Improved Statistical Models for SMT-Based Speaking Style Transformation

Graham Neubig, Yuya Akita, Shinsuke Mori, Tatsuya Kawahara

School of Informatics, Kyoto University, Japan

1. Overview of Speaking-Style Transformation

Speaking Style Transformation (SST)

- ASR is generally modeled to find the verbatim utterance V given acoustic features X
- In many cases verbatim speech is difficult to read:

V

ya know when I was asked earlier about uh the issue of coal uh you under my plan uh of a cap and trade system ...

- In order to create usable transcripts from ASR results, it is necessary to transform *V* into clean text *W*

W

When I was asked earlier about the issue of coal under my plan of a cap and trade system, ...

Previous Research

- **Detection-Based Approaches**
  - Focus on deletion of fillers, repeats, and repairs, as well as insertion of punctuation
  - Modeled using noisy-channel models [Honal & Schultz 03, Maskey et al. 06], HMMs, and CRFs [Liu et al. 06]
- **SMT-Based Approaches**
  - Treat spoken and written language as different languages, and "translate" between them
  - Proposed by [Shitaoka et al. 04] and implemented using WFSTs and log-linear models in [Neubig et al. 09]
  - Is able to handle colloquial expression correction, insertion of dropped words (important for formal settings)

Research Summary

- Propose **two enhancements of the statistical model** for finite-state SMT-based SST
  - **Incorporation of context** in a noisy channel model by transforming context-sensitive joint probabilities to conditional probabilities
  - Allowing **greater emphasis on frequent patterns** by log-linearly interpolating joint and conditional probability models
- Evaluation of the proposed methods on both verbatim transcripts and ASR output for the Japanese Diet (national congress)

2. Noisy-Channel and Joint-Probability Models for SMT

Noisy Channel Model

- Statistical models for SST attempt to maximize $P(W|V)$
- Training requires a parallel corpus of *W* and *V*
  - It is generally easier to acquire a large volume of clean transcripts (*W*) than a parallel corpus (*W* and *V*)
  - Bayes' law is used to decompose the probabilities

$$\hat{W} = \operatorname*{argmax}_{W} P(W|V)$$

$$= \operatorname*{argmax}_{W} P_t(V|W)\, P_l(W)$$

Translation Model (TM) Language Model (LM)

- $P_l(W)$ is estimated using an *n*-gram (3-gram) model

Probability Estimation for the TM

- $P_t(V|W)$ is difficult to estimate for the whole sentence
  - Assume that the word TM probabilities are independent
  - Set the sentence TM probability equal to the product of the word TM probabilities

$$P_t(V|W) \approx \prod_i P_t(v_i|w_i)$$

- However, the word TM probabilities are actually not context independent

$P_t(\text{like}|\,\varepsilon\,)$

I like told him that I really like his new hairstyle.

$P_t(\text{like}|\,\varepsilon, H_1\,)$ **(large)** $\quad$ $P_t(\text{like}|\,\varepsilon, H_2\,)$ **(small)**

Joint Probability Model [Casacuberta & Vidal 2004]

- The joint probability model is an alternative to the noisy-channel model for speech translation

$$\hat{W} = \underset{W}{\operatorname{argmax}} P_t(W, V)$$

- Sentences are aligned into matching words or phrases

V=	*ironna*	*e-*	*koto*	*de*	*chumon*		*tsukeru*	*to*	*desu*	*ne*	*...*
W=	*iroiro na*		*koto*	*de*	*chumon*	*o*	*tsukeru*	*to*			*...*

- A sequence *Γ* of word/phrase pairs is created

Γ= ironnna/iroiro_na e-/ε koto/koto de/de chumon/chumon ε/o tsukeru/tsukeru to/to desu/ε ne/ε

Joint Probability Model (2)

- The probability of Γ is estimated using a smoothed *n*-gram model trained on Γ strings

$$P_t(W,V) = P_t(\Gamma) \approx \prod_{k=1}^{K} P_t(\gamma_k | \gamma_{k-n+1}^{k-1})$$

- Context information is contained in the joint probability
- However, this probability can only be trained on parallel text (an LM probability cannot be used)

$$\underset{W}{\operatorname{argmax}}\, P_t(W|V) \neq \underset{W}{\operatorname{argmax}}\, P_t(W,V) P_l(W)$$

- It is desirable to have a **context-sensitive** model that can be **used with a language model**

3. A Context-Sensitive Translation Model

Context-Sensitive Conditional Probability

- It is possible to model the conditional (TM) probability from right-to-left, similarly to the joint probability

$$
\begin{aligned}
P_t(V|W) &= \prod_{i=1}^{k} P_t(v_i | v_1, \dots, v_{i-1}, w_1, \dots, w_k) \\
&= \prod_{i=1}^{k} P_t(v_i | y_1, \dots, y_{i-1}, w_i, \dots, w_k)
\end{aligned}
$$

Context Information

Prediction Unit

$$
\begin{array}{ccccccc}
\cdots & v_{i-2} & v_{i-1} & \boxed{v_i} & v_{i+1} & v_{i+2} & \cdots \\
\cdots & w_{i-2} & w_{i-1} & w_i & w_{i+1} & w_{i+2} & \cdots
\end{array}
$$

Independence Assumptions

- To simplify the model, we make two assumptions
  - Assume that word probabilities rely only on preceding words

$$P_t(V|W) \approx \prod_{i=1}^{k} P_t(v_i | \gamma_1, \ldots, \gamma_{i-1}, w_i)$$

  - Limit the history length

$$P_t(V|W) \approx \prod_{i=1}^{k} P_t(v_i | \gamma_{i-n+1}, \ldots, \gamma_{i-1}, w_i)$$

$$\begin{array}{ccccccc} \cdots & v_{i-2} & v_{i-1} & \boxed{v_i} & v_{i+1} & v_{i+2} & \cdots \\ \cdots & w_{i-2} & w_{i-1} & w_i & w_{i+1} & w_{i+2} & \cdots \end{array}$$

Calculating Conditional Probabilities from Joint Probabilities

- It is possible to decompose this equation into its numerator and denominator

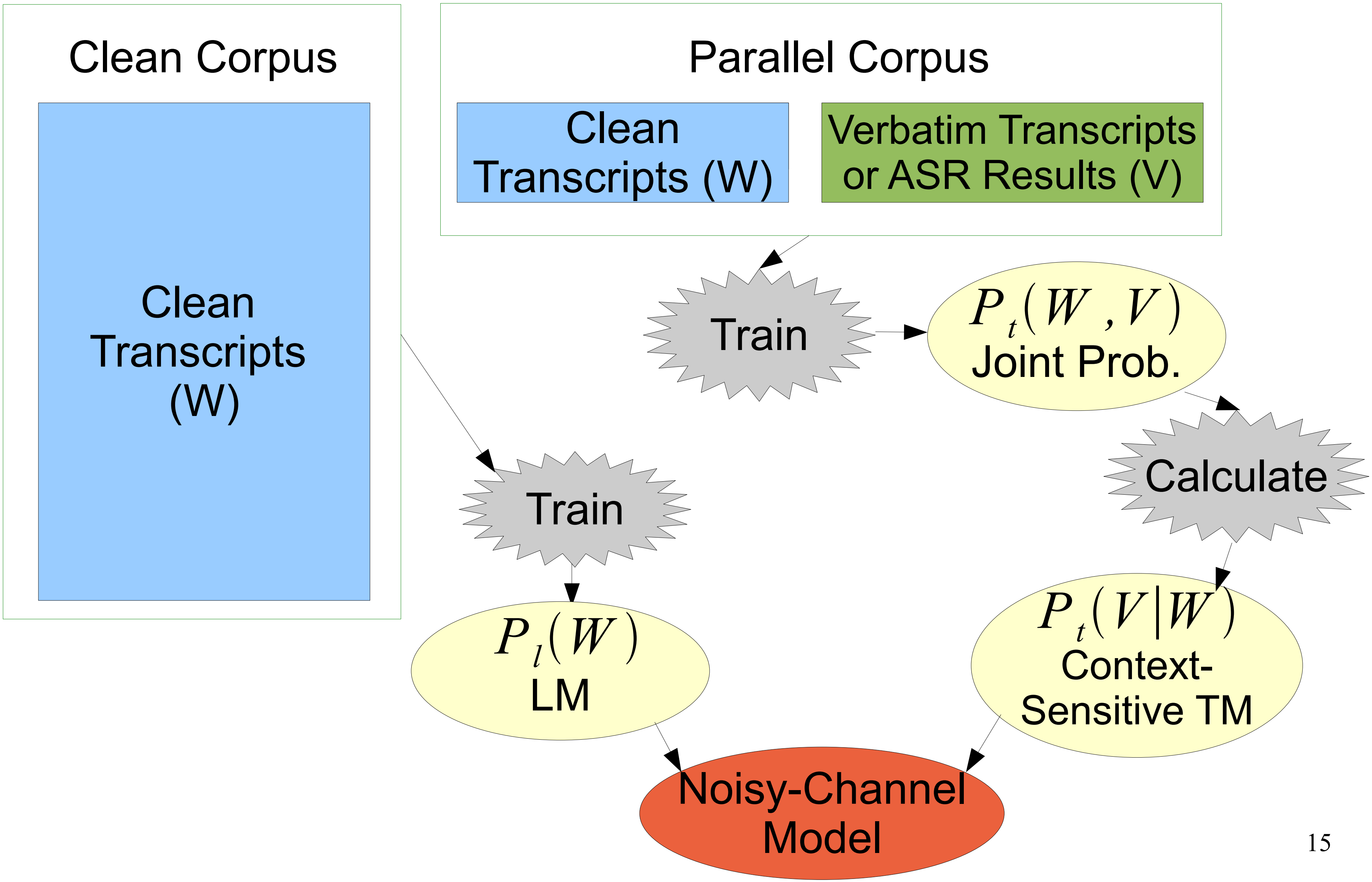
$$P_t(v_i|\gamma_{i-n+1},\ldots,\gamma_{i-1},w_i)=\frac{P_t(\gamma_i|\gamma_{i-n+1},\ldots,\gamma_{i-1})}{P_t(w_i|\gamma_{i-n+1},\ldots,\gamma_{i-1})}$$

- The numerator is equal to the joint *n*-gram probability, while the denominator can be marginalized

$$P_t(v_i|\gamma_{i-n+1},\ldots,\gamma_{i-1},w_i)=\frac{P_t(\gamma_i|\gamma_{i-n+1},\ldots,\gamma_{i-1})}{\sum_{\tilde{\gamma}\in\{\tilde{\gamma}:\langle\tilde{v},w_i\rangle\}}P_t(\tilde{\gamma}|\gamma_{i-n+1},\ldots,\gamma_{i-1})}$$

- This conditional probability **uses context information** and **can be combined with a language model**

Training the Proposed Model



Log-Linear Interpolation with the Joint Probability

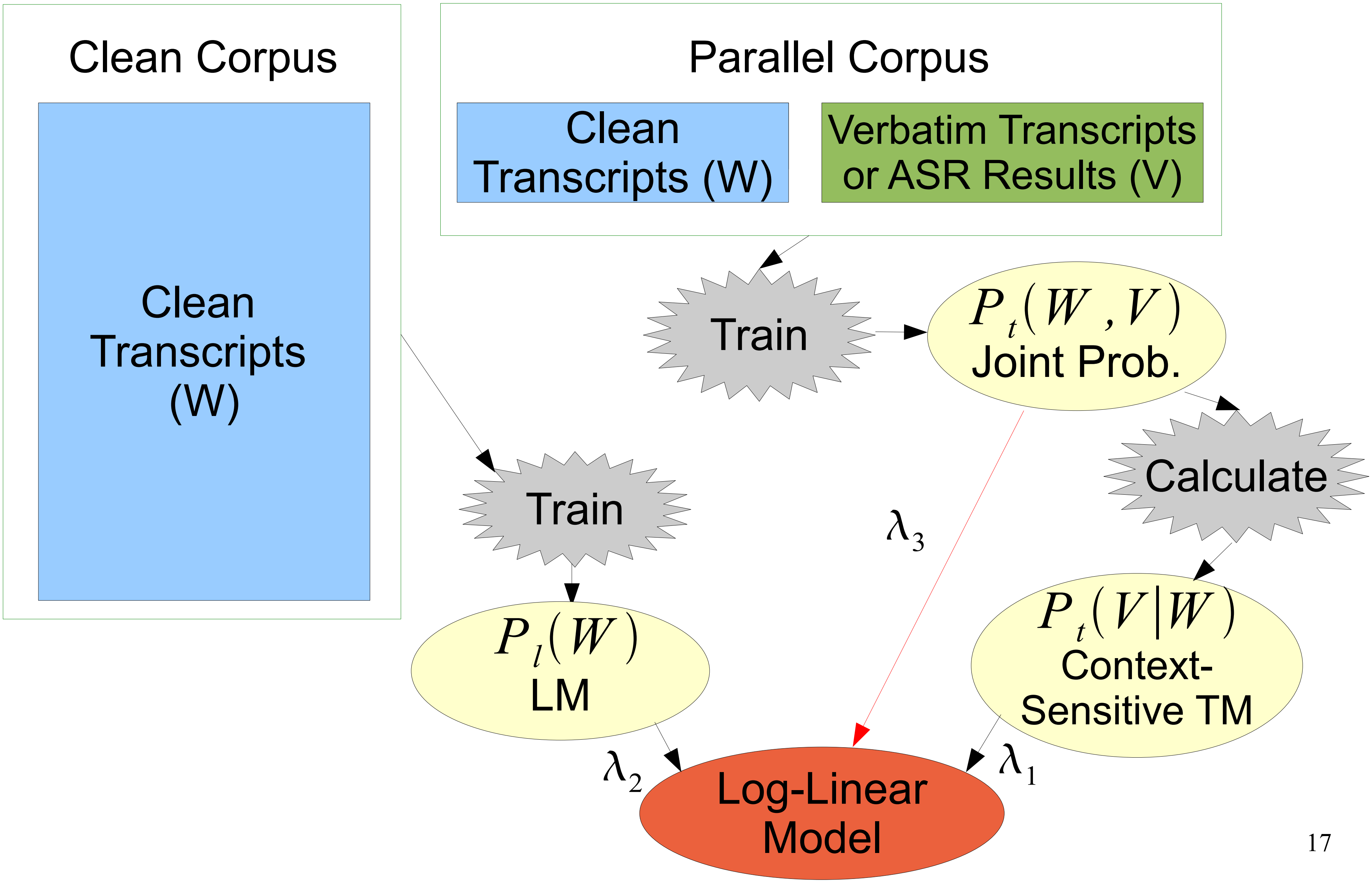
- The joint probability contains information about pattern frequency not present in the conditional probability

$c(\gamma_1) = 100$	$c(\gamma_2) = 1$	$P_t(v_1 \mid w_1) = P_t(v_2 \mid w_2)$
$c(w_1) = 1000$	$c(w_2) = 10$	$P_t(\gamma_1) \neq P_t(\gamma_2)$

- High-frequency patterns are more reliable
- The strong points of both models can be utilized through log-linear interpolation

$$\log(P(W|V)) \propto \underbrace{\boxed{\lambda_1 \log(P_t(V|W)) + \lambda_2 \log(P_l(W))}}_{\text{Noisy-Channel Model}} + \underbrace{\boxed{\lambda_3 \log(P_t(V,W))}}_{\text{Joint Probability}}$$

Training the Proposed Model



4. Evaluation

Experimental Setup

- Verbatim transcripts and ASR output of meetings from the Japanese Diet were used as a target

Data Type	Size	Time Period
LM Training	158M	1/1999 - 8/2007
TM Training	2.31M	1/2003 - 10/2006
Weight Training	66.3k	10/2006-12/2006
Testing	300k	10/2007

- TM training:
  - Verbatim system: Verbatim transcripts and clean text
  - ASR system: **ASR output and clean text**
- Baseline: noisy channel, 3-gram LM, 1-gram TM

Effect of Translation Models (Verbatim Transcripts)

- 4 models were compared
  - A) The context-sensitive noisy-channel model
  - B) **A** with log-linear interpolation of the LM and TM
  - C) The joint-probability model
  - D) **B** and **C** log-linearly interpolated
- Evaluated using edit distance from the clean transcript (WER), with no editing, the WER was **18.62%**

Model	LL	TM n-gram order		
		1-gram	2-gram	3-gram
A. Noisy-Channel (Noisy)		6.51%	5.33%	5.32%
B. Noisy-Channel (Noisy LL)	★	5.99%	5.15%	5.13%
C. Joint Probability (Joint)		9.89%	4.70%	4.60%
D. B+C (Noisy+Joint LL)	★	5.81%	4.12%	**4.05%**

Effect of Translation Models (ASR Output)

- The WER between ASR output and verbatim transcripts (ASR WER) was **17.10%**
- ASR output and clean transcripts was **36.10%**

Model	LL	TM n-gram Order		
		1-gram	2-gram	3-gram
A. Noisy-Channel (Noisy)		21.83%	21.00%	21.09%
B. Noisy-Channel (Noisy LL)	★	21.63%	20.97%	21.09%
C. Joint Probability (Joint)		28.61%	22.62%	21.98%
D. B+C (Noisy+Joint LL)	★	21.32%	20.04%	**20.03%**

- The noisy-channel model was more effective than the joint-probability model for ASR output

Comparison with Phrase-Based SMT (New Results)

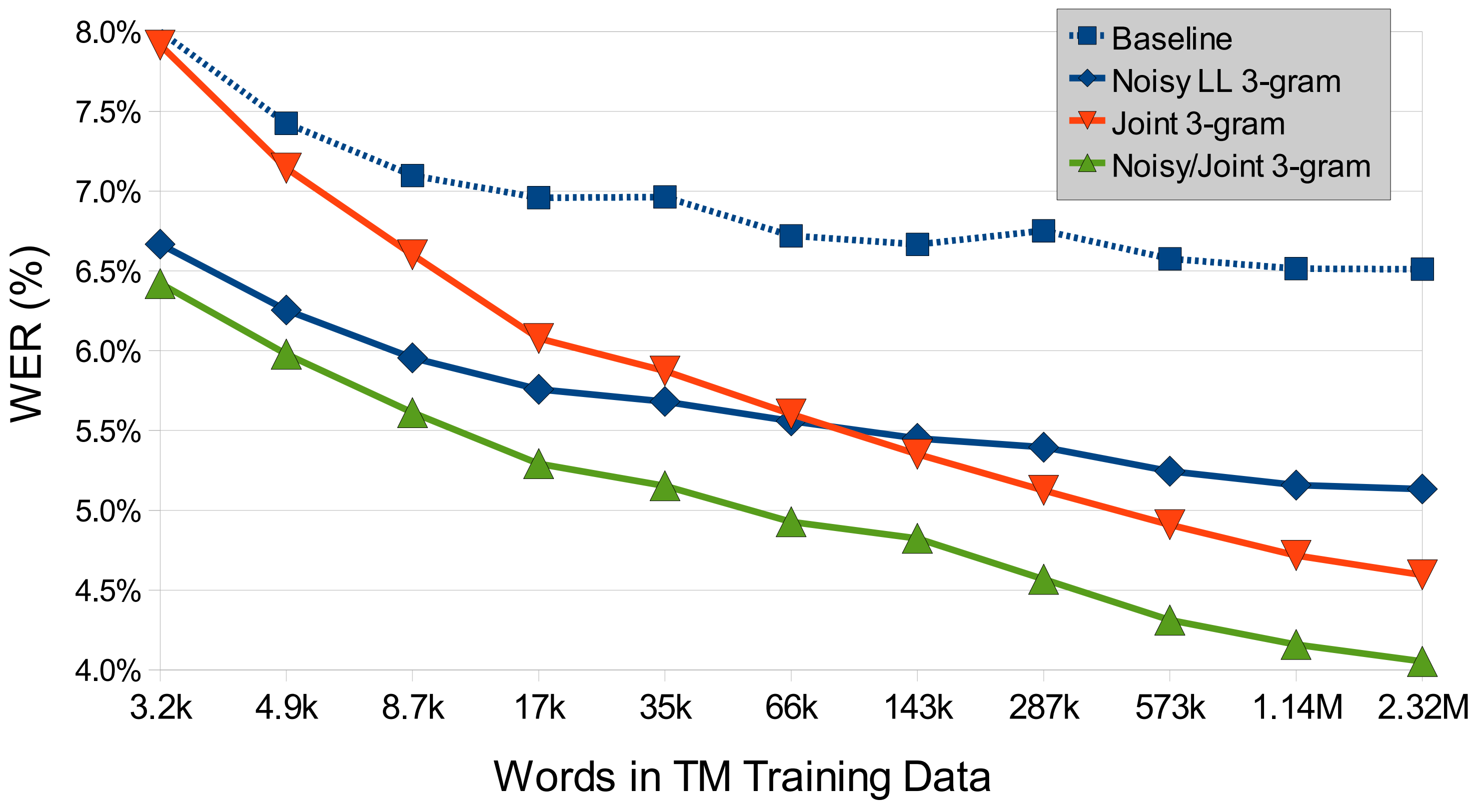
- The proposed techniques were also compared with Moses, a popular system for phrase-based SMT

Model	Verbatim WER	ASR WER
Baseline	6.51%	21.83%
Noisy LL (2-gram or 3-gram)	5.13%	20.97%
Noisy+Joint (2-gram or 3-gram)	**4.05%**	**20.03%**
Moses	5.45%	20.97%

- **Noisy LL** is able to achieve performance as good or better than **Moses**, while **Noisy+Joint** greatly outperforms it

Effect of Corpus Size (Verbatim Transcripts)

- The noisy-channel model is more effective with small data sizes, but the joint model improves rapidly



- Combining both allows for greater accuracy at all sizes

Conclusion

- We proposed two improved statistical models for SMT-based SST
- The proposed methods showed a significant improvement over the baseline for verbatim transcripts and ASR results
- Models transforming ASR output can be trained without using verbatim transcripts
- A promising future direction is tight coupling with a WFST-based ASR decoder

Thank you for listening.

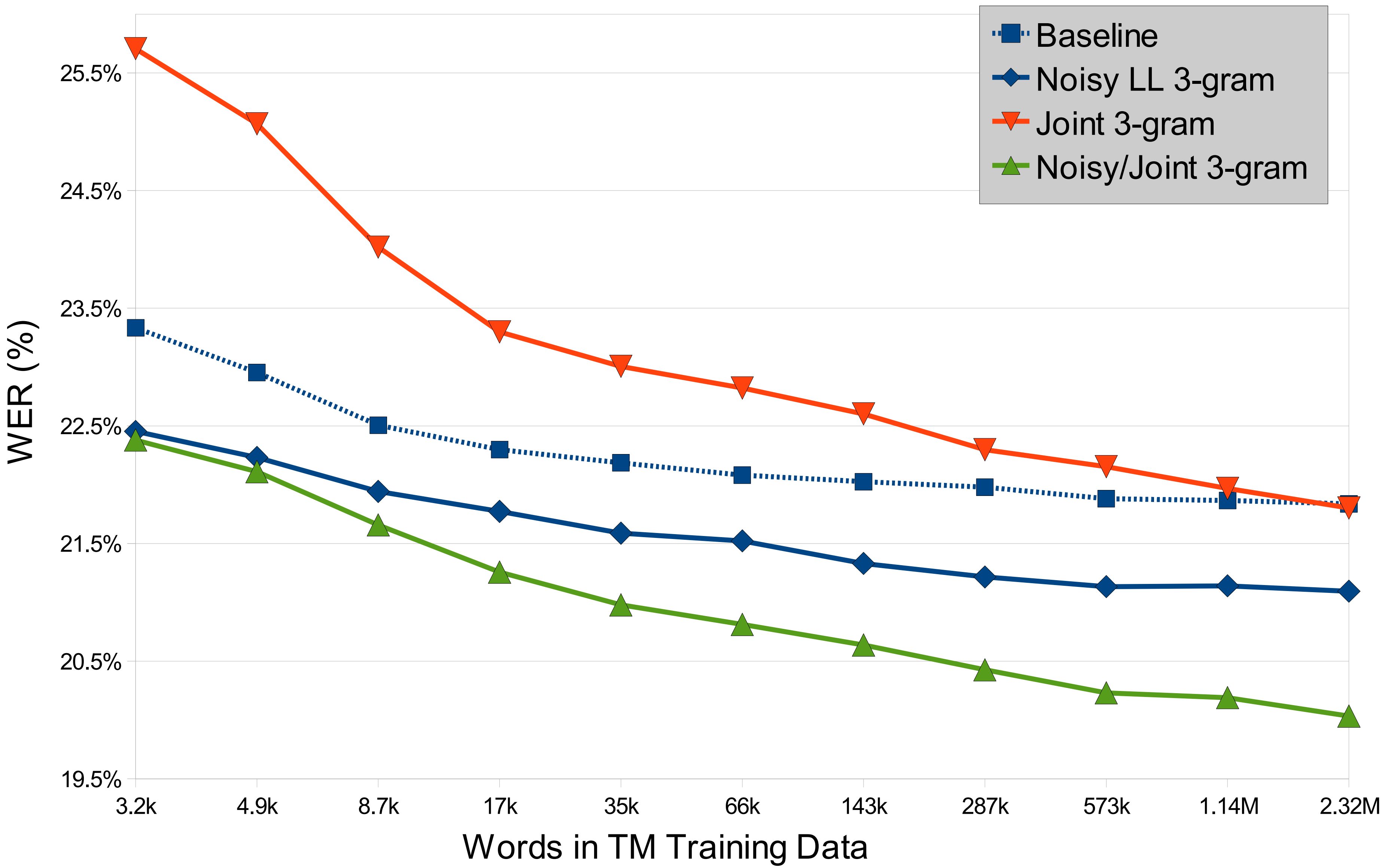
Target Phenomena

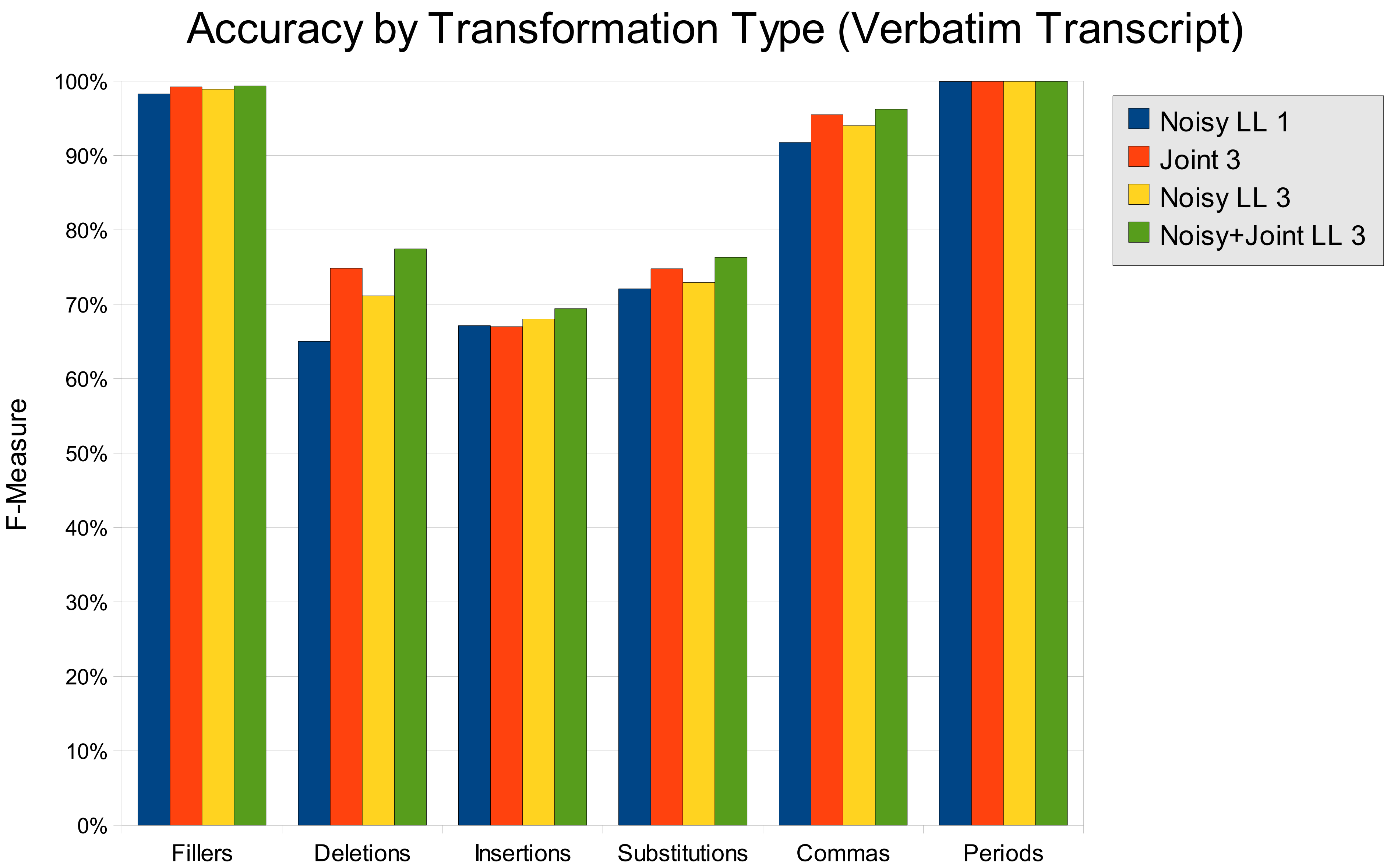
- **Deletion of Extraneous Words:** These include fillers (“um”), context-dependent deletions (“like”), repeats
- **Colloquial Expressions:** Expressions used in speech but less in writing (“ya'know”→“you know”, “*ironna*” → “*iroiro-na*”)
- **Insertion of Words and Punctuation:** Words are omitted in speech, but not in writing (“[did you] talk to the boss?”, “*chumon [o] tsukeru*”)

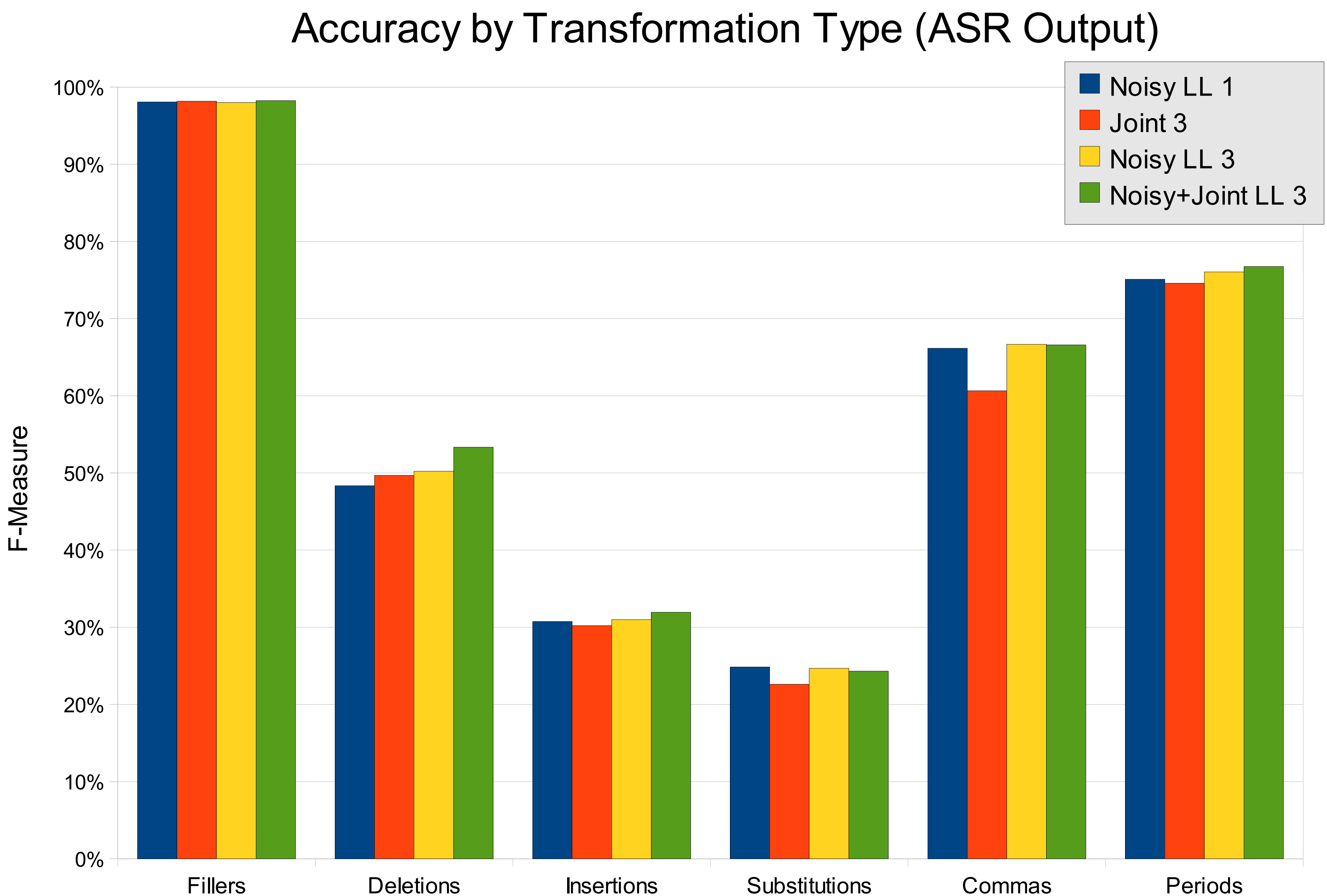
various	ahh	things	by	order	-obj	make	if	it is	
いろんな *ironna*	あー *a-*	こと *koto*	で *de*	注文 *chu-mon*		つける *tsukeru*	と *to*	です ね *desu ne*	…
いろいろ な *iroiro na*		こと *koto*	で *de*	注文 *chu-mon*	を *o*	つける *tsukeru*	と *to*		…
sub	fill				ins			non-fill	

- **Other Phenomena:** order reversal, repairs, fragments

Effect of Corpus Size (ASR Results)




Accuracy by Transformation Type (Verbatim Transcript)
F-Measure
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Fillers
Deletions
Insertions
Substitutions
Commas
Periods
Noisy LL 1
Joint 3
Noisy LL 3
Noisy+Joint LL 3


Accuracy by Transformation Type (ASR Output)
F-Measure
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
Fillers
Deletions
Insertions
Substitutions
Commas
Periods
Noisy LL 1
Joint 3
Noisy LL 3
Noisy+Joint LL 3