

# ATT-0: Submission to Generation Challenges 2011 Surface Realization Shared Task

**Amanda Stent**

AT&T Labs - Research  
180 Park Avenue, Bldg. 103  
Florham Park, NJ 07932  
stent@research.att.com

## 1 Introduction

This abstract reports on our submission to the shallow track for the Generation Challenges 2011 Surface Realization Shared Task. This system is intended to be a *minimal* system in the sense that it uses (almost) no lexical, syntactic or semantic information other than that found in the training corpus itself. The system architecture was motivated by work done on FERGUS (Bangalore and Rambow, 2000). The system uses three information sources, each acquired from the training corpus: is a localized tree model capturing information from the dependency tree; a trigram language model capturing word order information for words in the same subtree; and a morphological dictionary. In the sections below we briefly present each of these models.

### 1.1 Tree Model

The tree model contains a set of counts for localized tree paths in the dependency trees in the training data. During training, for each lemma we extract several kinds of tree path:

- **three deep, lexicalized** – root, part-of-speech (POS) tag, and phrase type for the lemma; root and phrase type for the two ancestors nearest the lemma in the dependency tree
- **three deep, partly lexicalized** – root, POS tag, and phrase type for the lemma; phrase type for the two ancestors nearest the lemma
- **three deep, not lexicalized** – POS tag and phrase type for the lemma; phrase type for the two ancestors nearest the lemma
- **two deep, not lexicalized** – POS tag and phrase type for the lemma; phrase type for its parent

For each tree path, we record whether the lemma on this path was a *left child* or *right child* of its parent in the dependency tree. We use only localized tree paths to minimize data sparsity.

During realization, we work our way from the most to the least specific tree path for each input lemma, stopping when we find a tree path in the tree model. We assign to the lemma the most frequently occurring relative position of this tree path (to the right or to the left of the head). We do not currently take n-best tree path positions.

**Use-lexicalized flag** We can set a flag in the system to cause realization to use only the non-lexicalized tree paths, or to use the lexicalized tree paths (backing off to the non-lexicalized ones). We experimented with both settings (see Table 1).

### 1.2 Language Model

The language model is a capitalization-invariant trigram language model with Good-Turing discounting acquired from the training corpus using the SRI language modeling toolkit (Stolcke, 2002).

During realization, for each node in the dependency tree having more than one *left child*, we pass the possible orderings of the left children to the language model. We take the top two orderings, if they have similar likelihood; otherwise we take only the top one ordering. If the language model finds no likelihoods for the alternative orderings, they are all retained. The same process is applied to the *right children* of a node in the dependency tree.

**Use-nbest flag** We can set a flag in the system to cause realization to use only the most likely word ordering from the language model, or to consider n-

System settings	Training data	Test data	Items	BLEU	NIST	Meteor	TER
Lexicalized, nbest	Train	Devel	1034	.670 (.344)	12.801	.975 (.435)	.146 (.418)
Non-lexicalized, nbest	Train	Devel	1034	.647 (.329)	12.685	.971 (.425)	.159 (.415)
Non-lexicalized, one-best	Train	Devel	1034	.623	12.587	.967	.174

Table 1: Automatic evaluation results. Single-best results are outside parentheses, 5-best are inside parentheses. *Lexicalized* = tree model has lexical information in tree paths.

best word orderings. Due to the vagaries of the testing software, we do not report results for different settings of this flag here. We used the same language model to rank order complete output sentences for the purposes of input to the testing software.

### 1.3 Morphological Dictionary

The morphological dictionary contains inflected forms found in the training data for each root form in the training data. It indexes root forms by part-of-speech and by verb tense, verb participle, number, and person (1/2/3) features. The person feature is approximated by assigning 1st person to first-person pronouns, 2nd person to second-person pronouns and leaving all other nouns alone.

We augment the morphological dictionary with 4 rules: add word-final *s* to plural nouns; add word-final *ed* to past tense verbs and past tense participles; add word-final *s* to present-tense singular verbs; add word-final *ing* to present tense participles. During post-processing of the entire sentence, we also add word-final *n* to the determiner *a* when it precedes a noun that starts with a vowel, remove multiple adjacent punctuation marks from the set  $\{?!.,;\}$  and ensure that the first letter of the sentence-initial word is upper case. This is the only information not found in the training data that we added to our system.

During realization, each input lemma is assigned inflection by looking up a tuple consisting of its root, POS tag and features in the morphological dictionary, or by using the rules mentioned above.

## 2 Results and Discussion

We evaluated the output of our surface realizer using the reference file and tool provided by Dominic Espinosa, which incorporates BLEU (Papenini and others, 2002), METEOR (Lavie and Denkowski, 2009) and TER (from TERp (Snover and others, 2009)). We used the subsets of the Penn Tree-

bank (Marcus et al., 1993) provided by the Linguistic Data Consortium and converted into dependency trees by Deirdre Hogan. Table 1 shows the output of the automatic metrics for the development data. The absence of lexicalized information in the tree paths causes only a slight drop in accuracy because the language model duplicates some of that information; it also adds efficiency. Tracking only one-best possibilities for all phrases also adds efficiency at a cost of accuracy.

We have not done a formal error analysis, but we did notice during development that punctuation marks, especially those that need to be matched (brackets, quotes), and missing entries in the morphological dictionary, are the source of many errors in our system. It would be easy to use an external morphological dictionary with this system; for these experiments we wanted to be minimalist about the resources we used.

## References

- Srinivas Bangalore and Owen Rambow. 2000. Using TAGs, a tree model, and a language model for generation. In *Proceedings of the TAG+5 Workshop*.
- Alon Lavie and Michael Denkowski. 2009. The METEOR metric for automatic evaluation of machine translation. *Machine Translation*, 23(2–3):105–115.
- Mithcell Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.
- Kishore Papenini et al. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the ACL*.
- Matthew Snover et al. 2009. Fluency, adequacy, or HTER? exploring different human judgments with a tunable MT metric. In *Proceedings of the Workshop on Statistical Machine Translation at the EACL*.
- Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *Proceedings of ICSLP*.