

# Phrase-Based Statistical Machine Translation Using Finite State Machines *with some links to ASR*

Bill Byrne

Cambridge University Engineering Department

The Johns Hopkins University Center for Language and Speech Processing

*wjb31@eng.cam.ac.uk*

27 May 2005

Work done with Shankar Kumar and Yonggang Deng

## Can We Pretend MT is ASR ?

Of course. We just need :

- ▶ Alignment models and estimation algorithms
- ▶ Training data: bitext, monolingual text
- ▶ Search algorithms for translation
- ▶ Some way to measure translation quality



## Five Easy Problems in Statistical Machine Translation

What's currently needed to build a basic phrase-based SMT system:

- ▶ Bilingual for SMT Training: Document and Sentence Alignment
- ▶ Models of Word and Phrase Alignment Within Sentences
- ▶ Language Models
- ▶ Translation – Model-Based Search Algorithms
- ▶ Automatic Measurement of Translation Quality



## Translation

Once the models are defined, translation is ‘trivial’ .

Its just search ...  $\hat{\mathbf{e}} = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{e}|\mathbf{f})$

One approach is to construct **specialized search algorithms**

- ▶ Depending on the underlying models, search can be by Viterbi,  $A^*$ , or other specialized search procedures

But decoder design and implementation is complex

- ▶ Small model changes might require large changes to a decoder
- ▶ Approximations in search imply inexact implementation of the models
  - ▶ Can only implement what can be implemented
  - ▶ May as develop models which lead to exact realizations
- ▶ **Decoder implementation takes effort away from ‘modeling’**

## Translation via Weighted Finite State Transducers

*Translation with Finite State Devices*, Knight & Al Onaizan, AMTA'98

- ▶ Implements the IBM models as WFSTs
  - ▶ word-to-word translation, word fertility, and permutation (reordering)

If the component models can be implemented as WFSTs which can be composed, building a decoder is trivial

- ▶ Can be limiting, but avoids special-purpose decoders
- ▶ The value of this modeling approach has been shown in ASR by the systems developed at AT&T
- ▶ Translation is performed using libraries of standard FSM operations
- ▶ Clear formulation of underlying model components
- ▶ Easy to work on modules in isolation



## Translation Template Model - TTM

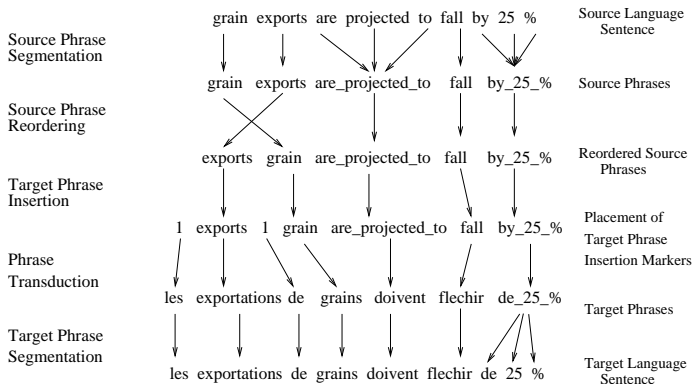
### Generative source-channel model of machine translation

- ▶ Takes the best of
  - ▶ Och&Ney's Phrase-based translation models
  - ▶ Knight&Al Onaizan WFST description of translation via IBM models
- ▶ Bibtex word **alignment** and **translation** under the model can be performed using standard WFST operations
  - ▶ Modular Implementation
  - ▶ No need for a specialized decoder - "the model is the decoder"
  - ▶ Can easily generate translation lattices and N-best lists

### Overall Goals:

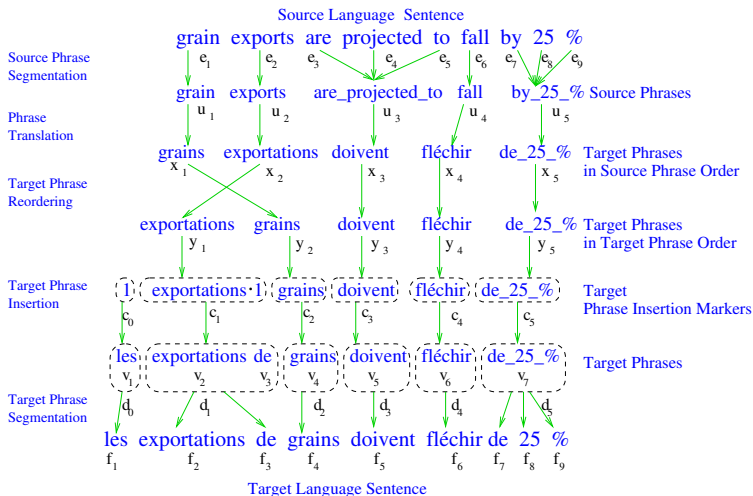
- ▶ Relatively good performance with models that are really quite simple
- ▶ Should be easy to use in translating ASR lattices
- ▶ Easy to teach

## TTM (2004) – Translation with Monotone Phrase Order



- ▶ Transformations via stochastic models implemented as WFSTs
- ▶ Implementation is direct using standard WFST operations

# TTM (2005) – Translation with Moving Target Phrase Order





## Phrase-to-Phrase Translation Models

### Alignment Template Models (Och et al. 1999)

- ▶ derived from ‘good’ word-level alignments, typically from IBM-4

IBM-4 F

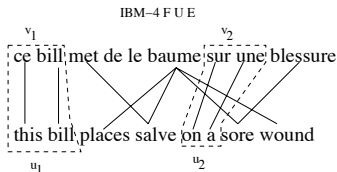
ce bill met de le baume sur une blessure

this bill places salve on a sore wound

IBM-4 E

ce bill met de le baume sur une blessure

this bill places salve on a sore wound



- ▶ **Phrase Pairs** are extracted to cover word alignment patterns
- ▶ Probability distributions are defined over phrase pair sequences



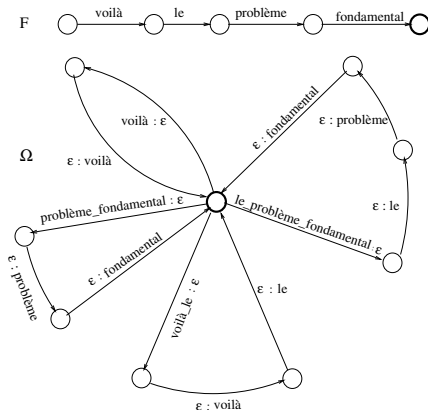
## The Phrase Pair Inventory

English Phrase $u$	French Phrase $v$	Phrase Transduction Probability $P(v u)$
hear_hear	bravo	0.8
	bravo_bravo	0.15
	ordre	0.05
terms_of_reference	mandat	0.8
	de_son_mandat	0.2

- ▶ Phrase Pair Inventory affects the performance of the TTM
  - ▶ Word Alignment Quality of underlying models
  - ▶ Coverage of phrases on the test set



## Target Phrase Segmentation Transducer $\Omega$



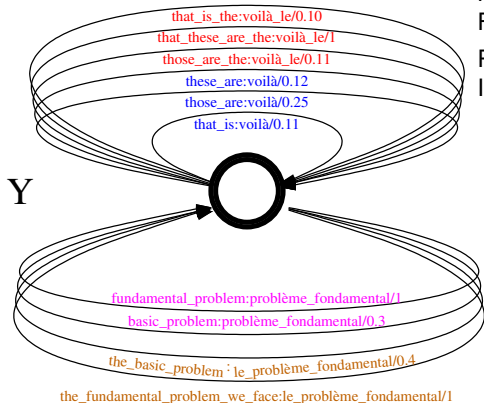
Based on the French phrases in the PPI

Assume  $F$  is the sentence to be translated

Phrase sequences that could have generated  $F: \Omega \circ F$

**voilà.le problème.fondamental**  
 voilà le.problème.fondamental

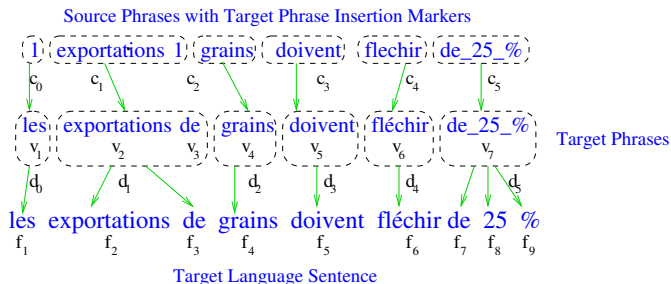
# Phrase Translation Transducer Y



Map English phrases into  
French phrases

Realizes the Phrase-Pair  
Inventory

## Target Language Phrase Insertion



Different phrase segmentations lead to different translations :

- ▶ Sequences  $c_0^K$  that could have generated  $F : Y \circ \Omega \circ F$

H1 `that_these_are_the_fundamental_problem :`  
`voilà_le_problème_fondamental`

...

H16 `that_is_the_basic_problem:`  
`voilà le_problème_fondamental`

## Procedures for Alignment and Translation (Monotone Phrase Order)

Given a French sentence  $f_1^J$  to be translated into English, we build the following transducers (in this order)

- ▶  $F$  to represent the French sentence
- ▶  $\Omega$  maps French phrases in our Phrase-Pair Inventory (PPI) to words in  $F$
- ▶  $Y$  maps English phrases to French phrases in  $\Omega$  with probabilities (PPI)
- ▶  $\Phi$  inserts French phrase insertion markers
- ▶  $W$  maps English words to English phrases seen in  $Y$   
(restricted phrase segmentation transducer)

If  $f_1^J$  is to be **aligned** with an English sentence  $e_1^I$ :

- ▶ Build an FSM  $E$  to represent  $e_1^I$

If  $f_1^J$  is to be **translated**:

- ▶ Build a n-gram backoff LM and compile it as a weighted acceptor  $G$

## Bitext Word Alignment and Translation Via WFSTs

TTM Generative Model:  $P(f_1^J, v_1^R, d_0^K, c_0^K, a_1^K, u_1^K, K, e_1^I)$

- ▶ MAP Alignment of a sentence pair  $f_1^J, e_1^I$

$$\{\hat{K}, \hat{u}_1^K, \hat{a}_1^K, \hat{c}_0^K, \hat{d}_0^K, \hat{v}_1^R\} = \operatorname{argmax}_{K, u_1^K, a_1^K, c_0^K, d_0^K, v_1^R} P(K, u_1^K, a_1^K, c_0^K, d_0^K, v_1^R | e_1^I, f_1^J)$$

- ▶ MAP Translation of a French sentence  $f_1^J$

$$\{\hat{e}_1^I, \hat{K}, \hat{u}_1^K, \hat{a}_1^K, \hat{c}_0^K, \hat{d}_0^K, \hat{v}_1^R\} = \operatorname{argmax}_{e_1^I, K, u_1^K, a_1^K, c_0^K, d_0^K, v_1^R} P(K, u_1^K, a_1^K, c_0^K, d_0^K, v_1^R | f_1^J)$$

### WFST Operations with Monotone Search

- ▶ Alignment

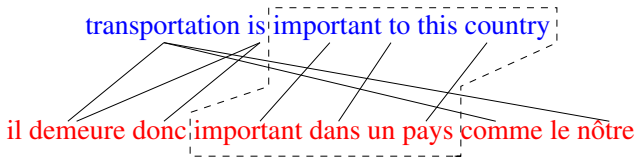
1. Generate the alignment lattice:  $\mathcal{B} = E \circ W \circ \Phi \circ Y \circ \Omega \circ F$
2. MAP Alignment : least cost path in  $\mathcal{B}$

- ▶ Translation

1. Generate the translation lattice:  $\mathcal{T} = G \circ U \circ \Phi \circ Y \circ \Omega \circ F$
2. MAP Translation : least cost path in  $\mathcal{T}$

## Problems with Bitext Word Alignment under the TTM

Consider extraction of phrase pairs from word alignments

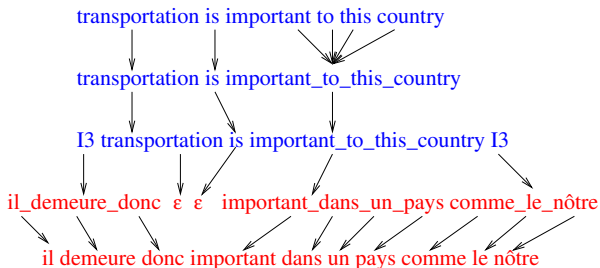


- ▶ Extracted PPI is not rich enough to cover all the sentence-pairs
  - ▶ If we discard the word alignments, this pair has probability zero under the model
  - ▶ In fact, most sentences from the training bitext have probability zero
- ▶ Solution:
  - **TTM already allows insertions of target phrases**
  - **In addition, we allow deletions of source phrases**





## Source Phrase Deletion in Bitext Word Alignment



- ▶ Novel use of phrase-based translation models for alignment
- ▶ TTM word alignments are very accurate: ↑ precision, ↓ recall

Supports parameter estimation procedures

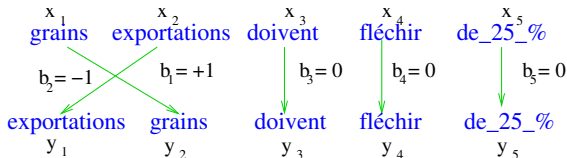
- ▶ Must be able to segment and align the bitext

EM requires

$$P(\underbrace{K, u_1^K, a_1^K, c_0^K, d_0^K, v_1^R}_{\text{Hidden Variables}} \mid \underbrace{e_1^I, f_1^J}_{\text{Training Data}})$$



## Phrase Swapping by WFSTs

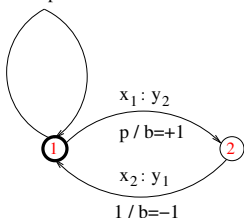


Associate a **jump sequence**  $b_1^K$  with each sequence  $y_1^K$

$$P(y_1^K | x_1^K, u_1^K, K, e_1^l) = P(b_1^K | x_1^K, u_1^K, K, e_1^l) = \prod_{k=1}^K P(b_k | x_k, u_k)$$

$$x_1 : y_1 / 1-p / b=0$$

$$x_2 : y_2 / 1-p / b=0$$



Inspired by work of Tillman

Input:  $x_1, x_2, b \in \{0, +1, -1\}$

Output:  $y_1, y_2$  with prob  $(1 - p)^2$   
 $y_2, y_1$  with prob  $p$

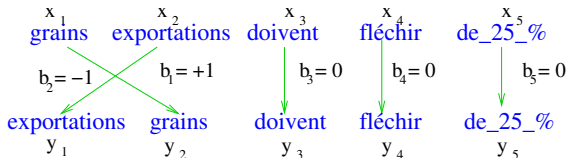
**MJ-1** : maximum jump of 1

Properly parameterized

Not degenerate

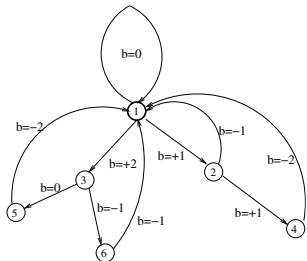


## Phrase Swapping by WFSTs



Associate a **jump sequence**  $b_1^K$  with each sequence  $y_1^K$

$$P(y_1^K | x_1^K, u_1^K, K, e_1^l) = P(b_1^K | x_1^K, u_1^K, K, e_1^l) = \prod_{k=1}^K P(b_k | x_k, u_k)$$



Inspired by work of Tillman

Input:  $x_1, x_2, b \in \{0, +1, -1\}$

Output:  $y_1, y_2$  with prob  $(1 - p)^2$   
 $y_2, y_1$  with prob  $p$

**MJ-2** : maximum jump of 2

Properly parameterized

Not degenerate

## Incorporating Reordering in Translation

- ▶ Reordering prior to translation:
  - ▶ English phrases in French phrase order
  - ▶ Difficult to realize with WFSTs
- ▶ Tried the opposite approach:
  - ▶ First generate a French phrase sequence in English phrase order
  - ▶ Next reorder this sequence into French phrase order under the Local Phrase Reordering Model
- ▶ Possible to realize with WFSTs both in alignment and translation
  - ▶ No English phrase reordering process
- ▶ Reordering is done prior to Insertion of Target Phrases
  - ▶ word alignments within phrases can span fairly long distances

Can perform embedded reestimation of reordering model parameters

- ▶ Phrase-pair dependent reordering probability :  $P(b_k | x_k, u_k)$
- ▶ Estimated via Viterbi approximation to EM
- ▶ Exact estimation: alignments are done under the translation model

## Training and Translation Procedures

1. Bitext Chunking
  - 1.1 Monotone alignment into coarse chunks of documents
  - 1.2 Divisive clustering into subsentence chunks
2. MTTK word alignment – Model 4 replacement
3. Construct component WFSTs for the TTM –
  - 3.1 extract phrases from the alignments using the ‘usual’ heuristics
  - 3.2 use phrase-pair induction under MTTK to augment the PPI
4. Translation lattice generation with pruned trigram →
5. Translation lattice rescoring with unpruned 4gram →
6. Minimum Error Training
  - 6.1 transducer weights optimized for BLEU on heldout data



## TTM and Phrase Reordering

Good gains from reordering in both Arabic→English and Chinese→English

- ▶ Jump-1 might be as good as Jump-2 (in this formulation)
- ▶ little gain in Chinese→English from parameter estimation

Translation under MJ-1 and MJ-2 reordering with a 4-gram LM.

Reordering Model	BLEU (%)					
	Arabic-English			Chinese-English		
	02	03	04	02	03	04
None	37.5	40.3	36.8	24.2	23.7	26.0
MJ-1 flat	40.4	43.9	39.4	25.7	24.5	27.4
MJ-1 VT	41.3	44.8	40.3	25.8	24.5	27.8
MJ-2 flat	41.0	44.4	39.7	26.2	24.8	27.8
MJ-2 VT	41.4	45.0	40.2	26.4	24.8	27.8



## Influence of Language Model Order on Reordering in Translation

BLEU Over Merged Eval Sets (Eval02, Eval03, Eval04)

	BLEU (%)					
	A-E			C-E		
	2g	3g	4g	2g	3g	4g
None	21.0	36.8	37.8	16.1	24.8	25.0
MJ-1	23.4	40.4	41.6	16.2	25.9	26.5
MJ-2	23.3	40.4	41.6	16.0	26.0	26.8

These Simple Reordering Models Benefit from Better Language Models



## Translation Performance with Reordering across Eval 04 Test Genres

Model	BLEU (%)					
	A-E			C-E		
	News	Eds	Spchs	News	Eds	Spchs
None	41.1	30.8	33.3	23.6	25.9	30.8
MJ-1 flat	44.7	31.6	35.3	24.3	27.6	32.9
MJ-1 VT	45.6	32.6	35.7	24.8	27.8	33.3
MJ-2 flat	45.2	31.9	35.2	24.8	27.8	33.4
MJ-2 VT	45.8	32.4	35.2	24.8	27.7	33.6

## Causes:

- ▶ Training / Test Mismatch ?
- ▶ Language model mismatch, in particular ?
- ▶ Less movement in Speeches and Editorials ?
- ▶ Different operating points ?





## Summary: Working Towards Integrated Modeling and Decoding

**Modeling:** Given an English Sentence  $e$  and a French sentence  $f$ , construct a joint distribution over their alignments.

$$P(e, a, f) = \underbrace{P(f|a, e)}_{\text{Translation Model}} \underbrace{P(a|e)}_{\text{Alignment Model}} \underbrace{P(e)}_{\text{Language Model}}$$

**Decoding (ideal):** Given  $f$ , find a translation  $\hat{e}$  and an alignment  $\hat{a}$  as

$$(\hat{e}, \hat{a}) = \operatorname{argmax}_{e, a} P(f|a, e) P(a|e) P(e)$$

**Decoding (really):** lacks integrated modeling and decoding

- ▶ Models are trained & alignments are generated over the training set
- ▶ The models are discarded, and the alignments are kept
- ▶ PPI, etc. are extracted from the alignments and used in translation

**Goal:** same models in alignment and translation – ‘what works’ for ASR

- ▶ needed for : MMI, clustering, context dependence, ...

## New! – TTM Tutorial is Available

German→English translation based on

- ▶ Europarl corpus
- ▶ Giza++ alignments
- ▶ AT&T FSM Toolkit
  - ▶ any FSM toolkit should work ...

Tutorial steps through building and using all the transducers

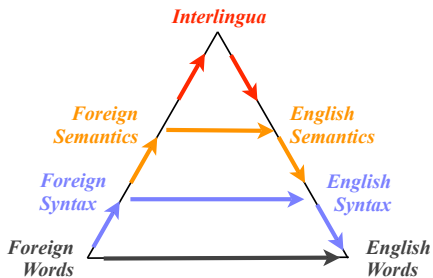
Very much an alpha version, but available to anybody who's interested



## The Machine Translation Pyramid

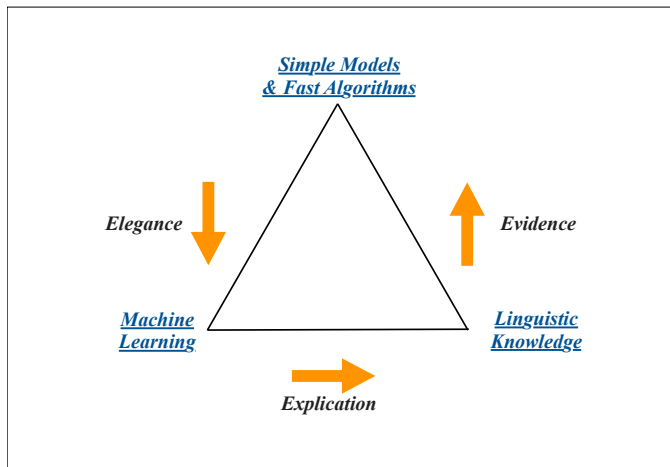
Casts the problem in familiar terms

- strengths and weaknesses of the formulation are obvious



## The *Statistical* Machine Translation Pyramid

Building good systems requires balancing competing concerns



Thanks!

