PARAMAX SYSTEMS CORPORATION: MUC-4 TEST RESULTS AND ANALYSIS

Carl Weir and Barry Silk [1]
Paramax Systems Corporation
Valley Forge Labs
Paoli, Pennsylvania
weir@prc.unisys.com
(215) 648-2369

INTRODUCTION

The data extraction system submitted by Paramax for evaluation in MUC-4 is a new implementation written in CLIPS, a forward-chaining system developed and maintained by NASA's Johnson Space Center [1]. Using CLIPS as a forward-chaining engine is desirable because it runs on a wide range of machines (including Sun Sparc stations, Apple Mac IIs, and PCs), it is available at little or no cost from the government, it is fast, and it comes with good documentation and support services.

The Paramax MUC-4 development team consisted of one Paramax staff member and one government employee on sabbatical at Paramax. The data extraction module was designed and implemented in less than two months, using less than four person-months of labor. Developing inference rules for the system did not require any linguistic expertise or any detailed knowledge of CLIPS—neither of the developers had any prior experience using CLIPS. All that was required was knowledge of the domain and the data extraction task to be performed.

TEST RESULTS

The Paramax MUC-4 system's ALL TEMPLATES score summaries for the TST3 and TST4 test sets are listed below. The Paramax system generated more spurious responses in each of the two tests than any other system: the average number of TST3 spurious responses for all systems participating in MUC-4 was 883 and the average number of TST4 spurious responses was 867; the Paramax system generated 2207 and 2240 spurious responses, respectively.

TEST	POS	ACT	COR	PAR	INC	ICR	IPA	SPU	MIS	NON	REC	PRE	OVG
TST3	1693	3264	607	225	225	14	154	2207	636	2224	42	22	68
TST4	1253	3148	529	184	195	2	117	2240	345	2050	50	20	71

Since the Paramax MUC-4 implementation is substantially different from the Paramax MUC-3 submission, the two systems are difficult to compare.[2] The rules developed for the MUC-4 system were initially based on rules developed for the MUC-3 system, but the MUC-3 and MUC-4 rule formalisms are significantly different in structure and functionality. In Figure 1, the TST2 scores for the Paramax MUC-3 system and the TST3 *progress* scores for the MUC-4 system are listed.[3]

An examination of the scores in Figure 1 indicates that improvements in recall between MUC-3 and MUC-4 have generally resulted in degraded precision scores. However, the P&R F scores for the MUC-3

[1] Barry Silk is a U.S. government employee on sabbatical at Paramax.

[2] The Paramax MUC-3 system was submitted by the Unisys Center for Advanced Informaton Technology (CAIT), which has since been renamed Paramax Valley Forge Labs R&D.

[3] NRaD (formerly NOSC) rescored the MUC-3 TST2 scores of veteran sites in order to calculate F measures. The TST2 scores listed in Figure 1 are these rescored results, not the ones that appear in the MUC-3 proceedings [2]. The MUC-4 TST3 *progress* scores differ slightly from the official MUC-4 TST3 scores; the differences result from minor adjustments which make the comparision with MUC-3 TST2 scores more meaningful.

TOTAL SCORES FROM MUC-3 EVALUATION														
SLOT	POS	ACT	COR	PAR	INC	ICR	IPA	SPU	MIS	NON	REC	PRE	OVG	FAL
template-id	85	47	35	0	0	0	0	12	50	36	41	74	26	
inc-date	81	32	21	7	4	0	7	0	49	4	30	76	0	
inc-loc	85	31	7	18	6	0	1	0	54	0	19	52	0	
inc-type	85	35	34	1	0	0	0	0	50	0	40	98	0	0
inc-stage	85	35	33	0	2	0	0	0	50	0	39	94	0	1
perp-inc-cat	59	25	15	0	7	0	0	3	37	23	25	60	12	9
perp-ind-id	80	14	5	1	4	0	1	4	70	32	7	39	28	
perp-org-id	49	43	12	1	6	2	1	24	30	28	26	29	56	
perp-org-conf	49	38	7	2	10	0	2	19	30	28	16	21	50	6
phys-tgt-id	53	17	10	2	0	0	2	5	41	47	21	65	29	
phys-tgt-type	53	17	9	1	2	0	1	5	41	47	18	56	29	0
phys-tgt-nation	4	0	0	0	0	0	0	0	4	82	0	*	*	0
phys-tgt-effect	36	17	5	3	1	0	0	8	27	58	18	38	47	2
hum-tgt-name	34	14	6	0	1	0	0	7	27	55	18	43	50	
hum-tgt-desc	102	32	14	0	10	1	0	8	78	23	14	44	25	
hum-tgt-type	106	38	15	4	7	0	4	12	80	20	16	45	32	2
hum-tgt-nation	10	2	0	0	0	0	0	2	10	75	0	0	100	0
hum-tgt-effect	82	21	0	12	5	0	8	4	65	27	7	28	19	1
MATCHED/MISSING	1053	411	193	52	65	3	27	101	743	549	21	53	24	
MATCHED/SPURIOUS	428	528	193	52	65	3	27	218	118	299	51	41	41	
MATCHED ONLY	428	411	193	52	65	3	27	101	118	200	51	53	24	
ALL TEMPLATES	1053	528	193	52	65	3	27	218	743	648	21	41	41	
SET FILLS ONLY	569	228	118	23	34	0	15	53	394	360	23	57	23	0
STRING FILLS ONLY	318	120	47	4	21	3	4	48	246	185	15	41	40	
TEXT FILTERING	60	40	36	*	*	*	*	4	24	36	60	90	10	10
F-MEASURES				P&R 27.77					2P&R 34.44			P&2R 23.27		

TOTAL SCORES FROM MUC-4 TST3 PROGRESS TEST														
SLOT	POS	ACT	COR	PAR	INC	ICR	IPA	SPU	MIS	NON	REC	PRE	OVG	FAL
template-id	115	189	86	0	0	0	0	103	29	14	75	46	54	
inc-date	111	84	40	19	25	2	19	0	27	4	44	59	0	
inc-loc	115	82	22	46	14	0	9	0	33	0	39	55	0	
inc-type	115	86	79	7	0	0	0	0	29	0	72	96	0	0
inc-stage	115	86	81	0	5	0	0	0	29	0	70	94	0	2
perp-inc-cat	75	80	42	0	13	0	0	25	20	15	56	52	31	25
perp-ind-id	87	64	19	0	21	4	0	24	47	35	22	30	38	
perp-org-id	59	90	31	0	10	8	0	49	18	27	52	34	54	
perp-org-conf	59	82	8	2	31	0	2	41	18	27	15	11	50	11
phys-tgt-id	66	60	17	2	11	1	2	30	36	49	27	30	50	
phys-tgt-type	66	60	16	4	10	0	3	30	36	49	27	30	50	2
phys-tgt-nation	2	0	0	0	0	0	0	0	2	114	0	*	*	0
phys-tgt-effect	40	57	9	8	3	0	5	37	20	57	32	23	65	6
hum-tgt-name	56	43	24	2	3	2	2	14	27	62	45	58	32	
hum-tgt-desc	135	186	45	9	35	6	8	97	46	13	37	27	52	
hum-tgt-type	146	184	47	26	20	0	23	91	53	13	41	33	49	7
hum-tgt-nation	17	0	0	0	0	0	0	0	17	102	0	*	*	0
hum-tgt-effect	124	145	39	21	9	0	19	76	55	26	40	34	52	7
MATCHED/MISSING	1388	1389	519	146	210	23	92	514	513	593	43	43	37	
MATCHED/SPURIOUS	1043	2627	519	146	210	23	92	1752	168	1133	57	22	67	
MATCHED ONLY	1043	1389	519	146	210	23	92	514	168	394	57	43	37	
ALL TEMPLATES	1388	2627	519	146	210	23	92	1752	513	1332	43	22	67	
SET FILLS ONLY	759	780	321	68	91	0	52	300	279	403	47	46	38	2
STRING FILLS ONLY	403	443	136	13	80	21	12	214	174	186	35	32	48	
F-MEASURES				P&R 29.11					2P&R 24.38			P&2R 36.11		

Figure 1: MUC-3 and MUC-4 Performance Comparison

TST2 and MUC-4 TST3 evaluations indicate an improvement of 1.34 in overall performance. F measures are determined using the following formula:

$F = \frac{(\beta^2+1.0)\times P\times R}{\beta^2\times P+R}$, where P is precision, R is recall, and β is the relative importance given to recall over precision.[4]

No analyses of statistical significance were performed among MUC-3 TST2 and MUC-4 TST3 performances. However, analyses of statistical significance were performed among F scores across systems participating in MUC-4. The results of these analyses indicate that for the P&R F measure (in which precision and recall are given equal weight), there was no significant difference in performance between the Paramax system and the system submitted by SRA. Similarly, on the 2P&R F measure (in which precision is given more weight), there was no significant difference in performance between the Paramax system and the systems submitted for evaluation by McDonnell-Douglas (MDC) and New Mexico-Brandeis (NM-BR). Finally, on the P&2R F measure (in which recall is given more weight), there was no significant difference in performance between the Paramax system and the system submitted by BBN. Appendix G provides additional information on F scores and how the analyses of statistical significance were performed.

ANALYSIS

The Paramax MUC-4 implementation satisfied the key goal of its developers: a fast rule development cycle. The Paramax MUC-3 system was implemented using a forward-chaining engine called Pfc, which is written in Prolog. Although the Pfc rule formalism has a number of interesting properties, including in particular a mechanism for easily escaping to Prolog in order to use Prolog's built-in factbase and to reason in a backward-chaining fashion, the system as a whole was inefficient. Processing a standard test set of 100 messages using the MUC-3 implementation required 40 hours of processing time running on three separate Sun workstations. In contrast, the Paramax MUC-4 system implemented in CLIPS can process 100 messages in just $3\frac{1}{2}$ hours on one Sun workstation. This dramatic improvement in the rule development cycle made it possible to achieve a respectable level of performance in a small amount of time.

The mid-range performance of the Paramax MUC-4 system could have been significantly improved if additional staffing had been available to better engineer the implementation.[5] After the MUC-4 test, it was determined that a bug existed in the preprocessing code for recognizing sentence boundaries—sentence endings terminated by double quotes were not recognized. Since sentence boundaries play a very important role in determining the relative likelihood of possible slot values, this problem had a significant impact on the accuracy of the system's slot value preferencing heuristics. The problem could have been easily resolved if enough staffing had been available to more carefully examine system output during training runs. Bugs in the forward-chaining rule base were also discovered after the MUC-4 test that would have been easy to correct and that had a dramatic cumulative impact on performance. Examples of such bugs are given in the Paramax MUC-4 system summary.

The Paramax system's high rate of spurious responses was caused by a poor performance in establishing coreference among event descriptions. This poor performance was caused in large part by a lack of time/staffing to develop routine heuristics for merging similar templates. For example, in some cases the Paramax system would generate two identical templates for the same message. In other cases, the same target would arise in two different templates of the same type for the same message (ie, the same building would be bombed, the same individual would be killed, and so forth). Improving the set of heuristics used to establish object coreference will be a top priority for the Paramax team in MUC-5. These improvements should result in a lower rate of spurious responses.

[4] In the case of P&R F scores, for which recall and precision are given equal weight, $\beta = 1.0$.

[5] No formal mechanism exists for determining the level of effort dedicated to the development of MUC-4 systems, and the informal estimates offered by the participating research groups are surely inaccurate. We estimate that implementations which performed better than the Paramax system generally involved double the staffing level—most such systems were developed with government support, which is not the case for the Paramax system.

CONCLUDING REMARKS

The Paramax MUC-4 system takes about $3\frac{1}{2}$ hours to process 100 messages on a Sparc2 with 32MB of memory and a normal CPU load (ie, with a text editor or two in use). The CLIPS-based data extraction component's average elapsed processing time per text in the MUC-4 TST3 data set is 1 minute, 47 seconds. This processing speed permits a fast rule development cycle, which is critical in building knowledge-based systems.

A failure to insure a rapid rule development cycle is a common mistake among research groups that are not accustomed to building large-scale text processing systems. This mistake was made by a number of research groups in MUC-3, and the Paramax team and other research groups, most notably SRI, rectified this mistake in MUC-4. The MUC-4 development strategies of Paramax and SRI were roughly similar: a rapid rule development cycle was insured by stripping away inefficient linguistic analysis techniques. The SRI MUC-4 system performed significantly better than the Paramax submission, but this is very likely a result of greater staffing resources than the consequence of some fundamental difference in approach.

For both the Paramax and SRI research teams, the decision to eliminate linguistic analysis techniques was more a recognition of the primary importance of satisfying the requirements of knowledge-based systems than it was a rejection of linguistic analysis as a useful methodology in text processing. Linguistic analysis is still clearly necessary for achieving finer-grained data extraction capabilities, but additional research must be performed to improve the efficiency and robustness of the techniques. Meanwhile, the data extraction capabilities of systems with only rudimentary linguistic analysis techniques are capable of generating data bases with sufficient detail to cause researchers to begin worrying about system development issues beyond the data extraction process itself. Paramount among these issues is the need to perform object coreference on the database level—in other words, to recognize that multiple database records are describing the same object. Until object cofererence on the data base level becomes a manageable problem, it will be difficult to use the data bases that are now being extracted.

The decision on the part of the Paramax team to build a completely new text processing implementation for MUC-4 was a difficult one to make. Although it was clearly necessary to achieve a fast rule development cycle, it was also clear that building a new implementation in only a couple of months with limited staffing was a high risk venture. But in retrospect, the Paramax team is confident that the right decision was made; system development requirements were prioritized and the need for a rapid rule development cycle came out on top.

What is truly surprising is that the Paramax MUC-4 system did as well as it did, given the level of effort that went into developing it. CLIPS has proven to be an excellent choice for building rule-based text analysis systems: it is an extremely fast forward-chaining engine, and it is easily integrated with other analysis components. Several CLIPS rule modules developed for the MUC-4 system can be reused, particularly the rules used to recognize proper names. Since the MUC-4 test, the Paramax team has implemented a specialized proper name database containing over 9,000 entries in C in order to reduce the size of the CLIPS fact base. This strategy should further improve the modularity and reasoning efficiency of the text processing system.

REFERENCES

[1] Software Technology Branch. CLIPS *Reference Manual.* Lyndon B. Johnson Space Center, September 1991. JSC-25012.

[2] DARPA, Software and Intelligent Systems Technology Office. *Third Message Understanding Conference (MUC-3).* Morgan Kaufmann, May 1991.