CRL/NMSU and Brandeis *MucBruce*: MUC-4 Test Results and Analysis

Jim Cowie, Louise Guthrie, Yorick Wilks
Computing Research Laboratory
New Mexico State University
&
James Pustejovsky
Computer Science Department
Brandeis University

INTRODUCTION

The Computing Research Laboratory (New Mexico State University) and the Computer Science Department (Brandeis University) are collaborating on the development of a system (*DIDEROT*) to perform data extraction for the Tipster project. This system is still far from fully developed, but as many of the techniques being used are domain —and in many cases language— independent, we have assembled them in a preliminary manner to produce a prototype system (*MucBruce*[1]), which handles the MUC-4 texts.

The overall system architecture is shown in Figure 1.

The development of the software and data used for MucBruce has been carried out over a three month period beginning at the end of February, 1992. The present version of the system relies extensively on statistically-based measures of relevance made both at the text and the paragraph level. Texts are tagged for a variety of features by a pipeline of processes. The marked texts and the paragraph relevancy information are used to allow a scan around keywords for appropriate slot filling strings. The system has been augmented since the dry-run with a parser which processes sentences which contain a word with an associated Generative Lexical Semantic (GLS) definition. This component was added by Brandeis late in the development process and has access to approximately 20 lexical definitions.

Our results reflect the extremely simplistic approach to identifying the slot fills in a text. We feel confident, however, that an expansion of the coverage of our GLS entries and the addition of further constraints to prevent template overgeneration will produce significant improvements. We have created a set of tagging and statistical techniques which will apply to any text type, given appropriate training data.

SYSTEM FEATURES

The system consists of three front-end components all of which are C or Lex programs:

- A text relevancy marker
- A paragraph relevancy marker
- A text tagging pipeline

and two MUC specific Prolog programs:

- A template constructor
- A template formatter

[1] We seem to have adopted a philosophical stance for our system nomenclature, and this particular Australian philosopher seemed to embody some of the ad hoc notions which, at the moment, glue our system together.

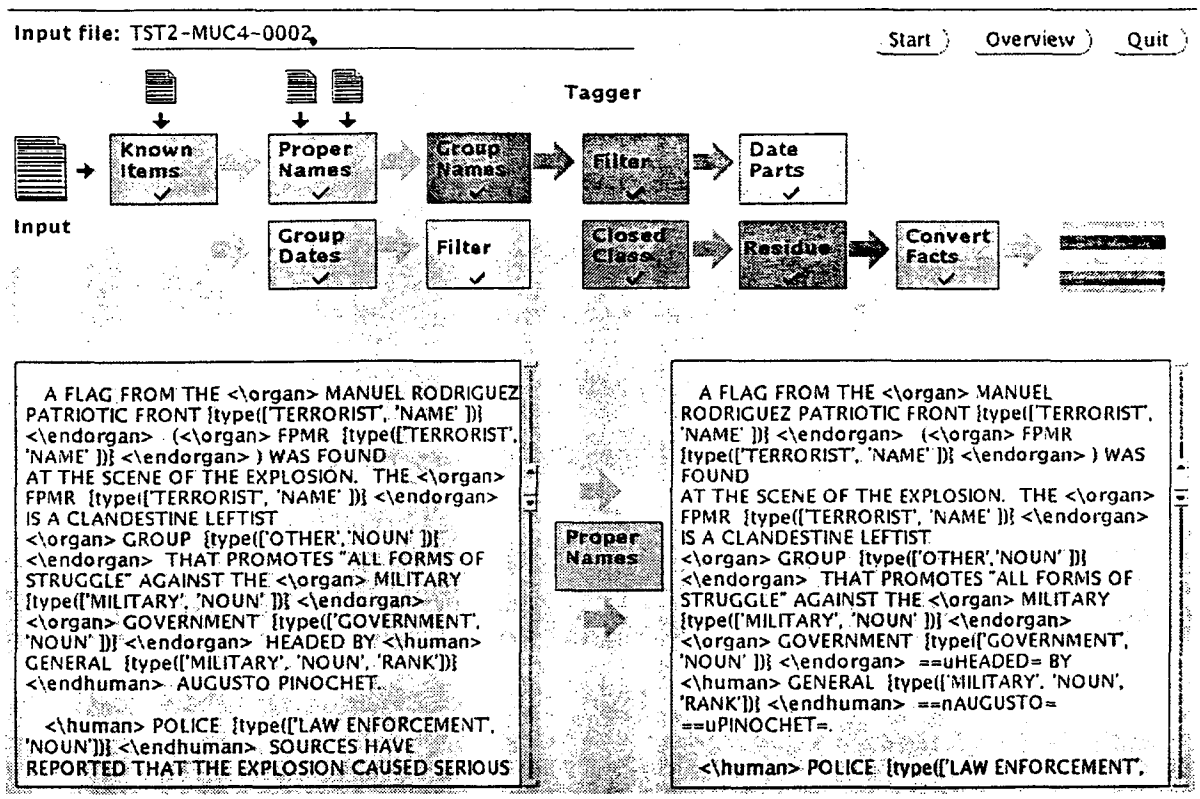

Figure 1: MucBruce - System Overview

One of our principal intentions is to automate as much as possible all the processes associated with the creation of a text extraction system. Our statistical techniques for relevant text recognition use word lists which are automatically derived from training data. Our text tagger uses proper name information derived from the key templates and other taggers for human names and dates are largely domain independent. We intend to derive the entire core lexicon for the system from Machine Readable Dictionaries and then to tune it against appropriate corpora.

OFFICIAL RESULTS

Our results are shown in tables 1 and 2. The results for test 4 are much poorer then those for test 3. We have not established any specific causes for this difference. For most of the individual slots we see some improvement in recall and a greater improvement in precision over the results of the dry run test. The MucBruce system is not parameterized in any way to affect recall or precision. To change these we would require modifying the parameters given to the text statistics programs. For MUC-4 we tried to improve precision at the expense of some recall. It is extremely difficult to measure the accuracy of the template predicting programs, as their performance can be easily masked by errors occurring in the template producing sections of the system. We need to run separate tests of these components to establish the exact relationship on performance of the text statistics, text marking and template producing components. We have not yet, however, had time to carry out these tests on the new MUC-4 data.

EFFORT SPENT

Approximately ten people have worked at one time or another on the MUC-4 system over the last three months. They were all, however, also working on other projects over this period. A rough estimate of the time involved would be six person-months. The major areas of work were in developing and refining

SLOT	POS	ACT	COR	PAR	INC	ICR	IPA	REC	PRE	OVG	FAL
template-id	110	135	58	0	0	0	0	53	43	57	
inc-date	105	131	40	13	4	0	13	44	35	56	
inc-loc	110	128	10	39	7	0	11	27	23	56	
inc-type	110	135	48	10	0	0	0	48	39	57	6
inc-stage	110	135	57	0	1	0	0	52	42	57	17
inc-instr-id	33	47	11	2	2	0	2	36	26	68	
inc-instr-type	54	47	11	4	1	1	0	24	28	66	1
perp-inc-cat	70	135	34	0	7	0	0	48	25	70	33
perp-ind-id	84	14	1	0	3	1	0	1	7	71	
perp-org-id	53	65	13	0	9	1	0	24	20	66	
perp-org-conf	51	0	0	0	0	0	0	0	*	*	0
phys-tgt-id	71	76	12	4	11	0	4	20	18	64	
phys-tgt-type	71	76	16	3	8	6	0	25	23	64	2
phys-tgt-num	70	0	0	0	0	0	0	0	*	*	
phys-tgt-nation	2	0	0	0	0	0	0	0	*	*	0
phys-tgt-effect	40	63	0	1	9	0	0	1	1	84	5
phys-tgt-total-num	0	0	0	0	0	0	0	*	*	*	
hum-tgt-name	56	52	14	4	2	2	4	28	31	62	
hum-tgt-desc	126	88	8	6	22	1	6	9	12	59	
hum-tgt-type	136	88	21	0	17	7	0	15	24	57	3
hum-tgt-num	135	0	0	0	0	0	0	0	*	*	
hum-tgt-nation	18	0	0	0	0	0	0	0	*	*	0
hum-tgt-effect	113	140	11	22	11	5	0	19	16	68	6
hum-tgt-total-num	1	0	0	0	0	0	0	0	*	*	
perp-total	258	214	48	0	19	2	0	19	22	69	
phys-tgt-total	254	215	28	8	28	6	4	12	15	70	
hum-tgt-total	585	368	54	32	52	15	10	12	19	62	
MATCH/MISS	1619	665	307	108	114	24	40	22	54	20	
MATCH/SPUR	928	1420	307	108	114	24	40	39	25	63	
MATCH ONLY	928	665	307	108	114	24	40	39	54	20	
ALL TEMPLATES	1619	1420	307	108	114	24	40	22	25	63	
SET FILLS	775	380	198	40	54	19	0	28	57	23	0
STRING FILLS	423	172	59	16	49	5	16	16	39	28	
TEXT FILTER	66	69	55	*	*	*	*	83	80	20	41

F-MEASURES: P&R 23.4, 2P&R 24.34, P&2R 22.54

Table 1: TEST 3 Summary Scores

the statistical techniques, designing and developing the tagging software and implementing a system which could use our current incomplete set of components. Work also went into designing and implementing an appropriate form for the Generative Lexical Semantic entries.

Our limiting factor was definitely time. In the last month we generalized the lexical entries in our tagging file. This meant our system was often likely to recognize partial strings as being appropriate fillers (e.g. GUERILLAS). We intended to avoid this problem by incorporating the BBN part of speech tagger (POST) into our MUC-4 system and to write code to glue together noun phrases occurring around our new general tags. All this code was written and tested just before the MUC-4 final test, but we were unable to incorporate it in time.

The training texts were used to generate our statistical information and word lists. The methods used are automatic and require only the setting of thresholds for word selection.

The system has improved its performance slightly since the dry run test. Many of our changes in isolation are detrimental and require the addition of other techniques to establish their usefulness.

SLOT	POS	ACT	COR	PAR	INC	ICR	IPA	REC	PRE	OVG	FAL
template-id	76	144	47	0	0	0	0	62	33	67	
inc-date	76	144	26	11	10	1	11	41	22	67	
inc-loc	76	140	1	41	5	0	9	28	15	66	
inc-type	76	144	37	10	0	0	0	55	29	67	8
inc-stage	76	144	45	0	2	0	0	59	31	67	22
inc-instr-id	32	54	15	0	1	0	0	47	28	70	
inc-instr-type	52	54	15	1	1	0	0	30	29	68	1
perp-inc-cat	61	144	34	0	4	0	0	56	24	74	38
perp-ind-id	55	10	0	0	3	0	0	0	0	70	
perp-org-id	53	71	7	1	10	0	1	14	10	75	
perp-org-conf	52	0	0	0	0	0	0	0	*	*	0
phys-tgt-id	60	72	10	8	12	0	8	23	19	58	
phys-tgt-type	60	69	11	6	11	1	4	23	20	59	2
phys-tgt-num	60	0	0	0	0	0	0	0	*	*	
phys-tgt-nation	2	0	0	0	0	0	0	0	*	*	0
phys-tgt-effect	44	48	0	0	11	0	0	0	0	77	4
phys-tgt-total-num	0	0	0	0	0	0	0	*	*	*	
hum-tgt-name	33	36	6	2	4	1	2	21	19	67	
hum-tgt-desc	71	81	8	3	13	0	3	13	12	70	
hum-tgt-type	74	81	6	8	10	0	6	14	12	70	3
hum-tgt-num	78	0	0	0	0	0	0	0	*	*	
hum-tgt-nation	6	0	0	0	0	0	0	0	*	*	0
hum-tgt-effect	72	131	1	24	6	0	3	18	10	76	6
hum-tgt-total-num	6	0	0	0	0	0	0	0	*	*	
inc-total	388	680	139	63	19	1	20	44	25	68	
perp-total	221	225	41	1	17	0	1	19	18	74	
phys-tgt-total	226	189	21	14	34	1	12	12	15	63	
hum-tgt-total	340	329	21	37	33	1	14	12	12	72	
MATCHED/MISS	1175	502	222	115	103	3	47	24	56	12	
MATCHED/SPUR	792	1423	222	115	103	3	47	35	20	69	
MATCHED ONLY	792	502	222	115	103	3	47	35	56	12	
ALL TEMPLATES	1175	1423	222	115	103	3	47	24	20	69	
SET FILLS	575	285	149	49	45	1	13	30	61	15	0
STRING FILLS	304	123	46	14	43	1	14	17	43	16	
TEXT FILTERING	56	80	53	*	*	*	*	95	66	34	61

F-MEASURES: P&R 21.82 2P&R 20.69 P&2R 23.08

Table 2: TEST 4 Summary Scores

CONCLUSIONS

The basic system is essentially domain-independent and around 80% of it should be directly usable in other applications. The module which needs the greatest amount of work is the template creator. Much of this will be replaced as we develop our system for Tipster. It would have been nice to see the effect of adding the part of speech tagger and the noun phrase recognizer to the system.

The test deadlines and the availability of the MUC-3 corpus have proved extremely useful to our research efforts, both encouraging us to get a robust working system together and to look critically at its performance.