

# Statistical Machine Translation of Parliamentary Proceedings Using Morpho-Syntactic Knowledge

Katrin Kirchhoff, Mei Yang, Kevin Duh

Department of Electrical Engineering  
University of Washington, Seattle, USA  
{katrin.yangmei,duh}@ee.washington.edu

## Abstract

This paper presents an overview of the University of Washington statistical machine translation system developed for the 2006 TCSTAR evaluation campaign. We use a statistical phrase-based system with multiple decoding passes and a log-linear probability model. Our main focus was on exploring the possibility of using morpho-syntactic knowledge (lemmas and part-of-speech tags) for word alignment, language modeling, processing out-of-vocabulary words, and reordering. Use of these knowledge sources led to substantial improvements for translation from English into Spanish and minor improvements for the opposite translation direction. In addition, we investigated hidden-event n-gram models for postprocessing of machine translation output.

## 1. Overview

The UW machine translation system for parliamentary proceedings is a statistical phrase-based system. Building on earlier experiences with a smaller Spanish to English translation task (Kirchhoff and Yang, 2005), the system was retrained from scratch for the 2006 TC-STAR EPPS tasks of Spanish-English and English-Spanish translation of verbatim and final text edition (FTE) data. The system uses the public-domain decoder Pharaoh (Koehn, 2004), which selects a translation  $e$  given a foreign sentence  $f$  according to a log-linear combination of  $K$  weighted model scores:

$$e^* = \operatorname{argmax}_e p(e|f) = \operatorname{argmax}_e \left\{ \sum_{k=1}^K \lambda_k \phi_k(e, f) \right\} \quad (1)$$

The overall structure of the system is shown in Figure 1. Two decoding passes are performed. In the first pass, n-best lists are generated using a combination of four translation model scores, a phrase transition penalty, a distortion score, and a trigram language model score. In the second pass, additional scores are provided by a 4-gram language model and additional models that depend on the translation direction. Scores are then again combined using a log-linear model (with a separate set of weights trained for the second pass) to identify the best translation hypothesis. Finally, a postprocessing step is performed to restore numbers and true case, clean up punctuation, etc.

Two different systems were developed. The baseline system (System 1) only uses the training data provided by the TC-STAR consortium and no additional data or annotation tools. The enhanced system (System 2) uses information from freely available morphological annotation tools for English and Spanish. This information is used in the rescoring and postprocessing steps; the precise nature of its integration differs depending on the translation direction and is described in detail below.

## 2. Baseline System

### 2.1. Data and Preprocessing

The data used for training all four systems consists of the parallel training corpus for the FTE condition provided

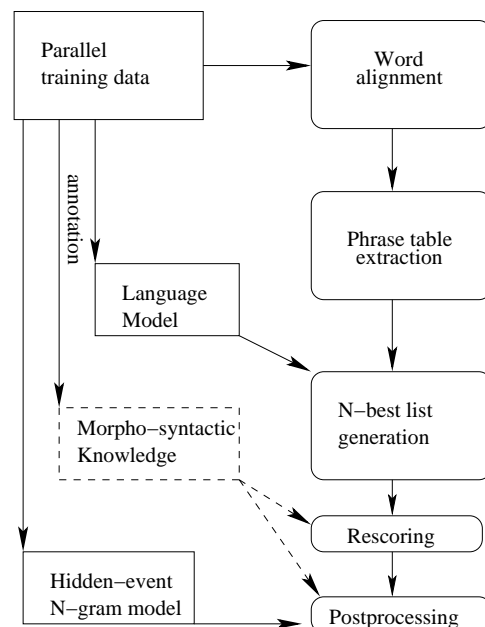


Figure 1: System overview. Dashed lines indicate components used by System 2 only.

by the TC-STAR consortium. No additional data (neither monolingual nor bilingual) was used; neither was the small amount of verbatim data provided as part of the training data. The data was preprocessed by tokenizing (separating punctuation signs from words), replacing numbers with a generic word class, lowercasing, and filtering out sentence pairs with large differences in length (sentences with a word count ratio greater than 3). The resulting training corpus has approximately 11.4M sentence pairs with 34M (Spanish) and 32.8M (English) words. Additional preprocessing steps were applied to the verbatim development and test data: contractions (on the English side) were expanded to full word forms (e.g. *don't* to *do not*), word fragments, disfluencies and fillers (*eh*, *uh*, etc.) were removed, and abbreviations and acronyms were normalized to match their

corresponding forms in the written data.

## 2.2. Word Alignment and Translation Model

Word alignment is performed by an IBM Model 4, trained using the publicly available software tool GIZA++ (Och and Ney, 2000).

The translation model is defined over a segmentation of source and target sentence into phrases:  $f = \bar{f}_1, \bar{f}_2, \dots, \bar{f}_M$  and  $e = \bar{e}_1, \bar{e}_2, \dots, \bar{e}_M$ . Phrase pairs are extracted from the word-aligned bitext using the method described in (Och and Ney, 2003), where word alignment is first performed in both translation directions and additional alignment points are added heuristically. The phrase length was limited to 7. Each phrase pair receives six translation model scores: first, two phrasal scores, i.e. the conditional probabilities of phrase  $\bar{f}$  given phrase  $\bar{e}$ ,  $P(\bar{f}|\bar{e})$ , and the reverse probability,  $P(\bar{e}|\bar{f})$ , which are obtained by maximum-likelihood estimation on the training data. Next, two lexical scores are provided,  $Score_{lex}(\bar{f}|\bar{e})$  and  $Score_{lex}(\bar{e}|\bar{f})$ , defined as follows:

$$Score_{lex}(\bar{e}|\bar{f}) = \prod_{j=1}^J \frac{1}{|\{j|a(i)=j\}|} \sum_{a(i)=j}^I p(f_j|e_i) \quad (2)$$

where  $j$  ranges over words in phrase  $\bar{f}$  and  $i$  ranges over words in phrase  $\bar{e}$ . Finally, a word transition and a phrase transition penalty are used; the latter is a constant added for each phrase used in the translation of a sentence, thus penalizing concatenations of shorter phrases.

## 2.3. Language Model

The language model used in the first pass is a trigram trained on the source language side of the parallel training corpus. The model is trained using SRILM (Stolcke, 2002) using modified Kneser-Ney smoothing and interpolation of bigram and trigram probabilities. For the second pass, a 4-gram model is trained using the same method.

## 2.4. Decoding and Rescoring

The decoder is run in non-monotone mode allowing phrase reordering. The limit on the number of positions by which a phrase may be moved is set to 4; a phrase distortion score is computed which is proportional to the number of word positions by which a phrase is shifted. Weights for the scores are trained on the 2006 development data using a modified implementation of the method in (Och and Ney, 2003). In the first pass, N-best lists of up to 2000 hypotheses are generated; larger n-best lists (5000 and 10,000 hypotheses) were not observed to yield any improvements. For rescoring, the 4-gram score is added and combination weights are trained for a second time (this time using the method proposed by (Nelder and Mead, 1965)) in order to rerank hypotheses to maximize the NIST BLEU score.

## 2.5. Postprocessing

Postprocessing involves re-inserting the actual numbers from the source files for the number variables (using the alignment information obtained during decoding) and restoring the original case. For the latter, the SRILM *disambig* tool is used (Stolcke, 2002), which changes vocabulary

Es: dentro de unos días, celebrará su cumpleaños, es decir, habrán pasado cuatro veces 15 años, pero está todavía detenida.

En-hyp: In a few days, held its 60th Birthday, is to say, will have four times past 15 years, but is still in prison.

En-ref: **She** will celebrate her 60th birthday within a few days, **that** is to say, that four times 15 years have passed, but **she** is still detained.

Figure 2: Translation problems arising from null subject in Spanish. Es = Spanish, En = English, hyp = hypothesis, ref = reference. Deleted pronouns are marked in boldface.

items according to a noisy channel model. Both the channel model and the language model required for this procedure are trained from the source language side of the mixed-case training data (since no alignment with the target language is required, the entire, unfiltered corpus is used). The language model is a 4-gram model. In addition, simple punctuation normalization is performed (e.g. deleting/inserting sentence-initial or sentence-final punctuation marks).

### 2.5.1. Hidden-Event N-gram Models

A notorious problem when translating between Spanish and English is the inaccurate translation of pronouns. Spanish is a so-called null-subject language which allows empty subject pronouns, whereas English requires an overt subject. In Spanish, verbal inflections signal number and person when the subject is missing but often, this information is still ambiguous and the actual referent of the pronoun needs to be inferred from the context. As a result, pronouns often receive the wrong translation or are not translated at all as shown in Figure 2.

This problem is quite widespread: whereas e.g. Spanish, Italian, and Portuguese are null-subject languages, other languages (like Japanese, Chinese, Turkish, or Finnish) also allow empty object pronouns, which are inferred entirely based on the surrounding context. Since there is a higher degree of implicit categories in spoken than in written language, this problem is likely to become more important as machine translation is moving towards increasingly natural spoken input (such as unconstrained dialogues) and a wider range of languages. Translations requiring pragmatic inference indicate a clear limitation of current statistical translation models that cannot be addressed by more training data or improved probability estimation.

In order to determine the potential improvement to be obtained from fixing problems caused by empty categories we conducted an oracle experiment where the translation output from the first pass was manually edited by inserting or deleting single function words that result from obvious cases of missing or “superfluous” categories in the Spanish input (a superfluous category in Spanish from the point of view of English would be e.g. the Spanish personal *a*).

total development set			
	NIST	BLEU (%)	PER
baseline	10.7	54.3	25.5
baseline +edits	10.9	55.1	25.1
subset of edited sentences only			
	NIST	BLEU (%)	PER
baseline	8.1	39.7	29.4
baseline +edits	8.2	41.8	24.5

Table 1: Case-insensitive NIST, BLEU (%) and PER scores for oracle experiments with function word insertions and deletions (“edits”) on the 2006 FTE dev set.

Table 1 shows the effect on the case-insensitive BLEU, NIST and PER scores. Though the overall effect on the total development set is small (0.7% absolute improvement in BLEU score), performance is drastically better when considering only the subset of sentences with inserted/deleted words.

One obvious solution to this problem would be to perform anaphora resolution on the input sentences, e.g. by applying a parser to the input and inserting overt pronouns in a rule-based fashion. The drawback is that fairly complicated disambiguation rules might be needed and that the translation model might have to be retrained on an entire training corpus processed in this fashion. The alternative considered here is to insert words in the translation output according to a statistical hidden-event model. Hidden-event N-gram models were first proposed by (Stolcke and Shriberg, 1996a; Stolcke and Shriberg, 1996b) as a way of detecting sentence boundaries and disfluencies in automatic speech recognition output. The model partitions the vocabulary or event set  $E$  into two (possibly overlapping) subsets: the set  $W$  of regular words and the set  $H$  of words that can be “hidden events” (typically a very small subset). During training, all events are observed; thus, training a model that predicts the joint probability of hidden and observed words is equivalent to training a standard N-gram model to predict event sequences:

$$P(e_1, \dots, e_T) \approx \prod_{t=n}^T P(e_t | e_{t-1}, \dots, e_{t-n+1}) \quad (3)$$

During testing, hidden events are hypothesized at specific locations in the observed word string and their posterior probability is computed by using a forward-backward dynamic programming procedure and the transition probabilities provided by the trained word/event N-gram model. The default is to hypothesize a hidden event before every observed word, but their location can also be constrained explicitly by preprocessing the input data. Hidden events are then applied to the output string (i.e. words are inserted or deleted) if their posterior probability exceeds a given threshold. In order to apply a hidden-event N-gram model to our problem, three tasks need to be solved: hidden event selection, selection of insertion/deletion sites, and tuning of probability thresholds for the two operations.

The deletion/insertion of individual words in the English output typically affects function words, such as pronouns

and prepositions. In order to narrow down the set of possible words we performed a string alignment of the translation output and the references on the development set and collected frequency counts of all deleted words in the hypotheses that occur between two correctly translated words, as shown in Figure 3. Similarly, candidates for deletions were found by considering insertions in the translation output between two correctly translated words. The set of the  $k$  most frequently inserted/deleted words was then selected as the hidden event vocabulary; here we used  $k = 15$ .

Since little reordering takes place between English and Spanish, standard dynamic programming with a Levenshtein distance function was found to be sufficient for this procedure. For languages with greater differences in word order or a weaker translation model, a different alignment strategy might be required that focuses on locally similar alignments.

Two posterior probability thresholds (one for insertions, one for deletions) were optimized on the development data (separately for the FTE and verbatim data) and range between 0.9 and 0.95. All hidden-event words have a priori the same probability of being inserted/deleted; their probability is thus only determined by the language model probabilities. The language model is a 4-gram trained on the English side of the training corpus.

### 3. Use of Morpho-Syntactic Information

For System 2, the training, development and test data were annotated using publicly available morphological analysis tools. For Spanish we used the FreeLing analyzer (Carreras et al., 2004), which provides a base form (lemma) and a morpho-syntactic tag for each word. Lemmatization is done in a rule-based fashion; tags are assigned using the HMM trigram tagger provided in the distributed version (ie the tagger was not retrained for the EPPS task). Its accuracy is reported to be around 95% on the corpora used for its development. Since reference POS annotations are not available for the TC-STAR data, the accuracy on our present corpus is unknown. The tool is able to additionally provide information about named entities, dates, quantities, etc.; however, these functions were not used. For English, we use the Porter stemmer (Porter, 1980) to obtain word stems and the maximum-entropy tagger of (Ratnaparkhi, 1996) for part-of-speech (POS) tagging.

The morpho-syntactic information is used in different ways, depending on the translation direction. For English-to-Spanish, we use a factored language model over words, lemmas and POS tags. For Spanish-to-English, we use a tag-based 6-gram language model; moreover, stem information is used to decompose out-of-vocabulary words prior to translation, and POS tag information is used for local re-ordering operations.

#### 3.1. Word Alignment

In order to reduce the number of parameters in the word alignment model, and to potentially improve the accuracy of the alignment, we tested an alignment model trained on the stemmed version of the training data instead of the full word forms. After converting all full word forms to their

HYP: we must remember \*\*\* because \*\*\* may have been forgotten.  
 REF: we must remember it because it may have been forgotten.

Figure 3: Word alignment for extracting hidden-event vocabulary.

	Es	En
full forms	123,392	68,689
base forms	100,767	45,846

Table 2: Vocabulary size and first-pass for full-form and stemmed versions of the corpus.

stems, an IBM model 4 was trained as usual, and the resulting alignment information was projected back to sentence pairs with full word forms. Phrases were then extracted as described above in Section 2.2. A comparison of the BLEU scores of the baseline system and the stemmed system in the first decoding pass did not show any difference. Further analysis showed that although 89% of all sentence pairs had some differences in alignment under the two different schemes, only one or two words per sentence were typically affected. Moreover, these are also typically adjacent, such that the impact on phrase extraction is fairly limited. However, the number of parameters in the alignment model was reduced significantly: the vocabulary size was decreased by about a third in English and by 20% in Spanish. Contrary to expectations, the vocabulary reduction was greater in English than in Spanish; this was because the English stemmer was more aggressive than the Spanish analyzer: the former truncates word forms whereas the latter outputs a base form which could in principle still be reduced further. The smaller vocabularies resulted in faster training and reduced memory requirements.

### 3.2. Morpho-syntactic Language Models

For the English to Spanish translation direction we trained a factored language model on Spanish that was used during rescoring. Factored Language Models (FLM) (Bilmes and Kirchoff, 2003) are a flexible modeling framework for utilizing diverse sources of information for predicting words. Words are conditioned not only on previous words but also on previous word features (factors), typically POS tags, morphological features such as affixes, stems, etc, or semantic or distributional classes. Previous experiments on using FLMs for machine translation into English (Kirchoff and Yang, 2005) did not show much improvement since English has little morphology. Spanish, which is morphologically more complex than English, is more likely to benefit from additional morphology information. We trained FLMs on the annotated training data to predict words from previous words, lemmas, and POS tags. The set of conditioning factors and the optimal backoff strategy were optimized using a Genetic Algorithms procedure (Duh and Kirchoff, 2004). In particular, the optimization is performed with respect to the set of oracle 1-best hypotheses from the N-best lists of the development set (see (Kirchoff and Yang, 2005) for more details). This ensures that the FLM is tuned to the type of errors made by the

translation model. The resulting FLM uses the following factors in predicting words: the two previous words, the previous lemma, and the previous POS tag. This model achieved a perplexity of 60.1 on the development set, which is only marginally better than the perplexity of the best word-based 4-gram (61.5). However, the FLM gained an improvement in BLEU score, as explained below in Section 4.

### 3.3. OOV Handling

Out-of-vocabulary (OOV) words (words not observed in the training data) are generally not a problem for machine translation when training and test data are well matched and the target language has a slow vocabulary growth rate. As can be seen from Table 3., the percentage of OOV words is low for both conditions and translation directions. It is slightly higher for the Spanish-English test data since different data sources (Spanish instead of European parliament proceedings) were included. We therefore attempted to translate OOVs only in the Spanish data. We use the lemma information provided by the annotation tool to map each OOV to its baseform, provided that a translation for that baseform can be found in the phrase table. Although this is in principle not suitable for general translation since the desired inflectional form of the word in question is ignored, it works well in practice for translation into English, since English makes few morphological distinctions. This method has been shown to be successful when translating from highly inflected languages (Yang and Kirchoff, 2006). On the present task, it reduces the OOV rate by 0.6/0.1% absolute (FTE) and by 4.6/0.3% (Verbatim). Although the overall impact in terms of BLEU score or word-error rate can be expected to be minimal due to the low overall percentage, OOV translation might improve the acceptability of the translations to human readers.

	English		Spanish	
	FTE	VBT	FTE	VBT
dev06	2.4/0.4	1.1/0.3	1.7/1.0	1.6/0.4
test06	2.6/0.4	2.2/0.3	3.2/0.6	6.2/1.1

Table 3: Percentage of OOV words (types/tokens).

### 3.4. Reordering

English and Spanish have fairly similar word orders, with a few notable differences: within noun phrases, the order of noun and adjectival modifiers is usually reversed, and different orderings of subject and verb may occur. Of these, the former is more frequent and can be handled by simple local reordering rules. This is done by POS tagging the English output and using simple pattern matching (regular expressions) to detect Noun-Adjective Phrase sequences, and reversing them unless the sequence is found in the phrase

table. This correctly reorders many noun phrases, however it also creates some wrong orderings in those cases where a word was erroneously tagged as a noun or adjective.

#### 4. Results

Tables 4 and 5 show the performance on the 2006 development data for the baseline and the enhanced systems. Case-sensitive BLEU, NIST and position-independent error rate (PER) are reported. First, performance for the Spanish-English direction with and without hidden-event n-gram postprocessing is compared (Table 4), showing slight improvements for the hidden-event model.

	BLEU (%)	NIST	PER
without NE-ngram - FTE	52.31	10.4	26.6
with HE-ngram - FTE	52.63	10.5	26.3
without NE-ngram - VBT	47.32	10.2	28.9
with HE-ngram - VBT	47.75	10.3	28.4

Table 4: Case-sensitive NIST BLEU score (%), NIST score and PER on the TC-STAR 2006 development set, for System 1 (Spanish-English direction) with and without hidden-event n-gram model for postprocessing.

The comparison of Systems 1 and 2 in Table 5 shows that System 2 gives a slight improvement in both conditions and both translation directions, though it is not statistically significant. For English-Spanish, the improvement is due only to the FLM; for Spanish-English, about two thirds of the increase in BLEU score are due to the tag-based 6-gram model and one-third to noun-adjective reordering. As expected, OOV handling did not make a noticeable difference. Table 6 shows the full results for the 2006 evaluation set. For the Spanish-English translation direction, we again observe very minor improvements from the morpho-syntactic system for both FTE and verbatim data. For English-Spanish, improvements in the FTE condition are on the same order; the improvements obtained by System 2 in the verbatim condition, however, are much larger (on the order of 1% absolute improvement in BLEU score). An analysis of the Spanish translation output showed that, compared to the baseline model, the FLM was able to assign higher probabilities to a larger number of n-grams that have correspondences in the references, due to the more robust back-off scheme.

#### 5. Summary

We have presented the UW statistical machine translation for the 2006 EPPS TC-STAR evaluation task. Among the new techniques we investigated were hidden-event n-gram models for postprocessing and using morpho-syntactic knowledge for word alignment, language modeling, translating out-of-vocabulary words and local reordering operations. We found that word stemming for alignment did not result in any improvement. For the English-Spanish translation direction, using morpho-syntactic knowledge in the form of a factored language model yielded a moderate gain in the FTE condition and a significant gain in the verbatim condition. For Spanish-English translation, we

observed minor improvements from the use of morpho-syntactic knowledge in the FTE condition and from hidden-event N-gram models throughout.

#### Acknowledgments

This work was funded by grant no. IIS-0308297 from the US National Science Foundation.

#### 6. References

- J.A. Bilmes and K. Kirchhoff. 2003. Factored language models and generalized parallel backoff. In *Proceedings of HLT/NAACL*, pages 4–6.
- X. Carreras, I. Chao, L. Padró, and M. Padró. 2004. Freeling: An open-source suite of language analyzers. In *Proceedings of the 4th Int. Conf. on Language Resources and Evaluation (LREC)*, Lisbon, Portugal.
- K. Duh and K. Kirchhoff. 2004. Automatic learning of language model structure. In *Proceedings of COLING*.
- K. Kirchhoff and M. Yang. 2005. Improved language modeling for statistical machine translation. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 125–128.
- P. Koehn. 2004. Pharaoh: a beam search decoder for phrase-based statistical machine translation models. In *Proceedings of AMTA*.
- J.A. Nelder and R. Mead. 1965. A simplex method for function minimization. *Computing Journal*, 7(4):308–313.
- F.J. Och and H. Ney. 2000. Giza++: Training of statistical translation models. <http://www-i6.informatik.rwth-aachen.de/och/software/GIZA++.html>.
- F.J. Och and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–52.
- M.F. Porter. 1980. An algorithm for suffix stripping. *Program*, 14(3):130–137.
- A. Ratnaparkhi. 1996. A maximum entropy part-of-speech tagger. In *Proceedings EMNLP*, pages 133–141.
- A. Stolcke and E. Shriberg. 1996a. Automatic linguistic segmentation of conversational speech. In *Proceedings of ICSLP*, pages 1005–1008.
- A. Stolcke and E. Shriberg. 1996b. Statistical language modeling for speech disfluencies. In *Proceedings of ICASSP*, pages 405–409.
- Andreas Stolcke. 2002. SRILM- an extensible language modeling toolkit. In *Proceedings of ICSLP*, pages 901–904.
- M. Yang and K. Kirchhoff. 2006. Phrase-based backoff models for machine translation of highly inflected languages. In *Proceedings of EACL*.

	English-Spanish			Spanish-English		
	BLEU (%)	NIST	PER	BLEU (%)	NIST	PER
System 1 - FTE	49.10	10.47	30.89	52.63	10.53	26.30
System 2 - FTE	49.44	10.52	30.80	52.97	10.56	26.23
System 1 - VBT	43.55	9.37	34.38	47.75	10.25	28.41
System 2 - VBT	44.14	9.47	34.03	48.05	10.26	28.32

Table 5: Case-sensitive NIST BLEU score (%), NIST score and PER on the TC-STAR 2006 development set.

English-Spanish											
	NIST*	BLEU*	WER*	PER*	NIST	BLEU	IBM	WER	PER	WNM-R	WNM-F
Sys 1 FTE	10.0	0.485	40.4	40.0	10.1	0.495	0.495	39.6	29.9	0.48	0.51
Sys 2 FTE	10.0	0.488	40.3	40.0	10.1	0.497	0.497	39.5	29.9	0.48	0.51
Sys 1 VBT	9.25	0.426	46.2	34.8	9.7	0.452	0.425	43.4	32.0	0.46	0.48
Sys 2 VBT	9.36	0.436	45.5	34.5	9.8	0.463	0.436	42.7	31.7	0.47	0.49
Spanish-English											
	NIST*	BLEU*	WER*	PER*	NIST	BLEU	IBM	WER	PER	WNM-R	WNM-F
Sys 1 FTE	10.2	0.461	43.5	31.3	10.4	0.477	0.462	42.4	29.9	0.74	0.71
Sys 2 FTE	10.2	0.465	43.3	31.4	10.4	0.481	0.466	42.2	29.9	0.74	0.72
Sys 1 VBT	9.8	0.442	46.3	33.1	10.2	0.467	0.442	44.1	30.7	0.66	0.70
Sys 2 VBT	9.8	0.443	46.2	33.1	10.2	0.468	0.443	44.0	30.7	0.66	0.70

Table 6: 2006 Evaluation results. FTE = final text edition, VBT = verbatim, Sys 1 = System 1, Sys2 = System 2. The scores are case-insensitive NIST, BLEU, word error rate (WER), position-independent word error rate (PER), and IBM BLEU score, their case-sensitive counterparts (indicated by \*), as well as weighted n-gram model recall and F-score.