

Efficient Path Counting Transducers for Minimum Bayes-Risk Decoding of Statistical Machine Translation Lattices

Graeme Blackwood, Adrià de Gispert, and William Byrne

Machine Intelligence Laboratory

Cambridge University Engineering Department



12th of July 2010

Introduction

- ▶ Our goal is efficient minimum Bayes-risk decoding over SMT lattices
- ▶ MBR can be used to improve the output of any SMT system
 - ▶ Based on posterior distribution over translation hypotheses
- ▶ MBR over evidence space \mathcal{E} under loss function $L(E, E')$ has the form¹

$$\hat{E} = \operatorname{argmin}_{E' \in \mathcal{E}} \sum_{E \in \mathcal{E}} L(E, E') P(E|F) \quad (1)$$

- ▶ We describe a novel implementation of MBR over lattices
 - ▶ Using path counting transducers to compute the required statistics
- ▶ This enables efficient decoding over even large SMT lattices

¹Shankar Kumar and William Byrne. *Minimum Bayes-risk decoding for statistical machine translation*. NAACL 2004.

Lattice Minimum Bayes-Risk Decoding

- ▶ Linearized lattice MBR² maximizes conditional expected gain:

$$\hat{E} = \operatorname{argmax}_{E' \in \mathcal{E}} \left\{ \theta_0 |E'| + \sum_{u \in \mathcal{N}} \theta_u \#_u(E') p(u|\mathcal{E}) \right\} \quad (2)$$


- ▶ $p(u|\mathcal{E})$ is “path posterior probability” of n -gram u

$$p(u|\mathcal{E}) = \sum_{E \in \mathcal{E}_u} P(E|F) \quad (3)$$

- ▶ Note that this is not the same as a conditional expected count:

$$c(u|\mathcal{E}) = \sum_{E \in \mathcal{E}} \#_u(E) P(E|F) \quad (4)$$

- ▶ **AIM:** Efficient and exact implementation of Equation (2)


²Roy Tromble, Shankar Kumar, Franz Och, and Wolfgang Macherey. *Lattice Minimum Bayes-Risk decoding for statistical machine translation*. EMNLP 2008. 

Path Posterior Probability Computation

- ▶ Path posterior probabilities can be computed using FSAs³
 - ▶ Intersect acceptor for $\Sigma^* u \Sigma^*$ with \mathcal{E} to obtain \mathcal{E}_u
 - ▶ Then sum path weight $P(E|F)$ for each $E \in \mathcal{E}_u$
 - ▶ Repeated one-by-one in sequence for each n -gram $u \in \mathcal{N}$
 - ▶ This “sequential” method can be slow for large $|\mathcal{N}|$

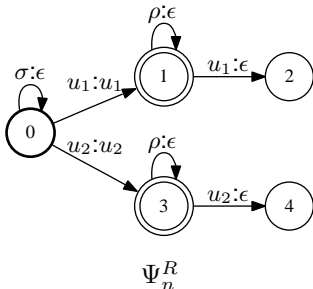
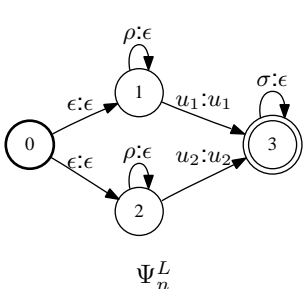
- ▶ We show that exact $p(u|\mathcal{E})$ can be computed simultaneously
 - ▶ Using a single counting transducer for each order $n = 1 \dots 4$

- ▶ We simplify counting by transducing lattice \mathcal{E} to lattice of n -grams \mathcal{E}_n
 - ▶ Easier to count unigrams in \mathcal{E}_n than to count n -grams in \mathcal{E}

³Roy Tromble, Shankar Kumar, Franz Och, and Wolfgang Macherey. *Lattice Minimum Bayes-Risk decoding for statistical machine translation*. EMNLP 2008. 

Efficient Path Counting 1

- ▶ Transducer Ψ_n computes simultaneously $p(u|\mathcal{E})$ for all $u \in \mathcal{N}_n$
- ▶ Example: Ψ_n^L and Ψ_n^R for $u_{1,2} \in \mathcal{N}_n$ for some order n :

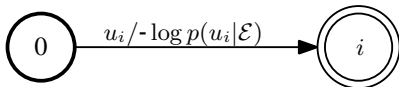


- ▶ Ψ_n^L counts **first** (left-most) occurrence of each n -gram $u \in \mathcal{N}_n$ on path
 - ▶ Has been previously used for counting unigrams in SMT lattices⁴
- ▶ Ψ_n^R counts **last** (right-most) occurrence of each n -gram $u \in \mathcal{N}_n$ on path
 - ▶ We show that Ψ_n^R is efficient and exact for n -gram orders $n = 1, \dots, 4$

⁴Cyril Allauzen, Shankar Kumar, Wolfgang Macherey, Mehryar Mohri, and Michael Riley. *Expected sequence similarity maximization*. NAACL 2010.

Efficient Path Counting 2

- ▶ We form weighted path counts acceptor $\mathcal{X}_n = \mathcal{E}_n \circ \Psi_n$
- ▶ Project output, map to log semiring, ϵ -removal, determinize, minimize
- ▶ \mathcal{X}_n has one arc from the start state for each $u \in \mathcal{N}_n$:



- ▶ $\mathcal{E}_n \circ \Psi_n$ can have many states and arcs for large $|\mathcal{N}_n|$
 - ▶ Slow log semiring ϵ -removal and determinization operations
- ▶ If Ψ_n^R is used instead of Ψ_n^L , then
 1. Each path in $\mathcal{E}_n \circ \Psi_n$ has a single non- ϵ output label u
 2. All paths leading to the same final state share the same output label u
- ▶ This allows a lattice traversal procedure to be used to compute $p(u | \mathcal{E})$
 - ▶ Simply requires propagating symbols as well as probabilities

Efficient LMBR Decoder Implementation

- ▶ We use exact values of $p(u|\mathcal{E})$ at all orders to compute

$$\hat{E} = \operatorname{argmax}_{E' \in \mathcal{E}} \left\{ \theta_0 |E'| + \sum_{n=1}^4 g_n(E, E') \right\}, \quad (5)$$

- ▶ $g_n(E, E')$ is partial gain associated with n -grams of order n
- ▶ Construct acceptor Ω_n to apply $g_n(E, E')$ to paths in \mathcal{E}
- ▶ Form \mathcal{E}_0 as \mathcal{E} with weight θ_0 on all arcs
- ▶ \hat{E} is maximum weight string in LMBR decoder automaton:

$$\mathcal{E}_0 \circ \Omega_1 \circ \Omega_2 \circ \Omega_3 \circ \Omega_4 \quad (6)$$

Lattice MBR Decoding Performance

- ▶ NIST MT Arabic→English translation task (constrained)
- ▶ HiFST: a hierarchical phrase-based lattice decoder⁵
- ▶ IBM BLEU scores for first-pass ML and lattice MBR translations:

	tune	test
ML	54.2	53.8
LMBR	55.0	54.6

⁵Gonzalo Iglesias, Adrià de Gispert, Eduardo R. Barga, and William Byrne. *Hierarchical phrase based translation with weighted finite state transducers*. NAACL 2009.

Lattice MBR Decoding Efficiency

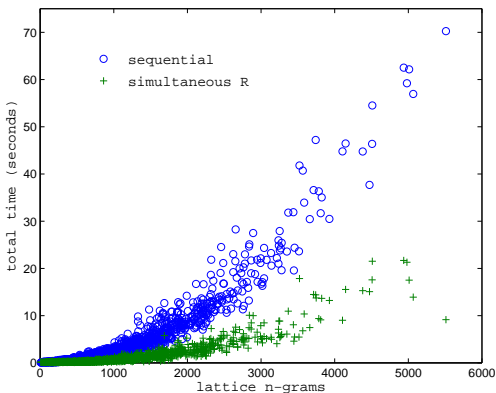
- ▶ Lattice MBR posteriors computation and decoding times (seconds):

		tune	test
Posteriors	sequential	3160	3306
	Ψ_n^L	6880	7387
	Ψ_n^R	1746	1789
Decoding	sequential	4340	4530
	Ψ_n	284	319
Total	sequential	7711	8065
	Ψ_n^L	7458	8075
	Ψ_n^R	2321	2348

- ▶ More efficient to count paths by final than by first occurrence
- ▶ Average MBR time is around 1.2 seconds/sentence

Total Lattice MBR Decoding Time Analysis

- ▶ Total lattice MBR time as a function of number of n -grams:



- ▶ Compares sequential and simultaneous Ψ_n^R on per-sentence basis

Summary

- ▶ Efficient implementation of linearised lattice MBR
- ▶ Based on path counting transducers and regular WFST operations
- ▶ We map to n -gram sequences to simplify higher-order counting
- ▶ Compute required statistics at each order simultaneously

- ▶ Poster Presentation: Venue A, Foyer, Board No. 8