics in Text Processing*, pp. 277–286. Stroudsburg, PA: Association for Computational Linguistics.

Chandrasegaran, S.K., Ramani, K., Sriram, R.D., Horváth, I., Bernard, A., Harik, R.F., & Gao, W. (2013). The evolution, challenges, and future of knowledge representation in product design systems. *Computer-Aided Design 45(2)*, 204–228.

Chen, H. (2010). *Learning semantic structures from in-domain documents*. PhD Thesis, Massachusetts Institute of Technology.

Curran, J.R., Clark, S., & Bos, J. (2007). Linguistically motivated large-scale NLP with C&C and Boxer. *Proc. 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pp. 33–36. Stroudsburg, PA: Association for Computational Linguistics.

Feigenbaum, E.A. (2003). Some challenges and grand challenges for computational intelligence. *Journal of the ACM 50(1)*, 32–40.

https://doi.org/10.1017/S0890060416000408 Published online by Cambridge University Press

Foltz, P.W., Kintsch, W., & Landauer, T.K. (1998). The measurement of textual coherence with latent semantic analysis. *Discourse Processes 25(2–3)*, 285–307.
Fraser, B. (1999). What are discourse markers? *Journal of Pragmatics 31(7)*, 931–952.
Giora, R. (2003). Segmentation and segment cohesion: on the thematic organization of the text. *Text-Interdisciplinary Journal for the Study of Discourse 3(2)*, 155–182.
Goller, C., Löning, J., Will, T., & Wolff, W. (2000). Automatic document classification—a thorough evaluation of various methods. *Proc. ISI 2000*, pp. 145–162. Cuernavaca, Mexico, October 10–14.
Grosz, B.J., & Sidner, C.L. (1986). Attention, intentions, and the structure of discourse. *Computational Linguistics 12(3)*, 175–204.
Gruber, T.R. (1989). Automated knowledge acquisition for strategic knowledge. *Machine Learning 4(3–4)*, 293–336.
Han, X., & Sun, L. (2012). An entity-topic model for entity linking. *Proc. 2012 Joint Conf. Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 105–115. Stroudsburg, PA: Association for Computational Linguistics.
Hearst, M.A. (1994). Multi-paragraph segmentation of expository text. *Proc. 32nd Annual Meeting on Association for Computational Linguistics*, pp. 9–16. Stroudsburg, PA: Association for Computational Linguistics.
Hoque, A.S.M., & Szecsi, T. (2007). Application of design-for-manufacture (DFM) rules in CAD/CAM. *Proc. 3rd I*PROMS Virtual Conf.*, Cardiff, July 2–13.
Hossain, M.S., & Angryk, R.A. (2007). Gdclust: a graph-based document clustering technique. *Proc. 7th IEEE Int. Conf. Data Mining Workshops, 2007/ICDM Workshops 2007*, pp. 417–422, Omaha, NE, October 28–31.
Kamp, H., & Reyle, U. (1993). *From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation theory*. No. 42. Berlin: Springer Science & Business Media.
Kataria, S.S., Kumar, K.S., Rastogi, R.R., Sen, P., & Sengamedu, S.H. (2011). Entity disambiguation with hierarchical topic models. *Proc. 17th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 1037–1045. New York: ACM.
Lascarides, A., & Asher, N. (2008). Segmented discourse representation theory: dynamic semantics with discourse structure. In *Computing Meaning*, pp. 87–124. Dordrecht: Springer.
Le Thanh, H., Abeysinghe, G., & Huyck, C. (2004). Automated discourse segmentation by syntactic information and cue phrases. *Proc. IASTED Int. Conf. Artificial Intelligence and Applications (AIA 2004)*, Innsbruck, Austria.
Li, Y., Chung, S.M., & Holt, J.D. (2008). Text document clustering based on frequent word meaning sequences. *Data & Knowledge Engineering 64(1)*, 381–404.
Liu, B., Li, X., Lee, W.S., & Yu, P.S. (2004). Text classification by labeling words. *Proc. AAAI*, Vol. 4, pp. 425–430. Cambridge, MA: MIT Press.
Liu, S., McMahon, C.A., & Culley, S.J. (2008): A review of structured document retrieval (SDR) technology to improve information access performance in engineering document management. *Computers in Industry 59(1)*, 3–16.
Liu, S., McMahon, C.A., Darlington, M.J., Culley, S.J., & Wild, P.J. (2006). A computational framework for retrieval of document fragments based on decomposition schemes in engineering information management. *Advanced Engineering Informatics 20(4)*, 401–413.
Liu, T.I., Yang, X.M., & Kalambur, G.J. (1995). Design for machining using expert system and fuzzy logic approach. *Journal of Materials Engineering and Performance 4(5)*, 599–609.
Loftus, C., Hicks, B., & McMahon, C. (2009). Capturing key relationships and stakeholders over the product life cycle: an email based approach. *Proc. 6th In. Conf. Project Life Cycle Management (PLM 09)*, Bath, July 6–8.
Loper, E., & Bird, S. (2002). NLTK: the natural language toolkit. Proc. ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics, Vol. 1. Stroudsburg, PA: Association for Computational Linguistics.
Madhusudanan, N., & Chakrabarti, A. (2014). A questioning based method to automatically acquire expert assembly diagnostic knowledge. *Computer-Aided Design 57*, 1–14.
Marx, W.J., Mavris, D.N., & Schrage, D.P. (1998). A knowledge-based system integrated with numerical analysis tools for aircraft life-cycle design. *Artificial Intelligence for Engineering, Design Analysis and Manufacturing 12(3)*, 211–229.
Mihalcea, R., Corley, C., & Strapparava, C. (2006). Corpus-based and knowledge-based measures of text semantic similarity. *Proc. AAAI*, Vol. 6. Cambridge, MA: MIT Press.
Miller, G.A. (1995). WordNet: a lexical database for English. *Communications of the ACM 38(11)*, 39–41.
Morris, J., & Hirst, G. (1991). Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics 17(1)*, 21–48.
Mozina, M., Guid, M., Krivec, J., Sadikov, A., & Bratko, I. (2008). Fighting knowledge acquisition bottleneck with argument based machine learning. *Proc. European Conf. Artificial Intelligence*, pp. 234–238, Patras, Greece, July 21–25.
Mu, J., Stegmann, K., Mayfield, E., Rosé, C., & Fischer, F. (2012). The ACODEA framework: developing segmentation and classification schemes for fully automatic analysis of online discussions. *International Journal of Computer-Supported Collaborative Learning 7(2)*, 285–305.
Nyberg, K. (2011). *Document classification using machine learning and ontologies*. MS Thesis, Aalto University, School of Science, Degree Programme of Information Networks.
Park, J.-H., & Seo, K.K. (2003). Knowledge-based approximate life cycle assessment system in the collaborative design environment. *Proc. 3rd Int. Symp. Environmentally Conscious Design and Inverse Manufacturing, 2003. EcoDesign'03*, Tokyo, December 11–13.
Passonneau, R.J., &. Litman, D.J. (1997). Discourse segmentation by human and automated means. *Computational Linguistics 23(1)*, 103–139.
Pevzner, L., & Hearst, M.A. (2002). A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics 28(1)*, 19–36.
Pokojski, J. (2006). *Knowledge Based Engineering and Intelligent Personal Assistant Context in Distributed Design, Intelligent Computing in Engineering and Architecture*, pp. 519–528. Berlin: Springer.
Qiu, L., Kan, M.Y., & Chua, T.-S. (2004). A public reference implementation of the RAP anaphora resolution algorithm. *Proc. 4th Int. Conf. Language Resources and Evalution*, Lisbon, Portugual.
Reynar, J.C. (1999). Statistical models for topic segmentation. *Proc. 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*. Stroudsburg, PA: Association for Computational Linguistics.
Stein, B. (2004). Topic identification: framework and application. *Proc. I-KNOW '04*, Graz, Austria, June 30–July 2.
Tofiloski, M., Brooke, J., & Taboada, M. (2009). A syntactic and lexical-based discourse segmenter. *Proc. ACL-IJCNLP 2009 Conf. Short Papers*. Stroudsburg, PA: Association for Computational Linguistics.
Venkatachalam, A.R., Mellichamp, J.M., & Miller, M.D. (1993). A knowledge-based approach to design for manufacturability. *Journal of Intelligent Manufacturing 4(5)*, 355–366.
Wijewickrema, C.M., & Gamage, R. (2013). An ontology based fully automatic document classification system using an existing semi-automatic system, *Proc. IFLA WLIC 2013*. Singapore: Future Libraries: Infinite Possibilities.
Xie, S.Q., PTu, P.L., & Zhou, Z.D. (2004). Internet-based DFX for rapid and economical tool/mould making. *International Journal of Advanced Manufacturing Technology 24(11–12)*, 821–829.
Zhang, W., Sim, Y.C., Su, J., & Tan, C.L. (2011). Entity linking with effective acronym expansion, instance selection, and topic modeling. *Proc. 23rd. Int Joint Conf. Artificial Intelligence*, pp. 1909–1914. Cambridge, MA: MIT Press.
Zheng, H.-T., Kang, B.-Y., & Kim, H.-G. (2009). Exploiting noun phrases and semantic relationships for text document clustering. *Information Sciences 179(13)*, 2249–2262.

N. Madhusudanan is a PhD student in the Virtual Reality Laboratory at the Centre for Product Design and Manufacturing at the Indian Institute of Science, Bangalore. He attained a Masters degree (research) in product design and manufacturing from the Centre for Product Design and Manufacturing at the Indian Institute of Science and a Bachelor's degree in mechanical engineering. He also has 2 years of experience in computer-aided design and product design and manufacturing devel-

opment and customization at Robert Bosch. His areas of interest are engineering design, artificial intelligence, CAD, knowledge-based engineering, design for assembly and manufacturability, and computer graphics.

Amaresh Chakrabarti is Professor and Chairman of the Virtual Reality Laboratory at the Centre for Product Design and Manufacturing at the Indian Institute of Science, Bangalore. Prior to joining the Indian Institute of Science, he was a Design Engineer with Hindustan Motors, Kolkata, and Research Associate and Senior Research Associate at the University of Cambridge. Dr. Chakrabarti has written hundreds of international journal articles and proceedings papers and has authored many books and book chapters. His interests are in design synthesis, creativity, ecodesign, sustainability, artificial intelligence in design, biologically inspired design, smart manufacturing, and design research methodology.

B. Gurumoorthy is a Professor in the Virtual Reality Laboratory at the Centre for Product Design and Manufacturing at the Indian Institute of Science, Bangalore. He received PhD and ME degrees in mechanical engineering from Carnegie Mellon University and a B.Tech. degree in mechanical engineering from the Indian Institute of Technology, Madras. His areas of interest are computer-aided design, product informatics, computational metrology, and computer-aided prototyping.