Machine Translation System Combination with MANY for ML4HMT

Loïc Barrault and **Patrik Lambert**

LIUM (Computing Laboratory)
University of Le Mans
France

ML4HMT 2011

MT System combination

- Studied for more than 15 years
- Improves the results, sometimes greatly
- Makes the most of MANY system differences and complementarity (hopefully)
  - Systems have different architectures (rule-based, example-based, phrase-based, syntax-based, hierarchical, . . .)
  - Diversity of models used (LM, TM)

Existing Work

- Hypothesis selection using information from nbest list [Hildebrand and Vogel, WMT'09]
- Syscomb with SMT system, by considering source text and systems outputs as bitext [Chen et al., WMT'09]
- Confusion Networks (CN)
  - [Rosti et al., ACL'07][Shen et al., IWSLT'08]
  - [Karakos et al., HLT'08][Matusov et al., EACL'06]
- Lattice based combination [Feng et al., EMNLP'09]
- MEMT [Heafield and Lavie, 2010]
- UPV: hypothesis space enhancement + MBR decoding
- etc.

Motivation

Why MANY ?

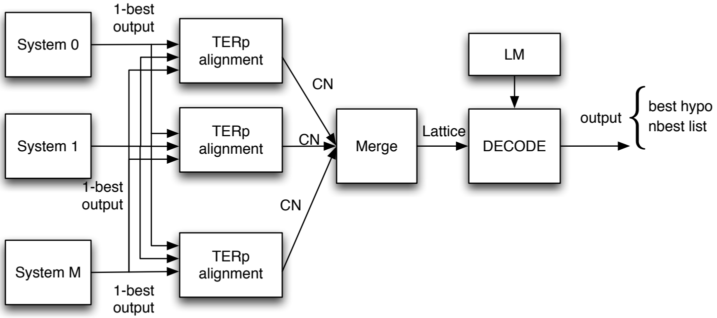
- Open Source
- *push-button* MT syscomb
- easy to use and extend

What is included in MANY ?

- Bash and Perl scripts integrated in Experiment Management System [Koehn, 2010]
- Main libraries
  - Incremental TERp (JAVA)
  - Decoder based on Sphinx4 library (JAVA)

System architecture

- Confusion Network (CN) based MT syscomb



- 3 steps
  - Alignment of 1-best hypotheses and construction of CNs
  - Construction of a lattice by merging CNs
  - Decoding of the lattice

TERp [Snover, WMT'09]

Algorithm

1. Calculate the WER between reference and hypothesis
2. Generate all possible shifts (for match, stem, synonym, paraphrase)
3. For each shift, calculate best score based on DP with match, insertion, deletion, substitution, shift, stem, synonym, paraphrase
4. Apply best shift if it does not degrade the score (or first one if several have same score)

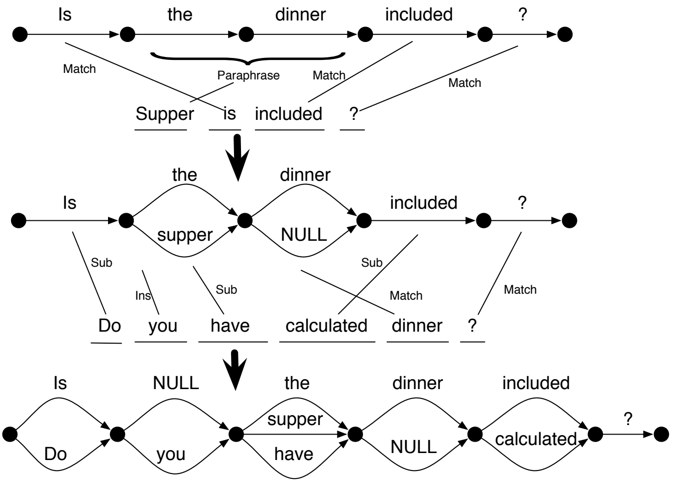
→ repeat steps 1 to 4 until no possible shift which improves score

- Default paraphrase table used
  - pivot-based extraction method [Bannard and Callison-Burch, ACL'05]
  - trained on Ar-En newswire bitext (1 million sentences)
  - Suggestion : use syntactic constraints to improve paraphrases quality [Callison-Burch, EMNLP'08]

Hypothesis Alignment

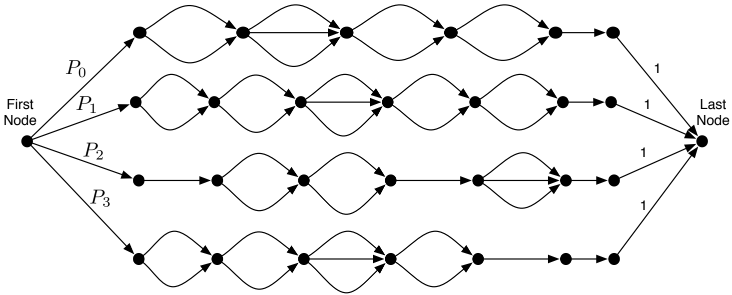
Incremental alignment of all MT system hypotheses against a backbone to create a confusion network (CN)

- Modified version of TERp:
  - alignment between a sentence and a CN
  - match when word in the hypothesis matches word in at least one arc of CN confusion set
- Default TERp weights: 0 for match, 1 for all other weights
- Remaining hypotheses aligned to CN beginning with the nearest in terms of TERp [Rosti et. al, WMT 08] (the order matters)
- Each system acts as backbone
  - no loss of information at this step (each backbone can be re-generated)
  - processing time increases dramatically with number of systems $\Rightarrow$ beware of scalability !


Is
the
dinner
included
?
Match
Paraphrase
Match
Match
Supper
is
included
?
the
dinner
Is
included
?
supper
NULL
Sub
Sub
Ins
Sub
Match
Match
Do
you
have
calculated
dinner
?
Is
NULL
the
dinner
included
supper
?
Do
you
have
NULL
calculated

Lattice

- Merge all CNs into 1 big lattice
- Adding first and last node
- First arcs are given *prior* probabilities (tuned)
- Last arcs are given probability 1



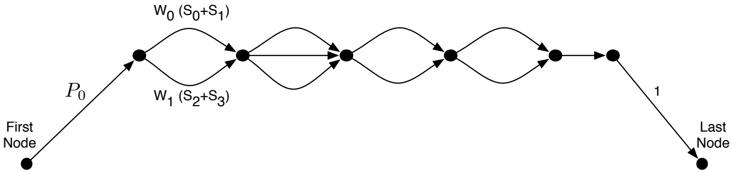
Decoding

- Token Pass decoder
- Probabilities computed in the decoder :

$$log(P_W) = \sum_i \alpha_i \ \log h_i(t)$$

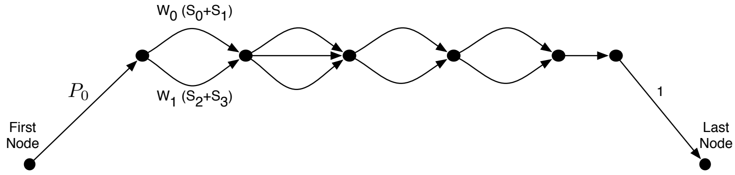
- Features considered for decoding:
  - LM probability, given by an n-gram language model.
  - Word penalty, depending on the hypothesis length (in words).
  - Null-arc penalty, depending on the number of null-arcs gone through
  - System weights: each word receives a weight corresponding to the sum of the weights of all systems which proposed it.
- Language model :
  - n-gram LM with server provided in SRILM
  - n-gram LM (ARPA or Sphinx binary format) $\Rightarrow$ released soon !
- Feature weights are optimised with MERT

Decoding Algorithm Illustration



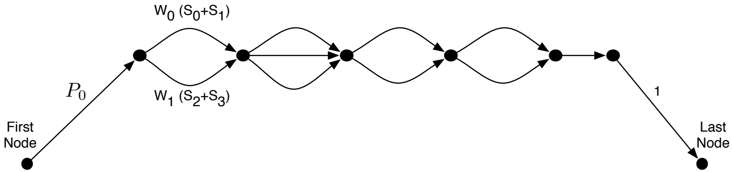
- At first node: 1 token $\{words; score\}$: $\{\varnothing; 0\}$

Decoding Algorithm Illustration



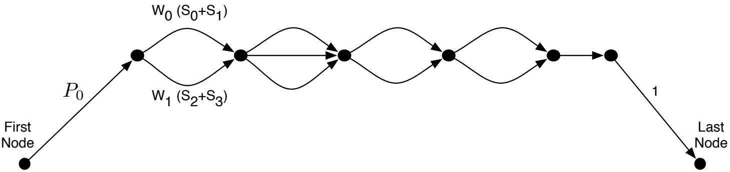
- At first node: 1 token $\{words; score\}$:
  $\{\varnothing; 0\}$
- At second node: 1 token:
  $\{\varnothing; P_0\}$

Decoding Algorithm Illustration



- At first node: 1 token $\{words; score\}$:
  $\{\varnothing; 0\}$
- At second node: 1 token:
  $\{\varnothing; P_0\}$
- At third node: 2 arcs to extend current token $\Rightarrow$ 2 tokens:
  $\{w_0; P_0 + P_0 + P_1 + LM(w_0| < s >) + word\ \ penalty(1)\}$
  $\{w_1; P_0 + P_2 + P_3 + LM(w_1| < s >) + word\ \ penalty(1)\}$

Decoding Algorithm Illustration



- At first node: 1 token $\{words; score\}$:
  $\{\varnothing; 0\}$
- At second node: 1 token:
  $\{\varnothing; P_0\}$
- At third node: 2 arcs to extend current token $\Rightarrow$ 2 tokens:
  $\{w_0; P_0 + P_0 + P_1 + LM(w_0| < s >) + word\ \ penalty(1)\}$
  $\{w_1; P_0 + P_2 + P_3 + LM(w_1| < s >) + word\ \ penalty(1)\}$
- Etc. Words and scores of each arc gone through are accumulated.

Task Description

- Task: combining the outputs of five MT systems: Joshua, Lucy, Metis, Apertium and Matrex.
- MT system outputs provided on development and test sets (WMT 2008 news test set divided in two).
- input of our combination system: one-best plain text output of each MT system, tokenised and with original case:
  - lower case for the Joshua output
  - true case for the rest of systems

Training and Tuning

- Language Model:
  - Trained on News Commentary corpus (4.3M words)
  - SRILM: 4-gram back-off language model with Kneser-Ney smoothing
- Tuning decoder weights:
  - Dev set hypotheses incrementally aligned with TERp default costs

  ⇒ lattice with the resulting confusion networks
  - Decoding of lattice of CNs tuned using MERT (towards BLEU)

  ⇒ decoder weights yielding best scoring combination output on dev set:

LM weight	Word penalty	Null penalty
0.032	0.23	0.010

Joshua	Lucy	Metis	Apertium	Matrex
0.013	0.27	-0.014	0.21	0.22

⇒ higher weight for words proposed by Lucy, then Matrex, Apertium, Joshua, and negative weight for Metis.

Evaluation

- test set hypotheses incrementally aligned with TERp default costs

$\Rightarrow$ lattice of CNs

- decoding the lattice with optimised weights

$\Rightarrow$ final combination output, evaluated on the test set

Evaluation: results

System	BLEU	TER	METEOR
Joshua	13.8	67.3	52.7
Lucy	**22.7**	62.0	**57.6**
Metis	9.1	80.0	41.4
Apertium	21.6	62.9	55.2
Matrex	20.2	**60.2**	56.5
MANY	24.4	58.5	56.2

- MANY vs best single system: +1.7 BLEU, -1.7 TER, -1.4 METEOR

Evaluation: results

System	BLEU	TER	METEOR
Joshua	13.8	67.3	52.7
Lucy	**22.7**	62.0	**57.6**
Metis	9.1	80.0	41.4
Apertium	21.6	62.9	55.2
Matrex	20.2	**60.2**	56.5
MANY	24.4	58.5	56.2

- MANY vs best single system: +1.7 BLEU, -1.7 TER, -1.4 METEOR
- Decision taken in decoder mainly depends on language model

$\Rightarrow$ restriction of LM training data size was a severe limitation

Evaluation: results

System	BLEU	TER	METEOR
Joshua	13.8	67.3	52.7
Lucy	**22.7**	62.0	**57.6**
Metis	9.1	80.0	41.4
Apertium	21.6	62.9	55.2
Matrex	20.2	**60.2**	56.5
MANY	24.4	58.5	56.2

- MANY vs best single system: +1.7 BLEU, -1.7 TER, -1.4 METEOR
- Decision taken in decoder mainly depends on language model

$\Rightarrow$ restriction of LM training data size was a severe limitation

- system ranking resulting from tuning consistent with METEOR score ranking, and close to BLEU or TER rankings.

1. Introduction
2. Architecture
3. ML4HMT Shared Task
4. Perspectives
   - Using MT System Extra Information in MANY
   - Extensions of MANY

Useful Information from MT systems

- Confidence score on each word/phrase given by the system
  - can be directly integrated into MANY (use confidence score instead of system priors)

Useful Information from MT systems

- Confidence score on each word/phrase given by the system
  - can be directly integrated into MANY (use confidence score instead of system priors)
- Decomposition in translation units (and their probabilities)
  - can be integrated as feature: score for phrases used depending on their probabilities

  $\rightarrow$ 1 additional feature (as LM), or 1 feature for each system (like priors)
  - can be used to avoid breaking phrases used by the MT systems

Useful Information from MT systems

- Confidence score on each word/phrase given by the system
  - can be directly integrated into MANY (use confidence score instead of system priors)
- Decomposition in translation units (and their probabilities)
  - can be integrated as feature: score for phrases used depending on their probabilities
  - $\rightarrow$ 1 additional feature (as LM), or 1 feature for each system (like priors)
  - can be used to avoid breaking phrases used by the MT systems
- Probability of each n-gram of system-specific LM
  - can be integrated as additional feature
  - we could also imagine a combined feature including TM+LM info representing the "opinion" of system i, in addition to the system priors

Useful Information from MT systems

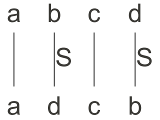
- Confidence score on each word/phrase given by the system
  - can be directly integrated into MANY (use confidence score instead of system priors)
- Decomposition in translation units (and their probabilities)
  - can be integrated as feature: score for phrases used depending on their probabilities
  - → 1 additional feature (as LM), or 1 feature for each system (like priors)
  - can be used to avoid breaking phrases used by the MT systems
- Probability of each n-gram of system-specific LM
  - can be integrated as additional feature
  - we could also imagine a combined feature including TM+LM info representing the “opinion” of system *i*, in addition to the system priors
- Enrich confusion sets with synonyms, paraphrases, etc. to extend search space.

Useful Information from MT systems

- General problem: combining heterogeneous features (phrase-pairs, trees, syntactic information, etc.)
- The feature calculated for a type of system information cannot be calculated for the other system outputs
  $\Rightarrow$ difficult to compare
- calculate a "system opinion" feature based on each system type of information. Weight optimisation can weight these different opinions

TERp limitations

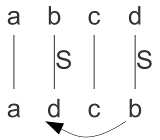
- Multiple shifts are not possible in the same iteration
- Only best shift explored (provided it does not worsen score)
- As a result, crossings often treated as substitutions:



<u>Cost</u>:
2 substitutions

TERp limitations

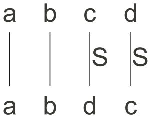
- Multiple shifts are not possible in the same iteration
- Only best shift explored (provided it does not worsen score)
- As a result, crossings often treated as substitutions:



Cost:
2 substitutions

TERp limitations

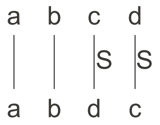
- Multiple shifts are not possible in the same iteration
- Only best shift explored (provided it does not worsen score)
- As a result, crossings often treated as substitutions:



Cost:
2 substitutions
+ 1 shift

TERp limitations

- Multiple shifts are not possible in the same iteration
- Only best shift explored (provided it does not worsen score)
- As a result, crossings often treated as substitutions:



Cost:
2 substitutions
+ 1 shift

$\Rightarrow$ double shift not possible and one shift worsens score

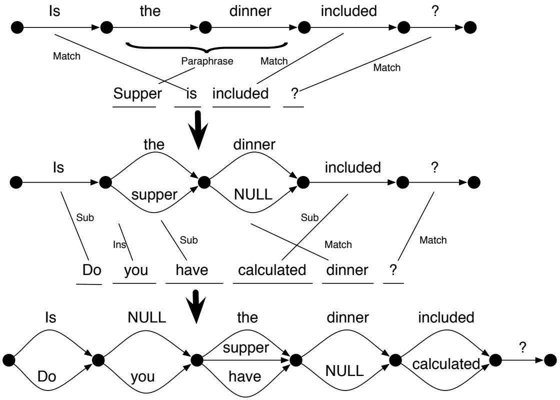
Extensions of the alignment module

In TERp, generate all possible shifts which do not degrade score

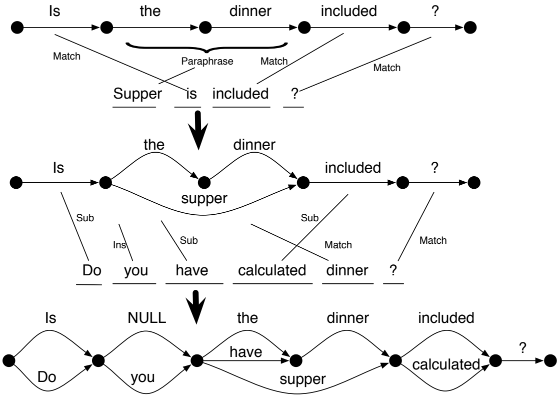
Tune TERp weights or use another aligner

- Editing costs can be tuned using Condor optimizer (available) (Condor not freely available any more $\Rightarrow$ not distributed with MANY)
- Experiments in progress with aligner based on linear models
- Issue: objective function used to tune editing or model costs
  - Cannot use TERp as objective function to optimise TERp
  - Use a pseudo-BLEU calculated on the confusion network
- No significant improvement with small number of systems (5)

Relax confusion network constraints



Is
the
dinner
included
?
Match
Paraphrase
Match
Match
Supper
is
included
?
the
dinner
Is
included
?
supper
NULL
Sub
Ins
Sub
Sub
Match
Match
Do
you
have
calculated
dinner
?
Is
NULL
the
dinner
included
supper
?
Do
you
have
NULL
calculated


Is
the
dinner
included
?
Match
Paraphrase
Match
Match
Supper
is
included
?
the
dinner
Is
included
?
supper
Sub
Ins
Sub
Sub
Match
Match
Do
you
have
calculated
dinner
?
Is
NULL
the
dinner
included
have
?
Do
you
supper
calculated

Decoder extensions

- weights on words $\Rightarrow$ confidence measure instead of system priors
- penalise bi-grams which do not appear in the system outputs [Rosti et. al, WMT 2011]

Conclusions

- MANY was run on five MT systems of different types
- The combination achieved a better BLEU score and TER score than the best single system (1.7 point gain in both cases), but a worse METEOR score
- We gave hints to integrate extra information about the systems in MANY
- We discussed some limitations and planned extensions of the current version of MANY