# A WFST-based Log-linear Framework for Speaking-style Transformation

Graham Neubig, Shinsuke Mori, Tatsuya Kawahara
*Graduate School of Informatics, Kyoto University, Japan*

## Overview

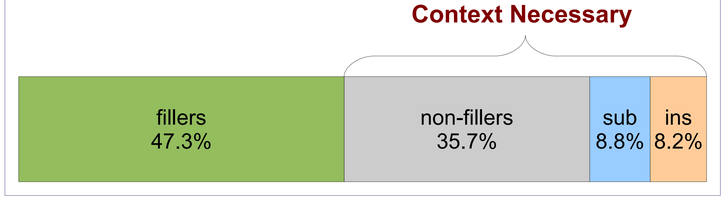
- **Objective:** Transform spoken-style language (V) into written style language (W) for the creation of transcripts
- **Approach:** Statistical machine translation to "translate" from verbatim text to written text
- **Innovations:**
  - *Log-linear modeling* for improved accuracy
  - Introduction of *features to handle common phenomena* in speaking-style transformation
  - *WFST-based implementation* for integration with WFST-based speech recognizers
- **Evaluation** on transformation of Japanese verbatim transcripts showed improvement over traditional methods

## Necessary Transformations

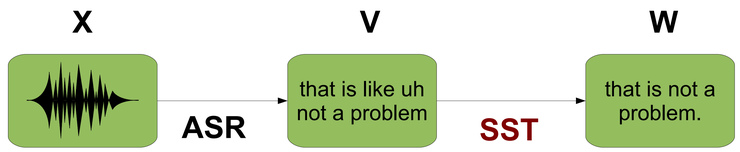
| various | ahh | things | by | order | -obj | make | if | it is | |
|---|---|---|---|---|---|---|---|---|---|
| いろんな<br>*ironna* | あー<br>*a-* | こと<br>*koto* | で<br>*de* | 注文<br>*chu-mon* | | つける<br>*tsukeru* | と<br>*to* | です ね<br>*desu ne* | … |
| いろいろ な<br>*iroiro na* | | こと<br>*koto* | で<br>*de* | 注文<br>*chu-mon* | を<br>*o* | つける<br>*tsukeru* | と<br>*to* | | … |
| sub | fill | | | | ins | | | non-fill | |

- **Filler Deletions:** Words that are consistently used as fillers: *"e-to" "ano-"*
- **Other Deletions:** Words that are fillers or not depending on context, repeats, repairs, etc.
- **Substitutions:** Colloquial expressions, etc.
- **Insertions:** Dropped words, particularly particles in Japanese: *"o" "wa" "ga"*

WER before transformation: 16.40%




## Speaking Style Transformation




- **Noisy Channel Modeling:**

$$\hat{W} = \underset{W}{argmax}\, P(W|V)$$
$$= \underset{W}{argmax}\, P(V|W)\,P(W)$$

Translation Model — Language Model

- **Log Linear Modeling:**

$$\hat{W} = \underset{W}{argmax} \log(P(V|W)) + \log(P(W))$$
$$= \underset{W}{argmax}\, \lambda_1 \log(P(V|W)) + \lambda_2 \log(P(W))$$

## SST-Specific Features

Extra features can be added to the log-linear model:

$$\hat{W} = \underset{W}{argmax}\, \lambda_1 \log(P(V|W)) + \lambda_2 \log(P(W)) + \lambda_3 f_3(V,W) + \ldots$$

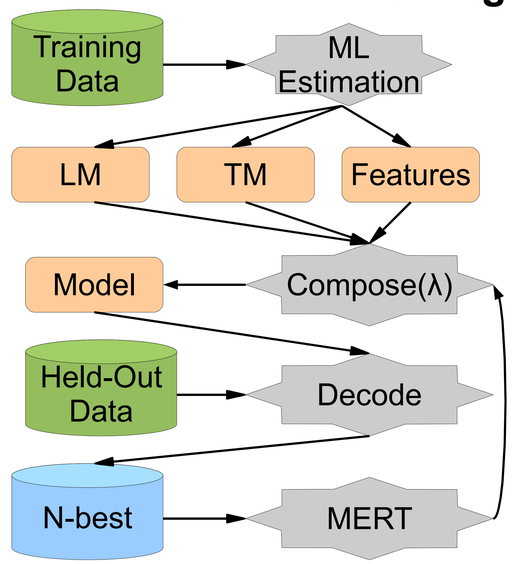
- **Filler Dictionary:** f(V,W) is equal to the number of fillers (from a 23-word list) present in W.
- **Transformation Groups:** f(V,W) is equal to the number of groups of words transformed.

that is like um uh maybe um not a problem

- **Transformation Types:** Insertions, deletions, substitutions are given separate penalties, allowing adjustment of the precision/recall of each type.
- **Decomposed Translation Model:** Use separate log-linear weights for each frequency used when calculating the translation model.

$$\log P(V|W) = \log \prod_{i=1}^{k} P(v_i, w_i)/P(w_i)$$
$$= \lambda_1 \log \prod_{i=1}^{k} P(v_i, w_i) - \lambda_2 \log \prod_{i=1}^{k} P(w_i)$$

## Training




3-gram Kneser-Ney LM

Each FST weighted separately during composition.

Beam-search decoding over FST models

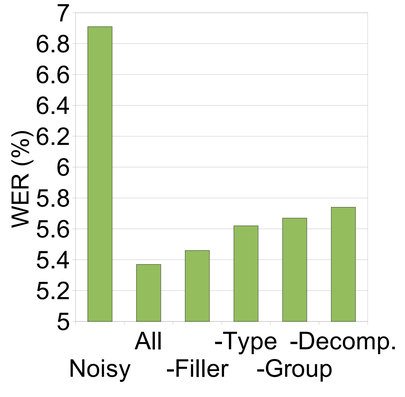
Minimum Error Rate Training on WER

**LM:** Kylm (http://www.phontron.com/kylm) **FST:** OpenFst (http://www.openfst.org)
**Decoder:** Kyfd (http://www.phontron.com/kyfd) **MERT:** Moses (http://www.statmt.org/moses)

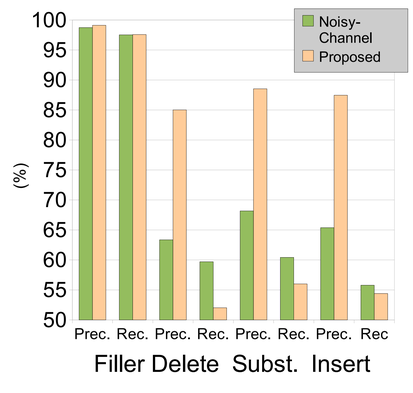
## Evaluation

Committee meetings of the Japanese National Diet
Verbatim transcripts as input, official transcripts as output

- **3.62M** sentences for LM training
- **56.2k** aligned sentences for TM training (**974** held-out)
- **7181** testing sentences from meetings after the training data




WER for noisy channel, all features, and each feature removed individually




Precision and recall for each type