Abstract

The need for accurate communication between customers and developers is critical in requirements engineering. If this communication is not accurate, misunderstandings occur with potentially serious consequences on product development. The most dangerous misunderstandings arise silently in the minds of engineers who remain convinced that their understanding is accurate, sometimes for protracted periods. Making this worse, misunderstandings are easily propagated unknowingly to others, thereby spreading the potential damage.

The communication with which we are concerned takes place in natural language. Previously, we have develop a requirements engineering technique called CLEAR that helps to address the problem of misunderstandings in natural language communications. It does this by tackling the basic issues using concepts from the theory of cognitive linguistics. CLEAR requires substantial effort to produce useful results, and this is a barrier to its adoption. In this paper, we present a method to generate and evaluate versions of CLEAR that are proportional, i.e., techniques that require less effort to use yet provide useful although not full performance. We present an example technique that provides identification of phrases critical to domain understanding, an activity that requires substantial human intervention in CLEAR. We compare the overall performance of the proportional technique with CLEAR.

1. Introduction

The need for accurate communication between customers and developers is critical in requirements engineering. If this communication is not accurate, misunderstandings occur with potentially serious consequences on product development. The most dangerous misunderstandings arise silently in the minds of engineers who remain convinced that their understanding is accurate, sometimes for protracted periods. Making things worse, misunderstandings are easily propagated unknowingly to others, thereby spreading the potential damage.

The vehicle that is used almost universally for communication in requirements engineering is natural language, usually English. Many times English documents are supplemented with formalisms such as tables and mathematics, but the basic structure of communication is English text.

English is known to be far from optimal in its role in engineering communication. Studies have revealed ambiguity, incompleteness, and inaccuracy in engineering documents written in English. The very high impact of misunderstandings is well known by practicing engineers who have to deal with their effects, yet English continues to be used.

Efforts to replace English with formal languages in requirements and specification have met with some success [2, 3]. Nevertheless, it is impossible to replace natural language with a formal language completely in any engineering document. Formal languages do not contain any real-world semantics, and so any formal statement only has a meaning in the closed world of the logic in which it is expressed. The real-world meaning in formal statements is derived from associated natural language, often identifiers and comments. The necessity of natural language can be seen by noting that if all the identifiers in a formal statement are changed to random strings and the comments are removed, the logical meaning of the statement is preserved but the real-world meaning is lost completely.

The field of cognitive linguistics has established many results that help to explain inaccuracy, incompleteness and ambiguity in natural language understanding, and basic research results in this field have been applied to the fundamental communication issues in requirement engineering [13, 14]. One of the results
of that work was the development of the Cognitive Linguistic Elicitation and Representation (CLEAR) method for defining the context associated with a specific application. The result of applying CLEAR is a CLEAR knowledge base in which the information necessary to understand the operating context for an application is defined. CLEAR attacks the problem of misunderstanding in requirements engineering directly by providing critical information that is accessible to both customers and developers and which is known to be free of certain classes of defects.

The CLEAR process is resource intensive, requiring prescribed activities by domain experts and others along with automated, tool-supported analysis. The effort required pays substantial dividends, but it is also an impediment to adoption. In this paper, we present an approach to proportional application of CLEAR, a technique that we refer to as Proportional Linguistic Analysis (PLA). Our goal with PLA is to allow engineers to adopt CLEAR progressively so as to reduce the risks associated with adoption. The research question that we answer in this paper is how to generate and evaluate applications of the complete linguistic process that are proportional and the degree to which CLEAR’s benefits are realized under such circumstances.

We begin by reviewing the CLEAR process and then we present our approach to deriving proportional versions of CLEAR. Next we present an example of this proportional approach together with an assessment of the approach. This assessment is conducted by comparing the results of a proportional version when applied to a specific project with the results obtained with CLEAR. Finally, we present our conclusions.

2. The CLEAR process

The CLEAR process is a comprehensive approach to the application of cognitive linguistics to the problem of eliciting and representing domain knowledge for use in software requirements documentation [13].

As we have noted, cognitive linguistics provides a foundation of results that can be used to both explain undesired phenomena observed in engineering environments and direct how they might be avoided. It elaborates cognitive states that must exist and events that must take place in order for communication to be pragmatically sufficient. In everyday life, such states and events arise spontaneously, and misunderstandings lead to circumstances with which we can usually cope. However, generally they do not in engineering, and we cannot know when they do. Requirements engineering is an activity that requires accurate and precise communication and that requires communication across domain boundaries.

CLEAR is built on the proposition that with appropriate guidance the necessary cognitive states and events can be determined and recorded under circumstances where this would not normally be the case. CLEAR accommodates natural human communicative habits and limits by defining, and guiding the construction of explicit processes and artifacts that allow necessary cognitive states to be achieved and necessary events to occur.

2.1. The basic process

The CLEAR process is both complex and extensive, and full details are available elsewhere [13]. A comprehensive description cannot be included here,
and so we present a brief summary. CLEAR consist of four phases (shown in Figure 1):

**Selection.** The terms and phrases that are important in the domain of interest are determined during the selection phase. This is done partially automatically by analysis of existing documents and partially manually. Selection uses a variety of heuristics derived from the information-retrieval literature to produce candidate sets of terms and phrases, and careful reviews by domain experts are used to prioritize candidate terms and complete the selection process.

**Elicitation.** In this phase, the semantics of the selected terms and phrases are collected, and the fundamental category graph is developed. Cognitive categories [11, 12, 9, 5] in the form of a graph are the basic structure that humans use for organizing semantic information, and in this phase of CLEAR a structure is developed to capture the category graph to the extent possible.

**Representation.** Explications for the selected terms and phrases are created using the Partial Reductive Paraphrase (PRP) technique (see below) in this phase. By design PRP produces explications that are free of an important set of definitional deficiencies.

**Integration.** This phase guides the assembly of the refined and enhanced representations into an integrated structure, the CLEAR knowledge base, permitting collective manipulation and analysis.

For any given application, the complete CLEAR process produces a structure that contains the domain information necessary for engineering tasks associated with that domain. As an example, consider the problem of developing requirements for a particular spacecraft. CLEAR might be applied to the domain of the navigation system. This will provide the necessary domain information to permit developers to understand navigation algorithms with sufficient precision that the right software can be specified with a high degree of assurance. Defective requirements are known to be a major source of system defects in high-assurance systems [6, 7].

CLEAR can be applied multiple times for the same application if that application involves multiple domains. This will be the case for the spacecraft example in which typical domains might be guidance, navigation, propulsion, scientific instruments, etc. The results of these multiple applications of CLEAR is a set of information repositories describing critical domain knowledge that facilitates the cross-boundary communication between the various engineering disciplines involved. In particular, CLEAR can be used in this way to enable effective communication between software engineers and all other engineering disciplines that are involved in the development of the spacecraft.

### 2.2. Creating explications

The development of CLEAR has shown that communication between domain experts and software developers can be analyzed rigorously using established techniques from the field of linguistics. Further, this communication has specific characteristics that are the root cause of communicative breakdowns across a domain boundary, and it is possible to identify these characteristics and mitigate their effects using the results of linguistic analysis. An example of these causes is **cognitive economy**. This mechanism is the ability of humans to quickly associate with an entity a large number of attributes that might not be readily observable [11].

When definitions of important terms and phrases are prepared in an ad hoc manner, they frequently suffer from weaknesses that reduce their value significantly. In the worst case, the result can be a misunderstanding by the reader of a term that has significant engineering importance. Three major weaknesses in definitions are:

**Obscurity.** An obscure definition presents a description that uses words no more significant to the consumer than the word being defined [4]. Since the consumer is lacking a form-meaning association allowing him or her to understand the target word to begin with, a definition that relies on words requiring additional associations not either already possessed by the consumer or themselves otherwise appropriately defined is not helpful. Scientific and engineering definitions are particularly vulnerable to obscurity. ‘[A]ir’ has been defined as “the mixture of invisible odorless tasteless gases (as nitrogen and oxygen) that surrounds the earth” [8]. A consumer not familiar with the term is not likely to be any more familiar with terms such as ‘nitrogen’ than with ‘air’ and thus will be limited in what he can interpret from the given description.

**Circularity.** A circular definition presents a description that uses one or more words that themselves are defined in terms of the target word [4]. Circularities can be divided into two classes; obvious circularities and insidious circularities. An obvious circularity uses the target word or a version of it in the description, such as “red: having the property of redness”. Understanding redness presupposes understanding red, but understanding red cannot be presupposed on the part of a consumer in search of a definition for red. Insidious circularities, on the other hand, are those for which there is hidden indirection that nevertheless results in a term relying eventually on itself for its own definition. For example, a published definition for ‘question’ relies on ‘request’, which relies on ‘ask’, which relies on ‘answer’, which relies back again on ‘question’ [10].
Otherwise non-predictiveness. Otherwise non-predictive definitions present descriptions that rely on devices which, while convenient to the definition writer, explicitly hinder the ability for a consumer to predict appropriate range of use of a word. Such devices include the disjunction ‘or’, subjunctives ‘may’, ‘might’ and ‘can’, umbrellas like ‘etc.’ and ‘esp.’ and hedges like ‘usually’ and ‘generally’. By resorting to devices like ‘etc.’, “...the lexicographer makes the definition untestable” [4]. The issue with such devices is that without criteria for the scope they are intended to cover, it is impossible to know what is included and what is not. Thus a consumer is unable to predict whether the word is appropriate in a given situation.

Partial Reductive Paraphrase (PRP) is a technique for the creation of accurate explications driven by the goal of eliminating the three problems listed above. PRP renders complex concepts accessible to those without prior experience of them or with insufficient or otherwise flawed experience of them, while taking into account needs and constraints of engineering environments. The content of PRP explications is guided by the material assembled in the category graph of CLEAR Phase 2, and the structure of PRP explications explicitly disallows faults that limit the accessibility of traditional definitions.

2.3. Experimental assessment of CLEAR

During its development, the complete CLEAR process was applied to a target application [13]. The application, which we refer to as the Walker Project, was a software-controlled, safety-critical medical device. Based on the notion of a walker as used by the elderly and others with ambulatory difficulty, it provides enhanced physical support and guidance to persons with limited ability in dynamic situations involving balance and motion. The Walker Project is a research project, but it has a number of properties in common with industrial software development. Its focus is the development of a new product from conception through design and implementation, with the ultimate goal of public commercial consumption. Further, its development brings together individuals with varying backgrounds and multiple domains of expertise who must all communicate their contributions sufficiently among each other for the device to be realized. In addition, its documentation practices represent one of many variations present across software development organizations. In particular, the requirements for the Walker Project are spread informally across a number of documents with varying purposes, as well as located implicitly in the minds of project personnel. For the CLEAR evaluation, access to all existing documentation pertaining to the project was available, as well as consultation with all of its personnel.

We used the Walker application and the results of applying CLEAR as the assessment vehicle for our proportional analysis techniques. The results from CLEAR were treated as the “gold” standard to which data from PLA was compared. We review the application in this section and refer to it in Section 4.

3. Proportional Linguistic Analysis

The CLEAR method has been demonstrated on the Walker Project. Its application requires extensive resources, but we hypothesize that it is, in fact, highly cost effective because it has the potential for reducing rework during development and defects in completed systems. Despite this, developers are, understandably, reluctant to adopt the whole method as a complete entity because of the various development risks associated with such a change. There would be considerable benefit in having a low-risk, low-cost adoption path.

With this issue in mind, we have begun the process of developing partial versions of CLEAR that will allow practitioners to preview CLEAR technology at low cost and with a low risk of process disruption.

3.1. Retroactive application of CLEAR

The first partial version of CLEAR, RetroCLEAR, is aimed at improving software requirements using defect data collected during system testing [15]. The key observation behind RetroCLEAR is that defect reports generated for faults found during testing provide a rich source of information regarding problematic phrases used in requirements documents. These reports indicate that faults often derive from instances of ambiguous, incorrect or otherwise deficient language.

RetroCLEAR guides the discovery of problematic phrases throughout a software requirements statement or specification, using defect reports and correction requests generated during testing to seed the detection process. We found that phrases identified as problematic this way have occurrence properties in specification documents that both allow the direction of resources to prioritize their correction, and generate insights characterizing more general locations of difficulty within the requirements. Our findings allowed us to make recommendations for more efficient and effective management of certain natural language issues in the creation and maintenance of requirements specifications. Further details of RetroCLEAR are available elsewhere [15].
3.2. Resources vs. return

Proportional Linguistic Analysis (PLA) is a natural continuation and generalization of the path started with RetroCLEAR. The idea behind PLA is illustrated in Figure 2. The graph shows a number of possible techniques that refer to as Proportional Linguistic Analysis Exploitation Techniques (PLAnETs). The graph’s X axis is the cost of using the PLAnET and the Y axis shows the value of the PLAnET’s results to developers. The goal is to create PLAnETs that lie on or above the diagonal thereby providing technology where the return is at least proportional to the cost of application.

Creation of PLAnETs is based on a framework that we have built which is a model of the CLEAR process. The framework includes all of the attributes of all elements of the CLEAR process. The attributes for a given element include the element’s functionality, its connections to other elements of the process, its inputs, its outputs, and all the parameters that can be set to characterize the element. As an example, the PLA framework for the Selection phase of CLEAR is shown in Figure 3.

PLAnETs are developed by deliberately restricting many of the attributes in the framework to either fixed values or to values that can be determined automatically. By doing so, we can reduce the effort needed to apply CLEAR because much of the process is either fixed or automated. The cost, of course, is that the value of CLEAR is reduced, but that trade-off is our intent. Fixing the value of an attribute or automating its determination is almost certain to be less effective than determining the attribute properly.

3.3. Creating proportional techniques

Creation of a specific proportional technique begins with the establishment of its goal and the acceptable level of effort that can be expended in its application. For example, an organization’s goal might be to have a PLAnET that just implements the CLEAR Selection process and does so with minimal expenditure of resources. A second organization might choose a complete implementation of CLEAR with a high level of automation. The first of these examples would be located on the lower left of Figure 2 and the second towards the upper right.

With the goal determined, the architecture of the PLAnET is developed by forming an instantiation of each relevant element of the CLEAR framework. For each element, either: (a) the element is given some fixed instantiation; (b) an automatic process is created to replace the manual element; or (c) a simplified manual element is created.

In the final step of PLAnET creation, any parameters that have been introduced by the instantiation process are either given values or a range of suitable values that will be tried in practice are set.

It is important to note that we have confidence in the results of applying the complete, original CLEAR process because it is based upon linguistic theory, and each part of CLEAR is present and has the form that it does for a reason. Thus, by definition, our process for creating a PLAnET weakens the validity of this link and thereby inevitably weakens fundamental confidence in the PLAnET’s validity.

We are dealing with this uncertainty by experimentation. As PLAnETs are developed, we will character-
ize their performance by experimental assessment. An example is presented in the next section.

4. An example of proportional analysis

Many different PLANETS can be created from CLEAR using the approach described in section 3.3. In order to investigate the concept further and to get preliminary assessment information, we created a PLANET to perform the Selection phase of CLEAR. We refer to this PLANET as the Automated Target Phrase Identification (ATPI) PLANET.

Recall that the Selection phase is responsible for identifying the targets upon which the later phases will operate. A critical property of target phrases is that they are the set of phrases whose understanding is essential for proper comprehension of the application being considered. By identifying the target phrases correctly and creating an appropriate knowledge basis including proper explications, we have a basis for accurate communication.

Inaccuracy in target-phrase selection would not necessarily be serious but it could be, and it would certainly be undesirable. Inaccuracy in this regard is difficult to determine, however, because the value of a specific phrase to a specific individual with a specific responsibility cannot be determined and varies between individuals. Thus determining the “correct” set of target phrases is impossible. What we can do is rest on the assumption that there are phrases that are generally better than others in facilitating effective communication, and that their identification is useful.

As an example of target-phrase identification and use, consider the spacecraft application mentioned earlier. Spacecraft navigation is a complex undertaking, and not an easy domain for engineers in other disciplines to comprehend. It is likely that application documentation for such a system would refer to a variety of coordinate systems, angles, distances, units, times, and numeric accuracies. The challenge with target-phrase identification is to find the complete set of target phrases about all of necessary concepts so that all are contained within the knowledge base.

If the ATPI PLANET is capable of identifying a useful set of targets in an efficient manner, it would provide a starting point for developing a CLEAR knowledge base. Our initial definition of “useful” was that the ATPI results enabled the creation of a knowledge base that provided value to a project. Because of the inevitable difference between the results that ATPI provides and those that CLEAR provides, we stress value to a project rather than similarity with CLEAR. Nevertheless, we used overlap with the results of CLEAR as an initial discriminator. Our definition of “efficient” was that the process be almost entirely automatic.

The ATPI PLANET was created using the approach described in section 3.3, and two versions were developed. The details of the PLANETS, their performance, and our interpretation of the results are presented in this section.

4.1. Creating the original PLANET

Starting with the CLEAR framework, we created the ATPI PLANET as follows. We created an automatic process of term selection based on the intuitive notion that the use of target phrases would have certain statistical characteristics. The CLEAR Selection phase contains an automatic element, and this motivated our

![Figure 3. PLA framework for CLEAR Selection phase](image-url)
automatic process. In CLEAR, however, selection also involves a variety of complementary manual elements. Of particular importance is the scoping element of CLEAR in which original source documents are examined manually for relevance and probably utility. We eliminated all of these manual elements in the ATPI PLAnET.

Although our goal with this PLAnET was just to achieve target phrase identification, it would be quite simple to develop a complete process that implemented all four CLEAR phases. For example, the Elicitation and Representation phases could be combined and reduced to a simple process in which project engineers develop explications in a totally traditional manner. Finally, CLEAR’s Integration phase could be reduced to merely documenting the explications in a sequential document.

We have hypothesized a number of statistical measures for term selection. The simplest and most obvious statistic is frequency of use, i.e., terms used frequently are likely to be targets. A second statistic is distribution of use, i.e., terms used across many documents are likely to be targets.

For the ATPI PLAnET we used the first statistic. To define an automated approach, we had to define two parameters (the parameter-setting elements of PLAnET creation). The first parameter is the set of phrase lengths that we expect candidate target phrases to have. We made that set one, two and three words.

The second parameter is the occurrence frequency cutoff point for candidate phrases, i.e., a frequency above which phrases are hypothesized to be possible targets. Clearly a phrase that was only used twice, for example, is unlikely to be a target phrase. We set this parameter at twenty, i.e., we considered any phrase occurring twenty times or more in the collection of scanned documents to be a candidate. This approach was used because actual frequencies of occurrence will be document-dependent. The cutoff point (twenty here) is set by looking at the entire list and assessing a cutoff point that does not include too many phrases nor phrases with an unacceptably low occurrence frequency.

Many phrases that occur frequently are not related to the application. Rather they are commonly-occurring phrases in English. We eliminated these from consideration using a list of known common phrases referred to as the *stop* list.

Other phrases occur that are related to the context. For example “University of Virginia” is a phrase that occurs frequently in the Walker Project documents but which is not relevant to requirements. We refer to such phrases as *structural*, and we eliminated them at various times by non-expert human judgement.

The final step in automatic target-phrase selection is to choose target phrases from the lists of candidates. For the ATPI PLAnET we used frequency of use as the discriminator, and we chose the fifteen most-frequently used phrases in each list as the proposed target phrases. If there were less than fifteen phrases in the list, we took the complete list.

4.2. Performance of the original PLAnET

The five original source documents that we obtained from the Walker Project were converted to plain text. For simplicity, no special consideration was given to figures, tables, or other embedded objects. The plain text materials were scanned using a prototype toolset that we developed based on Antconc [1] to locate phrases of interest. This generated lists of 306 one-word phrases, 110 two-word phrases, and 35 three-word phrases. Groups of words longer than three did not produce phrases with significant frequency.

These lists of phrases were filtered to remove any phrase that contained one of the words in the stop list. This reduced the target lists to lengths of 243, 37, and 14 respectively.

These lists were examined manually to remove obvious structural phrases, and then either the top fifteen phrases in each set or the complete set if the list contained less than fifteen entries were identified as the candidate target phrases. The resulting target phrases are shown in Table 1.

<table>
<thead>
<tr>
<th>walker</th>
<th>user</th>
</tr>
</thead>
<tbody>
<tr>
<td>control</td>
<td>system</td>
</tr>
<tr>
<td>support</td>
<td>project</td>
</tr>
<tr>
<td>model</td>
<td>data</td>
</tr>
<tr>
<td>intent</td>
<td>force</td>
</tr>
<tr>
<td>total</td>
<td>information</td>
</tr>
<tr>
<td>research</td>
<td>forces</td>
</tr>
<tr>
<td>motion</td>
<td></td>
</tr>
<tr>
<td>shared control</td>
<td>control system</td>
</tr>
<tr>
<td>user intent</td>
<td>cool aide</td>
</tr>
<tr>
<td>toe off</td>
<td>force moment</td>
</tr>
<tr>
<td>front wheel</td>
<td>navigational intent</td>
</tr>
<tr>
<td>obstacle avoidance</td>
<td></td>
</tr>
<tr>
<td>forces and moments</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: ATPI final phrase list

The comparison of the terms selected by the ATPI PLAnET and those selected by CLEAR is shown in Table 2. The comparison shows that the overlap between the lists was small and that the numbers of phrases of each length differed considerably.
Starting with the raw lists, we examined the results in detail to see what insights we might gain. We did not know at that point the significance or lack thereof of the phrases that had been identified. Analysis of these results led us to make significant revisions to the ATPI approach.

4.3. Creating the revised PLAnET

We made two changes to the PLAnET after seeing the results from the original version. The changes were to revise the selection statistic and to lower the cutoff points for candidate selection. We examine each of these in turn.

When we examined all the data, we found that many of the one-word phrases identified by ATPI were included only as part of longer phrases selected by CLEAR. For example, the phrase “walker” did not appear in CLEAR’s list of single-word phrases, but CLEAR did select the phrases “walker frame” and “walker intent”.

The distinction seen with phrases such as “walker” is exactly what one would expect with human insight operating in one case and not the other. The phrase “walker” is obviously important to this project yet it is so general that it is of very little value. With this in mind, we enhanced the statistical model of target phrase identification. In the enhanced statistical model, we hypothesize that very common single-word phrases are likely to be of little use in isolation because of their generality, but that their appearance in a multi-word phrase is likely to be very significant. Thus, starting with a set of candidate phrases chosen based on frequency, selection of multi-word target phrases might be improved if it were based on the presence of commonly occurring single-word phrases within the multi-word phrase.

Using this model, we modified the ATPI process for selecting two-word phrases to derive them from selected one-word phrases in the following way. In the revised ATPI PLAnET, one-word and three-word phrase selection was unchanged, and the original two-word selection was included also. A second set of two-word phrases, the derived two-word phrases, were selected by first determining a set of one-word root phrases. These were defined as the fifteen most frequently occurring one-word phrases. The set of two-word phrases was identified by locating the fifteen most frequently occurring two-word phrases that contained a root phrase either as the first or second word.

For the revised approach, we also increased the number of candidate phrases selected by lowering to ten the absolute occurrence frequency at which we assumed a phrase was a candidate. This was done because with the original cutoff (twenty), insufficient candidates were found to allow us to select the desired number of target phrases.

In summary, in the revised approach all source documents were scanned and all phrases of lengths one, two and three were identified. Phrases involving words from the stop list and involving words deemed structural were removed, and the lists were sorted based on occurrence frequencies. Fifteen additional two-word phrases were selected as the fifteen most commonly occurring two-word phrases that contained instances of the fifteen most-frequently-occurring single-word phrases either as the first or the second word.

4.4. Performance of the revised PLAnET

The derived two-word shared phrases selected by the revised ATPI PLAnET are shown in Table 3. The numbers are the ordering of the occurrence frequencies of the words. Five of the derived phrases were structural, and so they were removed. As might be expected, some two-word phrases were derived by multiple roots. For example, “control” derived “control system” and “system” also derived “control system.” This effect might be an indicator of the importance of the phrase, but we did not exploit the effect in the identification of target phrases. We plan to examine that possibility in the future.

In order to get an indication of the utility of the derived two-word phrase selection process, we asked a domain expert to assess the list in Table 3. The assessment was that phrases 1, 2, and 6 were extremely important, phrases 3, 5, and 10 were very important,

<table>
<thead>
<tr>
<th>Phrase</th>
<th>Intersection of Lists</th>
<th>ATPI Only</th>
<th>CLEAR Only</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 word</td>
<td>0</td>
<td>15</td>
<td>5</td>
</tr>
<tr>
<td>2 words</td>
<td>5</td>
<td>4</td>
<td>46</td>
</tr>
<tr>
<td>3 words</td>
<td>0</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>4+ words</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 2: ATPI vs. CLEAR comparison

<table>
<thead>
<tr>
<th>Phrase</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 word</td>
<td>1</td>
</tr>
<tr>
<td>2 words</td>
<td>2</td>
</tr>
<tr>
<td>3 words</td>
<td>3</td>
</tr>
<tr>
<td>4+ words</td>
<td>4+</td>
</tr>
</tbody>
</table>

Table 3: Two-word phrases derived from single words
phrases 4, 7, 8, 9, 11 and 12 are slightly important, and phrases 13, 14 and 15 were not important. This information is little more than anecdotal but the assessment does suggest the direction is valuable since five of the most-frequently-occurring six phrases were viewed as either extremely important or very important.

The second change that was made in the revised PLAnET was to lower the cutoff frequency for selection of target phrases for all three phrase lengths. In the original PLAnET, the cutoff was set to twenty, and in the revised version the cutoff was set to ten. This increased the number of candidate phrases that were selected.

Turning to the overlap with the lists by the revised PLAnET and the list generated by CLEAR, the details are contained in Table 4. Once again, the overlap with the list generated by CLEAR is small, and the effect of the two changes (revised statistic and lowered frequency) are similar.

4.5. Discussion

The fact that the PLAnET did not identify the same or highly similar sets of target phrases as CLEAR should not be surprising. CLEAR is a much more sophisticated process. The performance that we observed from the PLAnET is useful in several ways, specifically:

Critical Phase Identification. Four of the top six phrases that the revised PLAnET located were in the set of phrases that CLEAR selected. So, at least for this experiment, the PLAnET did identify correctly a significant number of the important target phrases. If this performance is confirmed by other experiments, it suggests that developing explications (even minimal ones) based on the top several phrases identified by the PLAnET would be valuable for the associated project. Selecting phrases that would not have been identified by CLEAR along with those that would does not mean a reduction in value because the additional phrases would inevitably have some value.

Overlap With Representation. CLEAR’s Representation phase generates additional important target phrases as a result of developing explications. Some of the phrases that the revised PLAnET identified were actually in the set that appeared in CLEAR’s Representation-phase results. Thus, the PLAnET is finding additional important material. However, when using the PLAnET one would not make the distinction between Selection and Representation (important though it is in the CLEAR model) because the goal of the PLAnET is merely to find target phrases as efficiently as possible.

Root Identification. The use of high-frequency single-word phrases (the root phrases) as a means of improving selection efficiency automatically is useful but not perfect. We noticed in the two-word phrases that the PLAnET identified, individual words that were clearly important roots that had not occurred with very high frequency. This information is beneficial in two ways. First, it could be used in a slightly extended PLAnET as input to a very short manual phase of root selection. This is a point at which a small effort involving expert human judgement might be very effective. Second, the PLAnET could be modified to use these one-word phrases rather than the high-frequency phrases that we have used.

There are some minor changes that need to be made to the approach we have been using to deal with relatively simple items. For example, in the detection of two-word phrases, “force and moment” and “forces and moments” were not conflated, and there are similar issues with tenses. These issues did not affect the basic results that we have observed.

5. Conclusion

In order to reduce misunderstandings in requirements engineering and thereby improve the quality of captured requirements, it is essential that everything possible be done to improve the accuracy of communication between the parties involved. This communication is in English, and by exploiting results in cognitive linguistics it is possible to both understand the sources of error and mitigate their effects.

CLEAR, a previously developed, comprehensive approach to facilitating human-to-human communication is quite resource intensive, and so we have developed an approach in which the benefits gained are proportional to the effort expended. For one element of the problem, namely target-phrase identification, we have developed a technique that is largely automated. We have assessed its performance by comparing it with
CLEAR and found that it provides useful performance with a mostly automatic and therefore inexpensive implementation. With these results, this new technique offers an approach to adoption of CLEAR technology that is low-risk and low-cost.

6. Acknowledgements

We thank Patrick Graydon for many helpful comments about this paper. This work was funded in part by NSF under contract number CCR-0205447.

7. References


