Towards the Supervised Machine Translation: Real Word Alignments and translations in a Multi-task Active Learning process.

Martha Alicia Rocha
División de Estudios
de Posgrado e Investigación
Instituto Tecnológico de León, México
mrocha@dsic.upv.es

Joan Andreu Sánchez
Instituto Tecnológico de Informática
Universidad Politécnica de Valencia, Spain
jandreu@dsic.upv.es

Abstract

We present a study on the Active Learning (AL) paradigm applied to a multi-task labeling scenario for Statistical Machine Translation (SMT). The main goal of this research is to show that the learning of a phrase-based model can be improved despite having a small corpus annotated with word-level alignments. We propose a simple scheme for supervised training of a SMT model with a Multi-task AL (MTAL) approach to get a bilingual corpus with word-level alignments from the scarce data. The main advantage of the AL paradigms is the intelligent sampling using informativeness functions throughout the entire labeling process. We experimented with the MTAL approach and this approach is compared with the Single-Task AL (STAL) approach. The STAL process was used to semi-supervised training of SMT models. We found out that the MTAL approach improved the efficiency obtained by STAL process. In order to compare the performance of both the STAL and MTAL approaches, we experimented with two types of passive learners named random sampling. An assessment of the entire labeling process was done in order to obtain an overall performance compared to passive learners. In the analysis of the experiments it was found that the MTAL outperformed the passive learners and the STAL approach.

1 Introduction

The supervised learning problem in many applications of Natural Language Processing (NLP) is carried out by using manually annotated samples. It is well known that annotating data manually is very time consuming and expensive yet it benefits the model performance.

Currently, Statistical Machine Translation (SMT) can be seen as a semi-supervised learning problem in which stochastic models are learned from large parallel corpus by using efficient parameter estimation algorithms (Koehn, 2010). This paper is focused in the training of Phrase-based Statistical Translation (PBST) models (Ortiz-Martínez et al., 2005) from word-level alignments that are manually generated by human beings.

Due to the cost of the generation of such alignments, we propose to use Active Learning (AL) techniques to reduce the human effort. The AL approach has been proposed in the literature for selecting the most informative samples to be annotated while reducing the annotation effort (Settles and Craven, 2008; Olsson, 2009; Settles, 2010).

For many language pairs parallel corpora are not available, even in domains that are different from usual domains for which enough training data is available. In such situations, the training process can seen as an adaptation task. In this context, the PBST models can be trained with an out-of-domain parallel corpus and then exploring a small monolingual corpus that is multi-annotated in a progressive form using AL.

Our main goal is to present a framework to obtain efficient models iteratively. This is the basis of interactive systems such as Computer Aided

Sima'an, K., Forcada, M.L., Grasmick, D., Depraetere, H., Way, A. (eds.)
Proceedings of the XIV Machine Translation Summit (Nice, September 2–6, 2013), p. 239–247.

Translation (CAT) systems and Computer Aided Word Alignment (CAWA) systems. CAWA systems are important for the post-edition of word-level alignment parallel corpus of interest for some NLP communities [1]

Current SMT systems are trained in a generative form (Koehn, 2010). First, a word-level alignment between a source sentence e and a target sentence f is automatically obtained for all sentence pairs of the training data by means IBM models. The word-based translation of IBM models are the base for the PBST systems. Second, phrase-based translation are obtained from the word-level alignments. All phrase pairs that are consistent with the word-level alignment are collected for computing the probability distributions over collected phrase pairs.

In addition, reordering models and language models are learned from training data. The reordering model, language model and phrase model are combined in a log linear model to compose a final SMT system. The training of the different parts of a SMT system takes time and needs large amounts of resources because the estimation of the parameters is done on very large corpora.

Recently it was shown (Gascó et al., 2012) that less data suitably selected obtained better translation models than using all data. The sentences were sampled from a huge parallel corpus and in a scenario where the labels are available. In contrast, another different scenario is when the labels are unavailable and labeled data is necessary. In this last scenario, the system queries a human to annotate samples and conform a parallel corpus with few effort. Our approach is related with the last scenario in which AL can help us to obtain informative samples that aim to reduce human labeling effort.

AL techniques have been recently used in SMT systems and significant improvements have been obtained (Haffari et al., 2009; Haffari and Sarkar, 2009; Gangadharaiah et al., 2008; Ambati et al., 2011). The critical point is to define a simple measure of informativeness and representativeness into the AL process that contributes to improve SMT systems and the consequent reduction of human effort. We present a simple entropy-based measure for two AL approach: Single-Task Active Learning (STAL) and mainly Multi-Task Active Learning (MTAL). It is simple because the measure does not use internal information extracted from the models (e.g. language models, models of alignments, translation models) that can be complex for a user with few knowledge of the models.

In AL, the word *task* usually means making la belling or annotation of samples. The STAL approach (Settles and Craven, 2008) is used to assess to the active learner for a one task. The MTAL approach (Reichart et al., 2008) was introduced as an extension to the STAL approach. In the MTAL approach several learners learn in parallel different aspects from each sample. In the MTAL approach, the samples are chosen according to several criteria that are defined by several active learners. The chosen samples are then annotated with multiple information for satisfying the needs of each active learner.

The SMT systems are built from several models, and each model can be considered as an active learner. In this research we propose a MTAL algorithm to train SMT systems. Two annotation tasks are considered in this paper: word-level alignments and translations. Our claim in this research is that the training process of a SMT system can be improved by providing more information than only the paired sentences. We showed that word-level alignments manually generated improved the PBST systems in a MTAL scenario.

Section 2 presents the STAL process and a function to measure the informativeness of sample based on entropy (Shannon, 2001). The MTAL approach for SMT proposed in this work is presented in Section 3. The evaluation of the MTAL protocol is explained in Section 4, and Section 5 shows the experiments that were performed. A final discussion is done in the conclusions section.

2 Single-task active learning for labeling data

AL seeks a response to unsupervised learning problems where the data may be abundant but the labels are scarce or expensive. In this scenario, the labeling of a data sample occurs after the algorithm has asked explicitly the corresponding label. The goal of active data selection is to reach the same accuracy as a supervised algorithm. In this sense, an oracle can be a human annotator that is

[1] p.e. *"Shared task at NAACL 2003"* `http://www.cse.unt.edu/~rada/wpt` and *"ACL 2005 workshops"* `http://www.cse.unt.edu/~rada/wpt05`

queried in different forms: membership query synthesis, stream-based selective sampling or pool-based sampling (Settles, 2010). The last framework is adopted for this research and we assume a monolingual corpus as a pool of unannotated sentences.

In the AL framework, a function $\rho()$ is used to measure the informativeness of an unlabeled training instance x. A general strategy of uncertainty sampling that has demonstrated to be a good assessment measure uses the entropy:

$$\rho_{\Theta}(x) = -\sum_{\widehat{y}} P(\widehat{y}|x;\Theta)\log P(\widehat{y}|x;\Theta) \quad (1)$$

where $\widehat{y}$ ranges for all possible labellings of x under the model Θ. Variable $\widehat{y}$ can be infinite, and therefore in SMT, it can be approximated with an n-best list obtained from a word graph (Hasan et al., 2007); Θ represents the model parameters that are used to account for input x. An initial set of labeled data $\mathcal{L}$ is used for obtaining Θ and a pool of unlabeled data $\mathcal{U}$ is supposed to be available. The AL algorithm selects the most informative instance according to:

$$x^* = \arg\max_x \rho_{\Theta}(x) \quad (2)$$

A density weighted method (Settles and Craven, 2008) can be used to reduce the bias in the sampling because an instance may lie on the decision boundary, but may not be representative of other instances in the distribution. For this purpose, the similarity is used for measuring the representativeness and a sample x^* is chosen according to:

$$x^* = \arg\max_x \rho_{\Theta}(x)\left(\frac{1}{|\mathcal{U}|}\sum_{u=1}^{\mathcal{U}}\text{sim}(x,x_u)\right)^{\beta} \quad (3)$$

where the function $\text{sim}(x,x_u)$ measures the similarity between x and x_u and β controls the importance of the density term.

Figure 1 shows the STAL pool-based sampling algorithm applied to a translation task. A trainTrad() function trains the language model Θ_{LM} and the translation models Θ_T[2]. The quality score of the current model was the BLEU (Papineni et al., 2002) in this paper and it was measured with a test set Γ with the function evaluate$_{\Theta_{LM},\Theta_T}()$[3]. The decoding$_{\Theta_{LM},\Theta_T}()$ function obtains the translation of unlabeled samples and the set $\widehat{\mathbf{Y}}$ of n-best translations $\widehat{y}$ of each sample x. The B most informative samples are selected into the **for** loop according to a strategy of sampling (see equation (2) and (3)), and then the selected samples are labeled with function labelT() and the corpus $\mathcal{L}$ and $\mathcal{U}$ were updated. Section 4 discusses the stopping criterion and adequate options for reaching a good performance with the AL algorithm.

Data: $\mathcal{L}$, $\mathcal{U}$, Γ, batch size B
Result: Θ_{LM},Θ_T
while *stopping criterion* **do**
  `// train models`
  [Θ_{LM}, Θ_T] = trainTrad($\mathcal{L}$)
  quality = evaluate$_{\Theta_{LM},\Theta_T}(\Gamma)$
  `// get hypothesis`
  $\widehat{\mathbf{Y}}$ = decoding$_{\Theta_{LM},\Theta_T}(\mathcal{U})$
  `// sampling`
  for *i=1 to B* **do**
    `// get best sample`
    $x^* = \arg\max_{x\in\mathcal{U}} \rho_{\Theta_{LM},\Theta_T}(x)$
    `// labeling and updating`
    $\mathcal{L} = \mathcal{L} \cup \langle x^*, \text{labelT}(x^*)\rangle$
    $\mathcal{U} = \mathcal{U} - x^*$
return Θ_{LM},Θ_T

Figure 1: STAL algorithm for PBST systems.

3 Multi-task AL for Supervised PBST systems

The STAL approach (see Section 2) can be extended to multiple labeling processes. A SMT can be seen as a system composed of several parts and where the Multi-Task AL (MTAL) (Reichart et al., 2008) approach can be used to annotate appropriate learning information for the specific parts of the system. This section discusses a MTAL approach for training PBST in a supervised way. In this paper we have considered two tasks: one task is the annotation of word-level alignments, and the other task is the annotation of translations. Note that in

[2]In fact, the function trainTrad() gets too the IBM model that is used to generate the PBST-reordered model (Θ_T).

[3]The function evaluate$_{\Theta_{LM},\Theta_T}()$ gets the translation of the corpus Γ and is evaluated with its real translation.

PBST system, the translation models are obtained from word-level alignments.

The main components of a PBST system are a language model Θ_{LM} and a translation model Θ_T. During the PBST training process, the Θ_{LM} and Θ_T models are obtained from paired sentences $\mathcal{L}_t$ that are aligned at word-level $\mathcal{L}_a$.

An initial corpus $\mathcal{L}_t$ is used to train a PBST system and to estimate the word-level alignments and conform an initial $\mathcal{L}_a$. The corpus $\mathcal{L}_t$ is a set of bilingual sentence-level alignments and $\mathcal{L}_a$ is a set of word-level alignments estimated with IBM models. Both inputs are necessary in the MTAL process.

Basically, three models were used to calculate the informativeness of sentences. The Θ_T and Θ_{LM} were used to get the set $\widehat{\mathbf{Y}}_t$ of n-best translations and the Θ_A model was used to get the set $\widehat{\mathbf{Y}}_a$ of n-best alignments $\widehat{\mathbf{y}}_a$ from the corpus $\mathcal{U}$. The n-best alignments $\widehat{\mathbf{y}}_a$ do not correspond necessarily to the n-best translations $\widehat{\mathbf{y}}_t$. Then, the sentences were ranked according to the following combined score:

$$\rho_{comb_\Phi}(x) = \sum_{i=1}^{|\Phi|} w_{\boldsymbol{\Theta}_i} \rho_{\boldsymbol{\Theta}_i}(x) \qquad (4)$$

where $\Phi = [\Theta_1, ..., \Theta_i, ..., \Theta_n]$ is the set of models and $w_{\boldsymbol{\Theta}_i}$ is a weighting factor associated to Θ_i (Reichart et al., 2008).

Figure 2 shows the algorithm that was designed to execute the proposed MTAL approach. Note that the algorithm follows the AL pool-based sampling mentioned in Section 2.

The set $\widehat{\mathbf{Y}}_t$ was obtained from phrase-based decoding and the set $\widehat{\mathbf{Y}}_a$ was obtained from word-level alignment decoding[4]. In fact the alignments were obtained as a by product of the decoding process in order to measure informativeness on the hypotheses.

The best ranked sentences were manually annotated with their translations and with their alignments. Finally these annotated sentences were added to $\mathcal{L}_t$ and $\mathcal{L}_a$. The function $\rho_{comb_{[\Theta_A,\Theta_{LM},\Theta_T]}}()$ used two weighting factors $[w_{\Theta_A}, w_{\Theta_{LM},\Theta_T}]$.

[4]The GIZA++ decoding is used to estimate $\widehat{\mathbf{Y}}_a$ only for one direction: source to target. Note that to estimate the alignments from source to target of $\mathcal{U}$ no the translation is needed.

Data: $\mathcal{L}_t$, $\mathcal{U}$, $\mathcal{L}_a$, Γ, batch size B
Result: Θ_{LM},Θ_T
while *stopping criterion* **do**
  // Estimate alignments
  $\Theta_A = \text{calAlign}(\mathcal{L}_t)$
  // Train LM and translator
  $[\Theta_{LM}, \Theta_T] = \text{trainTrad}(\mathcal{L}_t, \mathcal{L}_a)$
  // Assessment of translator
  $\text{quality} = \text{evaluate}_{\Theta_{LM},\Theta_T}(\Gamma)$
  // get hypotheses
  $\widehat{\mathbf{Y}}_t = \text{decoding}_{\Theta_{LM},\Theta_T}(\mathcal{U})$
  $\widehat{\mathbf{Y}}_a = \text{decoding}_{\Theta_A}(\mathcal{U})$
  // sampling
  for *i=1 to B* **do**
    // get best sample
    $x^* = \arg\max_{x \in \mathcal{U}} \rho_{comb_{[\Theta_A,\Theta_{LM},\Theta_T]}}(x)$
    // labeling and updating
    $\mathcal{L}_t = \mathcal{L}_t \cup \langle x^*, \text{labelT}(x^*)\rangle$
    $\mathcal{L}_a = \mathcal{L}_a \cup \langle x^*, \text{labelA}(x^*)\rangle$
    $\mathcal{U} = \mathcal{U} - x^*$
return Θ_{LM},Θ_T

Figure 2: MTAL algorithm for supervised PBST systems.

4 Assessment of the learning process

A basic goal of AL is to reduce the amount of effort in annotating training data to achieve an accurate model. To assess this goal, (Abe and Mamitsuka, 1998) and (Melville and Mooney, 2004) proposed two metrics to measure the efficiency of the data used for training. Also, to assess the deficiency, a metric was used to calculate the overall performance (OP) of a process with respect to other process. This is, the OP assesses and measures the global deficiency between processes (Baram et al., 2004).

4.1 Data utilization ratio

Suppose that a learning process is performed through a number of iterations p for achieving an optimal final model. A performance is reached in each iteration and it is determined by a quality measure q. Suppose we want to compare the efficiency of n different learning algorithms $\Psi = \{\psi_1, ..., \psi_k, ..., \psi_n\}$. The following function t computes the iteration i where a learning algo-

rithm ψ_k reaches the maximum value of q:

$$t(\psi_k) = \arg\max_i q_i(\psi_k) \quad (5)$$

where q_i is the quality of learning process ψ_k in the iteration i, for $0 \leq i \leq p$ and $1 \leq k \leq n$.

The following function t' computes the iteration in which a learning algorithm ψ_j reaches the same or better quality q than the algorithm ψ_k:

$$t'(\psi_k, \psi_j) = \min\{i : 0 \leq i \leq p \text{ and } q_i(\psi_j) \geq q_{t(\psi_k)}(\psi_k)\} \quad (6)$$

The Data Efficiency Ratio (DER) as:

$$\text{DER} = \frac{t(\psi_k)}{t'(\psi_k, \psi_j)} \quad (7)$$

The DER can be seen as a measure of the speed to achieve the same or better performance of learning algorithm ψ_j compared with the algorithm ψ_k. Note that the larger the DER is the better the ψ_j is with regard to ψ_k.

The Data Utilization Ratio (DUR) is equivalent to the DER, but the DUR can be seen as a degree of the exploitation of the data. Let us suppose that the top performance of the learning algorithm ψ_k is achieved in iteration i. The function $\zeta_i()$ calculates the number of instances that have been added to the initial corpus $\mathcal{L}_0$ when the iteration i is executed for the learning algorithm ψ_k:

$$\zeta_i(\psi_k) = |\mathcal{L}_i| - |\mathcal{L}_0| \quad (8)$$

where $i = t(\psi_k)$.

We defined another function that provide the number of instances added to the labeled corpus $\mathcal{L}_0$ in the earliest iteration l where the performance of the learning algorithm ψ_j is equal to or better than the learning process ψ_k:

$$\zeta'_l(\psi_k, \psi_j) = |\mathcal{L}_l| - |\mathcal{L}_0| \quad (9)$$

where $l = t'(\psi_k, \psi_j)$. The DUR ratio is written as:

$$\text{DUR} = \frac{\zeta_i(\psi_k)}{\zeta'_l(\psi_k, \psi_j)} \quad (10)$$

Note that the larger the DUR ratio is, the better the ψ_j is with regard to ψ_k.

4.2 Overall performance

While DER and DUR assess the use of the data, a metric proposed by (Baram et al., 2004) measures the deficiency between a pair of learning process (ψ_j, ψ_k). The definition of Overall Performance (OP) of a complete learning process ψ_j compared with the process ψ_k is defined as:

$$\text{OP}_p(\psi_j, \psi_k) = \frac{\sum_{i=0}^{p}(q_p(\psi_k) - q_i(\psi_j))}{\sum_{i=0}^{p}(q_p(\psi_k) - q_i(\psi_k))} = \frac{(p+1)q_p(\psi k) - \sum_{i=0}^{p} q_i(\psi_j)}{(p+1)q_p(\psi k) - \sum_{i=0}^{p} q_i(\psi_k)} \quad (11)$$

where p is the time of the complete process measured in iterations, q_p is the score of quality in the time p of a learning process in set Ψ and q_i is the score of quality of a learning process in set Ψ in the time i. Value OP is the ratio of the areas of two learning curves. A low value of OP indicates a high quality of process ψ_j with regard to ψ_k.

5 Experiments

To show the difference in performance between the STAL and the MTAL approaches, first, we performed an experiment with the STAL approach. We used the Information Density (*ID*) function according the equation (3) and it was compared with a *Random* selection.

Second, we carried out three different experiments with the MTAL approach: 1) a random selection, where both translations and word-level alignments were used (this experiment is referred as *Random+*), 2) a MTAL experiment where the weight factors went $w_{\theta_{\mathbf{A}}} = 0$ and $w_{[\theta_{\mathbf{LM}}, \theta_{\mathbf{T}}]} = 1$ according to equation (4), means that we only used the set $\widehat{\mathbf{Y}}_t$ to measure the informativeness for the sampling but the annotation was made to the translations and alignments (this experiment is referred as *non-combined-ID+*) and, 3) a MTAL experiment where the weight factors went $w_{\theta_{\mathbf{A}}} = w_{[\theta_{\mathbf{LM}}, \theta_{\mathbf{T}}]} = 1$ and sets $\widehat{\mathbf{Y}}_a$, $\widehat{\mathbf{Y}}_t$ were used to combine the informativeness according to equation (4) (this is referred as *combined-ID+*). These weight factors were assigned by hand, but in future work is expected to learn these weights automatically.

For the experiments, we used two corpus that had paired French-English sentences: the News-

Table 1: Characteristics of the initial corpus $\mathcal{L}_t$

	Fr.	En.
Sentences	37.8K	
Running words	714.8K	623.8K
Vocabulary	30.6K	25.1K

Table 2: Characteristics of pool $\mathcal{U}$

	Fr.	En.
Sentences	300	
Running words	5.3K	4.7K
Vocabulary	1.5K	1.3K
$\text{OoV}_{\text{NewsComm}}$	140	95

Commentary corpus[5] and the Hansards corpus[6]. The Hansards corpus was chosen because it was manually annotated with word-level alignments and it was used to simulate the annotation process. This corpus is small but it is important to say that larger public corpus is not available.

An initial set $\mathcal{L}_t$ of 37.8K sentences of the News-Commentary bilingual corpus was used to create the initial models Θ and estimate the alignments to create the initial corpus $\mathcal{L}_a$. The Table 1 shows the characteristics of the set $\mathcal{L}_t$.

The Hansards corpus was used to define three sets: a) a set of 300 sentences whose source sentences were used as the pool of monolingual corpus $\mathcal{U}$; b) a set of 47 sentences that was used as the tuning corpus, and c) a set of 100 paired sentences used as the test corpus Γ. Note that this is a small corpus, unfortunately there not exist many public corpora with annotated alignments. Therefore, the experiments in this section should be considered as a MTAL solution for task adaptation. See Tables 2 and 3 for some statistics of the corpus $\mathcal{U}$ and Γ. The $\text{OoV}_{\text{NewsComm}}$ were the words (singletons) of News-Commentary that were not in the pool and tuning sets of the Hansards corpus. The OoV_{tr} were the words (singletons) of training corpus that were not in the test corpus.

We used GIZA++[7] to estimate the word-level alignments. GIZA++ was also used to obtain the

Table 3: Characteristics of the test corpus Γ

	Fr.	En.
Sentences	100	
Running words	1.7K	1.6K
Vocabulary	665	608
OoV_{tr}	326	265

set $\widehat{\mathbf{Y}}_a$ for MTAL approach. The SRILM[8] toolkit was used to learn the Θ_{LM} model. Then, the Thot[9] toolkit was used to train the PBST model Θ_T. The set $\widehat{\mathbf{Y}}_t$ was obtained by the Carmel[10] finite-state transducer from word graphs generated by a Multi-Stack decoding (Ortiz-Martínez et al., 2006).

The manual annotation process, for simplicity, was simulated. When the corpus $\mathcal{L}_t$ and $\mathcal{L}_a$ were updated then a retraining process using the downhill simplex algorithm (Press et al., 2002) parameter tuning was done. The new models were evaluated with the corpus Γ.

During the AL sampling process (see Fig. 1 and 2) the parameter β in equation (3) was set to 1, and the batch size B was 20. The normalized edit distance (NED) (Vidal et al., 1995) was used to measure the similarity among sentences from $\mathcal{U}$. The size of the n-best lists was 1,000 both for the word alignments and translations. The scores of n-best lists were adequately normalized. The BLEU score was used to evaluate the quality of the models.

Figure 3 shows in the Y-axis the evaluation and the X-axis is the iteration of the algorithm. In each iteration, were added 20 sentences to the bilingual corpus $\mathcal{L}$ in the STAL algorithm and, $\mathcal{L}_t$ and $\mathcal{L}_a$ in MTAL algorithm. To give statistical significance, the BLEU score was validated with bootstrap resampling to 1,500 repetitions and 95% of confidence. The scores of the *Random* and *Random+* experiments are the average of ten repetitions.

Figure 3 (top) shows the BLEU score in the STAL scenario. The overall process indicated that the *ID* experiment was better in more iterations than the *Random* experiments. One of the problems of STAL approach is its instability. Schohn *et al.* (Schohn and Cohn, 2000) also observed that

[5] http://www.statmt.org/wmt10/shared-task.html
[6] http://www.cse.unt.edu/~rada/wpt/
[7] http://code.google.com/p/giza-pp/
[8] http://www.speech.sri.com/projects/srilm/
[9] http://sourceforge.net/projects/thot/
[10] http://www.isi.edu/licensed-sw/carmel/

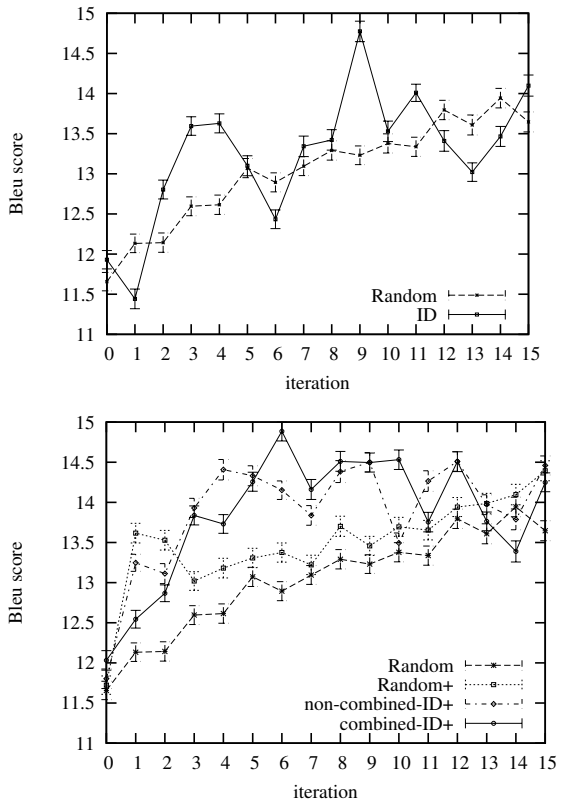

Figure 3: Comparative of BLEU learning curves for each iteration of STAL algorithm (top) and MTAL algorithm (down).

when the instances were selected at random from the pool of unlabeled data, the classifier performance increased monotonically. However, when the instances were added according to an AL strategy, the model performance may go up at a level above than achieved when using all available data. In the STAL experiment, peaks of this kind existed but there were also points that not outperformed the baseline (*Random*). This problem could be alleviated using multiple learners as discussed in this paper.

Figure 3 (below) shows the BLEU score in the MTAL scenario. The experiments showed that word-level alignments manually annotated by linguistic experts helped the learning process in the *Random+* experiment, that was better than *Random* experiment but the *Random+* performance was not increased monotonically . It is important to note that the *non-combined-ID+* and *combined-ID+* experiments were better than the *Random* and *Random+* experiments and more consistent than the *ID* experiment (see top of Fig.3) We see that the MTAL approach can help to reduce the labeled effort and obtain an optimal model. Also we note that the peaks mentioned above were reached in an early iteration. Next we explain the overall performance and data efficiency of the STAL and MTAL approaches.

Another important aspect is a performance analysis in the global process. Table 4 shows the OP measure both for the STAL and MTAL strategies. The OP was evaluated with the *Random* and *Random+* experiments according to equation (11) where the time of complete process p is 15. The OP measure for the *non-combined-ID+* and *combined-ID+* experiments were good with respect to *Random+* experiment and clearly better than the *Random* experiment. The low OP (see bold and italic value) indicates that the quality was better, that is, the alignments improved the results significantly over the *Random* experiment. As shown in Figure 3 the *non-combined-ID+* and *combined-ID+* experiments used informativeness that benefits the final model.

Table 4: OP ratio for the STAL and MTAL approaches.

	ψ_k	
Experiment (ψ_j)	*Random*	*Random+*
ID	0.90	1.32
non-combined-ID+	***0.54***	***0.60***
combined-ID+	0.66	0.74

The final discussion is concerned with DUR ratio. The DUR metric gives an idea of the efficiency of a measure of information of an AL approach with respect to random sampling. In addition, the DUR metric can help to take decisions about when stop the learning process. In (Olsson, 2009), it is proposed to use the target performance to decide stopping the process when it is reached or outperformed. We see that the AL approach in PBST can obtain an optimal performance with less data.

DUR ratio in Table 5 showed that the data efficiency was better in both *non-combined-ID+* and *combined-ID+* with respect to the *Random*.

The mentioned stopping criterion is useful when all data training is available but when the annotated data are not available, others criteria had been employed(Settles, 2010). In order to analyze the learning curves in this study the stop criterion was

when $\mathcal{U} = \emptyset$.

Table 5: DUR ratio for STAL and MTAL approaches.

	ψ_k	
Experiment (ψ_j)	*Random*	*Random+*
ID	1.56	1.67
non-combined-ID+	**3.5**	**3.75**
combined-ID+	2.8	3.0

Conclusions

The main goal of this research went show as real word-level alignments improved the phrase-based model estimation despite having a small corpus annotated with word-level alignments.

In this study we explored the advantage of the AL process in two ways: a single-task active learning and a multi-task active learning. It was observed in the experiments that the MTAL scenario was more stable than the STAL scenario. The usefulness of these approaches is to establish a framework to obtain multi-annotated corpus with minimal effort and incrementally in shared tasks to create linguistic resources.

A novelty of this study, unlike previous works(Haffari et al., 2009; Haffari and Sarkar, 2009; Gangadharaiah et al., 2008; Ambati et al., 2011), was to use the predictions to measure the informativeness of samples in a complex task such as SMT.

For future work, we are planning to use an efficient entropy calculation, because the word graphs were used to obtain the n-best list instead of all possible translations, and therefore the uncertainty measure was just an approximation. In addition, we plan to use new learners to chose appropriate sentences in a better form. We also plan to annotate in real time with CAWA and CAT system for extend or create multiannotated corpus.

Acknowledgments

This work was partially supported by the Spanish MEC under the STraDA research project (TIN2012-37475-C02-01), and by the Generalitat Valenciana under the grant Prometeo/2009/014. The first author is supported by the "División de Estudios de Posgrado e Investigación " of Instituto Tecnógico de León.

References

Abe, Naoki and Hiroshi Mamitsuka. 1998. Query learning strategies using boosting and bagging. In *Proceedings of the Fifteenth International Conference on Machine Learning*, ICML '98, pages 1–9, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Ambati, Vamshi, Stephan Vogel, and Jaime G. Carbonell. 2011. Multi-strategy approaches to active learning for statistical machine translation. In *Machine Translation Summit XIII*, pages 122–129, Xiamen, China.

Baram, Yoram, Ran El-Yaniv, and Kobi Luz. 2004. Online choice of active learning algorithms. *J. Mach. Learn. Res.*, 5:255–291, December.

Gangadharaiah, R., R. D. Brown, and J. Carbonell. 2008. Active learning in example-based machine translation. In *The 17th Nordic Conference on Computational Linguistics, (NODALIDA09)*, Odense, Denmark, May.

Gascó, Guillem, Martha-Alicia Rocha, Germán Sanchis-Trilles, Jesús Andrés-Ferrer, and Francisco Casacuberta. 2012. Does more data always yield better translations? In *Proceedings of the 13th European Chapter of the Association for Computational Linguistics*, pages 152–161.

Haffari, Gholamreza and Anoop Sarkar. 2009. Active learning for multilingual statistical machine translation. In *the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pages 181–189, Suntec, Singapore, 2-7 August.

Haffari, G., M. Roy, and A. Sarkar. 2009. Active learning for statistical phrase-based machine translation. In *Proc.HLT-NAACL*, pages 415–423, Boulder, Colorado, June. Association for Computational Linguistics.

Hasan, Saša, Richard Zens, and Hermann Ney. 2007. Are very large n-best lists useful for smt? In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, NAACL-Short '07, pages 57–60, Stroudsburg, PA, USA. Association for Computational Linguistics.

Koehn, Philipp. 2010. *Statistical Machine Translation*. Cambridge University Press, Edinburgh, UK, 1 edition.

Melville, Prem and Raymond J. Mooney. 2004. Diverse ensembles for active learning. In *In Proceedings of 21st International Conference on Machine Learning (ICML-2004*, pages 584–591. ACM Press.

Olsson, Fredrik. 2009. A literature of active machine learning in the context of natural language processing. Technical report, Swedish Institute of Computer Sciencies, 17 April.

Ortiz-Martínez, Daniel, Ismael García-Varea, and Francisco Casacuberta. 2005. Thot: A toolkit to train phrase-based statistical translation models. In *In Tenth Machine Translation*.

Ortiz-Martínez, D., I. García-Varea, and F. Casacuberta. 2006. Generalized stack decoding algorithms for statistical machine translation. In *Proceedings on the Workshop on Statistical Machine Translation, HLT-NAACL 2006*, pages 64–71.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Press, W.H., S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. 2002. *Section 10.4 Downhill Simplex Method in Multidimensions. Numerical Recipes in C++, The Art of Scientific Computing*. Cambridge University Press, USA, second edition edition.

Reichart, R., K. Tomanek, U. Hahn, and A. Rappoport. 2008. Multi-task active learning for linguistic annotations. In *Proceedings of ACL*, pages 861–869, Columbus, Ohio, U.S.A.

Schohn, Greg and David Cohn. 2000. Less is more: Active learning with support vector machines. pages 839–846. Morgan Kaufmann.

Settles, B. and M. Craven. 2008. An analysis of active learning strategies for sequence labelling tasks. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1069–1078.

Settles, Burr. 2010. Active learning literature survey. Technical report, University of Wisconsin-Madison, January.

Shannon, C. E. 2001. A mathematical theory of communication. *SIGMOBILE Mob. Comput. Commun. Rev.*, 5:3–55, January.

Vidal, Enrique, Andrés Marzal, and Pablo Aibar. 1995. Fast computation of normalized edit distances. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.