Structural Metadata at CUED: Progress Report

Marcus Tomalin, Sue Tranter, Phil Woodland,
& the CUED STT Team (including Ji-Hwan Kim)

May 21st 2003

Cambridge University Engineering Department

Progress: Jan 2003 - May 2003

- Specific CUED Structural Metadata Research:
  - Main focus on Slash Unit (SU) detection/classification.
  - Prosodic Feature Model (PFM).
  - SU Language Model (SULM).
  - SU Decoder.

- General MACEARS Structural Metadata Tasks:
  - Involved in SimpleMDE Annotation spec discussions.
  - Involved in the SimpleMDE pilot annotation.
  - Involved in the tool testing process.

Where were we in Jan 2003?

The CUED RT-03 dryrun SU system used word-time information and token-spotting algorithms:

N : gap of N seconds in transcriptions $\longrightarrow$ SU
SENT: SENT_START or SENT_END tag in STT output $\longrightarrow$ SU

```
QUES     = {WHAT WHY WHERE WHEN HOW DO ARE IS HAVE DID HAS REALLY}
CO-CONJ  = {AND BUT OR}
SUB-CONJ= {IF HOWEVER THEREFORE}
ART      = {THE A AN}
QUANT    = {ANY ALL MOST EVERY}
INCOMP   = {$CO-CONJ $SUB-CONJ $ART $QUANT}

RULE1: su = question if ( su-initial word == QUES )
RULE2: su = incomplete if ( su-final word == INCOMP )
RULE3: su = backchannel if ( su == BC+ )
RULE4: su = statement if (su not already classified)
```

Training and Test Data

Data Sets:

- Training data: subset of Treebank3 (TB3) corpus (c.90 hours).
- 'Held out' data: subset of TB3 corpus (c.1 hour).
- Test data: RT-03 dry-run test set.

Some problems with this:

- The training and test data sets are not annotated in exactly the same way
  - but we needed training data!
- Backchannels not labelled separately in the training data.
- Only the test data has reference ctm/mdtm files
  - so system tuning has to be performed upon the test data.

SU System Overview

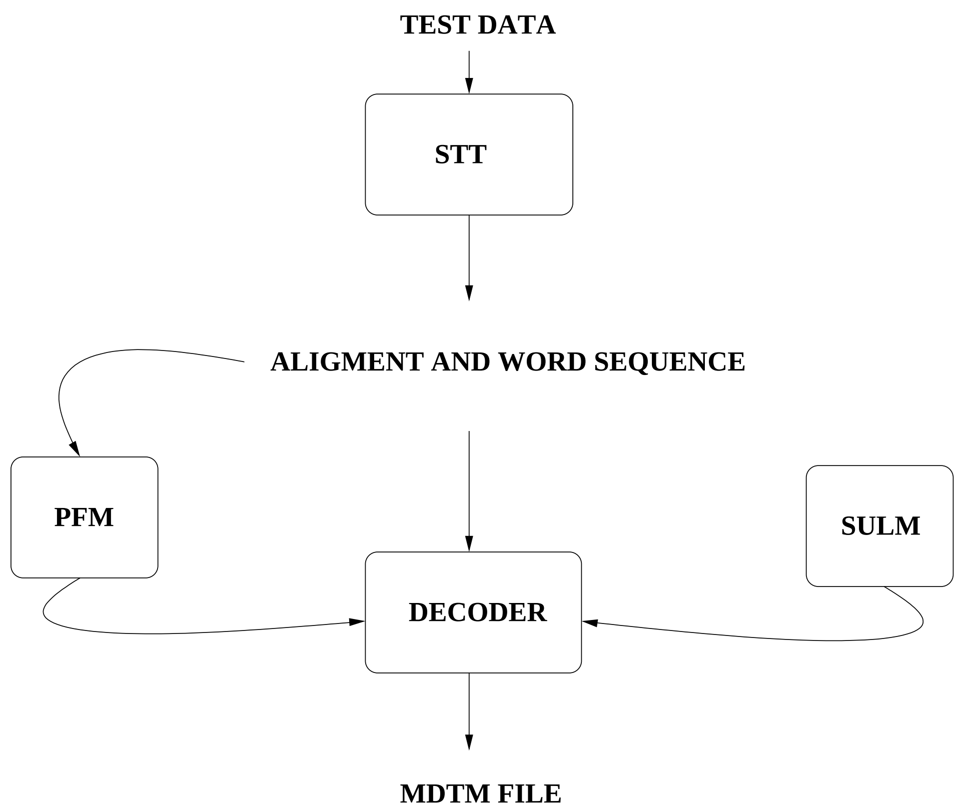

Figure 3: SU Detection System

The Prosodic Feature Model

The Prosodic Features (PFs):

Prosodic Feature	Description
Pause_Length	the pause length at the end of the word
Duration	the duration from the previous pause
Avg_F0_L	the mean of the good F0 values in left window
Avg_F0_R	the mean of the good F0 values in right window
Avg_F0_ratio	Avg_F0_L / Avg_F0_R
Cnt_F0_L	the number of good F0s in left window
Cnt_F0_R	the number of good F0s in right window
Eng_L	the RMS energy in left window
Eng_R	the RMS energy in right window
Eng_ratio	Eng_L / Eng_R

[Following Shriberg et al. 1998, Kim 2001]

The Prosodic Feature Model

4 SU types defined:

- **SU_S**: statement SU boundary
- **SU_Q**: question SU boundary
- **SU_I**: incomplete SU boundary
- **SU_N**: no SU-boundary

Steps in the PFM construction process:

- Convert training data into word sequences.
- Classify each word into one of the above SU sub-types.
- Obtain Forced Alignments for training data word sequences.
- Extract PF info using word start/end times.
- Construct CART decision tree using PFs and SU sub-type classification.

The Prosodic Feature Model

1456 Nodes: (728 non-terminal + 729 terminal)

Measures for determining the contribution of the PFs:

- Feature Appearance:
  the number of times a feature is used as a classifying feature.

- Feature Usage:
  the proportion of the number of times a feature is queried.

The Prosodic Feature Model

Prosodic Feature	Feature Appearance	Feature Usage
Pause_Length	180	**0.615**
Duration	115	0.094
Avg_F0_L	67	0.001
Avg_F0_R	62	0.014
Avg_F0_ratio	52	0.018
Cnt_F0_L	36	0.066
Cnt_F0_R	29	0.018
Eng_L	63	0.033
Eng_R	70	**0.116**
Eng_ratio	54	0.003

Table 1: Prosodic Feature Usage

The SU Language Model

Training Data Preparation:

Insert the required SU token after every word in the training data:

Example:

< s > OKAY **SU_S** ARE **SU_N** WE **SU_N** READY **SU_Q** I **SU_N** THINK **SU_N** WE **SU_N** SHOULD **SU_N** GIVE **SU_I** OKAY **SU_S** ... < /s >

Number of words in training data: 348,231

Three kinds of SULM were constructed:

- N-gram SULM
- Class-based SULM
- Interpolated N-gram + class-based SULM

The SU Language Model

SULM Type	Perplexity	Classes	Interpolation Weights
bg	29.1	N/A	N/A
tg	21.2	N/A	N/A
40cl-bg	28.3	40	N/A
40cl-tg	31.4	40	N/A
bg + 40cl-bg	27.7	40	$\sim$0.3, $\sim$0.7
tg + 40cl-tg	20.8	40	$\sim$0.9, $\sim$0.1
bg + 40cl-tg	28.3	40	$\sim$0.7, $\sim$0.3

Table 2: The SULMs

- Training data used to build SULMs.
- 'Held out' data used to obtain Perplexity (PP) values.
- The PPs are low (compared to typical STT perplexities) because the probability of the inter-word SU tokens is high.

The SU Decoder

The basic method used to combine the PFM and the SULM:

- Obtain STT output for test data.
- Obtain PFM scores (for the 4 SU sub-types) for each word in STT output.
- Create initial lattices using PFM scores and STT test data word sequences.
- Expand the initial lattices, using the SULM and standard lattice tools, to create a network.
- Select the best path (i.e., highest prob) through the expanded lattice.
- Output word and SU token sequence corresponding to the best path.
- Identify Backchannels in post-processing stage (token-spotting).

The SU Decoder

PFM scores are added to the arcs of the initial lattice:

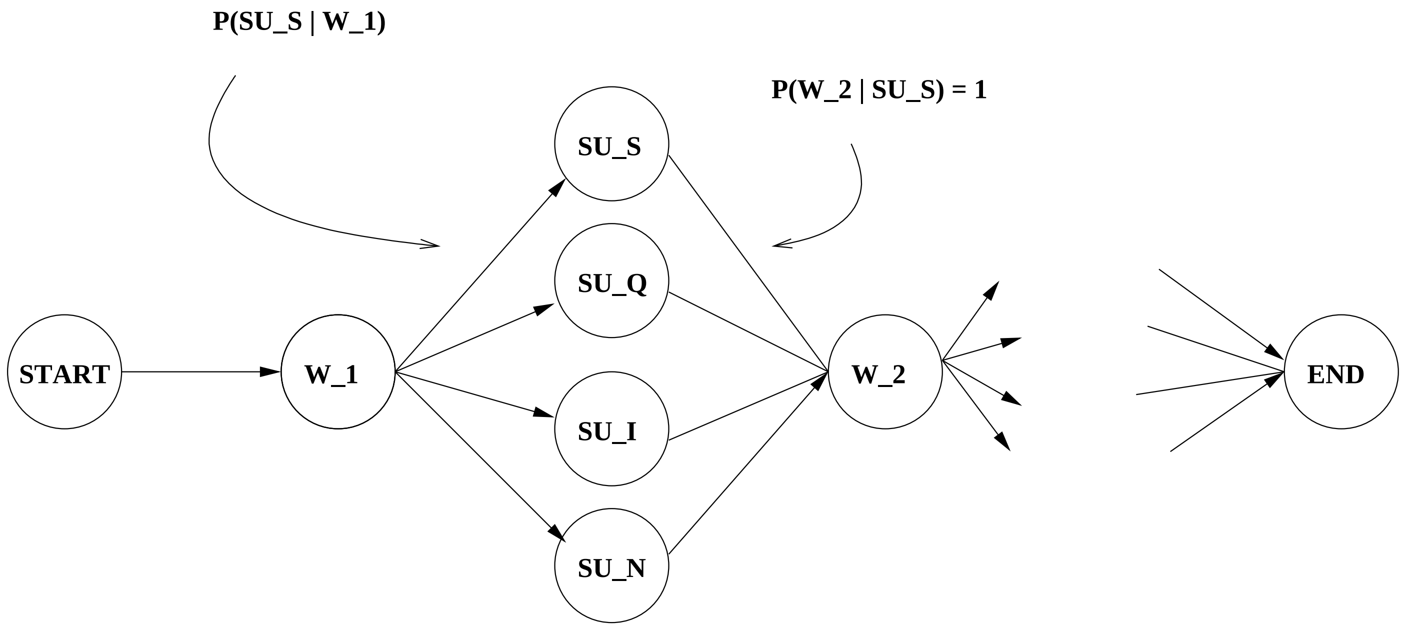

Figure 3: Initial SU Decoder lattice

The SU Decoder

The Grammar Scale Factor (GSF) constant weights the PFM and SULM scores:

$$\log \text{PFM\_score} + (\text{GSF} \times \log \text{SULM\_score})$$

The GSF can be varied (NB: this is tuning on the test data!)

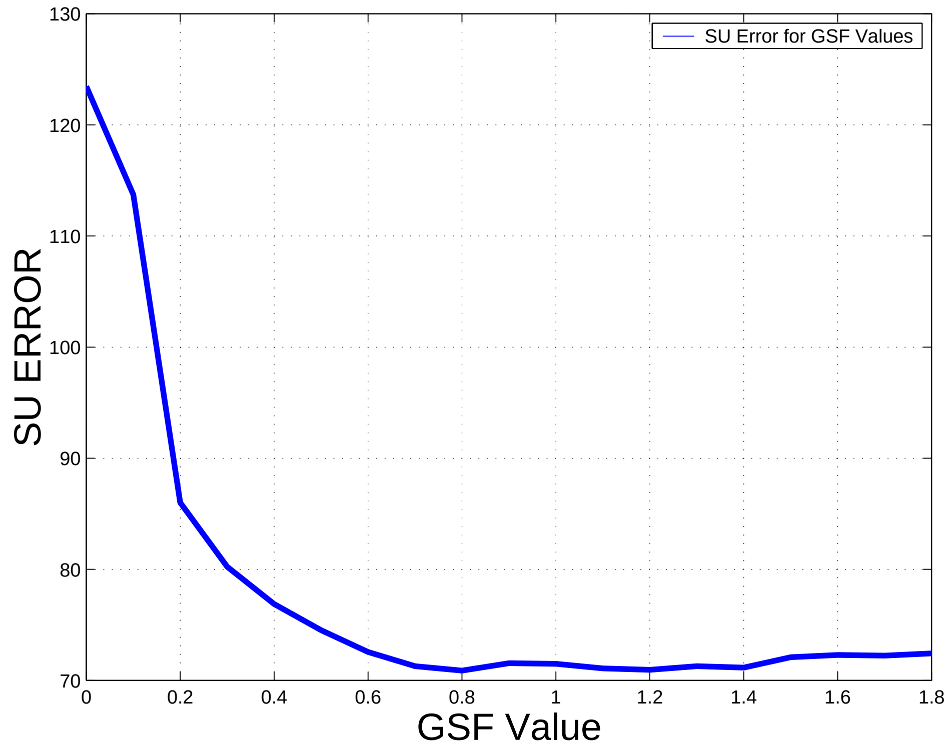

Figure 2: SU Error for Different Grammar Scale Factors†

SU Results

System	GSF	%Del	%Ins	%Sub	%Err
CUED Dryrun*	N/A	32.08	31.67	21.59	**85.34**
PFM	N/A	24.88	43.98	54.61	123.47
SULM_bg	N/A	81.30	6.32	3.56	91.19
SULM_40cl-tg	N/A	84.47	6.28	3.86	94.61
SULM_bg+40cl-tg	N/A	86.35	4.51	2.96	93.81
PFM+SULM_bg	0.8	38.94	16.41	15.20	**70.54**
PFM+SULM_40cl-tg	1.2	38.12	19.91	14.92	72.95
PFM+SULM_bg+40cl-tg	1.0	43.78	13.85	14.26	71.89

Table 3: SU Results†

∗ a debugged and tuned version of the dryrun system

† these results differ from those presented at the May workshop since they use a more recent version of the su-eval-v01.pl tool

Conclusions

- Standard Lattice-based Viterbi search techniques enable PFM and SULM scores to be combined easily.
- PFMs and SULMs model complementary information.
- Interpolated SULMs can be used to reduce SU %Err.
- Bigram SULMs give largest reductions in %Err when combined with the PFM (using the current training and test data!).
- The current CUED SU System achieves lower %Err values than the type of system used for the dryrun.

Future Plans

- Continue to participate in annotation/tools discussions.
- Develop the PFM (i.e., experiment with other kinds of features).
- Investigate different ways of calculating interpolation weights for SULMs.
- Explore different kinds SULMs (i.e., techniques for training with sparse data).
- Explore different lattice structures (i.e., 'skips' instead of **SU_N** tokens).
- Consider impact of STT performance upon the SU detection task.
- Use syntactic parsing techniques in post-processing stage to reassign SUs in decoder output to different sub-types.
- Start to focus on the disfluency subset of Structural Metadata tasks.