BIA: a Discriminative Phrase Alignment Toolkit

Patrik Lambert[1] *and Rafael Banchs*[2]

1. LIUM (Computing Laboratory)
University of Le Mans
France
–
2. Institute for Infocomm Research (I^2R)
Singapore

Machine Translation Marathon 2011

Introduction

Most Statistical Machine Translation (SMT) systems build translation models from word alignment trained:

- with word-based models

$\Rightarrow$ difficult to align some non-compositional multi-word expressions, compound verbs, etc

Introduction

Most Statistical Machine Translation (SMT) systems build translation models from word alignment trained:

- with word-based models

$\Rightarrow$ difficult to align some non-compositional multi-word expressions, compound verbs, etc

- in a completely separate stage

$\Rightarrow$ no coupling between word alignment and SMT system

Introduction

Most Statistical Machine Translation (SMT) systems build translation models from word alignment trained:

- with word-based models

$\Rightarrow$ difficult to align some non-compositional multi-word expressions, compound verbs, etc

- in a completely separate stage

$\Rightarrow$ no coupling between word alignment and SMT system

intrinsic alignment quality is poorly correlated with MT quality (Vilar et al. (2006)).
Lambert et al. (2007) suggested to tune the alignment directly according to specific MT evaluation metrics

Introduction

The BIA toolkit allows one to overcome these two limitations:

- implementation of discriminative word alignment framework by linear modelling (Moore, 2005; Liu et al., 2005, 2010), extended with phrase-based models and search improvements
- provides tools to tune the alignment model parameters directly according to MT metrics

Alignment Framework

- log-linear combination of feature functions calculated at the sentence pair level.
- searches alignment hypothesis $\hat{\mathbf{a}}$ which maximises this combination:

$$\hat{\mathbf{a}} = \arg\max_{\mathbf{a}} \sum_{m} \lambda_m h_m(\mathbf{s}, \mathbf{t}, \mathbf{a}), \quad (1)$$

- two-pass strategy:
  1. initial alignment of corpus
     (with BIA toolkit, with first set of features, or with another toolkit, e.g. GIZA++)
  2. alignment obtained in the first pass used to calculate a more accurate set of features, used to align the corpus in a second pass

Alignment Framework

Second-pass **alignment features**:

- phrase association score models with relative link probabilities (occurrences of link / occurrences of pair, source and target phrase)
- link bonus model, proportional to the number of links in **a**.
- source and target *word* fertility models giving the probability for a given *word* to have one, two, three or four or more links.
- distortion models counting the number and amplitude (difference between target word positions) of crossing links.
- A 'gap penalty' model, proportional to the number of embedded positions between two target words linked to the same source words, or between two source words linked to the same target words.

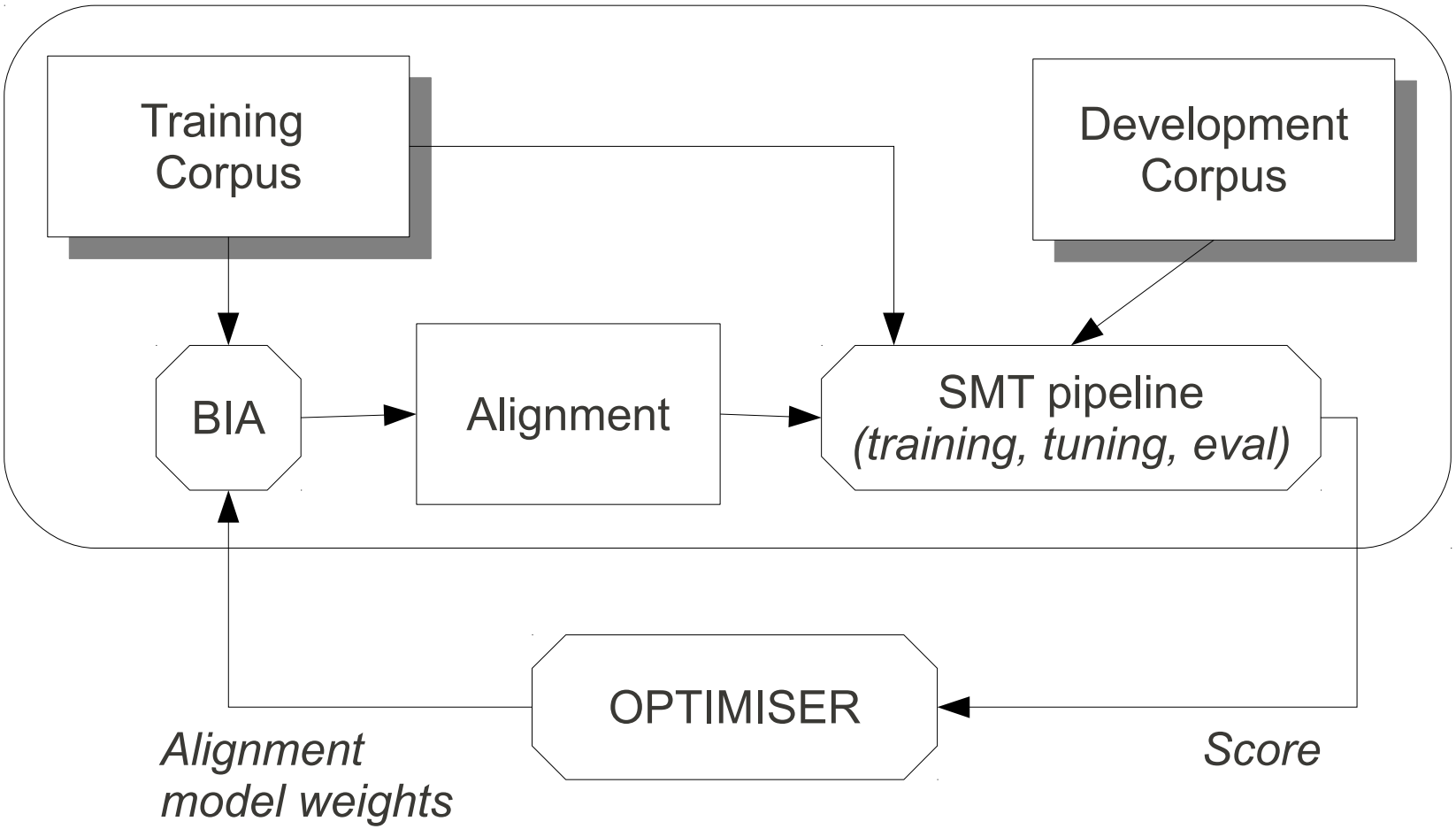
Alignment Framework

Second-pass **alignment features**:

- phrase association score models with relative link probabilities (occurrences of link / occurrences of pair, source and target phrase)
- link bonus model, proportional to the number of links in **a**.
- source and target *word* fertility models giving the probability for a given *word* to have one, two, three or four or more links.
- distortion models counting the number and amplitude (difference between target word positions) of crossing links.
- A 'gap penalty' model, proportional to the number of embedded positions between two target words linked to the same source words, or between two source words linked to the same target words.

Search: beam-search algorithm based on dynamic programming.

Alignment Tuning According to MT Metrics



Optimisers

Objective function evaluation (alignment+SMT pipeline) is time-consuming and gradient unknown:

- re-scoring not feasible
- estimation of gradient in all dimensions costly

$\Rightarrow$ use simpler methods

Simultaneous Perturbation Stochastic Approximation (SPSA):

- gradient estimation with only 2 evaluations of the objective function
- procedure in the general recursive stochastic approximation form: $\hat{\lambda}_{k+1} = \hat{\lambda}_k - \alpha_k \hat{\mathbf{g}}_k(\hat{\lambda}_k)$
- original SPSA algorithm has been adapted to achieve convergence after typically 60 to 100 objective function evaluations

Other tested optimiser: downhill simplex algorithm (Nelder and Mead, 1965)

Implementation overview

- The BIA (BIlingual Aligner) toolkit is implemented in C++ (with the Standard Template Library) and Perl and contains:
  - training tools (mostly in C++)
  - an alignment decoder (in C++)
  - tools to tune the alignment model parameters directly according to MT metrics (in Perl)
  - Perl scripts which pilot the training, tuning and decoding tasks
  - a sample shell script to run the whole pipeline (same as the one used to produce results presented after, but with sample data)
- tested in linux
- No multi-threading implemented. Parameter for number of threads to divide tasks by forking or submitting jobs to cluster (qsub).

Decoding: initialisation

- Load models in memory (into `hash_maps`)
- For each sentence pair, select a set of links to be considered in search:
  - the n best links for each source *and* for each target phrase are considered in search (typically $n = 3$).
  - store relevant information for each link (source and target positions, costs, ...) in specific data structure
  - arrange this set of considered links in stacks corresponding to each source (or target) word

Decoding: search

- State: alignment hypothesis (set of links)
- An hypothesis stack for each number of source+target words covered
- Basic beam-search algorithm:

```
insert initial state (empty alignment) in hypothesis stack
for each stack of links considered in search
   * for each state in each hypothesis stack
        for each link in link stack
           - expand current state by adding this link
           - place new state in corresponding hypothesis stack
   * perform histogram and threshold pruning of hypothesis stacks
```

- Fair comparison for hypotheses:
  - created by links corresponding to the same source (or target) word
  - having the same number of covered words

Implementation issues

- result depends on the order of introduction of the links in alignment hypotheses. Solutions:
  - future cost: should include cost of crossing links; no effective way to estimate this.
  - introduce most confident or less ambiguous links first
  - start from non-empty initial alignment (example: decode along source side, then target, re-decode taking the intersection as initial alignment) $\Rightarrow$ can now expand a state by deleting or substituting a link
  - multiple hypothesis stacks help decoding being more stable
- tuning process not very stable (optimisation algorithm can fall into a poor local maximum).

Experiments

- Spanish–English Europarl task: 0.55 (20k), 2.7 (100k), and 35 million words (full)
- Chinese–English tasks: FBIS (news domain), 3.7M words; BTEC (travel domain), 0.4M words
- Extrinsic evaluation (in BLEU score) of BIA toolkit + 9 other state-of-the-art alignment systems:
  - source-to-target and target-to-source IBM Model 4 alignment (GIZA++) and several combinations: intersection, union, grow-diag-final (GDF) and grow-diag-final-and (GDFA) heuristics
  - Berkeley aligner: (1) simple HMM-based; (2) HMM-based taking target constituent structure into account
  - Posterior Constrained Alignment Toolkit (PostCat)
  - BIA with second-pass models trained on GDFA combination
- BLEU scores: average over 4 MERT runs with different random seeds

Results

	EPPS				
Alignment	Full	100k	20k	FBIS	BTEC
Best other	**56.7**	51.4	46.2	**23.0**	34.8
Moses Default (GDF)	56.3	51.2	46.2	21.7	34.0
Initial (GDFA)	56.2	51.1	46.2	23.0	33.9
BIA	56.2	**51.7**	**46.6**	**23.0**	**35.2**

- in all cases, BLEU score achieved via BIA alignment at least as good as score achieved via alignment used to train BIA models.

Results

	EPPS				
Alignment	Full	100k	20k	FBIS	BTEC
Best other	**56.7**	51.4	46.2	**23.0**	34.8
Moses Default (GDF)	56.3	51.2	46.2	21.7	34.0
Initial (GDFA)	56.2	51.1	46.2	23.0	33.9
BIA	56.2	**51.7**	**46.6**	**23.0**	**35.2**

- in all cases, BLEU score achieved via BIA alignment at least as good as score achieved via alignment used to train BIA models.
- compared to Moses default alignment, BIA yielded a loss of 0.1 BLEU in one task, and gains of 0.4 to 1.3 BLEU in the other tasks

Results

	EPPS				
Alignment	Full	100k	20k	FBIS	BTEC
Best other	**56.7**	51.4	46.2	**23.0**	34.8
Moses Default (GDF)	56.3	51.2	46.2	21.7	34.0
Initial (GDFA)	56.2	51.1	46.2	23.0	33.9
BIA	56.2	**51.7**	**46.6**	**23.0**	**35.2**

- in all cases, BLEU score achieved via BIA alignment at least as good as score achieved via alignment used to train BIA models.
- compared to Moses default alignment, BIA yielded a loss of 0.1 BLEU in one task, and gains of 0.4 to 1.3 BLEU in the other tasks
- BIA always yielded best BLEU score of all alignment systems when its model weights had been tuned on the whole corpus

Results (II)

- Note: in all cases, better to do MERT at each tuning iteration than using tuning weights of SMT system trained on GDFA alignment
- Project wiki: tips to modify the mert-moses-new.pl script to reduce MERT time (max 12 iterations, 10 internal optimisations instead of 20, threshold value)
- Tuning time requirement: 130 min/iteration for Europarl 100k corpus with internal MERT and 8 threads (81 iterations: 7 days)

How-to-use guide

- Detailed instructions and examples in project wiki
- See also sample shell script (same options as the one to obtain the results presented)
- http://code.google.com/p/bia-aligner/

Conclusions and further work

- BIA toolkit:
  - discriminative phrase-based alignment decoder based on linear alignment models
  - training and tuning tools
  - alignment tuning may be performed according to MT metrics
- results on 5 tasks (in terms of BLEU score):
  - BIA alignment always at least as good as alignment used to train it
  - yield the best alignment of those computed when tuned on the whole corpus
  - our method not scalable to large corpora
- Further Work: scalability to any size corpora.

Project page

http://code.google.com/p/bia-aligner/

Training

- select linked phrases in first-pass alignment:
  - linked at least once
  - occurring more than N times in corpus
- count occurrences of links, source and target parts for theses phrases