SRI INTERNATIONAL: DESCRIPTION OF THE FASTUS SYSTEM USED FOR MUC-4

Jerry R. Hobbs, Douglas Appelt, Mabry Tyson, John Bear, and David Israel
SRI International
Menlo Park, California 94025
hobbs@ai.sri.com
(415) 859-2229

INTRODUCTION

FASTUS is a (slightly permuted) acronym for Finite State Automaton Text Understanding System. It is a system for extracting information from free text in English, and potentially other languages as well, for entry into a database, and potentially for other applications. It works essentially as a cascaded, nondeterministic finite state automaton.

It is an *information extraction* system, rather than a *text understanding* system. This distinction is important. In information extraction, only a fraction of the text is relevant. In the case of the MUC-4 terrorist reports, probably only about 10% of the text is relevant. There is a pre-defined, relatively simple, rigid target representation that the information is mapped into. The subtle nuances of meaning and the writer's goals in writing the text are of no interest. This contrasts with text understanding, where the aim is to make sense of the entire text, where the target representation must accommodate the full complexities of language, and where we want to recognize the nuances of meaning and the writer's goals.

The MUC evaluations are information extraction tasks, not text understanding tasks. The TACITUS system that was used for MUC-3 in 1991 is a text-understanding system [1]. Using it for the information extraction task gave us a high precision, the highest of any of the sites. However, our recall was mediocre, and the system was extremely slow. Our motivation in building the FASTUS system was to have a system that was more appropriate to the information extraction task.

The inspiration for FASTUS was threefold. First, we were struck by the strong performance that the group at the University of Massachusetts got out of a fairly simple system [2]. It was clear they were not doing anything like the depth of preprocessing, syntactic analysis, or pragmatics that was being done by the systems at SRI, General Electric, or New York University. They were not doing a lot of processing. They were doing the *right* processing.

The second source of inspiration was Pereira's work on finite-state approximations of grammars [3], especially the speed of the implementation.

Speed was the third source. It was simply too embarassing to have to report at the MUC-3 conference that it took TACITUS 36 hours to process 100 messages. FASTUS has brought that time down to 11 minutes.

The operation of FASTUS is comprised of four steps, described in the next four sections.

1. Triggering
2. Recognizing Phrases
3. Recognizing Patterns
4. Merging Incidents

The system is implemented in CommonLisp and runs on both Suns and Symbolics machines.

TRIGGERING

In the first pass over a sentence, trigger words are searched for. There is at least one trigger word for each pattern of interest that has been defined. Generally, these are the least frequent words required by the pattern. For example, in the pattern

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
1992		**00-00-1992 to 00-00-1992**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
SRI International: Description of the Fastus System Used for MUC-4	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **SRI International,333 Ravenswood Avenue,Menlo Park,CA,94025**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**8**	

take <HumanTarget> hostage

"hostage" rather than "take" is the trigger word. There are at present 253 trigger words.

In addition, the names of people identified in previous sentences as victims are also treated, for the remainder of the text, as trigger words. This allows us, for example, to pick up occupations of victims when they occur in sentences with no other triggers, as in

> Hector Oqueli and Gilda Flores were assassinated yesterday.
>
> Gilda Flores was a member of the Democratic Socialist Party (PSD) of Guatemala.

Finally, on this pass, full names are searched for, so that subsequent references to surnames can be linked to the corresponding full names. Thus, if one sentence refers to "Ricardo Alfonso Castellar" but does not mention his kidnapping, while the next sentence mentions the kidnapping but only uses his surname, we can enter Castellar's full name into the template.

In Message 48 of TST2, 21 of 30 sentences were triggered in this fashion. 13 of the 21 triggered sentences were relevant. There is very little penalty for passing irrelevant sentences on to further processing since the system is so fast, especially on irrelevant sentences.

Eight of the nine nontriggered sentences were irrelevant. The one relevant, nontriggered sentence was

> There were seven children, including four of the vice president's children, in the home at the time.

It does not help to recognize this sentence as relevant as we do not have a pattern that would match it.

The missing pattern is

<HumanTarget> be in <PhysicalTarget>

which would pick up human targets who were in known physical targets. In order to have this sentence triggered, we would have to take the head nouns of known physical targets to be temporary triggers for the remainder of the text, as we do with named human targets.

RECOGNIZING PHRASES

The problem of syntactic ambiguity is AI-complete. That is, we will not have systems that reliably parse English sentences correctly until we have encoded much of the real-world knowledge that people bring to bear in their language comprehension. For example, noun phrases cannot be reliably identified because of the prepositional phrase attachment problem. However, certain syntactic constructs can be reliably identified. One of these is the noun group, that is, the noun phrase up to the head noun. Another is what we are calling the "verb group", that is, the verb together with its auxilliaries and embedded adverbs. Moreover, an analysis that identifies these elements gives us exactly the units we most need for recognizing patterns of interest.

Pass Two in FASTUS identifies noun groups, verb groups, and several critical word classes, including prepositions, conjunctions, relative pronouns, and the words "ago" and "that". Phrases that are subsumed by larger phrases are discarded. Overlapping phrases are rare, but where they occur they are kept. This sometimes compensates for incorrect analysis in Pass Two.

Noun groups are recognized by a 37-state nondeterministic finite state automaton. This encompasses most of the complexity that can occur in English noun groups, including numbers, numerical modifiers like "approximately", other quantifiers and determiners, participals in adjectival position, comparative and superlative adjectives, conjoined adjectives, and arbitrary orderings and conjunctions of prenominal nouns and noun-like adjectives. Thus, among the noun groups recognized are

> approximately 5 kg
> more than 30 peasants
> the newly elected president
> the largest leftist political force
> a government and military reaction

Verb groups are recognized by an 18-state nondeterministic finite state machine. They are tagged as Active, Passive, Gerund, and Infinitive. Verbs that are locally ambiguous between active and passive senses, as the verb "kidnapped" the the two sentences,

Several men kidnapped the mayor today.
Several men kidnapped yesterday were released today.

are tagged as Active/Passive and Pass Three resolves the ambiguity if necessary.

Certain relevant predicate adjectives, such as "dead" and "responsible", are recognized, as are certain adverbs, such as "apparently" in "apparently by". However, most adverbs and predicate adjectives and many other classes of words are ignored altogether. Unknown words are ignored unless they occur in a context that could indicate they are surnames.

Lexical information is read at compile time, and a hash table associating words with their transitions in the finite-state machines is constructed. There is a hash table entry for every morphological variant of the words. Altogether there are 43,000 words in the hash table. During the actual running of the system on the texts, only the state transitions are accessed.

The output of the second pass for the first sentence of Message 48 of TST2 is as follows:

Noun Group:	Salvadoran President-elect
Name:	Alfredo Cristiani
Verb Group:	condemned
Noun Group:	the terrorist
Verb Group:	killing
Preposition:	of
Noun Group:	Attorney General
Name:	Roberto Garcia Alvarado
Conjunction:	and
Verb Group:	accused
Noun Group:	the Farabundo Marti National Liberation Front (FMLN)
Preposition:	of
Noun Group:	the crime

The verb groups "condemned" and "accused" are labelled "Active/Passive". The word "killing" which was incorrectly identified as a verb group is labelled as a Gerund. This mistake is common enough that we have implemented patterns to get around it in Pass Three.

On Message 48 of TST2, 243 of 252 phrases, or 96.4%, were correctly recognized. Of the 9 mistakes, 5 were due to nouns being misidentified as verbs or verbs as nouns. 3 were due to a dumb bug in the code for recognizing dates that crept into the system a day before the official run and meant that no explicit dates were recognized except in the header. (This resulted in the loss of 1% in recall in the official run of TST3.) One mistake was due to bit rot.

We implemented and considered using a part-of-speech tagger to help in this phase, but there was no clear improvement and it would have doubled the time the system took to process a message.

RECOGNIZING PATTERNS

The input to the third pass of FASTUS is a list of phrases in the order in which they occur. Anything that is not included in a phrase in the second pass is ignored in the third pass. The state transitions are driven off the head words in the phrases. In addition, some nonhead words can trigger state transitions. For example, "bomb blast" is recognized as a bombing.

We implemented 95 patterns for the MUC-4 application. Among the patterns are the following ones that are relevant to Message 48 of TST2:

killing of <HumanTarget>
<GovtOfficial> accused <PerpOrg>

bomb was placed by <Perp> on <PhysicalTarget>
<Perp> attacked <HumanTarget>'s <PhysicalTarget> with <Device>
<HumanTarget> was injured
<HumanTarget>'s body

As patterns are recognized, incident structures are built up. For example, the sentence

Guerrillas attacked Merino's home in San Salvador 5 days ago with explosives.

matches the pattern

<Perp> attacked <HumanTarget>'s <PhysicalTarget> in <Location>
<Date> with <Device>

This causes the following incident to be constructed.

Incident:	ATTACK/BOMBING
Date:	14 Apr 89
Location:	El Salvador: San Salvador
Instr:	"explosives"
Perp:	"guerrillas"
PTarg:	"Merino's home"
HTarg:	"Merino"

The incident type is an attack or a bombing, depending on the Device. There was a bug in this pattern that caused the system to miss picking up the explosives as the instrument. In addition, it is disputable whether Merino should be listed as a human target. In the official key template for this message, he is not. But it seems to us that if someone's home is attacked, it is an attack on him.

A certain amount of pseudo-syntax is done while patterns are being recognized. In the first place, the material between the end of the subject noun group and the main verb group must be read over. There are patterns to accomplish this. Two of them are as follows:

Subject {Preposition NounGroup}* VerbGroup

Subject Relpro {NounGroup | Other}* VerbGroup {NounGroup | Other}* VerbGroup

The first of these patterns reads over prepositional phrases. The second over relative clauses. The verb group at the end of these patterns takes the subject noun group as its subject. There is another pattern for capturing the content encoded in relative clauses:

Subject Relpro {NounGroup | Other}* VerbGroup

Since the finite-state mechanism is nondeterministic, the full content can be extracted from the sentence

The mayor, who was kidnapped yesterday, was found dead today.

One branch discovers the incident encoded in the relative clause. Another branch marks time through the relative clause and then discovers the incident in the main clause. These incidents are then merged.

A similar device is used for conjoined verb phrases. The pattern

Subject VerbGroup {NounGroup | Other}* Conjunction VerbGroup

allows the machine to nondeterministically skip over the first conjunct and associate the subject with the verb group in the second conjunct. Thus, in the sentence

Salvadoran President-elect Alfredo Cristiani condemned the terrorist killing of Attorney General Roberto Garcia Alvarado and accused the Farabundo Marti National Liberation Front (FMLN) of the crime.

one branch will recognize the killing of Garcia and another the fact that Cristiani accused the FMLN.

The second sort of "pseudo-syntax" that is done while recognizing patterns is attaching genitives, "of" complements, and appositives to their heads, and recognizing noun group conjunctions. Thus, in

> seven children, including four of the vice-president's children

the genitive "vice-president's" will be attached to "children". The "of" complement will be attached to "four", and since "including" is treated as a conjunction, the entire phrase will be recognized as conjoined noun groups.

In Message 48 of TST2, there were 18 relevant patterns. FASTUS recognized 12 of them completely. Because of bugs in implemented patterns, 3 more patterns were recognized only partially. One implemented pattern failed completely because of a bug. Specifically, in the sentence

> A niece of Merino's was injured.

the genitive marker took the system into a state in which it was not expecting a verb group.

Two more patterns were missing entirely. The pattern

> <HumanTarget1> <VerbGroup> with <HumanTarget2>

would have matched

> ... the attorney general was traveling with two bodyguards.

and consequently would have recognized the two bodyguards as human targets along with the attorney general.

The second pattern is

> <HumanTarget> be in <PhysicalTarget>

mentioned above.

A rudimentary sort of pronoun resolution is done by FASTUS. If (and only if) a pronoun appears in a Human Target slot, an antecedent is sought. First the noun groups of the current sentence are searched from left to right, up to four phrases before the pronoun. Then the previous sentences are searched similarly for an acceptable noun group in a left-to-right fashion, the most recent first. This is continued until the last paragraph break, and if nothing is found by then, the system gives up. A noun group is an acceptable antecedent if it is a possible human target and agrees with the pronoun in number. This algorithm worked in 100% of the relevant cases in the first 200 messages of the development set. However, in its one application in Message 48 of TST2, it failed. The example is

> According to the police and Garcia Alvarado's driver, who escaped unscathed, the attorney general was traveling with two bodyguards. One of them was injured.

The algorithm incorrectly identifies "them" as "the police".

MERGING INCIDENTS

As incidents are found they are merged with other incidents found in the same sentence. Those remaining at the end of the processing of the sentence are then merged, if possible, with the incidents found in previous sentences.

For example, in the first sentence of Message 48 of TST2, the incident

Incident:	KILLING
Perp:	–
Confid:	–
HTarg:	"Roberto Garcia Alvarado"

is generated from the phrase

killing of Attorney General Roberto Garcia Alvarado

while the incident

Incident:	INCIDENT
Perp:	FMLN
Confid:	Suspected or Accused by Authorities
HTarg:	–

is generated from the clause

> Salvadoran President-elect Alfredo Cristiani ... accused the Farabundo Marti National Liberation Front (FMLN)

These two incidents are merged, by merging the KILLING and the INCIDENT into a KILLING, and by taking the union of the other slots.

Incident:	KILLING
Perp:	FMLN
Confid:	Suspected or Accused by Authorities
HTarg:	"Roberto Garcia Alvarado"

Merging is blocked if the incidents have incompatible types, such as a KIDNAPPING and a BOMBING. It is also blocked if they have incompatible dates or locations.

There are fairly elaborate rules for merging the noun groups that appear in the Perpetrator, Physical Target, and Human Target slots. A name can be merged with a precise description, as "Garcia" with "attorney general", provided the description is consistent with the other descriptions for that name. A precise description can be merged with a vague description, such as "person", with the precise description as the result. Two precise descriptions can be merged if they are semantically compatible. The descriptions "priest" and "Jesuit" are compatible, while "priest" and "peasant" are not. When precise descriptions are merged, the longest string is taken as the result. If merging is impossible, both noun groups are listed in the slot.

We experimented with a further heuristic for when to merge incidents. If the incidents include named human targets, we do not merge them unless there is an overlap in the names. This heuristic results in about a 1% increase in recall. In Message 48 of TST2, the heuristic prevents the Bombing of Garcia Alvarado's car from being merged with the Bombing of Merino's home.

There were 13 merges altogether in processing Message 48 of TST2. Of these, 11 were valid.

One of the two bad merges was particularly unfortunate. The phrase

> ... Garcia Alvarado's driver, who escaped unscathed, ...

correctly generated an attack incident with no injury to the human target, the driver:

Incident:	ATTACK
Perp:	–
PTarg:	–
HTarg:	"Garcia Alvarado's driver"
HEffect:	No Injury

This was merged with the attack on Merino's home

Incident:	BOMBING
Perp:	"guerrillas"
PTarg:	"Merino's home"
HTarg:	"Merino"
HEffect:	–

to yield the combined incident

Incident:	BOMBING
Perp:	"guerrillas"
PTarg:	"Merino's home"
HTarg:	"Merino": "Garcia Alvarado's driver"
HEffect:	No Injury

That is, it was assumed that Merino was the driver. The reason for this mistake was that while a certain amount of consistency checking is done before merging victims, and while the system knows that drivers and vice presidents-elect are disjoint sets, the fact that Merino was the vice president-elect was recorded only in a table of titles, and consistency checking did not consult that table.

ERROR ANALYSIS

FASTUS made 25 errors on Message 48 of TST2, where a wrong answer, a missing answer, and a spurious answer are all counted as errors. (There is in principle no limit to the number of possible errors, since arbitrarily many spurious entries could be given. However, practically the number of possible errors is around 80. If no entries are made in the templates, that counts as 55 errors. If all the entries are made and are correct, but combined into a single template, that counts as 48 errors—the 24 missing entries in the smaller template and the 24 spurious entries in the larger.)

The sources of the errors are as follows:

Missing Patterns (2)	9
Bad Merges (2 of 13)	7
Military "armored car" Filtered Out	4
Answer Disputable	3
Bug in Existing Pattern	2
Bad Pronoun Resolution	1
Mysterious	1

Because of the missing patterns, we failed to find the children and the bodyguards as human targets. The bad merges resulted in the driver being put into the wrong template. The armored car was found as a physical target in the attack against Garcia Alvarado, but armored cars are viewed as military, and military targets are filtered out just before the templates are generated. The disputable answer is Merino as a human target in the bombing of his home.

We do not know to what extent this pattern of causes of errors is representative of the performance of the system on the corpus as a whole.

FUTURE DIRECTIONS

If we had had one more month to work on the MUC-4 task, we would have spent the first week developing a rudimentary pattern specification language. We believe that with about two months work we could develop a langauge that would allow a novice user to be able to begin to specify patterns in a new domain within hours of being introduced to the system. The pattern specification language would allow the user to define structures, to specify patterns in regular expressions interrupted by assignments to fields of the structures, and to define a sort hierarchy to control the merging of structures.

We would also like to apply the system to a new domain. Our experience with the MUC-4 task leads us to believe we could achieve reasonable performance on the new domain within two months.

Finally, it would be interesting to try to convert FASTUS to a new language. There is not much linguistic knowledge built into the system. What there is probably amounted to no more than two weeks coding. For this reason, we believe it would require no more than one or two months to convert the system to another language. This is true even for a language as seemingly dissimilar to English as Japanese. In fact, our approach to recognizing phrases was inspired in part by the bunsetsu analysis of Japanese.

SUMMARY

The advantages of the FASTUS system are as follows:

- It is conceptually simple. It is a cascaded finite-state automaton.
- The basic system is relatively small, although the dictionary and other lists are potentially very large.
- It is effective. Only General Electric's system performed significantly better than FASTUS, and it has been under development for a number of years.
- It has very fast run time. The average time for analyzing one message is less than 7 seconds.
- In part because of the fast run time, it has a very fast development time. This is also true because the system provides a very direct link between the texts being analyzed and the data being extracted.

FASTUS is not a text understanding system. It is an information extraction system. But for information extraction tasks, it is perhaps the most convenient and most effective system that has been developed.

ACKNOWLEDGEMENTS

The research was funded by the Defense Advanced Research Projects Agency under Office of Naval Research contracts N00014-90-C-0220, and by an internal research and development grant from SRI International.

REFERENCES

[1] Hobbs, Jerry R., Stickel, Mark, Appelt, Douglas, and Martin, Paul, "Interpretation as Abduction", SRI International Artificial Intelligence Center Technical Note 499, December 1990.

[2] Lehnert, Wendy, Claire Cardie, David Fisher, Ellen Riloff, and Robert Williams, 1991. "Description of the CIRCUS System as Used for MUC-3", *Proceedings*, Third Message Understanding Conference (MUC-3), San Diego, California, pp. 223-233.

[3] Pereira, Fernando, 1990. "Finite-State Approximations of Grammars", *Proceedings*, DARPA Speech and Natural Language Workshop, Hidden Valley, Pennsylvania, pp. 20-25.