From free-text repair action messages to automated case generation

Benoit Farley

Institute for Information Technology, National Research Council of Canada
Montreal Road
Ottawa, Ontario, Canada K1A 0R6
Benoit.Farley@iit.nrc.ca

Abstract

Free-text problem description messages and repair action messages, which are written down in airplanes log books by pilots and technicians, contain very valuable information about problem symptoms, the pieces of equipment involved in the repair and the actions performed on them, and the results of those actions on the symptoms. This information could be used by a diagnostic system to provide suggestions about the nature of the problem and the proper repair actions. We use natural language analysis and interpretation techniques to extract that information from the messages with an aim at using it to generate automatically new cases for the case-base reasoning performed by the IDS system at Air Canada.

Introduction

IDS (Integrated Diagnostic System) is a diagnostic system developed at the National Research Council of Canada (NRC) and currently in operation at Air Canada as a test implementation. The system monitors every message transmitted from Airbus 320 airplanes to the ground to Air Canada's Aircraft Information Monitoring System (AIMS) - either malfunction reports from the airplanes computers or digital messages typed on keyboards by the pilots. It sorts them into clusters representing distinct probable problems and where possible, and associates with each cluster an explanation -or diagnosis - in terms of probable causes along with suggestions on repair actions based on rules extracted from the A320 troubleshooting manual (TSM). IDS is also fed with transcripts of the hand-written notes of the airplanes log books describing problems reported by the pilots and the repair actions carried out by the ground technicians. Each entry in the log books, called a "snag", describes a problem and its repair actions.

Case-base reasoning (CBR) is used within IDS to refine the troubleshooting manual diagnosis and repair suggestions made by IDS. The information in the incoming AIMS messages from the airplanes is matched against the symptoms of past problems, recorded as cases, in order to find problems that could be similar to the one at hand. If such similar problems are found, then their causes and their repair actions, in conjunction with IDS's own TSM diagnosis, can be used to identify the problem and fix it. To build the cases, domain experts analyse the free-text snags and extract information about the pieces of equipment involved in the repairs and which actions were performed on those pieces. The snag problem descriptions provide information which can be used by the experts to associate them with the airplane AIMS messages and to the IDS clusters.

The case creation process depends heavily on the available time that the experts can devote to that activity, and at present, the case-base reasoning operates with a limited number of about seventy manually created cases. As a first step toward the automation of the case creation process, we have been applying natural language processing techniques, that is, grammatical analysis and semantic interpretation, to the free-text repair action messages (FTRAM) of the snags in order to identify the pieces of equipment and the actions executed upon them, and the results of tests and measurements.

In the first part of the paper, we describe the linguistic characteristics and peculiarities of the FTRAMs. In the second part, the various aspects of the design of our system are exposed.

The messages

The style of the language of the FTRAMs is fairly telegraphic, with one or more "sentences" per message, a lot of acronyms and abbreviations, omission of certain types of words like the definite article, and a fair amount of mispellings and typographical errors.

Abbreviations and acronyms

Abbreviations and acronyms are used extensively. In a sample set of 884 messages, nearly 90% of them contain at least one abbreviation or acronym. They make for nearly 27% of all the words, with approximately the same number of abbreviations and acronyms.

Abbreviations. Abbreviations are shortened representations of words, and sometimes of groups of words. They can be found in nearly 70% of the messages and make for approximately 13% of all the words. Many are formed by keeping only the consonants of the abbreviated word, others by keeping only the first few letters which make the word recognizable in the mind of the author of the message. In either case, the typical period may or may not be

present at the end, but generally, its use is consistant throughout a given message, depending on the author; either it is used for all abbreviations, or it is not used at all for any. Another form of abbreviation is represented by the following example where the slash (/) is used to separate the initial consonant of the two most important syllables of a word: "S/V" stands for "SERVICEABLE". The following is an example of a message with 3 abbreviations:

B IGNITION LEAD CHGD.,CHKD. SERV.

with "chgd." standing for "changed", "chkd." for "checked", and "serv." for "serviceable". In the case of abbreviations of verbs in the past participle and past tense with an "ed" ending, one often finds the abbreviation of the stem of the verb separated from the abbreviation "D" of the past ending by a space or some other sign. For example, the word "changed" can be found in the following forms: CHG D, CHG'D, CHG"D, CHG;D.

A number of words, which have the form of an acronym but are not to be found as approved acronyms in any official manual, appear repeatedly in messages. Examples of such words are "R/H" and "L/H", for "RIGHT HAND" and "LEFT HAND" respectively, "S/N" for "SERIAL NUMBER", "P/N" for "PART NUMBER", "AC" and "A/C" for "AIR CANADA", etc.

Acronyms. An acronym is a sequence of letters consisting of the initial letters of a group of several words and is used as an abbreviation for the group. It reads as a word, and typically functions like a noun. Acronyms can be found in about 75% of the messages and make for about 14% of all the words. Some are acronyms approved in the aerospatial industy in general, but most of them are specific to Airbus (the original equipment manufacturer) and to Air Canada (the operator of that equipment). For instance, the Airbus General Introduction Manual contains a table of 459 acronyms. Most of the acronyms are single words with alphabetical characters only (e.g. FADEC – Full Authority Digital Engine Control). A small number of acronyms are made of 2 or 3 words (e.g. FLT CTL – Flight Control) and certain ones contain non-alphabetical characters (e.g. FLT-T.O – Flexible Take Off, F/O – First Officer) which are sometines thrown off. For example, "T/R", the acronym for "thrust reverse", is often written "TR". The following is an example of a message with 3 acronyms:

#1 EIU RESET–CFDS. FADEC TESTED – NIL FAULT.

An interesting point, as shown in the above example with the acronym "CFDS.", is that acronyms are very often considered by the authors of the messages like abbreviations and treated as such, with a period at the end. One can also find acronyms with a period after every letter, as in "E.C.U.", instead of "ECU", because each letter stands for a word and is thus considered an abbreviation in itself. Another interesting point is the pluralization of acronyms by appending the mark of the plural, "s", to the acronyms. For example, one can find "ANTS" which stands for the plural of "ANT", the acronym for "ANTENNA". Here again, as in the case of past endings, the distinctive ending is sometimes separated from the root word by some punctuation sign, as in "MCD,S".

Typographical errors and mispellings

Close to 17% of the messages contain typographical errors, mispellings, and other peculiar typographical usages.

Typographical errors. Typographical errors are accidents, most of the time related to the use of the keyboard. They include typing a wrong character which is just next to the right one (eg. "EMG" instead of "ENG", "3NG" instead of "ENG"), typing a key without the shift key (eg. "31" instead of "#1"), typing a character a second time (eg. "SSERV" instead of "SERV"), inserting a character by inadvertently touching another key nearby while typing in a character (eg. "SER5V" instead of "SERV", "CHBECKED" instead of "CHECKED"), inverting characters (eg. "VLAVE" and "VAVLE" instead of "VALVE"), skipping over a character or even a word ("# AND #21" – a number should follow the first #, "RESITANCE" instead of "RESISTANCE").

Mispellings. Mispellings are due to a deficient knowledge of the spelling of words. This may take the form of letters being used instead of the right ones because they are phonetically associated with the desired sound (eg. "IGNITER" instead of "IGNITOR", "EVEDENCE" instead of "EVIDENCE"). Or of single consonants being typed when they should be double (eg. "FLAGED" instead of "FLAGGED", "CHANEL" instead of "CHANNEL").

Others. Another type of error related to character input is that words which should normally be separated by some space character are sometimes glued together, thus making a long unknown word (eg. "CALLUPPAGE" for "CALL UP PAGE", "ENG1" for "ENG 1"). Or on the contrary, additional and unnecessary characters can be inserted within a word. This can be seen in acronyms, where the letters that make the acronym are separated by spaces (eg. "E C U" instead of "ECU"). One also can find words cut up into pieces with hyphens and even with slashes. Sometimes the sign will separate a prefix from the root word, as in "RE-CALL" or "BORO/SCOPE", but some other times, there is no obvious reason, as in "OPERATIO-NAL" and "FURTH-ER".

Sentences

Each message contains one or more individual propositions, or "sentences", and there is a great deal of inconsis-

tency in how a sentence is separated from the next one. The end of a sentence is generally indicated by way of some punctuation sign, like the period (.), the comma (,), the hypen (-), and the slash (/). But very often there is no such indication at all and the next sentence begins right away, with just a space between it and the previous one. In a single message, more than one type of end marker may be found, like in the following example:

> #1 EIU RESET – CFDS. FADEC TESTED – NIL FAULT. #1 THRUST REVERSER OPERATION NORMAL ON BOTH CHANNELS

In this example, the hyphen is used twice to separate sentences, and the period is used once, after "NIL FAULT". Semantically, this message consists of two main parts: "1) Something has been done; then a test has been carried out; no fault has been found. 2) Everything is normal". The two main parts are separated by a period; the subparts of the first part are separated by hyphens. The analysis of a great number of such messages with multiple end-of-sentence markers tends to demonstrate that one will use one marker, either a semi-colon or a comma or a slash, to separate sentences which are semantically related, and another marker, most often a period, to separate sentences not as much semantically related, much like the semi-colon and the period in ordinary text, as demonstrated by the following example:

> #2 FADEC (A & B) MOTORING TEST CARRIED OUT; NO FAULT. ENG. GROUND RUN CARRIED; UNABLE TO DUPLICATE. CONS SERV.

Punctuation signs

In the two previous examples above, the period is also used to end the acronyms "CFDS" and "ENG", as if they were abbreviations. In fact, all punctuation signs used throughout the messages have different purposes. The period is used at the end of a sentence, at the end of an abbreviation or of what is considered to be an abbreviation, like an acronym or even like each letter of an acronym, in numerical identifiers (eg. "FCOM 1.70.90"), after page or figure numbers (eg. "FIG 1. PAGE 0. DETAIL A."), in numbers (eg. "AT 6.1 N1"). The slash is often used as a conjunction (eg. "J11/J12 CONNECTION AT ..."), as a word-initial separator in an abbreviation (eg. "P/N" for "PART NUMBER"), even in unexpected places like the acronym for a maintenance manual (eg. "/M/M 28-11-00") or as a syllable-initial separator in an abbreviation (eg. "S/V" for "SERVICEABLE"). The hyphen, in addition to being used as an end-of-sentence marker, is also used within various types of identifiers: maintenance manual section, part and serial numbers (eg. "P/N 78-30-324"). One may also find it between two syllables of a word, as the usual hyphenation, just as we do when we write and we get to the end of the line. This has two main causes, not obvious at first glance: first, as at the Toronto airport, the snag messages are transcribed from paper to computer by clerks who may duplicate exactly everything they see on the piece of paper they were given, such as an actual hyphen at the end of a line; second, the computer interface with which the snag messages are entered, may make its user have to use hyphens, even perhaps force them automatically.

Grammatical structures

One important aspect of grammar is to describe the different basic elements of a language (words and signs), how the words are formed and how they are classified: nouns, genders, verbs, tenses, acronyms, abbreviations, etc. Another important role of grammar is to describe how these elements can be organized with respect to one another in order for an utterance to be comprehensible, thus understandable. An utterance may be viewed as a structured cluster of parts, each with its own structure, purpose and meaning. By parts, we mean associations of basic elements, and by structure, we mean how those parts are associated, such that one can retrieve the meaning of the structured utterance. Such meaning is normally about some action on some object by some agent in some circumstances, or about the state of something. The role of the grammar is to capture the recurrent patterns of those associations, the regularities which make the language predictable and which make possible to attach meaning to every part of an utterance.

In order to assess the grammatical features of the language used in the FTRAMs, each of the 884 messages of our sample set was manually broken into propositions. We have adopted the following definition of a proposition: the grammatically most complete structured utterance (sequence of words) describing a fact or a state, that is, that can be understood as a statement about some fact or state. Where punctuation was without any doubt used as a sentence divider, we have considered it as such, even though the proposition thus determined was not complete. Where there was no punctuation at all and where punctuation could not be considered, we based our decisions on several factors: correct grammatical usage as far as possible, particular recurrent patterns that we discovered throughout the messages, domain knowledge, etc.

The messages are written in English. But it is not "correct" English as we are taught in schools or as it is described in grammar books. In addition to the use of many abbreviations and acronyms and the presence of many typographical errors and mispellings, the syntax is also treated roughly. Missing words, bad agreement between words, different usages of verbs, are typical and account for certain proposition structures commonly found throughout the messages.

Noun phrase + Past participle verb phrase. The most common structure, making up about 34% of all the propositions, consists of a noun phrase, most often denoting a piece of equipment or some test, followed by a verb in the past participle, like the following examples:

BMC GROUND TESTS CARRIED OUT

B IGNITION LEAD CHANGED

LEAK CHECKED

NO FAULTS INDICATED

These could be analysed as complex noun phrases, that is, as simple noun phrases post-modified by adjectival phrases with past participles as adjectives; this is a perfectly acceptable structure for noun phrases, which can be confirmed in any grammar book. But normally a noun phrase alone does not make a proposition. It does not say anything by itself ; it only "mentions" some thing. And unless the mere mention of something suffices to render a fact, a noun phrase needs a verb to convey some fact about it. If we read the first example above as a noun phrase, it mentions some tests with the particularity of having been carried out, but it does not say what happened to those particular tests. Another possible reading, the proper one, is: "BMC GROUND TESTS *HAVE BEEN* CARRIED OUT". Now, this is a valid proposition with a noun phrase as subject and a verb in the passive mood which says something about what happened with regard to that noun, except that the auxiliary "have been" (the verb "to be") is missing in our message in front of the main verb.

Noun phrase + Adjective or Prepositional phrase. Another structure used in 7,5% of the total propositions is a noun phrase followed by an adjective, an adjectival phrase, or a prepositional phrase, as in:

#1 THRUST REVERSER OPERATION NORMAL ON BOTH CHANNELS

SYSTEM SERVICEABLE

FADEC TEST OK

EGT NOTED STILL WITHIN LIMITS

GROUND RUN ON #2 ENGINE WITH IGNITION B SYSTEM ON

Here again, the propositions must be read as if there were some verb between the noun and the other part, otherwise we would be left with complex noun phrases which only mention some objects with some particular aspects, without stating any fact. So, the appropriate reading should be: "SYSTEM *(HAS BEEN FOUND)* SERVICEABLE", "FADEC TEST *(IS, INDICATES)* OK", "EGT NOTED *(IS)* STILL WITHIN LIMITS". Those with an adjective are less ambiguous than in the case of a noun phrase followed by a past participle, because normally an adjective precedes the noun it modifies. The fact that it follows it suggests that it is predicative.

If we put together all the propositions which have a noun phrase as the subject of an absent predicative verb, followed by a predicative part, either past participle, adjectival or prepositional, these amount to 41,5% of all the prepositions.

Active verb + Noun or Adjective or Prepositional phrase. The next most common structure is a sentence with no subject noun phrase, a verb in the active mood with an object and possibly some adverbials. This structure accounts for 22% of the propositions. Here are a few examples:

REPLACED VBV SENSOR

CHECKED WIRING AT J-5

SUSPECT CAKED OIL CAUSING BAD READING

Of these, one out of five has the verb "check" or the verb "test" followed by an adjective, occasionally by a prepositional phrase, like the following example:

CHECKED SERVICEABLE

This proposition, very often preceded by other propositions stating the replacement of a piece of equipment and its verification by some test, is ambiguous because it may be read in two different ways. The first reading is in the passive voice and goes like this: "(THE PIECE OF EQUIPMENT *HAS BEEN*) CHECKED SERVICEABLE." The other one is in the active voice, and reads like this: "(THE PIECE OF EQUIPMENT *HAS*) CHECKED SERVICEABLE." The ambiguity is due to the form "ed" of the verb, which is common to both the past tense and the past participle, and lies only in the surface syntax, since both have the same meaning. The latter reading may be less grammatical, and one might decide then to prefer the first reading in the passive voice. But in several propositions, the verb is in the present tense with the "-s" mark, which indicates definitely the active voice:

CHECKS SERVICEABLE

#2 OIL INDICATION CHECKS NORMAL ON GROUND RUN

CFDS FAULTS T3 SENSOR. T3 SENSOR CHANGED. BITE CHECKS SERVICEABLE (bite: built-in test equipment)

CHANGED VBV SENSOR. OPERATIONAL TEST OF FADEC SYSTEM CHECKS SERVICEABLE (fadec: full authority digital electronic control)

So, even though this is not a quite correct usage of the verb, it is being used that way and we have to take it into account. Other indications, like the absence of punctuation in front of the verb in an otherwise punctuated message,

also favor the "active voice" reading, and we have thus set the "CHECKED SERVICEABLE" propositions into the active category.

Adjectival phrase. Propositions consisting of only an adjective or adjectival phrase are the next most common ones and account for a little more than 12% of the total propositions. The noun that it modifies may be stated in another proposition earlier in the same message, or not at all, which means that it is to be found in the problem description message associated with the repair action message. They must be read as if they were the predicative part of a full sentence with a noun phrase as subject and a predicative verb (be, indicates, etc). Examples of adjectival propositions follow:

BLEED VALVE AT 11 OCLOCK POSITION GRINDING.-REPLACED

CHANGED

DEFERRED

Noun phrase. It was said that noun phrases by themselves do not convey any other meaning than the simple mention of some object or thing. This is normally true. The phrase "the red apple" mentions an apple that is red, but nothing about what happens or happened or will happen to that apple. But in very specific and circumscribed contexts, the mere mention of something is sufficient to induce the author's intended meaning in the reader's mind. In our messages, we can find about 10% of simple noun phrases and other substantives with no predicative modifiers. A large proportion of them mention tests, and we may interpret them as if one had added "have been carried out". The others may be part numbers, message identifiers, etc., and must be interpreted with the help of the surrounding propositions. For example,

ENGINE GROUND RUN

ENGINE 1 FADEC CHANNEL FAULT

S/N DNDV7541

AUTH #110241

Negative noun phrase. Another type of noun phrase propositions, about 6% of the total, have the noun phrase negated with a negation particle in front of them, which can be "no" or "nil", and must be interpreted as if one had added "have been found" or "has been achieved" or something similar, depending on the context, as in:

NO FAULTS

NIL FIX

NO LAST LEG FAULTS ON ECAM

Prepositional phrase. A very small percentage of the propositions, 0,5%, are prepositional phrases and like adjectival phrases, they will have to be interpreted with the help of the surrounding propositions:

WITHIN LIMITS

Complete sentence. Finally, we have found about 8% of all the propositions with a full sentence structure, that is, with a noun phrase subject, a verb and possibly complements, as in the following examples:

OPERATIONS CHECKED SERVICEABLE

THRUST REVERSER TEST FAULTS HCU

ENGINE FAILED

RESISTANCE CHECKED AS PER STEPS ARE IN SPECIFIED LIMITS

Table 1. summarizes the analysis of the structures of the propositions in our sample set. It indicates for each type of propositional structure the percentage of all the propositions exhibiting that structure.

Structure	% of total propositions
Noun phrase + Passive verb phrase	33,94%
Noun phrase + Adjectival phrase	5,74%
Noun phrase + Prepositional phrase	1,8%
Active verb + noun/prepositional/adjectival phrase	21,96%
Adjectival phrase	12,11%
Simple noun phrase / Substantive	9,85%
Negative noun phrase	6.2%
Prepositional phrases	0,5%
Noun phrase + Active verb phrase (complete proposition)	7,87%

Table 1. Proportional distribution of the propositional structures.

Most examples show another particularity of our propositions: the systematic absence of articles, be they definite or indefinite, for the sake of brevity and also because they are not necessary in this context for a proper understanding. This is not as problematic as the absence of other types of words, like prepositions. Prepositions are used normally to convey information about some particular circumstance under which an action takes place. It may indicate a time, a location, a manner, etc. A preposition is always followed by a noun phrase. It serves several purposes: first, it indicates the beginning of the circumstantial complement, it separates the modifier from the modified, and it links them together; second, it specifies the meaning of the link, the circumstance. If the preposition is missing, like in the two

following examples, it is more difficult to determine what the parts should be, and whether and how they connect. We will have to rely on some other strategies and information to palliate the absence of the preposition.

ENGINE GROUND RUN MINIMUM IDLE SPEED

In this example, we have two noun phrases which should be separated by the prepositions "at": "ENGINE GROUND RUN *(AT)* MINIMUM IDLE SPEED".

CFDS. TEST CARRIED OUT #2 A/B CHANNELS

In this second example, "on" is missing in front of "#2 A/B CHANNELS". This brings ambiguity: syntactically speaking, the message could be either one proposition with "carried out" as an active verb, "cfds test" as its subject and "#2 a/b channels" as its object, or two propositions each a noun phrase: "cfds test carried out" and "#2 a/b channels".

The processing

Each FTRAM is first preprocessed in order to determine the nature and properties of each word and token. Then the sequence of morphologically analysed items is syntactically analysed with a parser and checked against a grammar which describes the patterns of valid propositions. Finally, the result of the syntactic parsing is semantically interpreted. This is where we recognize information about pieces of equipment, actions upon them, tests carried out and results. The next sections explain in more details the diverse components of our processing system.

Dictionary and acronyms database

In order to be able to make any sense of what we read in any given domain, and to make wise decisions, one must know the words. If a word is unknown to the reader, how can that word trigger any mental image and domain knowledge necessary for a proper understanding? As the reader, given its goal of retrieving information about pieces of equipment and the actions upon them, our system must know the words that define them and the other tokens used in the messages, and this means a lot of them, several thousand words likely. In fact, in a total of 62523 messages, 35506 different tokens were found, and after removing numbers, single letters, prepositions, acronyms and abbreviations to keep only nouns, adjectives and verbs, there remained approximately 15000 different words, ranging from common nouns and verbs like "go", "do", "find", "noise", etc. to domain-specific words like "valve", "actuator", "engine", and son on. If one wants to retrieve information about pieces of equipment and the actions upon them, one must know the words that define them.

Our system uses a substantial English lexicon (Boguraev et al. 1987, Carroll and Grover 1989), derived semi-automatically with little manual intervention from the machine-readable version of a conventional learners' dictionary, and developed within the Alvey Natural Language Tools (ANLT) system (Grover, Carroll, and Briscoe 1993). It contains 28240 noun entries, 23273 verb entries, 9073 adjective entries, 1069 adverb entries, and 1008 entries for prepositions, modal and auxiliary verbs, and various function words. A word may have several entries, one for each of its typical usages.

A certain number of common recurrent abbreviations have been recorded into a lexicon as well. During our analysis of the messages, we have learned to recognize those forms as abbreviations, and we believed our system could be given that basic initial knowledge.

The acronyms are defined in a database which our system can access. It includes acronyms which are common to the aerospatial industry in general, Airbus acronyms, Air Canada acronyms, and IDS acronyms. Each acronym is defined along with its corresponding expression.

Preprocessing and morphological analysis

The main objective of this pass is to determine what each word is exactly or might be. Words may have different morphologies, depending on properties like grammatical number and gender for nouns, or person and tense for verbs. The morphological analysis allows to retrieve the root of the word and tell its grammatical features. Among other advantages, it helps keeping the dictionary smaller, because only the root form needs to be cited. In many cases, particularly in the English language, a word may have multiple personalities. For example, the word "tests" may be the noun "test" in the plural and the verb "test" in the third singular person of the present tense. The word "checked" may be the verb "check" in the past participle and the same verb "check" in the past tense.

If a word cannot be determined as a form of a root word in the lexicon, one checks in the acronyms database. Our analysis of the sample messages showed us that acronyms are often treated like nouns in the sense that they are often put in the plural by the addition of an "s" at the end, and sometimes with an apostrophe ('s) or a comma (,s): "vbvs" for "variable bleed valves". So, some form of morphological analysis is necessary too.

If a word cannot be determined either as a word in the dictionary or as an acronym, there is the possibility of its being an abbreviation. In our sample messages, we have found abbreviations of not only root words but also of plural nouns and conjugated verbs, like "chks" for "checks" and "chgd" for "changed". As in the case of the acronyms, the plural ending and the conjugation ending are often separated from the stem by some punctuation sign, and even by a space: "repl d" for "replaced. Abbreviations are used because one wants to reduce time and/or space and/or effort, but not at the expense of loosing understandability. It is thus assumed by the author that the reader will make

for it, either because the word is quite recognizable, or because the abbreviation, sometimes reduced to a couple of letters, has been used so commonly that is has become practically a word by itself, known and recognized on its own, idiomatic. In the latter case, it may be made part of the lexicon, standing similar to a synonym, as part of the necessary knowledge required to understand, as expected. In the former case, and also because the dictionary cannot possibly contain all of them, we have to rely on analysis. We have been looking for different strategies for identifying among the dictionary entries the actual word for which an abbreviation has been used, based on the analysis of a large number of abbreviations. For instance, the first letter of the word is always present. This reduces the search space only to the words beginning with that letter. Then we may check whether the abbreviation corresponds to some initial fragment of a word. If not, we may check whether we can find among the retained dictionary words one with a similar sequence of consonnants after removing the vowels and the typical noun and verb endings like "s", "d", etc. Double consonants are reduced to a single one, and composite consonants like "ch" are also reduced to "c", or "ck" to "k". These tests allow us for example to retrieve "check" from "ck", "chk", and "changed" from "chgd" or even "cgd". The level of confidence of having retrieved the right word is difficult to assess, and may depend on the number of retrieved candidates. In the case of multiple candidates, the determinant test will be at the parsing stage when trying to associate words, or later at the interpretation stage when one tries to make sense out of the syntactical associations made by the parser. An initial attempt has been made to add to the abbreviations lexicon any abbreviation that has been computed into a word and that contributed to a successful interpretation of the proposition which contains it.

In the event of a word not being determined either as a word in the lexicon, as an acronym, or as an abbreviation, we should consider the possibility of a mistyping or mispelling. This problem will be looked after in a second development phase. For the time being, at this stage, the proposition is rejected as unparsable. The message that contains it is not rejected though, because its other propositions may be parsable and information might be obtainable from them. In a second phase of development, we are thinking about keeping a record of the rejected propositions along with the unrecognized words, for later examination by the system manager who could then add the word to the lexicon and have the message analysed again.

Grammar and parsing

The structures of the messages, described in the preceding section, have been recorded into a set of grammar rules. These rules state what a message is composed of, what a valid proposition may be composed of, what the constituants of that proposition may be composed of, and so on down to the words, numbers and other basic tokens.

The first set of rules describe the composition of the message itself, holding for multiple- or single-proposition messages with or without end markers:

message : proposition + end_mark + message

message : proposition + message

message : proposition + end_mark

message : proposition

The next set of rules describe the composition of the propositions. For example, the rule which describes the most common proposition structure, as shown in the example "#1 EIU RESET", looks like this:

proposition : noun_phrase + passive_verb_phrase

In its turn, the noun phrase structure is described by a set of rules each describing a different combination. In the above example, the combination of the noun phrase "#1 EIU" is described by the following rules:

a) noun_phrase : identifier + non_ident_noun_phrase

b) noun_phrase : non_ident_noun_phrase

c) non_ident_noun_phrase : acronym

d) identifier : number_sign + identifier_token

e) number_sign : '#'

f) identifier_token : number

'#' and '1' form an identifier by rule d; 'EIU' is an acronym, which can make for a non idenfier noun phrase by rule c; the identifier '#1' and the non identifier noun phrase 'EIU' form a noun phrase by rule a. Similarly, a set of rules describes what a passive verb phrase may be, the simplest of which states that it may be a simple verb in the past participle, like "RESET". So, put together, the noun phrase "#1 EIU" and the passive verb phrase "RESET" form a proposition.

Our grammar is based on the Alvey grammar. The ANLT sentence grammar is written in a formalism similar to that of Generalized Phrase Structure Grammar (GPSG) (Gazdar et al. 1985), details of which are discussed in Boguraev et al. (1988), Briscoe et al. (1987a, 1987b) and Carroll et al. (1991). Most structures found in the messages are covered by this grammar. Rules for the structures with a missing auxiliary verb or predicative verb have been derived from existing ANLT rules. New rules had to be formulated to handle certain idiomatic expressions like references in a troubleshooting or maintenance manual, piece description with serial and part numbers, etc. We also had to modify the rules as to make them less strict

with regard to subject-verb agreement and also to allow for noun phrases with no article.

The biggest problem that we are facing is due to the lack of punctuation. In a message with no punctuation signs, the grammar, from a strictly syntactical point of view, may return more than one parse, that is, several different word associations and propositions. Let us look at an example:

CHANGED START VALVE OPERATION CHECKED OK

If there were a period after the word "valve", which is where in truth there should be one, there would be only two analysis, shown below with the noun phrases underlined:

1) changed <u>start valve</u>
   <u>operation</u> checked ok
2) changed <u>start valve</u>
   <u>operation checked ok</u>

The second parse is generated because a rule says that a proposition may be a single noun phrase, and another rule says that a noun phrase may be a noun followed by a verb in the past participle.

But since there is no punctuation to tell the parser where to stop, it will return all the possible parses including as many words as possible. Here is what the parser would returns in that case, with the noun phrases underlined:

1) changed <u>start valve operation checked ok</u>
2) changed <u>start valve operation</u>
   checked ok
3) changed <u>start valve</u>
   <u>operation</u> checked ok
4) changed <u>start valve</u>
   <u>operation checked ok</u>
5) changed <u>start</u>
   <u>valve operation</u> checked ok
6) changed <u>start</u>
   <u>valve operation checked ok</u>
7) changed
   <u>start valve operation</u> checked ok
8) changed
   <u>start valve operation checked ok</u>

The order in which the parses are generated depends on how the grammar rules have been written, and there is no guarantee that the first one will be the right one. For instance, in the above list, the right one is the third one.

Each successful grammatical parse is associated with a semantic expression which represents unambiguously the meaning of the input proposition. In the Alvey formalism, the semantic expressions are computed in the form of logical forms by means of rule-to-rule translations expressed in the lambda calculus (Dowty, Wall, and Peters 1981). Every grammar rule generates a logical form which is the combination of the logical forms of its constituents. The resulting logical forms provide an event-based (Davidson 1967, Parsons 1990) and unscoped (e.g. Alshawi, 1992) representation of the compositional semantics of the input sentence. As an example, here is the expression generated for the parse "changed start valve":

```
(DECL
  (some (x1) (and (sg x1) (person x1))
    (some (x2) (and (sg x2) (START-VALVE x2))
      (some (e1) (PAST e1) (CHANGE e1 x1 x2)))))
```

It says that it is a declarative statement to the effect that there is some person, and there is some start valve, and there is some past event, and the event is that of the person changing the start valve.

Semantic interpretation

After the parser has analysed a message and returned a list of grammatically valid parses, each parse is semantically interpreted, through its associated lambda-calculus logical form, with the ultimate goal of retrieving information about pieces of equipment and actions upon them, and verifications carried out and their results.

But this is also where one determines whether some word association proposed by the parser makes any sense, and where one rejects the parses which do not make sense. Let us look again at the message "CHANGED START VALVE OPERATION CHECKED OK". One of its parses, "changed <u>start valve operation</u>", will see the word "changed", which is described in the domain knowledge representation as an action to be done on a piece of equipment, interpreted as such over the word "operation", which is not classified as a piece of equipment. This parse will then be rejected on that basis, and the next candidate parse will be tried for interpretation. The interpretation of "changed <u>start valve</u>" will succeed because "valve" is classified as a piece of equipment and "changed" can now be interpreted successfully.

A proposition like "changed start valve" can be interpreted easily on its own, even though there is no subject for the verb, since it may be assumed at the interpretation level that it was done by the technician who wrote the FTRAM or some other technician, to the extent that it matters at all. The important information, that is, the piece of equipment and what was done to it, is there. But, as it was shown before, a number of propositions consist of only a noun phrase, a substantive, like "AUTH #110241", or an adjective, like "CHANGED", and these cannot be interpreted on their own, at least, not fully. They need to be associated with the interpretation of the surrounding propositions in the same message, or in the context of the information retrieved eventually from the problem description messages. For example, in the message:

REQS 80-10-401 – NIL YVR

the interpretation of the second proposition "NIL YVR" needs the information from the interpretation of the first proposition in order to understand that the piece referred to is not available at the Vancouver airport.

What is missing in the propositions in terms of words and syntactical linking operators such as prepositions for example, must be compensated with domain knowledge in order to associate correctly the so-unlinked parts of speech. We must assume that the authors of the messages, as far as they were concerned, did express something that was supposed to make sense. And it effectively makes sense if you assume, while writing it, that the people who will read it know the domain and the context and that the mere mention of a given "thing", a test for example, will generate in the reader's mind what the author actually intended to say. This suggests that our system, just like the people to whom the messages are addressed, must have a sufficient knowledge of the domain in order to make the appropriate semantic associations and derive the intended meaning of the messages.

The domain of the FTRAMs must therefore be represented. Basically, this means that each word of the dictionary likely to be encountered in a message must be semantically classified. For example, a number of nouns represent tests, and so a "test" class is created into the knowledge base with every test noun put under it. Other nouns represent pieces of equipment, and so a "piece of equipment" class is also created. The same thing happens with verbs. Some verbs denote a replacement action, and so, there is a "replace" action class, and so on. The task of identifying the class for each dictionary word, despite the limited number of classes, is an arduous one, due to the number of words involved. A great many words may though be treated automatically. Effectively, all the pieces of equipment mentioned in the troubleshooting manuals and the maintenance manuals are listed in a database for the IDS project, along with every useful information, such as serial numbers, part numbers, manufacturer, etc. The words in the dictionary can thus be checked against that list in the piece database. A similar database exists for tasks, tests, and testing equipment. Here follow as an example two entries in our knowledge base:

```
(make-sense START-VALVE piece-of-equipment)
(make-sense CHANGE replacement-action)
```

The format of these entries corresponds to the format used by our semantic interpretation module. In the course of a research project on natural language understanding in a 3D visual environment at the National Research Council of Canada, called Spoke'n'Image (Smith, Farley, and Ó Nualláin 1996), we have developed a semantic interpretor for Alvey lambda-calculus logical forms. Written in CLOS, it consists in a set of methods, special functions each of which is specialized in the interpretation of a specific class of objects, class of actions, class of grammatical operators, etc., representing the domain knowledge.

Let us return to the proposition "changed start valve". We repeat here the lambda-calculus expression generated during its parsing:

```
(DECL
  (some (x1) (and (sg x1) (person x1))
    (some (x2) (and (sg x2) (START-VALVE x2))
      (some (e1) (PAST e1) (CHANGE e1 x1 x2)))))
```

The first element of each paranthesized expression is a symbol associated with a "sense" class in the knowledge base. When interpreting that logical form, the subform (START-VALVE x2) is thus interpreted by a special method for pieces of equipment, such as the following:

```
(defmethod lexical-interpretation
  ((sense piece-of-equipment) bindings var other-vars)
```

This method sets the concept property of the variable x2 to 'start-valve'. Then a little later, the subform (PAST e1) is interpreted and sets the time property of the event variable e1 to 'past'. Finally, the subform (CHANGE e1 x1 x2) is interpreted by a special method for replacement actions, such as the following:

```
(defmethod lexical-interpretation
  ((sense replacement-action) bindings event-var others)
```

This method expects and verifies that the concept of the variable x2, passed in the argument others, is a piece-of-equipment. If that verification fails, then the interpretation fails as well, and the proposition is rejected. If it succeeds, then the interpretation continues and checks for the time of the event. A 'past' attribute indicates that the piece was actually changed; the interpretation succeeds and results in the creation of a structure 'PIECE-ACTION' instanced in the case of our example to (START-VALVE REPLACEMENT-ACTION). A 'future' or 'conditional' attribute would indicate that the piece has not been changed, and so, the interpretation would succeed but without resulting in the creation of a 'PIECE-ACTION' structure.

Conclusion

We have presented in this paper how we are proceeding in the task of extracting specific information, namely pieces of equipment, actions upon them, and results of those actions, from free-text repair action messages describing what was done to fix problems that occurred on an airplane. That information, along with the information eventually extracted from the associated problem description

messages, will be used to generate automatically cases that will support a case-based reasoning module in IDS. Our approach, based on a grammatical analysis of the messages and their semantic interpretation, calls upon natural language processing and understanding techniques.

We have first given a thourough description of the lexical and syntactical characteristics of the language used in the messages. We have then described the different steps and corresponding parts of our system. Most parts have been or are being implemented. Their integration has not been achieved yet, so that an assessment of the degree of efficiency of our approach with regard to our objectives cannot be made at this moment. Yet preliminary tests on the grammatical analysis of scores of messages with made-up dictionaries of acronyms and abbreviations have proven our grammar rules and our parser to generate the adequate parses with their corresponding semantic expressions. A very basic knowledge base with a few semantic entries and corresponding interpretation methods has demonstrated the adequacy of our approach.

Reference

Alshawi, H. ed. 1992. *The Core Language Engine*. MIT Press. Cambridge, MA.

Boguraev, B., Briscoe, E., Carroll, J., Carter, D. and Grover, C. 1987. The derivation of a grammatically indexed lexicon from the Longman Dictionary of Contemporary English. In Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics, 193-200. Stanford, California.

Boguraev, B., Carroll J., Briscoe E., and Grover C. 1988. Software Support for Practical Grammar Development. In Proceedings of 12th International Conference of Computational Linguistics, 54-58. Budapest, Hungary.

Briscoe, E., Grover C., Boguraev B., and Carroll J. 1987a. Feature Defaults, Propagation and Reentrancy. In Klein, E. and van Benthem, J. eds. Categories, Polymorphism and Unification, 19-34. Centre for Cognitive Science, University of Edinburgh.

Briscoe, E., Grover C., Boguraev B., and Carroll J. 1987b. A Formalism and Environment for the Development of a Large Grammar of English. In Proceedings of the 10th International Joint Conference on Artificial Intelligence, 703-708. Milan, Italy.

Carroll, J., and Grover C. 1989. The Derivation of a Large Computational Lexicon for English from LDOCE. In Boguraev, B., and Briscoe E. eds. Computational Lexicography for Natural Language Processing. Longman, London.

Carroll, J., Briscoe E., and Grover C. 1991. A Development Environment for Large Natural Language Grammars, Technical Report No. 233, Computer Laboratory, University of Cambridge.

Davidson, D. 1967. The Logical Form of Action Sentences. In Rescher N. ed. The Logic of Decision and Action. Univeristy of Pittsburgh Press, Pittsburgh.

Dowty, D., Wall R., and Peters S. 1981. *Introduction to Montague Semantics*. Reidel, Dordrecht.

Gazdar, G., Klein E., Pullum G., and Sag I. 1985. *Generalized Phrase Structure Grammar*. Blackwell, Oxford.

Grover, C., Carroll J., and Briscoe E. 1993. The Alvey Natural Language Tools grammar (4th release), Technical Report 284, Computer Laboratory, Cambridge University, UK.

Parsons, T. 1990. *Events in the Semantics of English: a study in subatomic semantics*. MIT Press. Cambridge, MA.

Smith, A.G., Farley, B., and Ó Nualláin, S. 1996. Visualized models for language understanding. In Proceedings of European Conference on Artificial Intelligence, Workshop on Processes and Representations between Vision and Natural Language, 43-50. Budaperst, Hungary.