The ITC-irst Statistical Machine Translation System for IWSLT-2004

Nicola Bertoldi, Roldano Cattoni, Mauro Cettolo, Marcello Federico

ITC-irst,
via Sommarive 18, Povo (TN) - Italy
{bertoldi,cattoni,cettolo,federico}@itc.it

Abstract

Focus of this paper is the system for statistical machine translation developed at ITC-irst. It has been employed in the evaluation campaign of the International Workshop on Spoken Language Translation 2004 in all the three data set conditions of the Chinese-English track. Both the statistical model underlying the system and the system architecture are presented. Moreover, details are given on how the submitted runs have been produced.

1. Introduction

This paper reports on the participation of ITC-irst to the evaluation campaign organized by the International Workshop on Spoken Language Translation (IWSLT) 2004. The Statistical Machine Translation (SMT) system developed at ITC-irst was applied to all the three data set conditions of the Chinese-English track.

The ITC-irst SMT system implements an extension of the IBM Model 4 as a log-linear interpolation of statistical models, which estimate probabilities in terms of *phrases*. The use of phrases rather than words has recently emerged as a mean to cope with the limited context that Model 4 exploits to guess word translation (lexicon model) and word positions (distortion model) [1, 2, 3, 4, 5, 6, 7].

While parameters of the models are estimated exploiting statistics of phrase pairs extracted from word alignments, the weights of the interpolation are optimized through a training procedure which directly aims at minimizing translation errors on a development set.

Decoding is implemented in terms of a dynamic programming algorithm.

The paper is organized as follows. Next section details the statistical model underlying the system. Sections 3 and 4 briefly describe the search and the segmentation algorithms, respectively. Section 5 gives an overview of the system architecture. Finally, in Section 6 experimental set-ups of the evaluation campaign runs and results are presented and discussed.

2. Statistical Machine Translation

The advantages of the statistical translation approach are advocated by the many papers on the subject, which followed its first introduction. Of course, there have been also attempts to overcome some of its shortcomings, e.g. the use of limited context within the foreign string to guess word translations and word positions. Recently, several research labs have reported improvements in translation accuracy by shifting from word- to phrase-based SMT. In particular, statistical phrase-based translation models have recently emerged, which rely on statistics of phrase pairs. Phrase pairs statistics can be automatically extracted from word-aligned parallel corpora [5]. In the following subsections, we introduce the SMT framework and the Model 4. Then, we briefly describe a method for extracting phrase pairs. Finally, a novel phrase-based translation framework is presented which is tightly related to Model 4.

2.1. Log-linear Model

As originally proposed by [8], the most likely translation of a foreign source sentence $\mathbf{f}$ into English is obtained by searching for the sentence with the highest posterior probability:

$$\mathbf{e}^* = \arg\max_{\mathbf{e}} \Pr(\mathbf{e} \mid \mathbf{f}) \tag{1}$$

Usually, the *hidden* variable $\mathbf{a}$ is introduced:

$$\mathbf{e}^* = \arg\max_{\mathbf{e}} \sum_{\mathbf{a}} \Pr(\mathbf{e}, \mathbf{a} \mid \mathbf{f}) \tag{2}$$

which represents an *alignment* from source to target positions.
The framework of maximum entropy [9] provides a mean to directly estimate the posterior probability $\Pr(\mathbf{e}, \mathbf{a} \mid \mathbf{f})$. It is determined through suitable real valued feature functions $h_i(\mathbf{e}, \mathbf{f}, \mathbf{a}), i = 1 \ldots M$, and takes the parametric form:

$$p_{\boldsymbol{\lambda}}(\mathbf{e}, \mathbf{a} \mid \mathbf{f}) = \frac{\exp\{\sum_i \lambda_i h_i(\mathbf{e}, \mathbf{f}, \mathbf{a})\}}{\sum_{\mathbf{e},\mathbf{a}} \exp\{\sum_i \lambda_i h_i(\mathbf{e}, \mathbf{f}, \mathbf{a})\}} \tag{3}$$

The maximum entropy solution corresponds to values λ_i that maximize the log-likelihood over a training sample T:

$$\boldsymbol{\lambda}_* = \arg\max_{\boldsymbol{\lambda}} \sum_{(\mathbf{e},\mathbf{f},\mathbf{a})\in T} \log p_{\boldsymbol{\lambda}}(\mathbf{e},\mathbf{a} \mid \mathbf{f}) \qquad (4)$$

Unfortunately, a closed-form solution of (4) does not exist. An iterative procedure converging to the solution was proposed by [10]; an improved version is given in [11].
If the following feature functions are chosen [12]:

$$\begin{aligned} h_1(\mathbf{e},\mathbf{f},\mathbf{a}) &= \log \Pr(\mathbf{e}) \\ h_2(\mathbf{e},\mathbf{f},\mathbf{a}) &= \log \Pr(\mathbf{f},\mathbf{a} \mid \mathbf{e}) \end{aligned}$$

exploiting eq. (3), eq. (2) can be rewritten as:

$$\mathbf{e}^* = \arg\max_{\mathbf{e}} \Pr(\mathbf{e})^{\lambda_1} \sum_{\mathbf{a}} \Pr(\mathbf{f},\mathbf{a} \mid \mathbf{e})^{\lambda_2} \qquad (5)$$

where λ_i's represent scaling factors of factors.
In eq. (5), English strings $\mathbf{e}$ are ranked on the basis of the weighted product of the language model probability $\Pr(\mathbf{e})$, usually computed through an n-gram language model [13], and the marginal of the translation probability $\Pr(\mathbf{f},\mathbf{a} \mid \mathbf{e})$.
In [8, 14] six translation models (Model 1 to 6) of increasing complexity are introduced. These alignment models are usually estimated through the Expectation Maximization algorithm [15], or approximations of it, by exploiting a suitable parallel corpus of translation pairs. For computational reasons, the optimal translation of $\mathbf{f}$ is computed with the approximated search criterion:

$$\mathbf{e}^* \approx \arg\max_{\mathbf{e}} \Pr(\mathbf{e})^{\lambda_1} \max_{\mathbf{a}} \Pr(\mathbf{f},\mathbf{a} \mid \mathbf{e})^{\lambda_2} \qquad (6)$$

2.2. Model 4

Given the string $\mathbf{e} = e_1, \ldots, e_l$, a string $\mathbf{f}$ and an alignment $\mathbf{a}$ are generated as follows: (i) a non-negative integer ϕ_i, called *fertility*, is generated for each word e_i and for the null word e_0; (ii) for each e_i, a list τ_i, called *tablet*, of ϕ_i source words and a list π_i, called *permutation*, of ϕ_i source positions are generated; (iii) finally, if the generated permutations cover all the available source positions exactly once then the process succeeds, otherwise it fails.
Fertilities fix the number of source words to be aligned to each target word, and the total length of the foreign string. Moreover, as permutations of Model 4 are constrained to assign positions in ascending order, it can be shown that if the process succeeds in generating a triple $(\phi_0^l, \tau_0^l, \pi_0^l)$, then there is exactly one corresponding pair $(\mathbf{f},\mathbf{a})$, and vice-versa. This property justifies the following decomposition of Model 4:

$$\begin{aligned} p_\theta(\mathbf{f},\mathbf{a} \mid \mathbf{e}) &= p(\phi_0^l, \tau_0^l, \pi_0^l \mid e_0^l) && (7) \\ &= p(\boldsymbol{\phi}, \boldsymbol{\tau}, \boldsymbol{\pi} \mid \mathbf{e}) && (8) \\ &= p(\boldsymbol{\phi} \mid \mathbf{e}) \cdot p(\boldsymbol{\tau} \mid \boldsymbol{\phi}, \mathbf{e}) \cdot p(\boldsymbol{\pi} \mid \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{e}) && (9) \end{aligned}$$

where

$$\begin{aligned} p(\boldsymbol{\phi} \mid \mathbf{e}) &= \prod_{i=1}^{l} p(\phi_i \mid e_i)\, p(\phi_0 \mid \sum_{i=1}^{l} \phi_i) && (10) \\ p(\boldsymbol{\tau} \mid \boldsymbol{\phi}, \mathbf{e}) &= \prod_{i=0}^{l} p(\tau_i \mid \phi_i, e_i) && (11) \\ p(\boldsymbol{\pi} \mid \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{e}) &= \frac{1}{\phi_0!} \prod_{i=1}^{l} p(\pi_i \mid \phi_i, \bar{\pi}_{\rho(i)}) && (12) \end{aligned}$$

with

$$\begin{aligned} p(\tau_i \mid \phi_i, e_i) &= \prod_{k=1}^{\phi_i} p(\tau_{i,k} \mid e_i) && (13) \\ p(\pi_i \mid \phi_i, \bar{\pi}_{\rho(i)}) &= p_{=1}(\pi_{i,1} - \bar{\pi}_{\rho(i)}) \times \\ &\quad \times \prod_{k=2}^{\phi_i} p_{>1}(\pi_{i,k} - \pi_{i,k-1}) && (14) \end{aligned}$$

In eq. (9), the first factor is the *fertility model* $p(\boldsymbol{\phi} \mid \mathbf{e})$ - see eq. (10) - and represents step (i): fertilities of $e_1, \ldots, e_l$ are generated for each word according to the distributions $p(\phi_i \mid e_i)$, while the fertility of e_0 is generated through a Binomial distribution $p(\phi \mid m')$. The remaining factors, the *lexicon model* $p(\boldsymbol{\tau} \mid \boldsymbol{\phi}, \mathbf{e})$ - see eq. (11) - and the *distortion model* $p(\boldsymbol{\pi} \mid \boldsymbol{\phi}, \boldsymbol{\tau}, \mathbf{e})$ - see eq. (12) - correspond to step (ii): tablets for cepts[1] are generated according to eq. (13), and permutations π_i, with the exception of π_0, are generated according to eq. (14). The latter relies on two probability tables: $p_{=1}(\cdot)$, which considers the distance between the first generated position and the *center* [2] of the most recent cept; $p_{>1}(\cdot)$, which considers the distance between any two consecutively assigned positions of the permutation. Finally, positions for e_0 are generated at random over the residual ϕ_0 positions, with probability $\frac{1}{\phi_0!}$. It is worth remarking that the here considered distortion model omits some dependencies specified in [8].

2.3. Phrase-pair Extraction

The here used method exploits the so called *union alignments* between sentence pairs of the training corpus [5]. Given strings $\mathbf{f} = f_1, \ldots, f_m$ and $\mathbf{e} = e_1, \ldots, e_l$, a direct alignment $\mathbf{a}$ (from $\mathbf{f}$ to $\mathbf{e}$) and an inverted alignment $\mathbf{b}$ (from $\mathbf{e}$ to $\mathbf{f}$), the union alignment is defined as:

$$\mathbf{c} = \{(j,i) : a_j = i \ \vee \ b_i = j\} \qquad (15)$$

It is easy to verify that while $\mathbf{a}$ and $\mathbf{b}$ are many-to-one alignments, $\mathbf{c}$ is a many-to-many alignment. Moreover, the union

[1] A *cept* is a target word (including e_0) with positive fertility. A not-cept word may only generate an empty tablet and an empty permutation with probability 1.

[2] $\bar{\pi}_{\rho(i)}$ is defined as the ceiling of the mean position assigned to the most recent cept, whose index is defined by $\rho(i)$.

alignment does not necessarily cover all source and target positions.

Given a source-target sentence pair $(\mathbf{f}, \mathbf{e})$ and a union alignment $\mathbf{c}$, let J and I denote two closed intervals within the positions of $\mathbf{f}$ and $\mathbf{e}$, respectively. We say that I and J form a *phrase pair*[3] under $\mathbf{c}$ if and only if $\mathbf{c}$ aligns all source positions J with target positions contained in I, and all target positions I with source positions contained in J.

Given a parallel corpus provided with Viterbi alignments in both directions, we can compute all phrase pairs occurring in its sentences:

$$\mathcal{P} = \{(\tilde{f}^p, \tilde{e}^p) : p = 1, \ldots, P\} \tag{16}$$

Practically, in order to limit the number of phrases, the maximum length of I and J is limited to some value k. Note that the set $\mathcal{P}$ also includes phrase pairs with one single target word.

2.4. Phrase-based Model

We assume that the target vocabulary is augmented by including all target phrases in $\mathcal{P}$. Hence, the search criterion (6) is modified as follows:

$$\tilde{\mathbf{e}}^* = \arg\max_{\tilde{\mathbf{e}}} \Pr(\tilde{\mathbf{e}})^{\lambda_1} \max_{\mathbf{a}} p_\theta(\mathbf{f}, \mathbf{a} \mid \tilde{\mathbf{e}})^{\lambda_2} \tag{17}$$

where $\tilde{\mathbf{e}}$ ranges over all strings of the augmented target vocabulary.

Our phrase-based language model $\Pr(\tilde{\mathbf{e}})$ is a simple extension of a n-gram word-based language model.

The phrase model exploits a counting probability measure defined on the phrase sample $\mathcal{P}$. Hence, the relative frequency of a given phrase pair $(\tilde{f}, \tilde{e})$ in the sample $\mathcal{P}$ is interpreted as the probability of the phrase pair, given the training data. Basic probabilities of the translation model relying on statistics over $\mathcal{P}$ are summarized in Table 1. $\tilde{f}(\tau)$ trivially transforms τ into a phrase.

The implicit assumption that the tablet must correspond to a source phrase, i.e. it must cover consecutive positions, is made explicit by the distortion model. In fact, it assigns the first tablet position the same probability given by the Model 4 distortion model, but constrains successive positions to be adjacent.

3. Decoding Algorithm

Given the source sentence $\mathbf{f} = f_1^m$, the optimal translation $\tilde{\mathbf{e}}^*$ is searched through the approximate criterion (17).

According to the *dynamic programming* paradigm, the optimal solution can be computed through a recursive formula which expands previously computed partial theories, and recombines the new expanded theories. A theory can be described by its *state*, which only includes information needed for its expansion; two partial theories sharing the same state are identical (undistinguishable) for the sake of expansion, i.e. they should be recombined.

Table 1: Phrase-based model: fertility, lexicon, and distortion probabilities.

$$\begin{aligned}
N(\tilde{f}, \phi, \tilde{e}) &= \sum_{p=1}^{P} \delta(\tilde{f}^p = \tilde{f})\, \delta(\tilde{e}^p = \tilde{e})\, \delta(|\tilde{f}^p| = \phi) \\
N(\phi, \tilde{e}) &= \sum_{\tilde{f}} N(\tilde{f}, \phi, \tilde{e}) \\
N(\tilde{e}) &= \sum_{\phi} N(\phi, \tilde{e})
\end{aligned}$$

$$\begin{array}{ll}
\text{Fertility Model:} & \tilde{p}_S(\phi \mid \tilde{e}) = \dfrac{N(\phi, \tilde{e})}{N(\tilde{e})} \\
\text{Lexicon Model:} & \tilde{p}_S(\tau \mid \phi, \tilde{e}) = \dfrac{N(\tilde{f}(\tau), \phi, \tilde{e})}{N(\phi, \tilde{e})} \\
\text{Distortion Model:} & \tilde{p}_S(\pi \mid \phi, \bar{\pi}) = p_{=1}(\pi_1 - \bar{\pi}) \times \\
& \qquad \displaystyle\prod_{k=2}^{\phi} \delta(\pi_k - \pi_{k-1} = 1)
\end{array}$$

More formally, let $Q_i(s)$ be the best score among all partial theories of length i sharing the state s, $pred(s)$ the set of partial theories which are expanded in a theory of state s, and $G(s', s)$ the cost for expanding a partial theory of state s' into one of state s. The score Q^* of the optimal solution $\tilde{\mathbf{e}}^*$ can be computed by explicitly searching among optimal solutions fixing the length i and the state s, i.e.:

$$\begin{aligned}
Q^* &= \max_{\tilde{\mathbf{e}}} \Pr(\tilde{\mathbf{e}}) \max_{\mathbf{a}} p_\theta(\mathbf{f}, \mathbf{a} \mid \tilde{\mathbf{e}}) && (18) \\
&= \max_{i} \max_{\tilde{e}_1^i} \Pr(\tilde{e}_1^i) \max_{\mathbf{a}} p_\theta(\mathbf{f}, \mathbf{a} \mid \tilde{e}_1^i) && (19) \\
&= \max_{i,s} Q_i(s) && (20)
\end{aligned}$$

Henceforth, the score $Q_i(s)$ can be defined recursively with respect to the length i as follows:

$$Q_i(s) = \max_{th' \in pred(s)} Q_{i-1}(s(th')) * G(s(th'), s) \tag{21}$$

with a suitable initialization for $Q_0(s)$.

Given the model described in the previous section, the state $s(th) = (\mathcal{C}, \bar{\pi}, \tilde{e}', \tilde{e})$ of a partial theory th includes the coverage set, $\mathcal{C}$, the center of the last cept, $\bar{\pi}$, and the last two output phrases, $\tilde{e}' and \tilde{e}$. A theory of state $s = (\mathcal{C}, \bar{\pi}, \tilde{e}', \tilde{e})$ can be only generated from one of state $s' = (\mathcal{C} \setminus \pi, \bar{\pi}', \tilde{e}'', \tilde{e}')$, i.e. a new output phrase $\tilde{e}$ is added with fertility $\phi = |\pi|$, and ϕ positions are covered. Notice that if $\phi = 0$ the center remains unaltered, i.e. $\bar{\pi}' = \bar{\pi}$. The possible initial states $s = (\pi_0, \bar{\pi}_0, \epsilon, \epsilon)$ correspond to partial theories with no target phrases and with all ϕ_0 positions in $\mathcal{C} = \pi_0$ covered by the null phrase $\tilde{e}_0$. Notice that $\bar{\pi}_0$ is not used in the computation.

[3] In order to distinguish between words and phrases and between word-based and phrase-based models, the latter will be identified with the symbol˜ through all the rest of the paper.

Hence, eq. (21) relies on the following definitions:

$$\begin{aligned}G(s',s) &= G((\mathcal{C}\setminus\pi,\bar{\pi}',\tilde{e}'',\tilde{e}'),(\mathcal{C},\bar{\pi},\tilde{e}',\tilde{e}))\\ &= p(\tilde{e}\mid\tilde{e}'',\tilde{e}')\times\\ &\quad\times\begin{cases}p(\phi_i,\tau_i,\pi_i\mid\tilde{e},\bar{\pi}') & \text{if } \pi\neq\emptyset\\ p(\phi_i=0\mid\tilde{e}) & \text{if } \pi=\emptyset\end{cases}\end{aligned}\tag{22}$$

$$Q_0(s) = Q_0(\pi_0,\bar{\pi}_0,\epsilon,\epsilon)\tag{23}$$

$$= p(\phi_0\mid m-\phi_0)\,p(\tau_0\mid\tilde{e}_0)\,\frac{1}{\phi_0!}\tag{24}$$

In order to reduce the huge number of theories to generate, some methods are used, which affect the optimality of the search algorithm:

- *Comparison with the best theory*: theories are pruned, whose score is worse than the so-far best found complete solution, as theory expansion always decreases the score.

- *Beam search*: at each expansion less promising theories are also pruned. In particular, two types of pruning define the beam:

  - *threshold pruning*: partial theories th whose score $Q_i(s(th))$ is smaller than the current optimum score Q^*_{curr} times a given factor T, i.e.

    $$\frac{Q_i(s(th))}{Q^*_{\text{curr}}} < T\ ,\tag{25}$$

    are eliminated;

  - *histogram pruning*: hypotheses not among the top N best scoring ones are pruned.

  These criteria are applied, first to all theories with a fixed coverage set, then to all theories of fixed output length.

- *Reordering constraint*: a smaller number of theories is generated by applying the so-called IBM constraint on each additionally covered source position, i.e. by selecting only one of the first 4 empty positions, from left to right.

Figure 1 shows how theories are generated, recombined and pruned during the search process.

4. Chinese Segmentation

The Chinese word segmentation problem can be formulated as follows. Let

$$x_1^n = x_1, x_2, \ldots, x_n \qquad x_i \in \Sigma\tag{26}$$

be a string of characters (observations) representing a Chinese text, where Σ denotes the set of Chinese characters. We assume that the text is produced by concatenating words which are independent and identically distributed according to a distribution $P(w)$, defined over strings w of Σ:

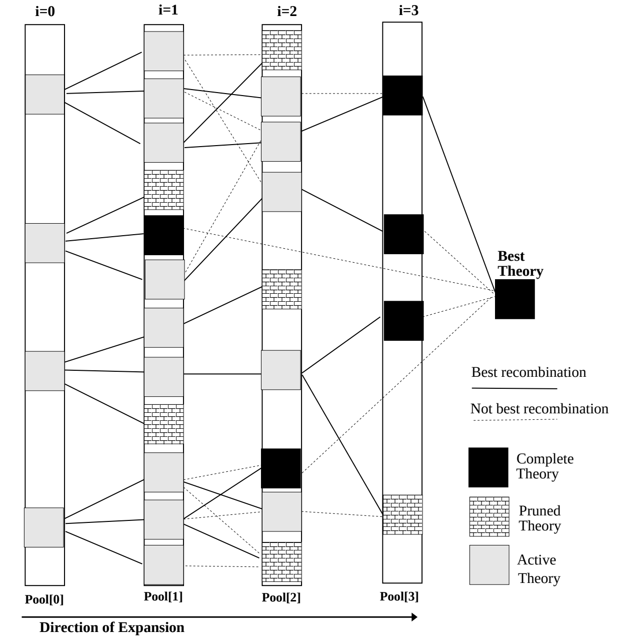

Figure 1: Expansion, recombination and pruning of theories during the search process.

$$\underbrace{x_1\cdots x_{n_1-1}}_{P(w_1)}\ \underbrace{x_{n_1}\cdots x_{n_2-1}}_{P(w_2)}\ \cdots\ \underbrace{x_{n_c}\cdots x_n}_{P(w_c)}\tag{27}$$

Hence, segmentation is the task of guessing the number of words c and of detecting the transition points $n_1^c = n_1$, $n_2 \ldots n_c$ within the original string. From a statistical perspective, we look for the segmentation which maximizes the text log-likelihood:

$$L^*(x_1^n) = \max_{c,n_1^c} L(x_1^n; c; n_1^c)\tag{28}$$

$$= \max_{c,n_1^c} \sum_{i=1}^{c+1} \log P(w = x_{n_{i-1}}^{n_i-1})\tag{29}$$

where $1 = n_0 < n_1 < n_2 < \ldots < n_c < n_{c+1} = n+1$. The maximization in eq. (29) can be solved by dynamic programming, while the word model can be defined as follows. Elementary statistics suggests that simple and effective word models can be built from word occurrence statistics collected within a large corpus of segmented texts. However, while just relying on word counting can be optimal in a closed-vocabulary situation, smoothing word probabilities with other less specific features can improve performance on texts including never observed words. Here, we present a word model including statistics of words, word lengths, and character sequences. More specifically, we assume the fol-

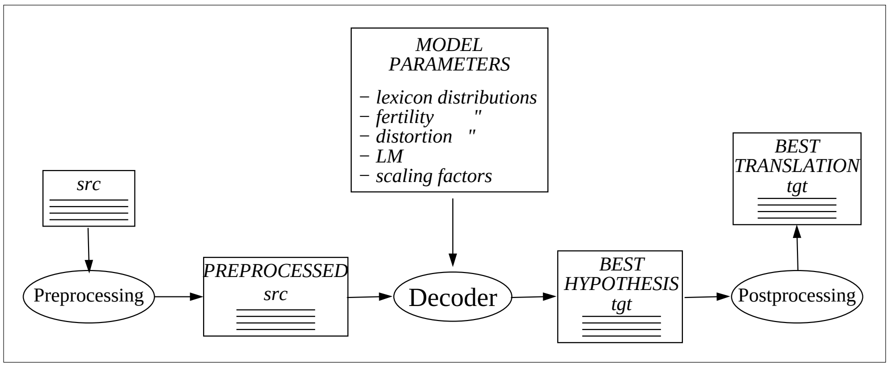

Figure 2: The architecture of the ITC-irst SMT system at run time: after preprocessing, the input sentence is sent to the decoder that, given the model parameters, searches for the best hypothesis. A final postprocessing step provides the actual translation.

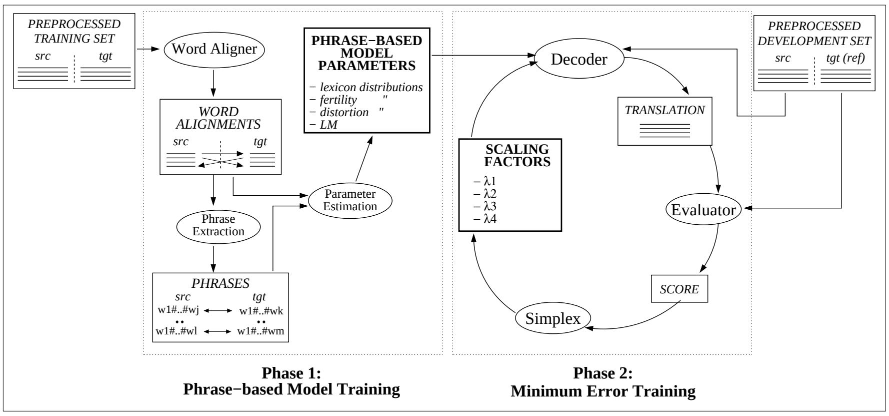

Figure 3: The two-phase architecture of the training system: first, the distributions of the components of the phrase-based model are estimated by means of alignments (left side). Then, the scaling factors of the components are computed by a minimum error training loop (right side).

lowing *back-off* word model over Σ^+:

$$P(w = x_1^l) = \begin{cases} (1-\lambda)\,\tilde{p}(w) & \text{if } \tilde{p}(w) > 0 \\ \alpha\,\lambda\,p(l, x_1^l) & \text{otherwise} \end{cases} \qquad (30)$$

where $\tilde{p}(w)$ is an empirical word distribution estimated on a segmented text sample, $\lambda \in (0,1)$ is a smoothing factor, α is a normalization term to ensure that $\sum_{w\in\Sigma^+} P(w) = 1$, and $p(l, x_1, \ldots, x_l)$ is a character-based language model. The character n-gram model is defined by:

$$p(l, x_1^l) = \tilde{p}(l) \prod_{i=1}^{l+1} p(x_i \mid x_{i-1}, l) \qquad (31)$$

where $\tilde{p}(l)$ is the empirical word-length distribution of the training data, $p(x_i \mid x_{i-1}, l)$ is a length conditional bigram language model, and x_0 and x_{l+1} are set to the conventional character \$ to model word boundaries. Bigram probabilities are estimated from a sample of words by applying the well-known Witten-Bell smoothing method [16].

5. System Architecture

The architecture of the ITC-irst SMT system at run time is shown in Figure 2. After a preprocessing step, the sentence in the source language is given as input to the decoder, which outputs the best hypothesis in the target language; the actual translation is obtained by a further postprocessing.

Preprocessing and postprocessing consist of a sequence of actions aiming at normalizing text and are applied both for preparing training data and for managing text to translate. The same steps can be applied to both source and target sentences, accordingly with the language. Input strings are tokenized, and put in lowercase. Text is labeled with few classes including cardinal and ordinal numbers, week-day and month names, years and percentages.

As training and decoding assume sentences divided into words, Chinese sequence of ideograms are segmented by means of the algorithm described in Section 4.

Parameters of the statistical translation model described in Section 2 can be divided into two groups: the parameters of each basic phrase-based model and the weights of their log-linear combination. Accordingly, the training procedure

Table 2: Experiments for the selection of additional training data. Results are given on the development set CSTAR-2003.

System	Additional Data		BLEU	NIST	MWER	MPER
name	monolingual	bilingual				
`baseline`			0.3001	7.0157	50.8	41.5
`lm-btec`	BTEC		0.3509	7.5099	47.2	38.1
`lm-db1`	BTEC, DB1		0.3466	7.4475	47.6	38.3
`lm-db2`	BTEC, DB2		0.3460	7.4427	47.1	38.3
`tm-btec`	BTEC	BTEC	0.4311	8.5336	42.0	33.3
`tm-db3`	BTEC	BTEC, DB3	0.4574	8.7890	39.7	30.5

of the system, shown in Figure 3, consists of two separate phases.

In the first phase, distributions of the components of the phrase-based models are computed starting from a parallel training corpus. After preprocessing, Viterbi alignments from source to target words, and vice-versa, are computed by means of the GIZA++ toolkit [1]. Phrase pairs are then extracted taking into account both direct and inverse alignments (see section 2.3), and the phrase-based distributions are estimated (section 2.4).

In the second phase the scaling factors of the log-linear model are estimated by the so-called *minimum error training* procedure. This iterative method searches for a set of factors that minimizes a given error measure on a development corpus. The simplex method [17] is used to explore the space of scaling factors. A detailed description of the minimum error training approach is reported in [18].

6. Experiments

ITC-irst participated to all the three data conditions of the Chinese-English track: Supplied, Additional, and Unrestricted data. According with the evaluation specification, in the last two conditions monolingual and bilingual training data can be added to the supplied corpus of 20K sentences. Experiments on a development set were performed to select these corpora in order to optimize performance of the system.

System development was performed on the CSTAR-2003 evaluation set, and then blindly applied to the IWSLT-2004 test set. No optimization has been done with respect to the post-processing required by the IWSLT-2004 evaluation campaign (e.g. absence of punctuation). The system has been trained in a standard way (e.g. with punctuation and with lower-case letters) and the required post-processing was simply applied to the output sentences as final step. The development of the system was done by considering the BLEU score, both in the data selection and in the optimization of the scaling factors.

6.1. Selection of additional training data

Adding data for training the system is an hard issue. Using more training data usually improves performance of the baseline system, provided these data are close enough to the domain of the test set. However, an exhaustive exploration of corpora available for the IWSLT evaluation for finding the best combination for training the system is unfeasible. Hence, first we searched for the best monolingual resources consisting of the English part of parallel corpora. Successively, we tried the effectiveness of additional bilingual resources. Note that no optimization of scaling factors is made in this phase.

The upper half of Table 2 summarizes the results of the selection of additional monolingual resources. Monolingual data are used only for estimating the language model. The `baseline` system was trained on the Supplied data.

Among the available corpora, the Basic Traveling Expression Corpus (BTEC) [19], a collection of 162K parallel sentences in several languages, is the closest to the task domain[4]. Using it, performance improvement over the baseline is about 17% relative. The impact on performance of other corpora was explored by training different language models on them, and combining the estimated models in a mixture [20]. Two groups of additional data are considered: DB1 mostly composed by news corpora[5] and DB2 consisting of press releases released by the Hong Kong Special Administrative Region[6]. In both cases, small relative decrements (-1.2% and -1.4%) of the BLEU score were observed. This behavior can be explained by the specificity of BTEC, whose domain - tourism - is different from those of the other corpora. Accordingly, the language model estimated over BTEC is used for all the following experiments.

Even more challenging is the selection of bilingual resources. In order to avoid constraints given by the Additional data condition, we worked under the Unrestricted data condition, that permits the use of any parallel corpus for training.

Two translation systems are trained on different sets of bilingual resources: `tm-btec` and `tm-db3` (see lower half of Table 2). The first system extends the supplied data with BTEC; the second one with a selection of other corpora available from LDC (DB3[7]). The `tm-btec` system signifi-

[4] In fact, both development and test sets are extracted from BTEC.

[5] The corpora are available from LDC: LDC2002E17, LDC2002E58, and LDC2002E18.

[6] LDC2003E25 and LDC2000T46.

[7] LDC2002E17, LDC2002L27, LDC2003E25, LDC2002E58, and

Table 3: Official results of the IWSLT-2004 evaluation campaign. Comparison between different types of Chinese segmentation.

Data Condition	Segmentation	BLEU	NIST	MWER	MPER
Supplied	Supplied	0.3156	7.1604	53.1	45.3
	Special	0.3493	7.0973	50.8	43.0
Additional	Supplied	0.3499	7.5199	51.0	43.3
	Special	0.3514	7.3958	49.7	42.0
	Full	0.3490	6.6185	51.9	44.5
Unrestricted	Supplied	0.3774	7.0880	50.0	43.4
	Special	0.4118	7.0908	47.7	41.0
	Full	0.4409	7.2413	45.7	39.3

cantly outperformes previous systems. The increment of the BLEU score is about 43% and 23% relative, with respect to the `baseline` and `lm-btec` systems, respectively. Performance of `tm-db3` system scored better than `baseline` and `lm-btec`, too.

The constraints on the use of training data for the three conditions and the above reported results on the development set suggested the employment of the following systems for the evaluation campaign: the `baseline` system in the Supplied data condition, the `lm-btec` in the Additional data condition, and the `tm-db3` in the Unrestricted data condition. The scaling factors that minimize the errors on the development set were estimated through the procedure mentioned in Section 5 and employed for the official evaluation.

6.2. Official evaluation

In developing the Chinese-English MT system for the IWSLT-2004 evaluation campaign we had to face the problem of having different Chinese word segmentation in the training corpora and in the test set. By assuming that each available data set provides its own segmentation, and that no knowledge is given about its characteristics, an interesting issue is to understand which choice is the best between (i) exploiting the provided segmentation or (ii) removing the provided segmentation and homogeneously re-segmenting all data.

Three types of segmentation were taken into account:

1. *Supplied*, the original Chinese segmentation provided in the training and test corpus was not changed and data were used as they were. This means that the segmentation step was skipped during the preprocessing.

2. *Special*, Chinese segmentation was applied from scratch by training the segmentation model (Section 4) on a 7K-entry word-frequency list extracted from the supplied data.

3. *Full*, Chinese segmentation was applied from scratch by training the segmentation model on a 44K-entry word-frequency list supplied by LDC.

Table 3 reports automatic scores on the official test set for each data condition and for each segmentation type. Concerning the Supplied data condition, results show that the Special segmentation outperforms the *Supplied* one in terms of BLEU score; the relative improvement is more than 10%. It is worth noticing that the *Full* segmentation is not permitted according to the Supplied data conditions. A reason for the large difference in performance is probably due to the fact that training and testing data were manually segmented by different people. Hence, the two data sets reflect different ways of interpreting the concept of word, which is quite frequent in Chinese. Hence, the approach of automatically re-segmenting all the data with one model produces the positive effect of making training and testing data more consistent.

By looking at the Additional data condition, we notice that the three segmentation modalities give comparable results.

In the Unrestricted data condition, results show that the *Full* Segmentation method achieves the best performance. The BLEU score relative improvement is about 17% and 7% with respect to *Supplied* and *Special* segmentations, respectively. This is not surprising because (i) the size of the training set is much larger than in the Additional data condition and (ii) the training set contains data much closer to the Chinese dictionary used by the segmenter. These numbers appear to confirm that the manual segmentation of the test set exhibits some differences with respect to the segmentation typically found in the LDC documents or even in the IWSLT-2004 supplied training set.

7. Acknowledgments

This work was partially financed by the European Commission under the project PF-STAR - Preparing Future Speech Translation Research (IST-2001-37599, http://pfstar.itc.it), and by the Province of Trento (FDR-PAT program) under the project WebFAQ.

LDC2002E18.

8. References

[1] F. J. Och and H. Ney, “Improved statistical alignment models,” in *Proc. of the 38th Annual Meeting of the Association for Computational Linguistics*, Hongkong, China, October 2000, pp. 440–447.

[2] K. Yamada and K. Knight, “A syntactic-based statistical translation model,” in *Proc. of the 39th Meeting of the Association for Computational Linguistics (ACL)*, Toulouse, France, 2001, pp. 523–530.

[3] D. Marcu and W. Wong, “A Phrase-based, Joint Probability Model for Statistical Machine Translation,” in *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2002.

[4] S. Vogel, Y. Zhang, F. Huang, A. Venugopal, B. Zhao, A. T. a nd M. Eck, and A. Waibel, “The CMU statistical machine translation system,” in *Proc. of the Machine Translation Summit IX*, New Orleans, LA, 2003.

[5] P. Koehn, F. J. Och, and D. Marcu, “Statistical phrase-based translation,” in *Proc. of HLT-NAACl 2003*, Edmonton, Canada, 2003, pp. 127–133.

[6] C. Tillmann, “A projection extension algorithm for statistical machine translation,” in *Proc. of Empirical Methods in Natural Language Processing (EMNLP)*, Sapporo, Japan, 2003, pp. 1–8.

[7] D. Marcu, “Towards a unified approach to memory- and statistical-based machine translation,” in *Proc. of the 39th Meeting of the Association for Computational Linguistics (ACL)*, Toulouse, France, 2001, pp. 378–385.

[8] P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer, “The Mathematics of Statistical Machine Translation: Parameter Estimation,” *Computational Linguistics*, vol. 19, no. 2, pp. 263–313, 1993.

[9] A. Berger, S. Della Pietra, and V. Della Pietra, “A Maximum Entropy Approach to Natural Language Processing,” *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.

[10] J. Darroch and D. Ratcliff, “Generalized Iterative Scaling for Log-Linear Models,” *The Annals of Mathematical Statistics*, vol. 43, no. 5, pp. 1470–1480, 1972.

[11] S. Della Pietra, V. Della Pietra, and J. Lafferty, “Inducing features of random fields,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 380–393, 1997.

[12] F. Och and H. Ney, “Discriminative training and maximum entropy models for statistical machine translation,” in *ACL02: Proc. of the 40th Annual Meeting of the Association for Computational Linguistics*, PA, Philadelphia, 2002, pp. 295–302.

[13] F. Jelinek, *Statistical Methods for Speech Recognition*. MIT Press Cambridge, Massachusetts, London, England, 1997.

[14] F. J. Och and H. Ney, “A systematic comparison of various statistical alignment models,” *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[15] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum-likelihood from incomplete data via the EM algorithm,” *Journal of the Royal Statistical Society, B*, vol. 39, pp. 1–38, 1977.

[16] I. H. Witten and T. C. Bell, “The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression,” *IEEE Trans. Inform. Theory*, vol. IT-37, no. 4, pp. 1085–1094, 1991.

[17] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1992.

[18] M. Cettolo and M. Federico, “Minimum Error Training of Log-Linear Translation Models,” In these proceedings.

[19] T. Takezawa, E. Sumita, F. Sugaya, H. Yamamoto, and S. Yamamoto, “Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world,” in *Proc. of 3rd International Conference on Language Resources and Evaluation (LREC)*, Las Palmas, Spain, 2002, pp. 147–152.

[20] N. Bertoldi, F. Brugnara, M. Cettolo, M. Federico, and D. Giuliani, “Cross-task portability of a broadcast news speech recognition system,” *Speech Communication*, vol. 38, no. 3-4, pp. 335–347, 2002.