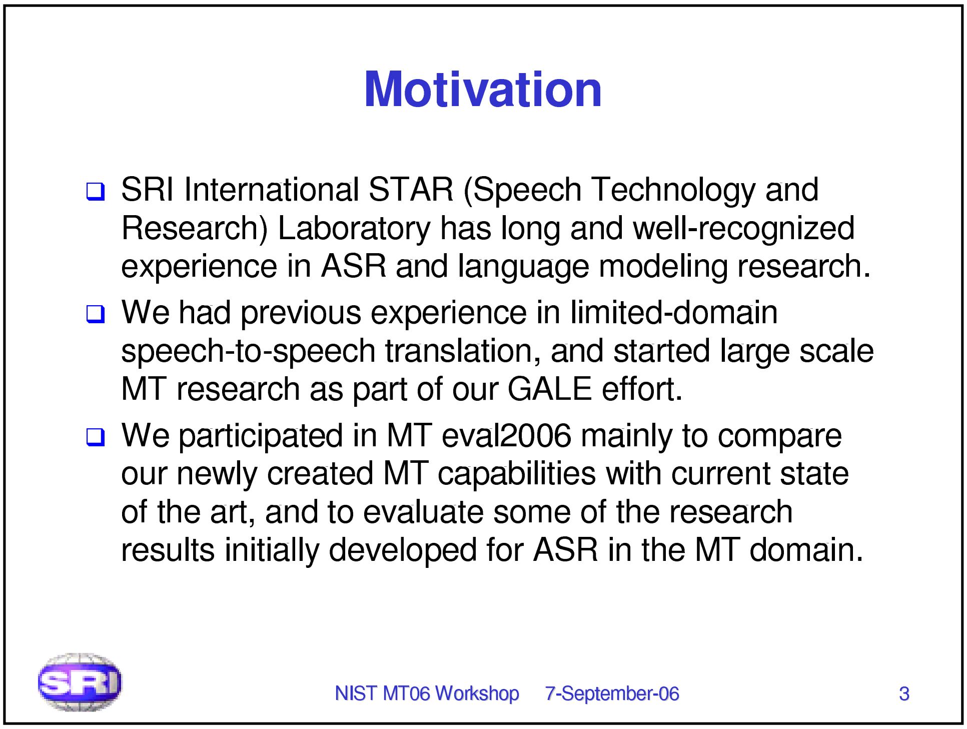
Machine Translation Research @ SRI

Jing Zheng
Wen Wang
Andreas Stolcke

Speech Technology and Research Lab
SRI International

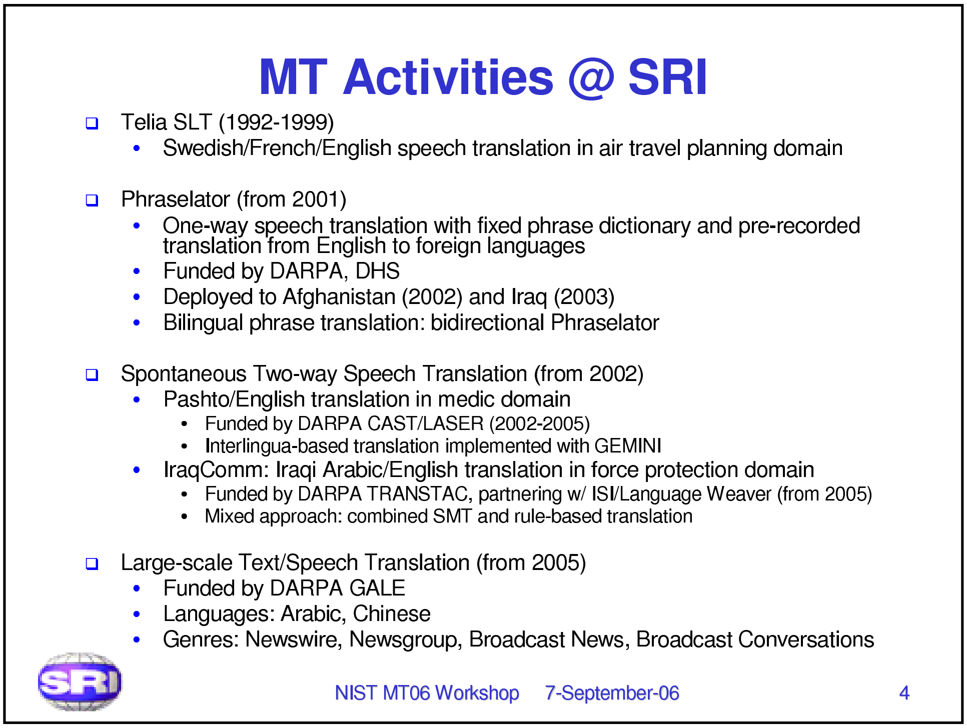
Talk Outline

- ❑ Motivation
- ❑ MT Activities @ SRI
- ❑ SRI's EVAL06 Arabic-English system
- ❑ Language Modeling
- ❑ Summary & Future work



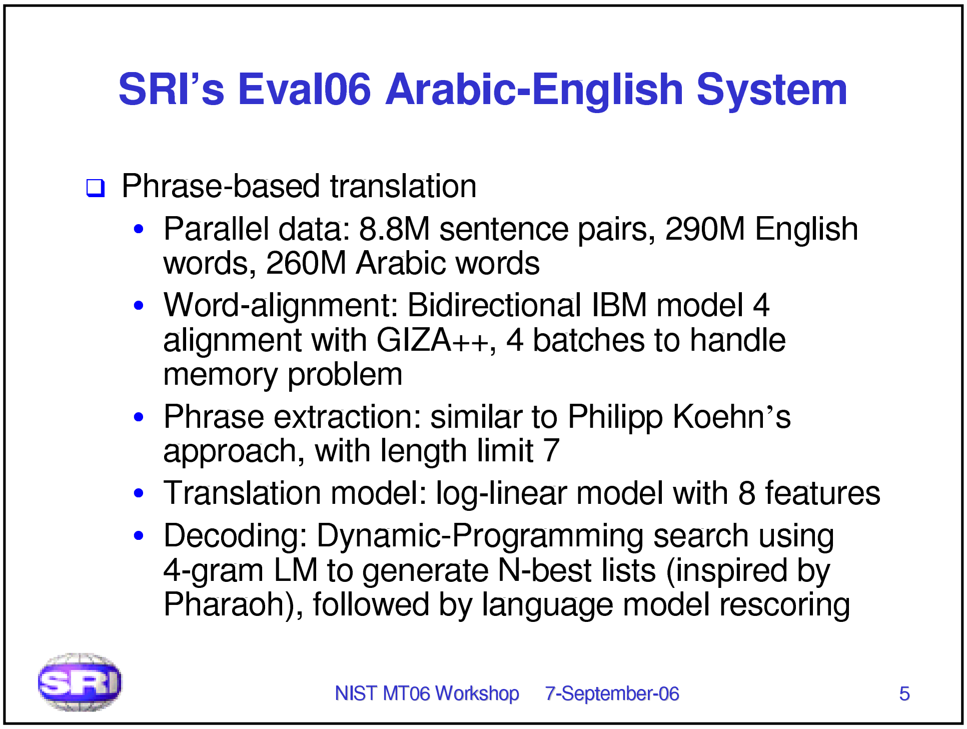
Motivation

- SRI International STAR (Speech Technology and Research) Laboratory has long and well-recognized experience in ASR and language modeling research.
- We had previous experience in limited-domain speech-to-speech translation, and started large scale MT research as part of our GALE effort.
- We participated in MT eval2006 mainly to compare our newly created MT capabilities with current state of the art, and to evaluate some of the research results initially developed for ASR in the MT domain.



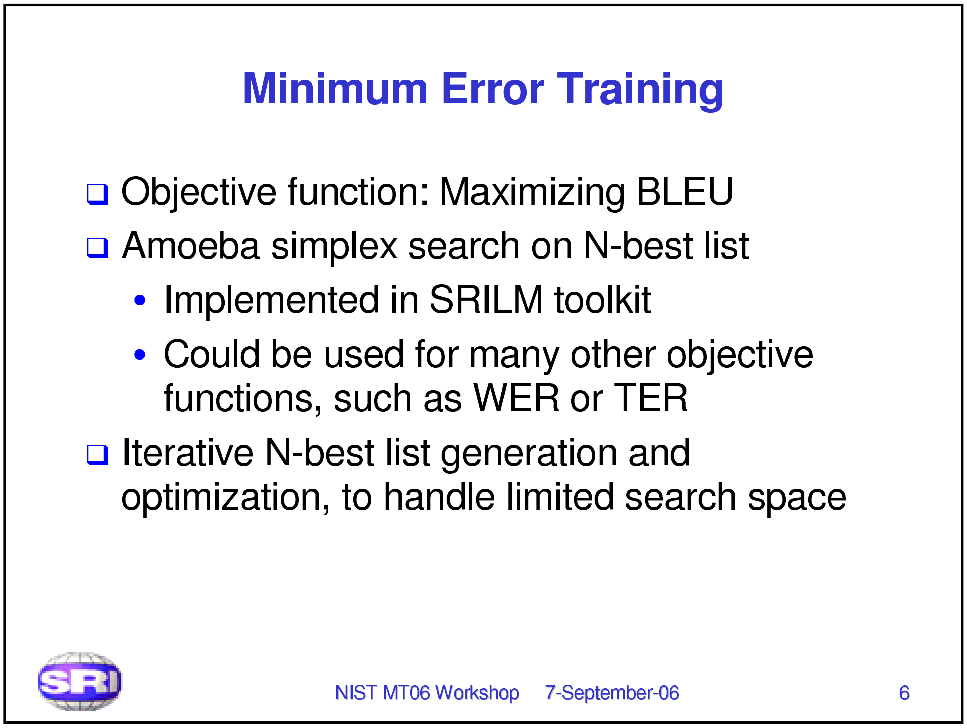
MT Activities @ SRI

- Telia SLT (1992-1999)
  - Swedish/French/English speech translation in air travel planning domain
- Phraselator (from 2001)
  - One-way speech translation with fixed phrase dictionary and pre-recorded translation from English to foreign languages
  - Funded by DARPA, DHS
  - Deployed to Afghanistan (2002) and Iraq (2003)
  - Bilingual phrase translation: bidirectional Phraselator
- Spontaneous Two-way Speech Translation (from 2002)
  - Pashto/English translation in medic domain
    - Funded by DARPA CAST/LASER (2002-2005)
    - Interlingua-based translation implemented with GEMINI
  - IraqComm: Iraqi Arabic/English translation in force protection domain
    - Funded by DARPA TRANSTAC, partnering w/ ISI/Language Weaver (from 2005)
    - Mixed approach: combined SMT and rule-based translation
- Large-scale Text/Speech Translation (from 2005)
  - Funded by DARPA GALE
  - Languages: Arabic, Chinese
  - Genres: Newswire, Newsgroup, Broadcast News, Broadcast Conversations



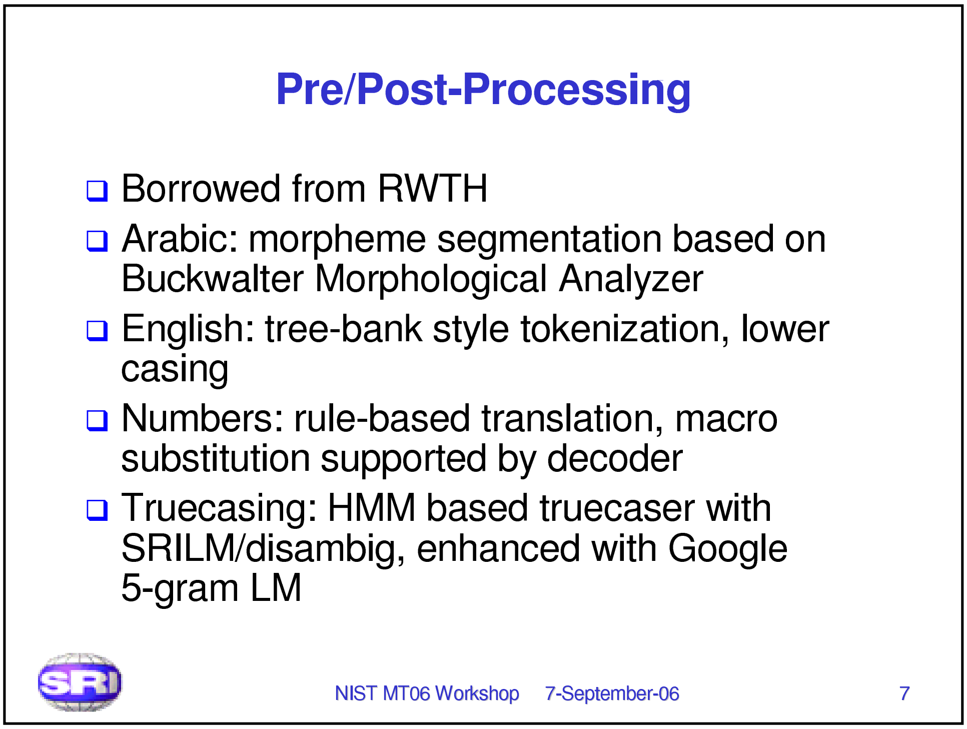
SRI’s Eval06 Arabic-English System

- ❑ Phrase-based translation
  - Parallel data: 8.8M sentence pairs, 290M English words, 260M Arabic words
  - Word-alignment: Bidirectional IBM model 4 alignment with GIZA++, 4 batches to handle memory problem
  - Phrase extraction: similar to Philipp Koehn’s approach, with length limit 7
  - Translation model: log-linear model with 8 features
  - Decoding: Dynamic-Programming search using 4-gram LM to generate N-best lists (inspired by Pharaoh), followed by language model rescoring



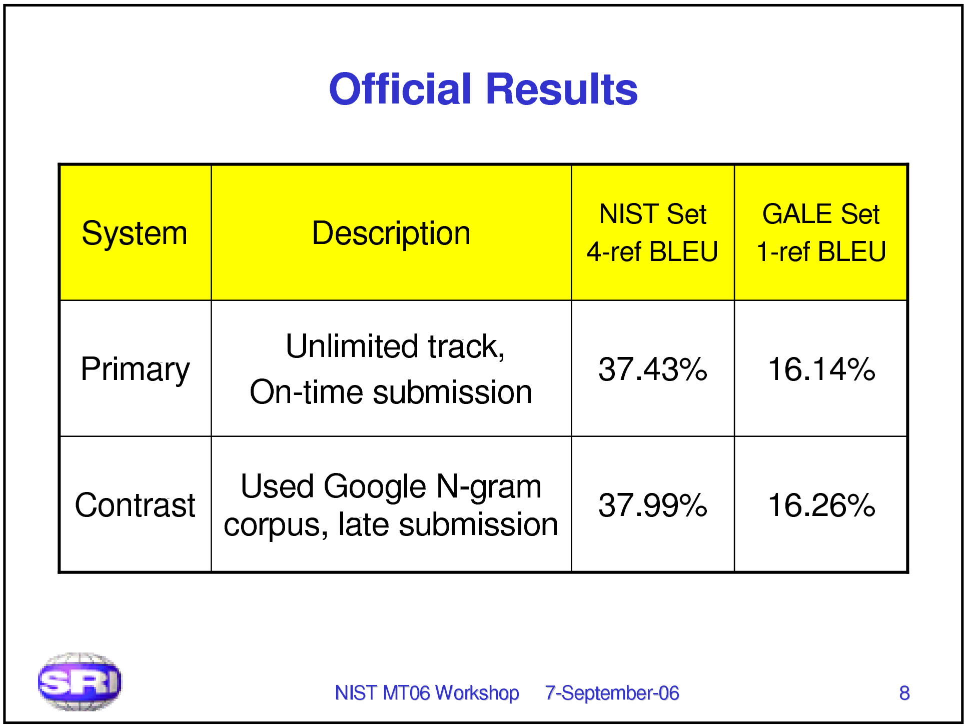
Minimum Error Training

- ❑ Objective function: Maximizing BLEU
- ❑ Amoeba simplex search on N-best list
  - Implemented in SRILM toolkit
  - Could be used for many other objective functions, such as WER or TER
- ❑ Iterative N-best list generation and optimization, to handle limited search space



Pre/Post-Processing

- Borrowed from RWTH
- Arabic: morpheme segmentation based on Buckwalter Morphological Analyzer
- English: tree-bank style tokenization, lower casing
- Numbers: rule-based translation, macro substitution supported by decoder
- Truecasing: HMM based truecaser with SRILM/disambig, enhanced with Google 5-gram LM



Official Results

System	Description	NIST Set 4-ref BLEU	GALE Set 1-ref BLEU
Primary	Unlimited track, On-time submission	37.43%	16.14%
Contrast	Used Google N-gram corpus, late submission	37.99%	16.26%

Language Modeling

- Rescoring with 3 type of language models:
  - Modified Kneser-Ney smoothed word 5-gram LM
  - Deleted interpolation smoothed word 5-gram LM based on structured LM counts (NgramCountLM)
  - Grammar based SARV LM (order 5)
- In primary system: we used 2 knowledge sources:
  - Word LM: 5gram KN smoothed LM interpolated with 5gram count LM, trained on about 900M word corpora
  - SARV LM: trained on 600M word corpora
- In contrast system: we added Google N-gram LM (NgramCountLM) based on the 1 Tera-word 5-gram corpus as an additional knowledge source.

Grammar-based Language Modeling

- **Approach:** integrate multiple-layer, linguistically-motivated knowledge sources with word identity, such as syntactic constraints, morphological features, and lexical features that have synergy with syntax (e.g., enforcing *number* agreement, modeling *wh-movement*).

- **Underlying grammar:** Constraint Dependency Grammar (CDG) (Maruyama 1990, Harper et al 1995)
  - A CDG is a four-tuple, (**Σ, R, L, C)**, where:
    - **Lexical categories (Σ)**: noun, verb, determiner, etc.
    - **Roles (R)**: governor, need1, need2, etc. (Need roles are created to ensure grammatical requirements are met, e.g., a transitive verb requires an object).
    - **Labels (L)**: subject, object, vp, pp, etc.
    - **Constraints (C)**: rules of the grammar (unary and binary constraints) which determine which role values can be assigned to the roles of a word in a sentence. For example, determiners must be governed by nouns.

Super ARV: Lexicalization of CDG Rules

- A **SuperARV (Super Abstract Role Value)**: a **join** operation on all of the dependents related to one word in a CDG parse with additional positional ordering information on all of the word's dependents, plus the morphological features and lexical features that have synergy with syntax, under the current word use environment.

  <C, F, (R, L, UC, MC)+, DC>
  - **C:** lexical category;
  - **F:** a vector of feature types and values;
  - **(R, L, UC, MC)+:** one or more entries of role value assignment constraints:
    - **R:** role id (e.g., governor (G), need1 (N1), need2 (N2))
    - **L:** role label (e.g., subject)
    - **UC:** positional relation between the word and this modifiee, i.e., to the left, to the right, pointing to itself
    - **MC:** modifiee constraints (e.g., lexical category of the modifiee)
  - **DC:** positional relations among the word and all of its modifiees

SARV LM: Almost-parsing CDG-based LM

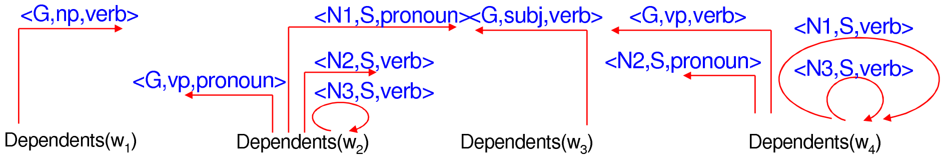
- Class language model using SuperARVs as classes: **jointly** predict a sequence of words and their SuperARVs (Wang et al, 2002, 2004):

$$\Pr(W_N T_N) = \prod_{i=1}^{N} \Pr(w_i t_i \mid W_{i-1} T_{i-1}) = \prod_{i=1}^{N} \Pr(t_i \mid W_{i-1} T_{i-1}) \bullet \Pr(w_i \mid W_{i-1} T_i)$$

- The word n-gram probability is computed as the sum of joint probabilities over all <word, SARV-tag> prefixes.
- After applying this LM on a word sequence, we obtain an **almost-parse** since only limited succeeding work, i.e., specifying modiees for each SuperARV, is required to obtain a complete dependency grammar parse:
  - Strength: tightly integrate linguistic constraints with context word identity information for word prediction, less computationally expensive compared to a full parser based LM
  - Weakness: Not as good as a full parser-based LM on modeling long-distance dependencies

Illustration: Applying SARV LM

<s>	w_1	w_2	w_3	w_4	</s>
	what	**did**	**you**	**learn**	
	t_1	t_2	t_3	t_4	
	pronoun case=common behavior=nominal type=interrogative agr=3s	verb type=past voice=active inverted=yes type=none gapp=yes mood=whquestion agr=all	pronoun case=common behavior=nominal type=personal agr=2s	verb vtype=infinitive voice=active inverted=no gapp=yes mood=whquestion agr=none	



<G,np,verb>

<N1,S,pronoun><G,subj,verb>

<G,vp,verb>

<N1,S,verb>

<N2,S,verb>

<N2,S,pronoun>

<G,vp,pronoun>

<N3,S,verb>

<N3,S,verb>

Dependents(w_1)

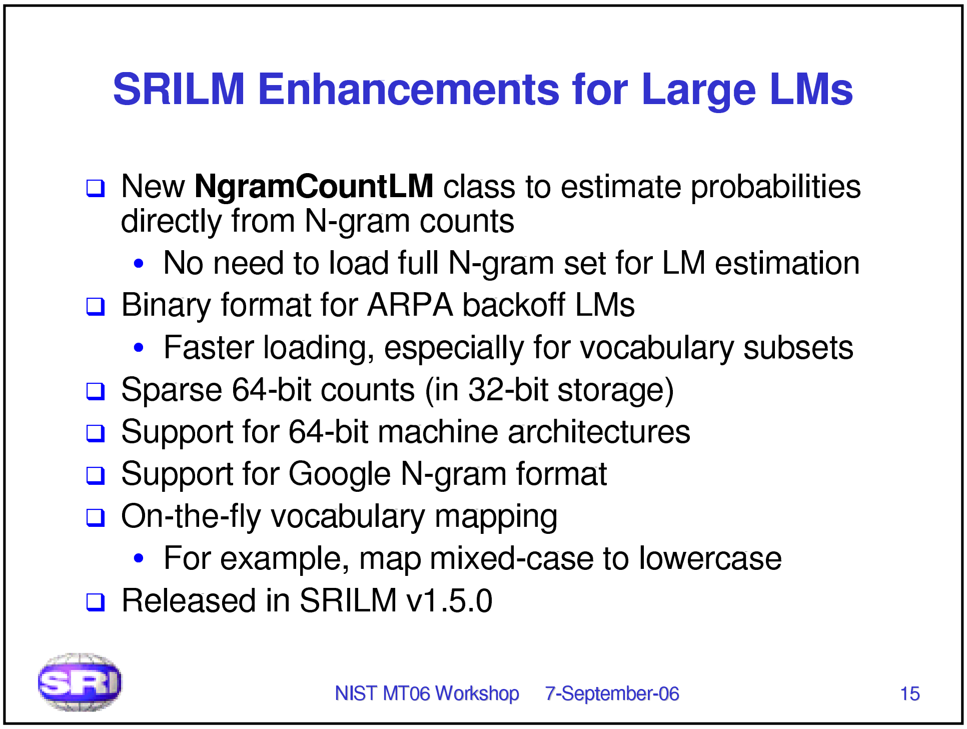
Dependents(w_2)

Dependents(w_3)

Dependents(w_4)

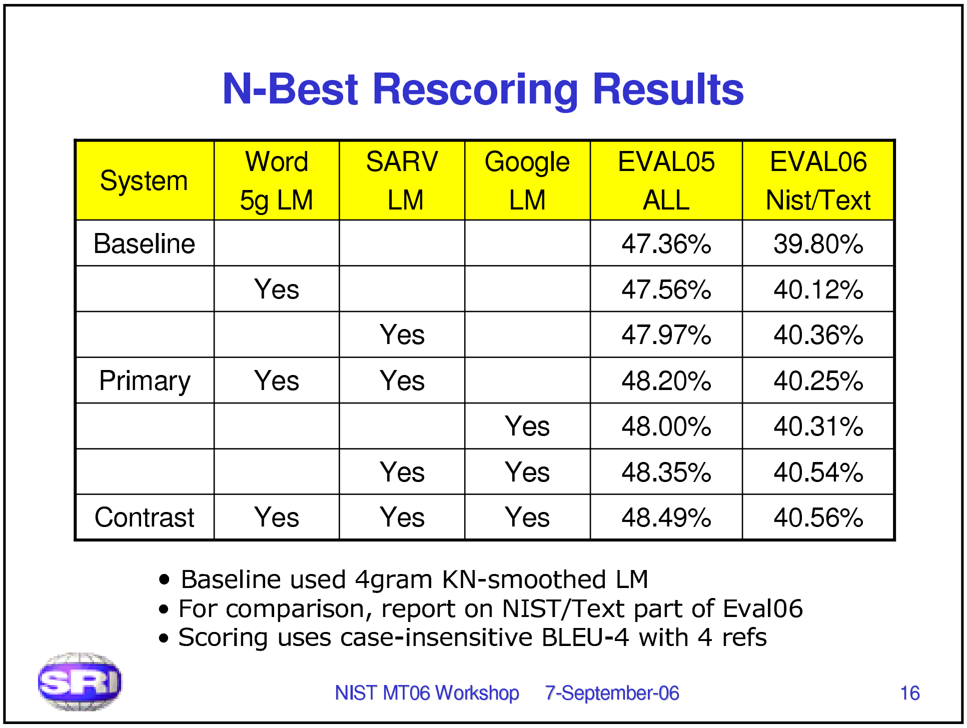
Google N-gram LM

- Obtained Google 1 Tera-word 5-gram corpus via LDC
  - Vocabulary size: 11M
  - 314M bigrams, 976M trigrams, 1.3G 4grams, 1.2G 5grams
  - 25 GB on disk (gzipped)
- Smoothed using deleted interpolation on 1/3 of LM tuning data
- Implemented using SRILM “NgramCountLM” class.
- Rescore batches of 100 N-best lists, loading only N-grams for vocabulary subset
- Two LM versions used:
  - All-lowercase for MT rescoring
  - Mixed case for truecasing
- Thanks to Thorsten Brants @ Google for clarifying text processing and replacing corrupted data files.



SRILM Enhancements for Large LMs

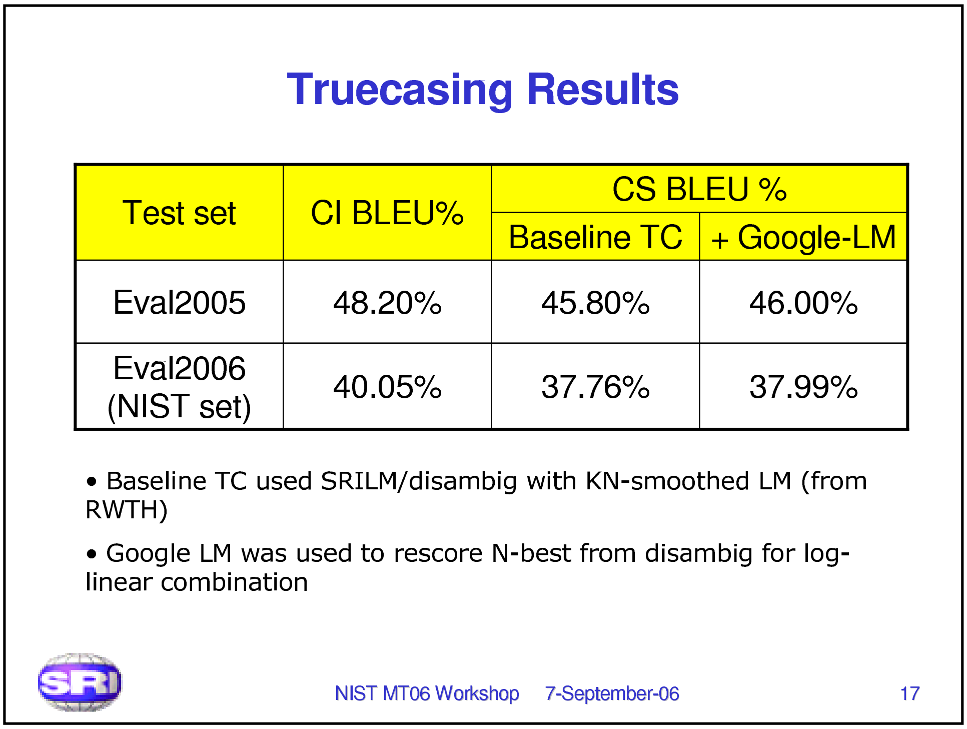
- New **NgramCountLM** class to estimate probabilities directly from N-gram counts
  - No need to load full N-gram set for LM estimation
- Binary format for ARPA backoff LMs
  - Faster loading, especially for vocabulary subsets
- Sparse 64-bit counts (in 32-bit storage)
- Support for 64-bit machine architectures
- Support for Google N-gram format
- On-the-fly vocabulary mapping
  - For example, map mixed-case to lowercase
- Released in SRILM v1.5.0



N-Best Rescoring Results

System	Word 5g LM	SARV LM	Google LM	EVAL05 ALL	EVAL06 Nist/Text
Baseline				47.36%	39.80%
	Yes			47.56%	40.12%
		Yes		47.97%	40.36%
Primary	Yes	Yes		48.20%	40.25%
			Yes	48.00%	40.31%
		Yes	Yes	48.35%	40.54%
Contrast	Yes	Yes	Yes	48.49%	40.56%

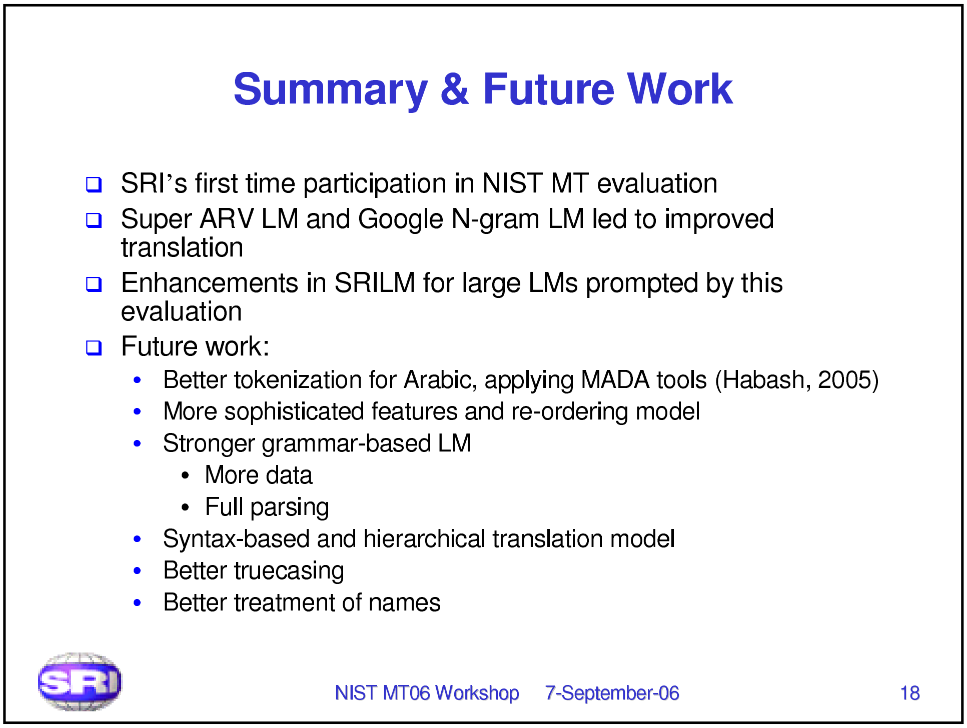
- Baseline used 4gram KN-smoothed LM
- For comparison, report on NIST/Text part of Eval06
- Scoring uses case-insensitive BLEU-4 with 4 refs



Truecasing Results

Test set	CI BLEU%	CS BLEU %	
		Baseline TC	+ Google-LM
Eval2005	48.20%	45.80%	46.00%
Eval2006 (NIST set)	40.05%	37.76%	37.99%

- Baseline TC used SRILM/disambig with KN-smoothed LM (from RWTH)
- Google LM was used to rescore N-best from disambig for log-linear combination



Summary & Future Work

- SRI's first time participation in NIST MT evaluation
- Super ARV LM and Google N-gram LM led to improved translation
- Enhancements in SRILM for large LMs prompted by this evaluation
- Future work:
  - Better tokenization for Arabic, applying MADA tools (Habash, 2005)
  - More sophisticated features and re-ordering model
  - Stronger grammar-based LM
    - More data
    - Full parsing
  - Syntax-based and hierarchical translation model
  - Better truecasing
  - Better treatment of names