SRA: DESCRIPTION OF THE IE^2 SYSTEM USED FOR MUC-7

Chinatsu Aone, Lauren Halverson, Tom Hampton, Mila Ramos-Santacruz
SRA International, Inc.
4300 Fair Lakes Court
Fairfax, VA 22033-4232
{aonec, laur, hamptont, mila}@sra.com

INTRODUCTION

In our planning for MUC-7, we made the decision to concentrate our efforts in two areas: improvements to our infrastructure, both in system architecture and the supporting set of tools; second, maximizing the performance of each information extraction (IE) module, including the introduction of trainable learning-based modules. We felt that achieving our performance goals in extraction depended on improvements in these areas. Specifically, our goals when entering MUC-7 were to:

- Increase the accuracy in the Template Element (TE) task and the Template Relation (TR) task sufficiently for operational use, i.e., F-Measures of 85% and 80% respectively,
- Increase the accuracy and portability in the Scenario Template (ST) task significantly.

In order to achieve these goals, we took several important steps, which included:

- A new IE system developed in the past year
- A flexible, modular system architecture
- A set of annotation tools and development tools to speed up development, and enable higher accuracy
- A high-performance phrase and link tagger
- A hybrid, trainable discourse module.

The result was a new high-performance information extraction system, **Information Extraction Engine (IE^2)**. It has a flexible and modular architecture and allows optimal speed in the development of all system modules.

In the end, this new system achieved the highest score in each of the three tasks we entered: TE, TR, and ST, as shown in Table 1. In particular, TE showed an operational performance level, while TR was almost at that level. ST, by contrast, still presents fundamental problems yet to be solved by the community, but we believe that we have demonstrated an effective strategy for enhancing performance on this task.

	Recall	Precision	F-Measure
TE	86	87	**86.76**
TR	67	86	**75.63**
ST	42	65	**50.79**

Table 1: SRA's Scores for TE, TR and ST

SYSTEM ARCHITECTURE

About a year ago, we set out to develop a new IE system. One of the first things we worked on was the design of a new system architecture. The two top requirements were modularity and flexibility. We wanted a modular architecture so that each module could be developed, tested, and improved independently. This modular architecture not only speeds up the system development process but also provides the ability to replace an existing module with a new one very easily. We also wanted a flexible workflow so that development of a module in the latter stage of IE processing does not require processing input through all the previous modules. This flexible workflow not only cuts development time, but also enables simultaneous development of multiple modules without dependency problems.

We chose SGML-marked up texts as the input and output of each module, which follows the spirit of the TIPSTER architecture. This enabled us to spell out system interface requirements between two modules much more clearly than previously possible, and reduced mis-communication between modules. The IE^2 system is truly modular, and any module can be replaced as long as the new module follows the system interface requirements. Its workflow is also flexible so that one can start processing at any point. For example, the person who is testing the discourse module can just input the previous SGML output of the event tagging module without re-processing any of the previous modules. Figure 1 shows the new system architecture. We discuss each module in more detail in the next section.

SYSTEM DESCRIPTION

SRA's IE^2 System used five modules for the TE and TR tasks and six for the ST task, as shown in Figure 1. The three core modules, namely NameTag, PhraseTag, and EventTag, use SRA's multilingual information extraction engine called TurboTag.

Entity Name Recognition

For MUC-7, we used NetOwl Extractor 3.0, a commercial entity name recognition software by IsoQuest, a subsidiary of SRA. NetOwl Extractor 3.0 recognizes names of people, organizations, and places, as well as time and numeric expressions. It was configured to follow MUC-7 NE guidelines.

In addition to its name recognition capability, we used its sub-typing and name alias recognition capabilities. Extractor 3.0 provides subtypes of organizations (e.g., company, government, military, etc.) and places (e.g., country, province, city, etc.) It also provides links among aliases of organizations (e.g., "TCI" for "Tele-Communications Inc."), people (e.g., "Goldstein" for "Irving Goldstein"), and places (e.g., "U.S." for "United States").

Custom NameTag

SRA developed custom patterns to perform additional name recognition using the TurboTag engine. The Custom NameTag tags artifact names (vehicle) necessary for the TE, TR and ST task. It also performs semantic classification of vehicle names into AIR, GROUND, and WATER. They are further classified into

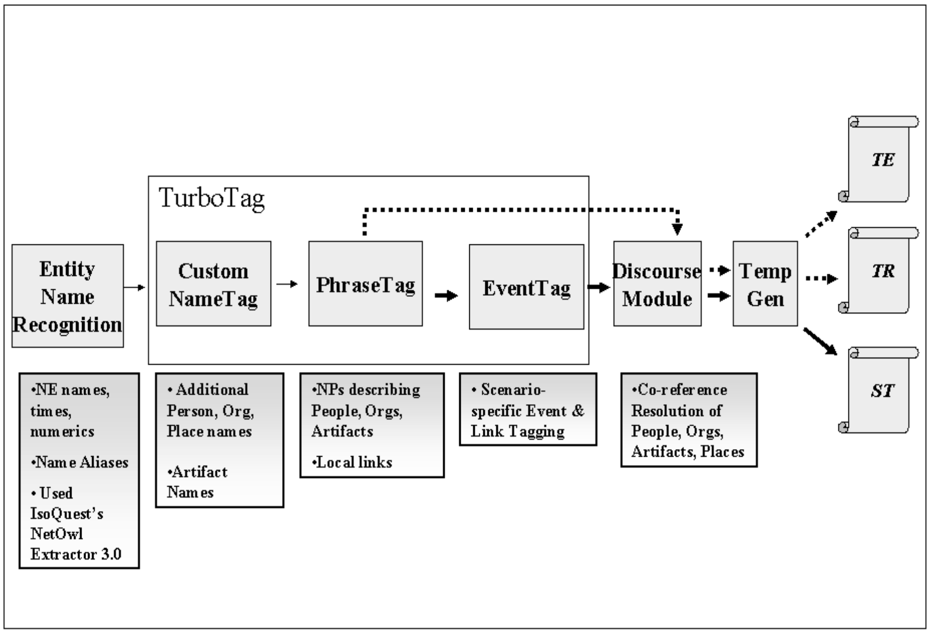

Figure 1: IE^2 System Architecture

subtypes: plane, helicopter, space shuttle, rocket, missile, space probe, space station, satellite, and capsule for AIR, car and tank for GROUND, and ship and submarine for WATER.

During the formal test period, we also developed additional patterns to recognize missing names of people, organizations, and places for the launch domain.

PhraseTag

In order to achieve operational-level accuracy in TE and TR, we decided that we needed a high-performance noun phrase tagger especially targeted to recognize noun phrases describing people (PNP), organizations (ENP), and artifacts (ANP). Thus, we developed PhraseTag, which tags these three types of NPs. It recognizes not only simple NPs but also complex NPs with post modifiers such as relative clauses, reduced relatives, and prepositional phrases. In a blind test set, it achieves around 80% F-Measure, including the post-modifier attachments.

Additionally, PhraseTag adds local *links* between phrases with high accuracy. During the design phase of IE^2, our analysis of MUC-6 data led us to believe that most links necessary to recognize for the TE and TR tasks are *local*. That is, by targeting one's effort at good recognition of such linguistic structures as appositives (e.g., "Irving Goldstein, director general and chief executive of Intelsat") and copula sentences (e.g., "International Technology Underwriters of Bethesda, Maryland, is one insurer in this consortium"), one can achieve high accuracy in TE and TR. As these structures are constrained to local contexts, it is much easier to recognize them with high accuracy than structures involving long-distance dependencies.

PhraseTag recognizes four local links, namely **employee_of**, **location_of**, **product_of**, and **owner_of**. Links are registered in the SGML attributes *AFFIL*, *LOC*, *MAKER* and *OWNER*, respectively, as shown below. The **owner_of** relation is not a part of the TR task, but we used it in the ST task to extract owners of vehicles and payloads.

- **employee_of**
  <PNP AFFIL=1>an analyst at <ENTITY ID=1>ING Barings</ENTITY> in Mexico City</PNP>
- **location_of**
  <ENP LOC=2><PLACE ID=2>Paris</PLACE> insurer</ENP>
- **product_of**
  <ANP MAKER=3>a satellite built by <ENTITY ID=3>Loral Corp.</ENTITY> of New York for Intelsat</ANP>
- **owner_of**
  <ANP OWNER=4>an <ENTITY ID=4>Intelsat</ENTITY> satellite</ANP>

Note that in these examples, the unique identifier (ID) of the ENTITY or PLACE is being written as attribute information on the surrounding noun phrase tags with the appropriate label.

Other types of link, such as the parent/child-organization relationship (e.g., " Apache Corp., which owns a majority stake in Hudson Energy"), can be easily added to IE^2.

EventTag

For the ST task, we decided to take a bottom-up, minimalist approach in MUC-7 rather than using a general-purpose full noun phrase or sentence parser. The EventTag module starts with a set of syntactically-oriented rule templates. We show some examples in Table 2 and Table 3.

Then, an IE developer fills in the templates using example phrases and sentences from the training texts. Scenario-specific NP macros, such as $Vehicle and $Payload, are used to fill Arg's, while scenario-specific verb lists (e.g., LauchTransitiveV, AttackV) and noun lists (e.g., LaunchN, AttackN) are used to fill verbs and nouns in the templates respectively. Some examples from the launch event are shown in Table 4.

Our plan was to develop and integrate a trainable rule generalization module under TIPSTER 3, and apply it to an initial set of manually-coded rules to significantly boost ST accuracy, especially recall. Unfortunately, this effort did not start early enough to be completed and integrated with EventTag. However, we are currently making progress, and believe that the rule generalization module will improve the ST accuracy substantially.

Discourse Module

In the past year, we also developed a new discourse module for co-reference resolution that is both *trainable* and *configurable*. We wanted an automatically trainable system to ease portability and achieve high accuracy, as well as a configurable system to optimize the benefit of discourse resolution for different IE tasks. Our goal was to develop a discourse module which can improve specific IE tasks, rather than develop a generic one which may perform well overall but may not necessarily increase the IE performance. For this, we aimed at developing a high-precision discourse module.

Template 1	Arg1	IntransitiveV			
Template 2	Arg1	IntransitiveV	prep	Arg2	
Template 3	Arg1	TransitiveV	Arg2		
Template 4	Arg1	TransitiveV	Arg2	prep	Arg3
Template 5	Arg1	PassiveV	by	Arg2	
Template 6	Arg1	PassiveV			

Table 2: ST Rule Template Examples (Verbs)

Template 7	Arg1	noun			
Template 8	noun	of	Arg1		
Template 9	noun	of	Arg1	prep	Arg2

Table 3: ST Rule Template Examples (Nouns)

Template 7	$Vehicle + LaunchN	"the Arian 5 *launch*"
Template 8	LaunchN + of + $Payload	"today's failed *launch* of a satellite"
Template 1	$Vehicle + FailureIntransitiveV	"A Chinese rocket *exploded*"
Template 6	$Payload + LaunchPassiveV	"a second satellite to be *launched*"

Table 4: Launch Rules

The new Discourse Module employs three co-reference resolution strategies so that it can be configured for its best performance for different IE tasks. The *rule-based* strategy uses the CLIPS engine. We have developed rules to resolve definite NPs (ANP, PNP, ENP) and singular personal pronouns (e.g., "he," "his," "him"). The *machine learning* strategy uses C50, which is a decision tree implementation [4], to learn co-reference resolution automatically from a tagged corpus. This is a re-implemented (in C++) and improved version of SRA's previous machine learning-based co-reference resolution module [2]. The third strategy is a hybrid method where the module first applies the rule-based strategy to narrow down the possible antecedents and then applies the machine learning strategy to order the possible antecedent candidates.

The Discourse Module is also configurable so that one can select a set of anaphora types to resolve for each task. Currently the module resolves:

- name aliases (artifacts, people, organizations, places),
- definite NPs (ANP, ENP, PNP), and
- singular personal pronouns.

Name alias resolution for people, organizations, and places is performed in addition to that performed by NetOwl Extractor 3.0 in order to increase recall. Appositive links are always made by PhraseTag because of its local nature.

For TE, TR, and ST, we resolved name alias and appositive anaphora. However, we performed definite NP and pronoun resolution only for TR and ST because doing so did not increase the TE score in experimenting with the formal training texts. We think this is because the discourse module was trained on the dry-run training texts (i.e., aircraft crash domain), and the lack of training on the formal test domain (i.e., spacecraft launch domain) made the resolution of definite NPs and pronouns less accurate. In fact, the module had increased the TE score on the dry-run test texts.

This indicates the delicate trade off between recall and precision in IE. For the TE task, the definite NP resolution should theoretically increase the recall of the DESCRIPTOR slot. However, as most descriptors are actually found locally, either in appositives or copula constructions, it takes a very high-precision discourse module to improve the recall of this slot without hurting precision. The discourse module with definite NP and pronoun resolution did however increase the ST score by 2 points in the formal test.

The Discourse Module is still relatively new, and it needs to be trained on more texts. We plan to perform additional experiments to make it a high-precision system that helps IE tasks.

```
<PERSON ID=5>Jeff Bantle</PERSON>, <PNP REF=5 AFFIL=12><ENTITY ID=12>NASA</ENTITY>'s
mission operations directorate representative for the shuttle flight</PNP>.
```

Figure 2: Simplified Discourse Module output

TempGen

SRA's new template generator, TempGen, is implemented in JAVA, and is considerably easier to configure and customize than our previous versions. This module takes SGML output of the Discourse Module and maps it into TE, TR or ST templates. It uses an SGML-to-template mapping configuration file so that mapping is more declarative and easier to customize. For instance, for a phrase like "Jeff Bantle, NASA's mission operations directorate representative for the shuttle flight," the Discourse Module produces the simplified SGML output in Figure 2.

TempGen takes this output and produces two TE templates (cf., Figure 3 and Figure 4), and one TR template (cf., Figure 5). The PNP's REF register holds the ID of the person the PNP refers to (i.e., Jeff Bantle). Thus, the PNP is used for ENT_DESCRIPTOR of the TE template for "Jeff Bantle." The PNP's AFFIL register holds the person's affiliation, and is used for the EMPLOYEE_OF TR for the person (i.e., Jeff Bantle) and its employer (i.e., NASA).

For ST, TempGen integrates a Time Module, which interprets and normalizes time expressions according to the MUC-7 time guidelines. TempGen also performs event merging. While the Discourse Module takes care of the merging of noun phrases describing payloads and vehicles in the launch domain, the TempGen makes decisions on whether or not to merge two launch events based on the consistency of payloads, vehicles, time, and location participating in the two events. Event merging is a complex operation because the accuracy of the merging operation depends on various factors including:

- accuracy in the co-reference resolution of payloads and vehicles,
- correct interpretation of time phrases (e.g., "two days ago"),
- correct inference on whether two time/location expressions are consistent.
  (e.g., "yesterday morning" vs. "on Wednesday," "Florida" vs. "Miami").

  For instance, in the example below, knowing that "Wednesday" and "tomorrow" are the same day is crucial for event merging.

  > "China plans to send a satellite into orbit *Wednesday* ... In *tomorrow*'s launch, a Long March 3 rocket will carry an Apstar-1A satellite."

Our plans to improve event merging include:

```
<ENTITY-9601120403-13> :=
    ENT_NAME:  Jeff Bantle
               Bantle
    ENT_TYPE:  PERSON
    ENT_CATEGORY: PER_CIV
    ENT_DESCRIPTOR: NASA's mission operations directorate representative for
                    the shuttle flight
```

Figure 3: TE for Jeff Bantle

```
<ENTITY-9601120403-4> :=
    ENT_NAME:   National Aeronautics and Space Administration
                NASA
    ENT_TYPE:  ORGANIZATION
    ENT_CATEGORY: ORG_GOVT
```

Figure 4: TE for NASA

```
<EMPLOYEE_OF-9601120403-44> :=
    PERSON:  <ENTITY-9601120403-13>
    ORGANIZATION:  <ENTITY-9601120403-4>
```

Figure 5: TR for Jeff Bantle and NASA

- enhancing the Discourse Module for co-reference resolution of noun phrases and the Time Module for time interpretation,
- incorporating event merging as a part of the trainable Discourse Module.

ANNOTATION AND DEVELOPMENT TOOLS

One of the crucial aspects of developing a high-performance IE system is to have the right tools for the right modules. In general, IE development requires *annotation* tools for creating training examples and *development environments* for evaluating and debugging the output of a given IE module. Having the right tools not only speeds up development significantly, but also enables high accuracy.

SRA has considerable experience in building annotation and development tools for multilingual information extraction. Annotators of MUC-6, MUC-7, MET-1, and MET-2 used SRA's Named Entity Tool for NE annotation, and SRA's Discourse Tagging Tool for the Coref annotation [1]. SRA's Name Entity Tool currently works with Chinese, Japanese, Arabic, Thai, Russian, and all the Romance alphabet languages. In the past year, we built the following three new tools to support development for the TE, TR, and Coref tasks. All the tools were implemented in JAVA to support multilingual, cross-platform capabilities.

Annotation Tool: A GUI-based template-level annotation tool which can create TE, TR, and ST type templates. The tool is easily configurable from the GUI, and can be integrated with IE^2 to function also as a post-editing tool. Figure 6 shows the Annotation Tool.

Template Tool: A GUI-based template-level development environment which enables inspection and debugging of complex template structures. The tool is designed to allow an IE developer to run any portion of the IE system from the GUI, and score the results using a scoring program for immediate feedback. Figure 7 shows the Template Tool.

Discourse Debugger: A GUI-based discourse development environment which allows evaluation of complex co-reference links. The tool lets an IE developer run the discourse module on any document or document sets, scores the results, and can display both the system-generated links and the links from the key graphically. Figure 8 illustrates the Discourse Debugger.

We believe that our ability to create training examples quickly with our annotation tools and evaluate system performance in an effective manner using our development environments contributed to our good performance in MUC-7.

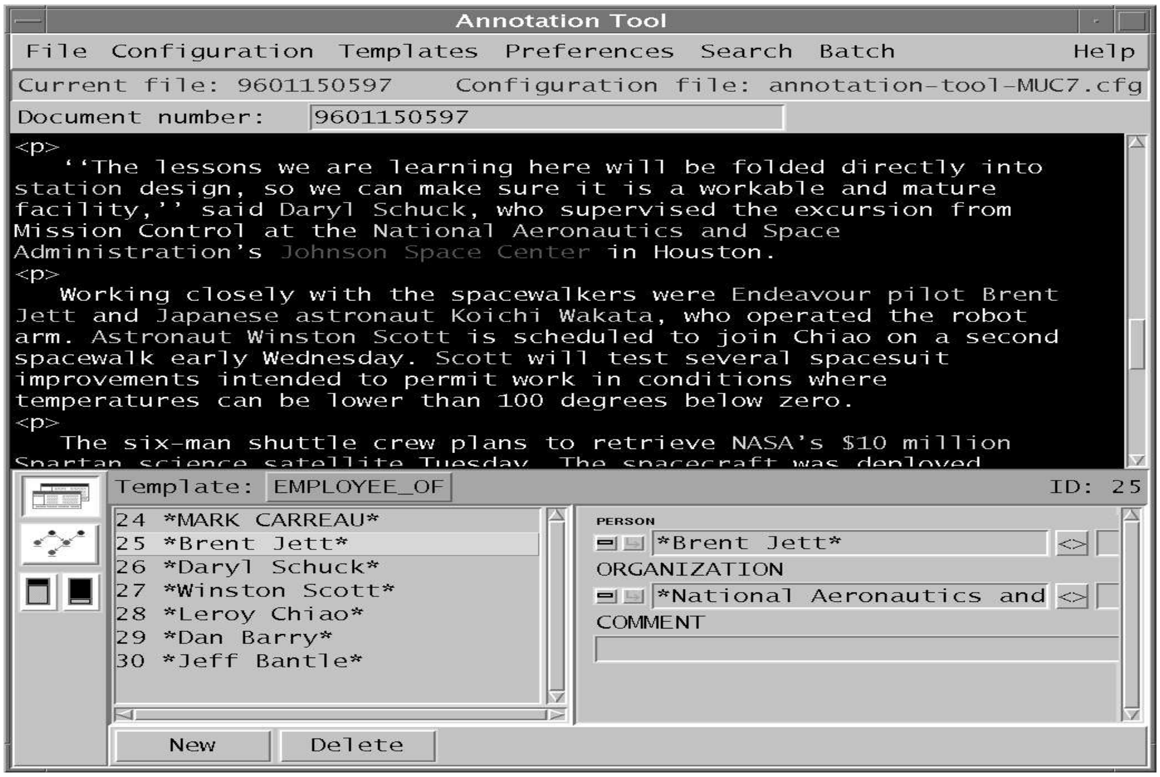

Figure 6: Annotation Tool

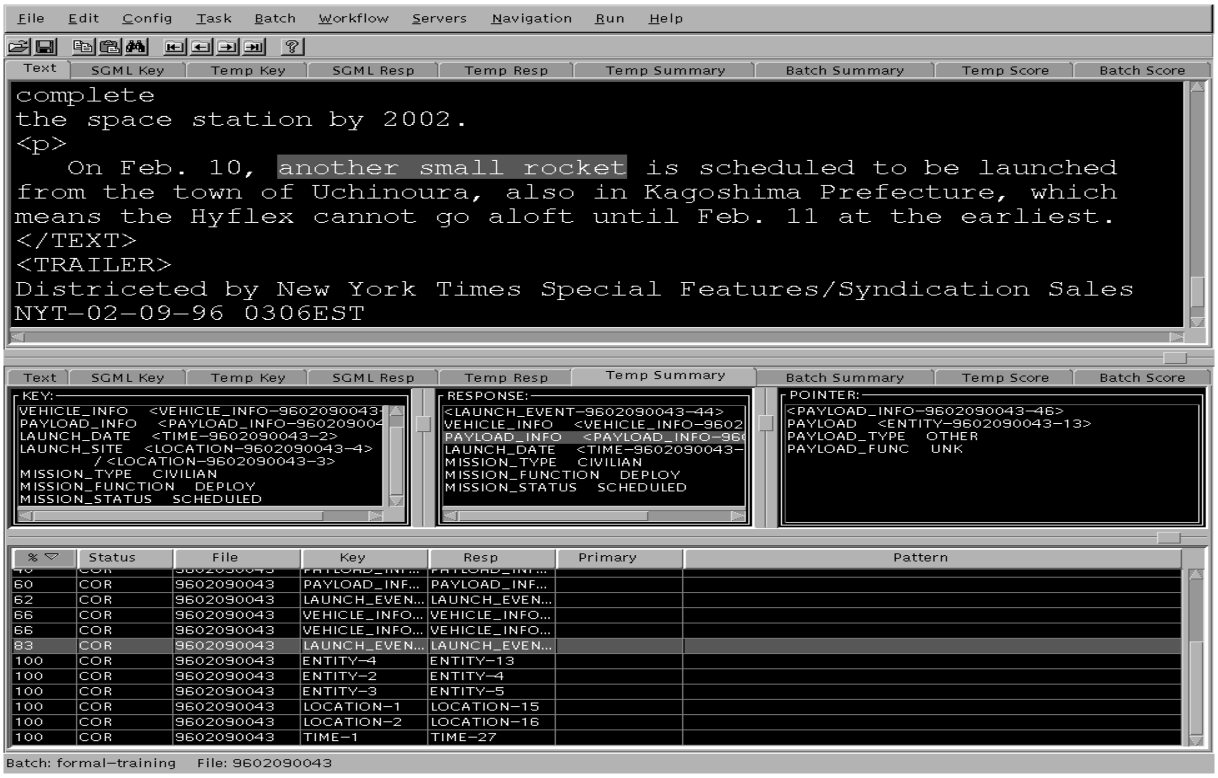

Figure 7: Template Tool

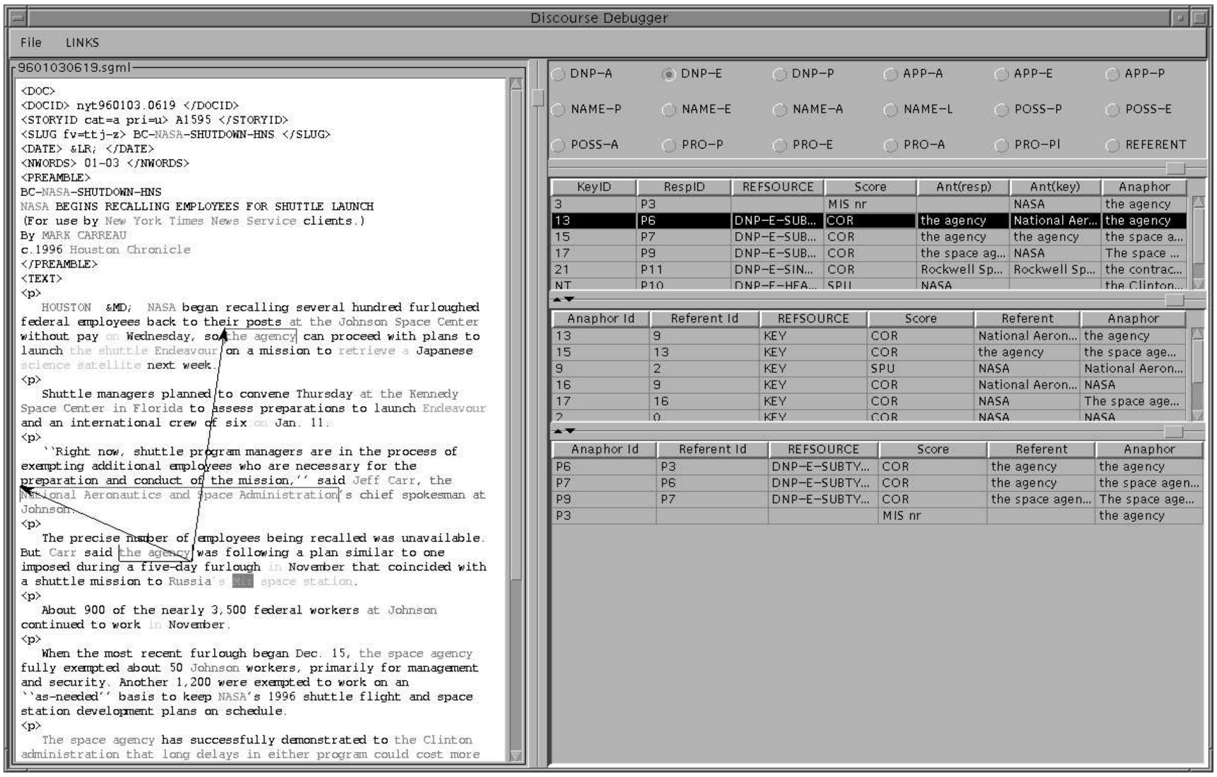

Figure 8: Discourse Debugger

EVALUATION RESULTS

IE^2 performed very well on its three tasks (cf., Table 1). In TE, it **achieved operational quality**, performing well above 85% F-M. Both the TE and TR scores were statistically significantly higher than the second best scores in each task. TR was almost at the operational level. In fact, the **employee_of** relation scored a 87.59% F-M (Recall=82, Precision=94). While ST did not achieve operational-level accuracy during the four-week time limit, it still achieved the highest score among the ST participants. We believe that we have an effective strategy for enhancing performance on this task, as described in the "Lessons Learned" Section.

SYSTEM FACTSHEET

SRA's TurboTag and the Discourse Module are written in C^{++}, TempGen in JAVA. In processing 100 test texts on a SUN Ultra (167 MHz) with 128 MB of RAM, the system achieved the following performance figures:

- TE: 11 min., 17 sec. (with Coref, add 5 min., 38 sec.)
- TR: 18 min., 59 sec.
- ST: 19 min., 22 sec.

We could increase throughput further easily by optimizing the TempGen code, distributing processes, and/or by performing a trade-off with development-time flexibility.

LESSONS LEARNED

Lessons for TE

Our system IE^2 performed very well in TE (Recall=86; Precision=87;F-M=86.76). These results are especially positive for two reasons. First, the TE score shows a distinct improvement over that of MUC-6 (80% F-M). This is so in spite of the fact that the TE task is more complex in MUC-7 than in MUC-6.[1] Second, given the domain-independent design and implementation of our TE module, we believe that these results would carry over to other text types without much performance degradation.

In our view, to go even higher – to achieve 90% F-M in TE – requires two things:

- Artifacts are a fairly new extraction target for the field, at least by comparison with persons, organizations, and places. In addition, there were far fewer artifact occurrences in the training texts than of the other name categories. More training examples for artifacts will increase their yield significantly and improve the recognition of their descriptors.
- We can improve the recognition of long-distance links by increasing the accuracy of definite NP co-reference. In future work, we will focus on trainable discourse.

Lessons for TR

IE^2 also did well in the TR task (Recall=67; Precision=86; F-M=75.63). The most exciting sub-result was in the **employee_of** relation, which showed excellent performance (Recall=82; Precision=94; F-M=87.59).

This piece of extraction technology is almost, but not quite, at the operational level. To achieve TR of 80 or above, we need:

- More training examples, particularly for the **product_of** relation. There were only 77 training examples of this in the dry-run training texts.
- As with TE, we need improved discourse, specifically better accuracy of definite NP and pronoun co-reference. This would improve performance on examples such as:
  - "International Technology Underwriters" — "*its* chief executive" (**employee_of**)
  - "Intelsat" — "*the company's* Washington headquarters" (**location_of**)

Lessons for ST

Our ST scores (Recall=42; Precision=65; F-M=50.79) reflect the intrinsically difficult nature of the ST task. We regard the following as the most important challenges to be overcome for ST to show improvements up towards 75%.

- **Rule Generalization:** Knowing how to extend rule coverage beyond what is explicitly contained in the training data is a critical goal to reach. For example, while there were 3000 TE templates and 750 TR templates in the dry-run training keys, there were only 80 launch events in the formal training keys. The current system has the intrinsic limitation that it encodes extraction rules from examples in the training keys and from what the IE developer can intuit. Corpus-based learning algorithms are required to extend the coverage of extraction rules much further while maintaining their accuracy.

[1] MUC-7 requires the extraction of two additional elements: artifact TE templates and a descriptor for person TE templates.

- **Event Merging:** The elements of an ST event are frequently scattered over a text. We need better techniques for understanding what and how to merge partial descriptions of the same event. Enhancing the current co-reference resolution of noun phrases will definitely help the merging of event arguments. In addition, we plan to incorporate event merging as a part of the Discourse Module, treating events also as anaphora, and use a learning algorithm to acquire event merging rules from a corpus.
- **Time Interpretation:** Understanding the proper sequence of sub-events is critical for understanding the structure of the overall event and for event merging in particular. Fewer examples and less time devoted to the Time Module made it less mature. However, we are certain that we can improve the time interpretation to increase the ST score significantly.

SUMMARY AND FUTURE DIRECTIONS

The IE field has shown distinct progress in the last few years. At MUC-6, several of the systems showed strong results in name recognition. At the current MUC, TE and TR have both been shown to be highly practicable IE tasks. TE in particular is already performing at an operational level.

Very important to our success in MUC-7 was the quality of our new architecture, new tools, a new phrase/link tagging module, and an improved discourse module. The new architecture emphasized modularity and flexibility. A modular architecture means that each module can be developed, tested, and improved independently of the other. We also made the workflow as flexible as possible so that development of a "back-end" module does not require processing input through all the previous modules.

However, to continue progress in extraction technology, we believe that the following track should be followed:

- Continue development of learning-based, trainable modules:
  - **Automated learning of names and phrases.** Although name recognition is already performing at a high accuracy level, this does not answer the question of how to port a name recognizer quickly and cost effectively to new languages or new text types. In some cases a pattern matching-based name/phrase tagger is the best choice, while in other cases, a learning-based system is more suitable. SRA has already developed RoboTag, a decision-tree-based multilingual learning system for names, and has shown promising results [3]. We plan to continue enhancing RoboTag, and extend it to phrase recognition.
  - **Trainable discourse module.** Discourse has to be better, both to get TE to 90%, to make TR operational (greater than 80% F-M), and to make event merging more successful in ST. We plan to pursue a *hybrid* strategy of both automated and manual acquisition of co-reference resolution rules.
  - **Trainable event rule generalization module.** Increasing the "reach" of extraction rules is critical for next-generation extraction systems, particularly for ST. We are currently working on a trainable EventTag under the TIPSTER 3 effort.
- Develop a tightly integrated Annotation Toolset for efficient and consistent tagging of training texts for *all* tasks.

WALKTHROUGH (nyt960214-0509)

SRA's system performed very well on nyt960214-0509. Table 5 shows the scores for this article in the three relevant tasks. This text illustrates fairly well the strengths of SRA's system as well as some shortcomings.

TE (94% F-M)

With respect to TE's, below are the errors the system made:

	Recall	Precision	F-Measure
TE	95	93	**93.99**
TR	74	96	**83.93**
ST	67	50	**57.14**

Table 5: Scores for the Walkthrough Text

- It got some wrong aliases. The system listed "Bloomberg" as an alias of "Bloomberg L.T". It is an alias of "Bloomberg Business News". Similarly, the system did not recognize "News Corporation" as an alias of "News Corp."
- It failed to get the correct descriptor extent in two cases because of PhraseTag errors. Instead of "one insurer in this consortium", the system reported "this consortium" as the descriptor. Instead of "A Chinese rocket carrying an Intelsat satellite", the system reported "A Chinese rocket carrying an Intelsat satellite exploded as it" as the descriptor.
- It got a bad ENT_CATEGORY for "Space Transportation Association": instead of ORG_CO, it got ORG_OTHER.
- It failed on one country normalization. Instead of "French Guiana", it output "French Guyana".
- It output the wrong LOCAL_TYPE for Xichang: AIRPORT instead or CITY/PROVINCE.

And here is what the system did correctly:

- It recognized relatively long and complex descriptors. We indicate the descriptors with brackets:

  "Intelsat is [a global supplier of international satellite communication services]"

  "Virnell Bruce, [spokeswoman for Lockheed Space and Strategic Missiles in Bethesda, Maryland]."

  "Eric Stallmer, [spokesman for the Space Transportation Association of Arlington, Virginia], which represents U.S. rocket makers who compete with the Chinese."

  "Bloomberg Information Television, [a unit of Bloomberg L.P., the parent of Bloomberg Business News], was in negotiations for carriage of its 24-hour news service on the satellite destroyed today, it a company spokesman said."

- It recognized indefinite NP's that are not associated with a name, but are specific:

  "today's failed launch of [a satellite built by Loral Corp. of New York for Intelsat]"

  "[a company spokesman] said"

TR (84% F-M)

As for TRs, here are the errors the system made:

- Some of the problems stem from the TEs. "Space Transportation Association" got the wrong ENT_CATEGORY. As a result, the scoring program matched it with the wrong TE in the keys. Moreover, two TRs which "Space Transportation Association" participates in were scored as partially correct: the **employee_of** TR (for Eric Stallmer) and **location_of** TR (for Arlington).

- The system failed to recognize four long-distance relations. One relation is implicit in the descriptor "a company spokesman," where "company" refers to "Bloomberg Information Television."

  > "*Bloomberg Information Television*, a unit of Bloomberg L.P., the parent of Bloomberg Business News, was in negotiations for carriage of its 24-hour news service on the satellite destroyed today, *a company spokesman* said."

  Similarly, the second relation is implicit in the descriptor "company spokesman," where company refers to "News Corporation."

  > "This failure will not affect *News Corporation*'s launch plans for the direct-to-home satellite service" in Latin America, said *company spokesman* Howard J. Rubenstein in a statement."

  In the third case, the link requires the anaphoric resolution of a possessive pronoun across paragraphs: "its chief executive," where "its" refers to "International Technology Underwriters." Currently, the anaphoric resolution module only resolves personal possessive pronouns.

  > "*International Technology Underwriters* of Bethesda, Maryland, is one insurer in this consortium. it The company is 80 percent owned by Paris insurer Axa SA and 20 percent by Prudential Reinsurance Holdings Inc. of Newark, New Jersey.
  > *Its* chief executive, former space shuttle astronaut Rick Hauck, wouldn't comment on the size of International Technology's loss. The company insures about 20 to 30 satellites a year."

  In the fourth case, the system missed the relation between "Intelsat" and "Washington." Resolving "the company" to "Intelsat" would fix this problem.

  > "His comments came at *the company*'s Washington headquarters"

- The system did not output a PRODUCT_OF TR for "Long March 3B", because of an error in the TR template generation.

Most of the errors are easily fixable, and the system got the rest of the relations correctly, with 84% as the overall F-M for this text.

ST (57% F-M)

The system successfully recognized all the launch events mentioned and their respective payloads and vehicles in this text. However, the system over-generated three more templates for two reasons. First, EventTag did not recognize two generic discussions of a satellite launch in this text. Consequently, it outputs launch events for the following two sentences.

> "U.S.-made rockets are not yet powerful enough alone to send a satellite as heavy as the one launched today into orbit."

> "Communications satellites typically cost about $150 million to $300 million to build and launch."

Second, the system did not recognize the launch event in (s2) below to be co-referent with the launch event in (s1). This merging was particularly difficult because of the use of the indefinite noun phrase "a satellite" (instead of a definite "the satellite") in the second sentence.

> (s1) A Chinese rocket carrying an Intelsat satellite exploded as it was being launched today, delivering a blow to a group including Rupert Murdoch's News Corp. and Tele-Communications Inc. that planned to use the spacecraft to beam television signals to Latin America.

(s2) The China Great Wall Industry Corp. provided the Long March 3B rocket for today's failed launch of *a satellite* built by Loral Corp. of New York for Intelsat.

In addition, the Time Module failes to interpret certain time descriptors correctly. For instance, the module converted "later this month" to the default document date.

```
<TIME-9602140509-99> :=
    START:  14021996
    END:  14021996
    DESCRIPTOR:  later this month
```

ACKNOWLEDGEMENTS

We would like to thank John Maloney, Michael Niv, and Bob Schlosser for their help with various aspects of the MUC-7 effort.

REFERENCES

[1] Aone, Chinatsu and Bennett, Scott W. Discourse Tagging Tool and Discourse-tagged Multilingual Corpora. In *Proceedings of International Workshop on Sharable Natural Language Resources (SNLR)*, 1994.

[2] Aone, Chinatsu and Bennett, Scott William. Evaluating Automated and Manual Acquisition of Anaphora Resolution Strategies. In *Proceedings of 33rd Annual Meeting of the ACL*, 1995.

[3] Bennet, Scott W, Aone, Chinatsu, and Lovell, Craig. Learning to Tag Multilingual Texts Through Observation. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing (EMNLP-2).*, 1997.

[4] Quinlan, J. Ross. *C4.5: Programs for Machine Learning.* Morgan Kaufmann Publishers, 1993.