Project funded by the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme FP7/2007-2013/ under REA grant agreement no. 317471.

Project reference: 317471

Project full title: EXPloiting Empirical appRoaches to Translation

Improved Translation Workflow

Authors: Santanu Pal (USAAR)

Contributors: Josef van Genabith (USAAR), Marcos Zampieri (USAAR), Anna Zaretskaya (UMA)

Document Number: EXPERT_D2.2_20150508

Distribution Level: Public

Contractual Date of Delivery: 30.10.2014

Actual Date of Delivery: 08.05.2015

Contributing to the Deliverable: WP2

WP Task Responsible: USAAR

EC Project Officer: Concepcion Perez-Camaras

Abstract:
The last decade has seen phenomenal growth in research and development activities in Machine Translation (MT), particularly so for statistical MT (SMT). Researchers have proposed many different approaches to component technologies in SMT, starting from pre-processing techniques to core SMT processes to post-editing, all of which bring some improvements over baseline SMT. However, despite these immense developments in MT research, from a realistic viewpoint, all these different techniques are not compatible with each other. In a sense, MT workflow is less investigated than the MT processes. Many research have not gone into how to make the best of these individual techniques in an ideal framework, given some resources and constraints for any language pair.
In this study we investigate towards an ideal MT workflow. We have studied how different paradigms, like Translation Memory, Example-Based MT and SMT can be harmonized to the best effect. We also carried out study into how different component technologies, like word alignment, phrase alignment, can be hybridized by making use of the state-of-the-art techniques to achieve improved MT.
This study also includes an investigation into optimal human-machine interactive MT by taking humans in the loop. We study both how post-editor feedback can be directly integrated into the system, as well as, how automatic post-editing tools can be developed by making use of the post-edited data.

Table of Contents

1 Introduction

Translation tools are progressively changing due to technological advancement. This is also impacting on translators' work practice. By definition, translation tools are computer software that facilitates translators' work in terms of ease of use, faster project delivery, saving translators' time and cost due to (partial) automation. Automatic translations or translation suggestions produced by these tools may not always be correct. Because of this and many other reasons (including the fear of job loss due to progressive automation) translation tools are not always widely accepted in traditional translation workflows. This is a well-known problem for the translation industry. Over the last 10-15 years, Machine Translation (MT) has made considerable progress, in large part due to research and development of statistical approaches to MT (SMT). In many cases, MT services provide a convenient support not only for professional translators but also for common users. Various free online MT engines or commercial engines are available, such as Google Translator, Systran, Microsoft Bing translator, etc. Case studies have shown that the deployment of a MT engine can be beneficial for all sides, including clients, translators and language service providers (LSP) in terms of cost and time saving. However, the oval picture remains mixed: on the one hand, often MT is cheap and easy to use, while on the other hand, in many cases, the quality of translation is not always satisfactory: sometimes professional translators prefer to translate from scratch.

The main aim of the research in the WP is to study real-life needs and problems confronted by translation technology users, including both professional translators and readers of translations. Based on this the translation system functionality should be optimized in terms of user requirements rather than forcing users to change how they work with the technology. The research also investigates whether and how existing technology such as Statistical Machine Translation (SMT), Example Based Machine Translation (EBMT) and Translation Memory (TM) systems can fulfill the user's requirements. The user involvement provides a systematic evaluation and error analysis of the new technologies developed within the EXPERT project.

In this deliverable, we describe the initial work on exploring the use of translation memories and related translation automation and support tools including various forms of MT in real life environments to find evidence about needs and problems encountered in real-life conditions by users of translation technologies, including both professional translators and readers of translations. We will also provide a thorough study of existing SMT, EBMT and TM systems from a user perspective to establish the requirements of different types of users and to what extent their requirements are supported by existing technologies.

In order to achieve this objective, we describe our approach in two complementary ways. Technology components are usually combined into translation workflows: part of our research will therefore concentrate on the (i) components, while the other part of our research will focus on the (ii) workflows. Specifically, in the first part of our research we focus on the design and implementation of component technology based on user requirements in order to provide MT component technologies that are effectively and user friendly. In the second part of our research we concentrate on identifying optimal workflows, based on a combination of different components of the component technology into the workflow using a plug and play methodology.

The workflow research will also take into account some methods that are available in cognitive science (such as eye tracking). In this deliverable, we focus on part one of our research program, the component technologies those are the basis of the workflows. Individual components of the component technology are easy to connect and chain up into MT workflows by using plug and play methods.
We designed, developed, tested and evaluated core component MT workflow technologies, guided by the following research questions:

- RQ1: How can existing resources and data be optimally used?
- RQ2: What would the ideal hybrid implementation of MT be?

- RQ3: How can human interaction be implemented in existing MT workflows?
- RQ4: How could human involvements be optimized?

2 Component Technologies

Our component technology has been designed with three major components: a data preprocessor, hybrid machine translation and post editing. The MT components of the component technology are based on corpus-based machine translation (CBMT). The data-preprocessing module mainly relies on data acquisition and effective preprocessing of the existing translation training data. Many natural language processing tasks, such as CBMT heavily rely on bilingual parallel corpora. Recently, CBMT has delivered increasingly better quality translations. SMT is a kind of CBMT based on probabilistic translation models. SMT heavily relies on good quality word alignment and phrase alignment tables representing the system's translation knowledge acquired from a bilingual corpus.

There are many CBMT approaches that have been proposed in the last few decades such as Translation Memory (TM) (Kay, 1980), Example-based Machine Translation (EBMT) (Carl and Way, 2006) and SMT (Brown et al., 1993; Vogel et al., 1996; Marcu, 2001, Koehn et al., 2003; Koehn, 2010).

Out of these, in terms of large-scale evaluations, SMT is the most successful and efficient MT paradigm. The quality of SMT mainly relies on good quality word alignment as well as optimal phrase pair estimation, both of which can be achieved by using large amounts of sentence- and word-aligned parallel corpora. However, SMT for low-resource language pairs often produces inferior quality translation. A major problem of SMT is scarcity of available parallel training data. SMT for many language pairs, such as English to Indian languages, suffers from the scarcity of parallel data.
Comparable corpora provide a possible solution to this data scarceness problem to some extent. Comparable documents are not strictly parallel: the corpus consists of bilingual documents but they are not sentence-aligned, as sentences of comparable corpora are not really translations, but the comparable documents convey similar information on the same topic and hence some sentential or sub-sentential level of parallelism (i.e. actual translations) between the documents in comparable corpora.
Recently, comparable corpora have received considerable attention as a valuable resource for mining and acquiring parallel data, which can play an important role in improving the quality of machine translation (MT) (Smith et al. 2010). The extracted parallel texts from comparable corpora are typically added to the training corpus as additional training material that is expected to improve the performance of SMT systems, specifically for low-resource language pairs.

Data preprocessing also plays a crucial role in CBMT. In our research we effectively preprocessed MWE, namely NEs, and paraphrases and supplied them as additional information sources to a state-of-the-art PB-SMT system. We analyzed how single-tokenization of two types of MWEs, namely NEs and compound verbs, as well as their prior alignment can boost the performance of PB-SMT. Additionally, reordering poses a big challenge in SMT between distant language pairs. We also present how reordering between distant language pairs can be handled efficiently in PB-SMT.

MWE are defined as "idiosyncratic interpretations that cross word boundaries (or spaces)" (Sag et al., 2002). Traditional approaches to word alignment following IBM Models (Brown et al., 1993) do not work well with MWEs, especially with NEs, due to their inability to handle many-to-many alignments. Firstly, the IBM Models only carry out alignment between words and do not consider the case of complex

expressions, such as multi-word NEs. Secondly, the IBM Models only allow at most one word in the source language to correspond to a word in the target language (Marcu, 2001, Koehn et al., 2003).
In another well-known word alignment approach in SMT, Hidden Markov Model (HMM) alignment (Vogel et al., 1996), the alignment probabilities depend on the alignment position of the previous word. This approach does not explicitly consider many-to-many alignment either. In our research we address this many-to-many alignment problem indirectly. Our objective is to see how to best handle the MWEs and NEs in SMT.

Conventionally, TM systems store source and target language translation pairs for reusing previous translations originally created by human translators. Conceptually speaking, EBMT is closely related to TM technology. The difference between the two approaches is that EBMT extracts translations of fragments from the translation model and combines them to produce a new translation.

Each approach has its own method of acquiring and using translation knowledge from the bilingual translation examples, along with its own advantages and disadvantages. The knowledge representation process, in both EBMT and SMT, uses very different techniques in order to extract translation resources. The SMT phrases essentially operate on n-grams, rather than grammatical phrases as in EBMT. Many researchers have investigated the combination of these different MT approaches in Hybrid MT approaches to achieve better performance. Our Hybrid MT system described in this deliverable is one such approach. We also propose the improvement of word alignment quality by combining three word alignment tables (i) GIZA++ alignment (ii) Berkeley Alignment and (iii) Rule based alignment. Our objective is to realize the effectiveness of the Hybrid model in word alignment in order to enhance the quality of translation in the SMT system. In the present work, we have implemented a rule based alignment model by considering several types of chunks that are automatically extracted on the source side. Each individual source chunk is translated into the target chunk and then validated by the target chunks on the target side. The validated source-target chunk pairs are added to the rule based alignment table. The work has been carried out in three ways: (i) three alignment tables are combined together by the set union technique and (ii) extra alignment pairs are added into the alignment table. This is a well-known practice in domain adaptation in SMT (Eck et al., 2004; Wu et al., 2008). (iii) Update the alignment table through a semi-supervised alignment technique. We preprocess the parallel training corpus by single tokenizing multiword NEs . The preprocessing of the parallel corpus results in improved MT quality in terms of automatic MT evaluation metrics.

2.1 Preprocessing data

2.1.1 Data acquisition from external resources

Good performance of CBMT and especially SMT is usually achieved with huge parallel bilingual training corpora, because the translations of words or phrases are computed based on the bilingual data. However, in case of low-resource language pairs such as English-Bengali, the performance is affected by insufficient amounts of bilingual training data. Recently, comparable corpora became widely considered as valuable resources for machine translation. Though very few cases of sentential level parallelism are found between two comparable documents, there are still potential parallel phrases in comparable corpora. Mining parallel data from comparable corpora is a promising approach to collect more parallel training data for SMT. We propose an automatic alignment of English-Bengali comparable sentences from comparable documents. We exploit the multilingualism of Wikipedia. The most important fact is that this approach does not need any domain specific corpus. We have been able to improve the BLEU score of an existing domain specific English-Bengali machine translation system by 11.14% (relative over baseline system). We further improve the approach using a novel textual entailment method and

distributional semantics for text similarity. Subsequently, we apply a template-based phrase extraction technique to aligned parallel phrases from comparable sentence pairs. The effectiveness of our approach is demonstrated by using parallel phrases extracted in this fashion from our comparable training data as additional training examples for an English-Bengali PB-SMT system. Our system achieves significant improvement in terms of translation quality (2.92 points BLEU, 26.64% relative) over the baseline system (the research reported in Pal et al., 2014a; 2015).

2.1.2 Effective use of Data

Data pre-processing plays a crucial role in PB-SMT. In our research we show how single-tokenization of two types of MWEs, namely NEs and compound verbs, as well as their prior alignment can boost the performance of PB-SMT. Single-tokenization of compound verbs and NEs provides significant gains over the baseline PB-SMT system. Automatic alignment of NEs substantially improves the overall MT performance, and thereby the word alignment quality indirectly. For establishing NE alignments, we transliterate source NEs into the target language and then compare them with the target NEs. Target language NEs are first converted into a canonical form before the comparison takes place. MWEs contribute to major lexical ambiguity problems for any language and pose a big challenge in SMT. We present the role of MWE in improving the performance of a PB-SMT system. In one of our approaches automatically aligned MWEs have been incorporated indirectly, i.e., added as additional parallel examples with the parallel corpus. In another of our approaches we have directly incorporated MWEs into the word alignment model. Both approaches boost the performance of the PB-SMT system. For MWE alignment, we have used a bidirectional pre-trained PB-SMT system which was trained on the same parallel corpus. For alignment validation we have used a string level edit distance mechanism. At the end, we bootstrap the whole procedure with a single iteration to obtain more MWE alignments. Our system achieves significant improvements with 7.0 BLEU points absolute, and 64.1% relative improvement, over the English-Bengali baseline system and bootstrapping with a single iteration results in 9.24 BLEU points absolute, and 84.7% relative improvement over the baseline system on an English-Bengali translation task. We also applied a similar approach except the prior one-to-one MWE and NE alignment for our WMT-2014 Manawi system. We participated in the English-Hindi (EN-HI) and Hindi-English (HI-EN) language pair and achieved 79.20 for the Translation Error Rate (TER) score[1] for EN-HI, the lowest among the competing systems. Our main innovations are (i) the usage of outputs from NLP tools, viz. a bilingual MWEs extractor and NE recognizer to improve SMT quality and (ii) the introduction of a novel filter method based on sentence-alignment features. The Manawi system showed the potential of improving translation quality by incorporating multiple NLP tools within the MT pipeline.

Handling out of vocabulary (OOV) words is also a big issue in CBMT approaches. SMT suffers from out of vocabulary (OOV) words and less frequent words especially when only limited training data are available or training and test data are in different domains. In our work, we propose a convenient way to handle OOV and rare words using a paraphrasing technique. Initially we extract paraphrases from a bilingual training corpus with the help of comparable corpora. The extracted paraphrases are analyzed by conditionally checking the association of their monolingual distribution. Bilingual aligned paraphrases are incorporated as additional training data into the PB-SMT system. Integration of paraphrases into PB-SMT system results in significant improvements (the research reported in Pal et al., 2014b).

Likewise, other lexical pre-processing, word order, otherwise known as reordering, pose major challenges in MT. The problem of reordering between distant languages has been approached with prior reordering of the source text at chunk level to simulate the target language ordering. Prior reordering of

[1] Lower TER often results in better translation

the source chunks is performed in our work by following the target word order suggested by word alignment (the research reported in Pal et al., 2014c). The test set is then reordered using monolingual MT trained on source and reordered source. This approach of prior reordering of the source chunks was compared with pre-ordering of source words based on word alignments and the traditional approach of prior source reordering based on language-pair specific reordering rules. The effects of these reordering approaches were studied on an English–Bengali translation task, a language pair with different word order. From the experimental results we found that word alignment based reordering of the source chunks is more effective than the other reordering approaches, and it produces statistically significant improvements over the baseline system on BLEU. On manual inspection we found significant improvements in terms of word alignments.

2.2 Ideal Hybridization in Machine Translation

2.2.1 Hybrid Word Alignment

The proposed hybrid word alignment model (Pal et al., 2013a) provides most informative alignment links, which are offered by both unsupervised and semi-supervised word alignment models. Two unsupervised word alignment models, namely GIZA++ and the Berkeley aligner, and a rule based word alignment technique **(**Pal et al., 2010; 2011; 2012; 2013c**)** are combined together. The unsupervised alignment models are trained on the surface form as well as the root form of the training data and provide alignment tables for the corresponding training data. The rule-based aligner is aimed towards aligning NEs and chunks. NEs are aligned through transliteration using a joint source-channel model. Chunks are aligned employing a bootstrapping approach by translating the source chunks into the target language using a baseline PB-SMT model and subsequently validating the chunk hypotheses using a fuzzy matching technique against the target corpus. Experiments are carried out after single-tokenizing the multi-word NEs. Our best system provides significant improvements over the baseline as measured by BLEU (10.25 BLEU points absolute, 93.86% relative over baseline system).
The hybrid word alignment model is described as the combination of three word alignment models as follows:

Word Alignment Using GIZA++

GIZA++ (Och and Ney, 2003) is a statistical word alignment tool, which incorporates all the IBM 1-5 models. GIZA++ facilitates fast development of SMT systems. In case of low-resource language pairs the quality of word alignments is typically quite low and also deviates from the independence assumptions made by the generative models. Although huge amount of parallel training data enables better estimation of the model parameters, a large number of language pairs still suffer from the unavailability of sizeable amounts of parallel data. GIZA++ has some drawbacks. It allows at most one source word to be aligned with each foreign word. To resolve this issue, a number of techniques have already been applied, including the following one. The parallel corpus is aligned bi-directionally; then the two alignment tables are reconciled using different heuristics e.g., intersection, union, and most recently grow-diagonal-final. The grow-diagonal-final-and heuristics has been applied for bidirectional alignment in our research. In spite of these heuristics, the word alignment quality for low-resource language pairs is still low and calls for further improvement. We describe our approach of improving word alignment quality in the following three subsections.

Word Alignment Using Berkley Aligner

Recent advances in word alignment are implemented in the Berkeley Aligner (Liang et al., 2006), which allows both unsupervised and supervised approaches to align words from parallel corpora. We initially train on the parallel corpus using unsupervised technique. We then make a few manual corrections to the alignment table produced by the unsupervised aligner. Then we apply this corrected alignment table as gold standard training data for the supervised aligner. The Berkeley aligner is an extension of the Cross Expectation Maximization word aligner. The Berkeley aligner is a very useful word aligner because it allows for supervised training, enabling us to derive knowledge from an already aligned parallel corpus or we can use the same corpus by updating the alignments using some rule based methods. Our approach deals with the latter case. The supervised technique of the Berkeley aligner helps us to align those words, which could not be aligned by our rule based word aligner.

Rule Based Word Alignment

Our rule based aligner aligns NEs and chunks. Figure 1 shows the architecture of our rule-based system. For NE alignment, we first identify NEs from the source side (i.e. English) using Stanford NER[2]. The NEs on the target side (i.e. Bengali) are identified using a method described in (Ekbal and Bandyopadhyay, 2009). The accuracy of the Bengali NER is much poorer than that of English NER due to several reasons: (i) there is no capitalization cue for NEs in Bengali; (ii) most of the common nouns in Bengali are frequently used as proper nouns; (iii) suffixes (case markers, plural markers, emphasizers, specifiers) get attached to proper names as well in Bengali. A Bengali shallow parser[3] has been used to improve the performance of NE identification by considering proper names as NE. In our architecture, the NER and shallow parser are jointly employed to detect NEs in Bengali sentences. The source NEs are then transliterated using a modified joint source-channel model (Ekbal et al., 2006) and aligned to their target side equivalents following the approach of Pal et al. (2010). Since Bengali NEs differ in their choice of matras (vowel modifiers), both the NEs found in the Bengali sentence as well the transliterated (i.e., Bengali) NEs are transformed into canonical form after omitting their 'matras'. The transliterated NEs are then matched with the corresponding parallel target NEs and finally we align the NEs if a match is found. After identification of multiword NEs on both sides, we pre-processed the corpus by replacing space in NEs with the underscore character ('_'): this ensures that the multiword NEs are single tokenized and considered as a single unit. We have used underscore ('_') instead of hyphen ('-') since there already exist some hyphenated words in the corpus. The use of the underscore ('_') character also facilitates de-tokenization of single-tokenized NEs after decoding.

Automatic Alignments of NEs through Transliteration

We extract the source and target (single token) NEs from the NE-tagged parallel translations in which both sides contain at least one NE. Then we first create an NE parallel corpus. In the example mentioned below, we extract the NE translation pairs given in (1) from the sentence pair shown in (1), where the NEs are shown in italics.

> (1a) *Kirti_Mandir* , where *Mahatma_Gandhi* was born , today houses a photo exhibition on the life and times of the *Mahatma* , a library, a prayer hall and other memorabilia .
> (1b) কির্তী_মন্দির , যেখানে মহাত্মা_গান্ধী জন্মেছিলেন , বর্তমানে সেখানে মহাত্মার জীবন ও সেই সময়ের ঘটনাসমূহের একটি চিত্রপ্রদর্শনশালা , একটি লাইব্রেরী ও একটি প্রার্থনা ঘর এবং অন্যান্য স্মৃতিবিজড়িত জিনিসপত্র আছে।

[2] http://nlp.stanford.edu/software/CRF-NER.shtml#Download
[3] http://ltrc.iiit.ac.in/showfile.php?filename=downloads/shallow_parser.php

(2a) Kirti_Mandir Mahatma_Gandhi Mahatma
(2b) কির্তী_মন্দির মহাত্মা_গান্ধী মহাত্মার

Next, we try to align the extracted source and target NEs, as illustrated in example (2). If both sides contain only one NE then the alignment is trivial, and we add such NE pairs to seed another parallel NE corpus that contains examples having only one token in both sides. Otherwise, we establish alignments between the source and target NEs using transliteration. We use the joint source-channel model of transliteration (Ekbal et al., 2006) for this purpose.

If both the source and target side contains *n* number of NEs, and the alignments of *n*-1 NEs can be established through transliteration or by means of already existing alignments, then the n^{th} alignment is trivial. Similarly, for multi-word NEs, intra-NE word alignments are established through transliteration or by means of already existing alignments. For a multi-word source NE, if we can align all the words inside the NE with words inside a target NE, then we assume they are translations of each other.
Since the source side NER is much more reliable than the target side NER, we transliterate the English NEs, and try to align them with the Bengali NEs. We take the 5 best transliterations produced by the transliteration system for an English word, and compare them against the Bengali words. Here, we first normalize both Bengali words, target NEs and the transliterated source ones, because Bengali NEs often differ in their choice of *matras* (vowel modifiers). We transform Bengali NEs into a canonical form by dropping the *matras*, and then compare the results; if they match, then we align the English NE word with the Bengali NE word.

(3) নিরজ (ন + ি+ র + জ) -- নীরাজ (ন + ী + র + া + জ)

The example in (3) illustrates the procedure. Assume we are trying to align “Niraj” with “নীরাজ”. The transliteration system produces “নিরজ” from the English word “Niraj” and we compare “নিরজ” with “নীরাজ”. Since the consonant sequences match in both words, “নিরজ” is considered a spelling variation of “নীরাজ”, and the English word “Niraj” is aligned to the Bengali word “নীরাজ”.
In this way, we achieve word-level alignments, as well as NE-level alignments. Example (4) shows the alignments established from example (1). The word-level alignments help to establish new word / NE alignments. Word and NE alignments obtained in this way are added to the parallel corpus as additional training data.

(4a) Kirti-Mandir — কির্তী-মন্দির
(4b) Kirti — কির্তী
(4c) Mandir — মন্দির
(4d) Mahatma-Gandhi — মহাত্মা-গান্ধী
(4e) Mahatma — মহাত্মা
(4f) Gandhi — গান্ধী
(4g) Mahatma — মহাত্মার

Automatic Chunk alignment

For chunk alignment, the source sentences of the parallel corpus are parsed using the Stanford POS tagger. The chunks of the sentences are extracted using the CRF chunker[4]. The chunker detects the boundaries of noun, verb, adjective, adverb and prepositional chunks in the sentences. In case of prepositional phrase chunks, we have taken a special approach: we have expanded the prepositional

4 http://crfchunker.sourceforge.net/

phrase chunk by examining a single noun chunk followed by a preposition or a series of noun chunks separated by conjunctions such as 'comma', 'and' etc. For each individual chunk, the head word is identified. Similarly, target side sentences are parsed using a shallow parser. The individual target side Bengali chunks are extracted from the parsed sentences. The head words for all individual chunks on the target side are also marked. If the translated head word of a source chunk matches with the headword of a target chunk then we hypothesize that these two chunks are translations of each other.
The extracted source chunks are translated using a baseline SMT model trained on the same corpus. The translated chunks are validated against the target chunks found in the corresponding target sentence. During the validation process, if any match is found between the translated chunk and a target chunk then the source chunk is directly aligned with the original target chunk. Otherwise, the source chunk is ignored in the current iteration for any possible alignment and is considered in the next iterations.

The extracted chunks on the source side may not have a one to one correspondence with the target side chunks. The alignment validation process is focused on the proper identification of the head words and not between the translated source chunk and target chunk. The matching process has been carried out using a fuzzy matching technique. If both sides contain only one chunk after aligning the remaining chunks then the alignment is trivial. After aligning the individual chunks, we also establish word alignments between the matching words in those aligned chunks. Thus we get a sentence level source-target word alignment table.

Figure 2 shows how word alignments are established between a source-target sentence pair using the rule based method. Figure 2.a shows the alignments obtained through the rule based method. The solid links are established through transliteration (for NEs) and translation. The dotted arrows are also probable candidates for intra-chunk word alignments; however they are not considered in the present work. Figure 2.b shows the gold standard alignments for this sentence pair.

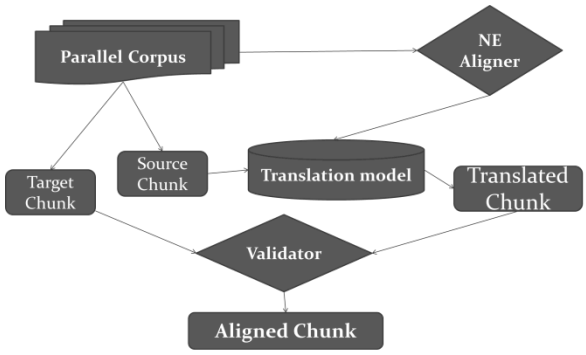

Figure 1: System architecture of rule based aligner.

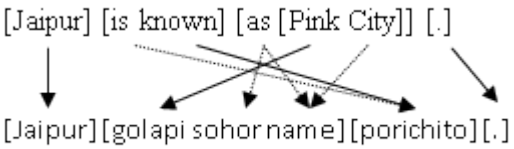

Figure 2.a: Rule based alignments

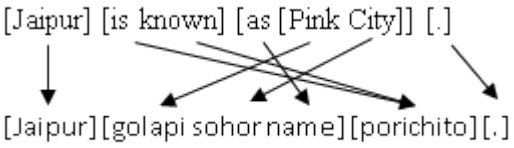

Figure 2.b: Gold standard alignments

Hybrid Word alignment Model

Our hybrid word alignment method combines word alignments produced by three different kinds of word aligners: Giza++ with grow-diag-final-and (GDFA) heuristic, the Berkeley aligner and our rule based aligner. We have followed two different strategies to combine the three different word alignment tables.

Union

In the union method all the alignment tables are put together and duplicate entries are removed. Taking union of the alignments should improve the recall of the word alignment.

ADD additional Alignments

In this method, we consider either of the alignments generated by GIZA++ (A1) or Berkeley aligner (A2) as the standard alignment as the rule-based aligner (A3) fails to align many words in the parallel sentences. For any set of alignments $\{A_1, A_2, \ldots A_n\}$, we propose an alignment combination method as described in algorithm 1.

ALGORITHM 1

> **Step 1:** Choose a standard alignment table (A_s) from the set of alignment tables $\{A_1, A_2 \ldots A_n\}$ with the exception that any rule based alignment cannot be assigned to A_s.
> **Step 2:** Correct the alignments in A_s using the remaining (n-1) alignment tables. Take intersection of the other n-1 alignment tables. E.g., for three alignment tables A_1, A_2 and A_3, if A_2 is assigned to A_s then find additional alignments from A_1 and A_3 using $A_1 \cap A_3$ and add these additional entries to A_s.

Berkeley Semi-supervised Alignment

The correctness of the alignments is verified by manually checking the performance of the various alignment systems. We start with the combined alignment table, which is produced by Algorithm 1. Initially, we take a subset of the alignments, a set of 500 alignments from the combined alignment table, which was manually inspected and corrected. Then we train the Berkeley supervised aligner with this labeled data. A subset of the unlabeled data, i.e., alignments collected from the combined alignment table, is aligned with this supervised model. The output is then added as additional labeled training data for the supervised training method for the next iteration. Using this bootstrapping approach, the amount of labeled training data for the supervised aligner is gradually increased. The process is continued until there are no more unlabeled training data left. In this way we refine the whole alignment table for the entire parallel corpus. The process is carried out in a semi-supervised manner.

The manual correction process involves correction of one-to-one, one-to-many, many-to-one and many-to-many alignments. To optimize the manual effort involved we focus only on one-to-one alignment correction, other types of correction are automatically taken care of by the system during the iterative process. We manually inspected 500 alignments and observed that the quality of the one-to-one alignments is better than the other kinds of alignments. Table 1 shows statistics over the 500 manually inspected alignments.

Alignment	Accuracy
1:1	83.2%
1:2	67.4%
1:3	49.1%

Table 1: Word Alignment Accuracy

Since the one-to-one alignment list has better accuracy, the one-to-one alignments are considered initially for correction in the 1st Iteration. In the 1st iteration of the statistical model, the manually checked 500 alignments are used with the large set of alignment. At the end of Iteration 1, it was found that the accuracy of both the one-to-one and one-to-many mapped word alignments increase as more and more words are now correctly aligned. After an in depth study of the one-to-one aligned pairs for a few words, it was found that the number of incorrectly aligned entries before the 1st iteration were more than the correctly aligned entries. A detailed analysis of the word alignment quality after 1st iteration exposed that not only the accuracy of one-to-one world alignments is improved by this process, but also that the accuracy of other kinds of word alignments also improves. The example given below depicts the improvement the in word alignment.

English sentence: This variety is replicated in the food, architecture, music and culture of Brazil.
Bengali Sentence (English gloss): Brajilera khadya, parikaṭhamo , sangita , sanṣkr̥tite ei baicitra pratiphalita haya .

Word	Word alignment position in iteration 1	Word alignment position in iteration 2
NULL	7	7
This	9	9
variety	10	10
Is	NA	12
replicated	8 11 12	11
In	NA	NA
The	NA	NA
Food	2	2
,	NA	NA
architecture	4	4
,	5	5
music	NA	NA
And	NA	NA
culture	NA	8
Of	NA	NA
Brazil	1	1
.	13	13

Table 2: Word alignment improvement with iterations

The example in Table 2 shows that, before the first iteration the word “replicated” is aligned to 3 Bengali words in the target side while the word “culture” remains unaligned. After the first iteration, the word “culture” is correctly mapped to the target word “sangaskritite”, as these one to one mapped words are

manually corrected in the training alignment set, the system identifies the correct alignment pairs during the successive iterations. In iteration 2, the system correctly aligns “culture” with “sangaskritite”, “is” with “hay” and “replicated” with “pratiPalita”.

For the successive iterations the correction of one-to-one mapped word alignments are preferred again. With each increment in the iteration process, the correction effort is gradually less and the accuracy of the one-to-many as well as other types of word alignment increases.
The hybrid word alignment model has been incorporated into the SMT workflow as shown in Figure 3.

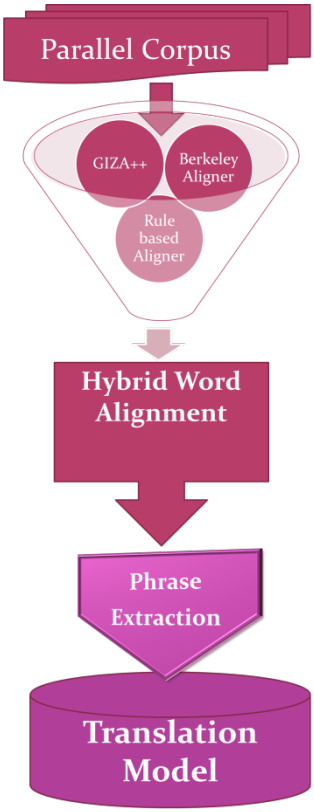

Figure 3: Translation model using Hybrid word alignment.

2.2.2 Hybrid Phrase Alignment

Traditional PB-SMT systems derive phrase pairs directly from the training corpus according to purely statistical method. Thus PB-SMT phrase pairs may not follow traditional syntactic constituents of a sentence; they are just n-grams. In our approach we experiment with restricting the phrase extraction module to extract phrase pairs that begin and end at chunk boundaries and to use these to augment standard SMT phrase tables. Our approach ensures that the phrase pairs thus extracted thus are not just n-grams; they are linguistically motivated and incorporate syntactic knowledge to some extent. This section describes the English-Bengali machine translation task towards examining the effects of linguistically motivated shallow phrases (chunks) incorporated in state-of-the-art PB-SMT Systems. Linguistically guided phrase pairs are extracted from the training corpus. Afterwards, these phrase pairs are added to the standard n-gram based translation model of a PB-SMT system and the probabilities are normalized. We observed that inclusion of these linguistically motivated phrase pairs into the English–Bengali PB-SMT translation model leads to significant improvements in translation quality as measured

by automatic MT evaluation metrics. Our system achieved 3.18 BLEU point (29.7%) relative improvements over baseline system.

System Description

Phrase Extraction

The English sentences are POS-tagged and chunked using Stanford POS tagger and English CRF chunker[5], respectively. From the chunked sentences, we have extracted phrases. Chunks are identified on both sides of the training corpus. We modified the Moses phrase extraction scripts so that it extracts phrase pairs, which begin and end at chunk boundaries. The phrase pairs extracted thus are made up of chunks.

Source Side Extraction

The source sentences are chunked using the English CRF based chunker by considering their POS tags. After chunking the whole sentences, we have modified the prepositional chunks by extending their boundaries using some constraints such as prepositional chunk is followed by a noun chunk or a series of noun chunks separated by conjunction. In the example below the "PP(in 1855)" chunk contains the constituent chunk such as PP(in) and NP(1855) which are combined into a single PP chunk. The phrase extraction procedure is followed by the instruction from the user who has to give some minimum and maximum phrase length. In the example below the maximum phrase length is 5. The phrase extraction procedure is Overlap type i.e. a sliding window type where - the window slides over the sentence and extracts phrase pairs, which begin and end at chunk boundaries following algorithm 3. We have also extracted strict n-gram (n=7) linguistic phrases, which is a non-sliding type of extraction. The strict type of extraction has been reported in the algorithm 2. To avoid out of vocabulary items we have included all the individual chunks after running the algorithm 2 (non-overlap phrase extraction) and algorithm 3 (overlap phrase extraction).

ALGORITHM 2: Non-Overlap n-gram Phrase Extraction
maxPhL← Max-Phrase-Length;
for *i* = 1 to *m* chunks
  out_phrase ← $chunk_i$;
  length ← number words in $chunk_i$;
  j = i+1
  C ← $chunk_j$;
  P ← set of words in C;
  L← number words in P;
  if (length + L)<=maxPhL
    Concatenate C with out_phrase;
    length← length +L;
  else
    add out_phrase into List;
    out_phrase←null
  end if
end for

The algorithm (Overlap) 3 is slightly different from algorithm 2 (Non-Overlap); it takes chunk annotated sentence and maximum phrase length as input. The *outer for loop* delivers current $chunk_i$ to the *inner for loop*. The *inner for loop* reads next $chunk_{i+1}$ and then calculates the length of the $chunk_i$ and $chunk_{i+1}$ (i.e. $chunk_j$) and store in out_phrase, if the calculated length equals with maximum phrase length. If the

5 http://crfchunker.sourceforge.net/

calculated length is less than maximum phrase length then $chunk_{j+1}$ is concatenated with out_phrase until maximum phrase length equals. Otherwise proceed to the next $chunk_{i+2}$ and proceed to the next iteration.

Source phrase extraction Overlap type:

The republic of Colombia
Of Colombia was formally establish
Was formally establish in 1855
In 1855 .

ALGORITHM 3: Overlap n-gram Phrase Extraction

```
maxPhL← Max-Phrase-Length;
for i = 1 to m chunks
    out_phrase ← chunk_i;
    length ← number words in chunk_i;
    for j = i+1 to m chunks
          C ← chunk_j;
          P ← set of words in C;
          l← number words in P;
          if (length + l)<=maxPhL
                Concatenate C with  out_phrase;
                length← length +l;
          end if
    end for
    add  out_phrase into List;
    out_phrase←null
end for
```

Source-target Phrase Extraction files creation

Using the alignment file provided by GIZA++, we created an alignment file using grow-diag-final-and algorithm and created two files as directed in the Moses (Koehn, 2009) toolkit - extract.direct and extract.inv. The example below shows the steps of the phrase alignment file creation procedure by using the knowledge of word alignment produced by GIZA++. Using these two extracted files we have created a phrase table following a method similar to the one described in Koehn, 2003. The extracted phrase alignment file is pruned and those phrases are discarded that contain extra words on either source or target phrase that are not relevant to the phrase alignment.
In the below examples the phrase alignment no 6 is marked as erroneous because "*the republic*" has already been aligned with "প্রজাতন্ত্র (prajatantra)" but in the left hand side of phrase 6 does not contains "*the republic*".

Target sentence chunking:

NP(কলম্বিয়ার প্রজাতন্ত্র) NP(1855 খ্রীষ্টাব্দে) RBP(আনুষ্ঠানিকভাবে) JJP(প্রতিষ্ঠিত) VP(হয়েছিল) (.)

Source target word Alignment: 0-0 3-0 1-1 2-1 5-2 8-3 7-4 8-4 6-5 6-6 9-7

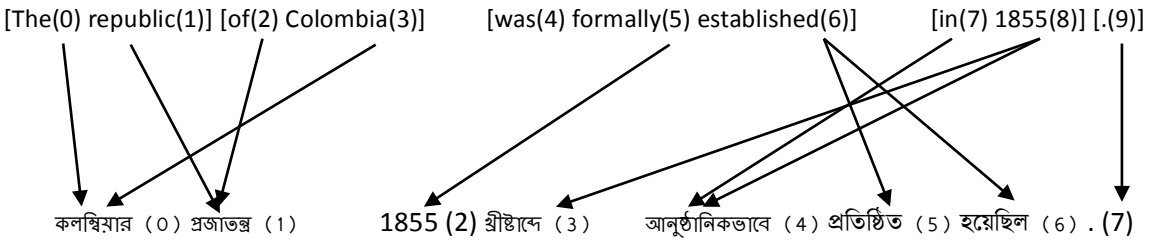

Figure 4: Word alignment provided by GIZA++

Phrase Alignment file

The republic ||| প্রজাতন্ত্র
Of Columbia ||| কলম্বিয়ার
3. Was formally established ||| প্রতিষ্ঠিত হয়েছিল
4. In 1855 ||| 1855 খ্রীষ্টাব্দে আনুষ্ঠানিকভাবে
5. The republic of Colombia ||| কলম্বিয়ার প্রজাতন্ত্র
6. Of Colombia was formally established ||| কলম্বিয়ার প্রজাতন্ত্র 1855 খ্রীষ্টাব্দে আনুষ্ঠানিকভাবে প্রতিষ্ঠিত হয়েছিল
7. Was formally establish in 1855 ||| 1855 খ্রীষ্টাব্দে আনুষ্ঠানিকভাবে প্রতিষ্ঠিত হয়েছিল
8. In 1855 . ||| 1855 খ্রীষ্টাব্দে আনুষ্ঠানিকভাবে .

Phrase table

We constrained the PBSMT phrase length to a maximum of 7 and a minimum of 4. After extracting all phrase pairs we calculated lexical weighting and phrase translation probabilities in both source and target direction.

The phrase translation probability corresponding to the phrase pair (*f, e*) is given by equation [1].

$$\emptyset(f|e) = \frac{count(f,e)}{\sum_{f_i}(f,e)} \quad [1]$$

Lexical weighting (*Lw*) is given by equation [2].

$$Lw(f|e,a) = \prod_{i=1}^{length(f)} \frac{1}{\{j|(i,j)\in a\}} \sum_{\forall(i,j)\in a} w(f_i|e_j) \quad [2]$$

where $w(f_i|e_j)$ signifies word translation probability.

To speed up decoding, we integrated phrases associated with their phrase translation probability and lexical weighting into the phrase table.

Decoder

We have used state-of-the-art Moses decoder to decode the test sentences. The state-of-the-art Moses decoder is initialized with an empty hypothesis; a new hypothesis is expanded by a sequence of un-translated foreign words and a possible target phrase translation is selected. As our translation system is biased towards linguistically motivated phrases, we decoded individual phrases and finally recombined all the fragment translations by updating hypothesis based on the same process followed by the Moses decoder.

2.2.3 Hybridization with Translation Memory, Example Based Machine Translation and Statistical Machine Translation

This part of the deliverable describes joint work between Dublin City University (DCU) and Saarland University (USAAR) as part of an EXPERT training visit. We describe the USAAR-DCU machine translation system submitted to the NLP Tools Contest at the International Conference on Natural Language Processing (ICON 2014). The shared task on SMT in Indian languages encompassed translating from five languages into Hindi in three different domains. Our best system achieved average BLEU scores (over three domains) of 29.25, 16.41, 35.72, 16.16 and 25.53 for Bengali, English, Marathi, Tamil and Telugu to Hindi translation respectively. The system combination output is the best performing system and outperforms from an individual component system. The main innovations are: (i) effective pre-processing and use of explicitly aligned bilingual terminology i.e. NEs , (ii) simple but effective hybridisation technique for using multiple knowledge sources. Our hybrid system (Pal et al., 2014d) has the potential to improve over the baseline SMT performance by incorporating additional knowledge sources such as the extracted bilingual NEs , translation memories, and phrase pairs induced from example-based methods. We report performance on three hybrid systems as well as results of a confusion network-based system combination that combines the best performance of each individual system within the multi-engine pipeline.

System description

Our Hybrid MT system consists of three basic steps as follows:

- Cleaning and clustering of sentences based on sentence length
- Effective preprocessing of data such as explicit alignment of bilingual terminology (e.g. NEs)
- Hybrid MT implementation with the preprocessed data

In order to combine (hybridize) multiple knowledge sources (SMT, EBMT, TM, and NE), the alignment of NEs is appended to the training set prior to the SMT training with Moses[6] (Koehn et al., 2007) and the entire training set as well as template-based EBMT phrases are also saved into that translation memory for future use at decoding time. We designed four (baseline and three hybrid) systems for each language pair and finally a system combination module to produce the optimal output. The four systems are as follows:

- **Baseline:** SMT
- **System 1:** Baseline SMT with Named Entity Alignment (NEA)
- **System 2:** NEA with EBMT (NEA-EBMT)
- **System 3:** System 2 with TM (TM-EBMT- SMT).

Preprocessing of the parallel corpus

The initial parallel corpus is cleaned and filtered using a semi-automatic process. We filter the parallel training data on maximum allowable sentence length of 100 and sentence length ratio of 1:2 (either direction). Our first objective was to use effective preprocessing of data and terminology identification

[6] Moses SMT Toolkit available http://www.statmt.org/moses/

and extraction such as NEs to improve the output quality of baseline PB-SMT system (Pal et al., 2010; Pal et al., 2013; Tan and Pal, 2014).

Named Entity Alignment

We initially identify NEs on both the source and target side of the POS-tagged parallel training corpus. We create a NE parallel corpus by extracting the source and target NEs from the NE-tagged (NNP or N_NNP) parallel translations in which both sides contain at least one NE. For example, we extract the NE translation pairs given in example (6) from the sentence pair shown in example (5), where the NEs are shown as italicized.

(5a) In/IN this/DT *Yamuna/NNP Bio/NNP Diversity/NNP Park/NNP* an/DT effort/NN has/VBZ been/VBN made/VBN to/TO grow/VB and/CC preserve/VB the/DT herbs/NNS produced/VBN in/IN the/DT *Yamuna/NNP* region/NN ./.

(5b) इनमें/DM_DMR *यमुना/N_NNP बायो/N_NNP डायवर्सिटी/N_NNP पार्क/N_NNP* में/PSP *यमुना/N_NNP* क्षेत्र/N_NN में/PSP उपजने/V_VM वाली/PSP वनस्पतियों/N_NN को/PSP एक/QT_QTC जगह/N_NN उगाने/V_VM और/CC_CCD संरक्षित/N_NN करने/V_VM की/PSP कोशिश/N_NN की/V_VM गई/V_VAUX है/V_VAUX ।/RD_PUNC

(6a) Yamuna Bio Diversity Park Yamuna

(6b) यमुना बायो डायवर्सिटी पार्क यमुना

Although the resulting bilingual NE table does not provide a perfect NE dictionary, it filters out NEs from the training sentences and improves word alignment at the start of the MT pipeline while it is appended as an additional training data to the given parallel corpus.

Example-based Phrase Extraction

We use EBMT techniques to extract additional phrase pairs from the training data to augment the SMT phrase pairs (baseline) in our experiments. We extract EBMT phrase pairs based on the work described in Cicekli and Guvenir (2001). They used a compiled approach of EBMT to automatically extract translation templates from sentence-aligned bilingual text by observing the similarities and differences between two example pairs. Their approach produces two types of translation templates: *generalized* and *atomic* translation templates. A generalized translation template replaces similar or differing sequences with variables while an atomic translation template does not contain any variable. We extract atomic translation templates as an additional phrase pair for our Hybrid MT system. Consider the following two English--Hindi translation pairs from the health domain data:

(7) patient feels weakness : रोगी को कमजोरी महसूस होती है

(8) patient feels restlessness : रोगी को बेचैनी महसूस होती है

These two examples share the word sequence *patient feels* and differ in the word sequence *weakness* and *restlessness* on the source side. Similarly, on the target side, the differing fragments are *कमजोरी* and *बेचैनी*. Based on these differing fragments, we extract the following sub-sentential phrase pairs in

(9) a. *weakness* : *कमजोरी* b. *restlessness* : *बेचैनी*.

We apply this process recursively to extract sub-sentential phrase pairs when more than one differing sequence is present in between a pair of sentences. The details of the algorithm can be found in Cicekli and Guvenir (2001). This particular approach has a cubic runtime complexity w.r.t. the number of sentences in the parallel corpus. It takes a significant amount of time to extract phrase pairs even from a small corpus. Therefore we used a heuristics to reduce the time complexity. We divided the entire corpus into *n* clusters based on the sentence length such that similar length sentences belong to the same cluster. We extract atomic translations from each of these clusters.

TM Implementation

Translation Memories (TM) are widely used as human translation tools. Since many translations are highly repetitive, finding existing translations for the whole or a part of the source sentence is important and TMs play a crucial role in reducing the workload of a translator. Our TM loads existing translations that are collected from the training data. The TM also contains EBMT phrases and parallel extracted NEs. The basic functions of the TM are:

- If the source sentence is found in the TM, it will immediately be replaced by the target text to generate the output sentence
- If a sequence of words is found in the TM, the source sequence will be replaced with the target word sequence and transferred to the next process in the MT pipeline

Hybrid System

The Hybrid approach combining TM, EBMT, and SMT is often investigated in terms of a multi-stage successive translation starting with NE substitution, Example based phrase substitution, and finally the application of the SMT technique to the remaining material in a segment to be translated. As mentioned previously, we implemented four different systems, namely Baseline **SMT**, Baseline SMT with NE alignment (**NEA**), NEA with EBMT phrase alignment (**NEA-EBMT**) and **TM-EBMT-SMT** hybrid system. In order to achieve optimal performance from the component modules, we finally generate a composite translation output using confusion network-based system combination (described below).

- **NEA System:** For the NEA system, we append extracted parallel NE list to the training data. This model has been trained on the same settings as the Baseline system described in page 20 and the training corpus contains parallel NE list as well as parallel baseline sentence pairs.
- **NEA-EBMT System:** In order to build the training corpus for this model, we appended the extracted parallel NE list and also added EBMT parallel phrases to the training sentence pairs. To build the PB-SMT model, we trained this model also on the same settings as the Baseline system, described in page 20.

- **TM-EBMT-SMT hybrid system:** In this system, we employ and maintain a translation memory. The TM repository consists of the parallel training data, the parallel NE list, and the EBMT parallel phrases. When a new sentence is entered for translation into the system, the MT system first checks in the stored TM, whether the translation is already present. If the input sentence is found in the TM then the output is immediately returned. If there are no exact matches found in the TM, then we find matches for a given sequence of tokens inside the TM repository. If there exist any matching sequences of source token strings in the input sentence, then the source sequence is immediately replaced with the target sequence from TM. The generated sequence serves as an input to the SMT system. The SMT system then produces the final translation output.

- **Post-processing:** As a final step, we generate transliterations of Out-Of-Vocabulary (OOV) words that remain un-translated from the source language. These OOV words may contain NEs. Our system post processed the output by replacing each OOV NE with a target language NE after looking up the extracted NE list from parallel corpus. Note that this is an additional MT system apart from the baseline and 3 Hybrid MT systems introduced above.

System Combination

Since we could only submit one system per language pair and per domain, we applied a system combination approach on the four MT system outputs described above. System Combination is a technique to combine translation hypotheses (outputs) from multiple MT systems. We implement the Minimum Bayes Risk coupled with Confusion Network (MBR-CN) framework described in Du et al. (2009). The MBR decoder (Kumar and Byrne, 2004) selects the best single system output from amongst the multiple outputs by minimizing BLEU (Papineni et al., 2002) loss. This output is referred to as the "backbone". A confusion network (Matusov et al., 2006) is built from the backbone while the remaining hypotheses are aligned against the backbone using the TER metric (Snover et al., 2006). In our experiments, 5 MT hypotheses (Baseline, NEA, NEA-EBMT, TM-EBMT-SMT, OOV Post-process) are fed to the System Combination framework from which one was selected as the backbone. The features used to score each arc in the confusion network are word posterior probability, target language model (3-gram, 4-gram), and length penalties. Minimum Error Rate Training (MERT) (Och, 2003) is applied to tune the CN weights.

Baseline Settings

The effectiveness of the present work is demonstrated by using the standard log-linear PB-SMT model as our baseline system. For building baseline system, we experimented with various maximum phrase lengths for translation model and n-gram settings for the language model. We found that using a maximum phrase length of 7 and a 5-gram language model produced best results in terms of BLEU scores for our baseline model (i.e. without the incorporation of outputs from the NLP tools).
The other experimental settings were: GIZA++ implementation of IBM word alignment model 4 with grow-diagonal-final-and heuristics for performing word alignment and phrase-extraction (Koehn et al., 2003). The reordering model was trained on msd-bidirectional (i.e. using both forward and backward models) and conditioned on both source and target language. The reordering model was built by calculating the probabilities of the phrase pairs being associated with the given orientation such as monotone (m), swap(s) and discontinuous (d). The 5-gram target language model with Kneser-Ney

smoothing (Kneser and Ney 1995) was trained using SRILM (Stolcke, 2002). Minimum Error Rate Training (MERT) (Och, 2003) was carried out on a held-out development set (devset). After the parameters were tuned, decoding was carried out on the held out test set.

Note that all systems described above, employ the same PB-SMT settings (apart from the feature weights which are obtained via MERT) as the Baseline system.

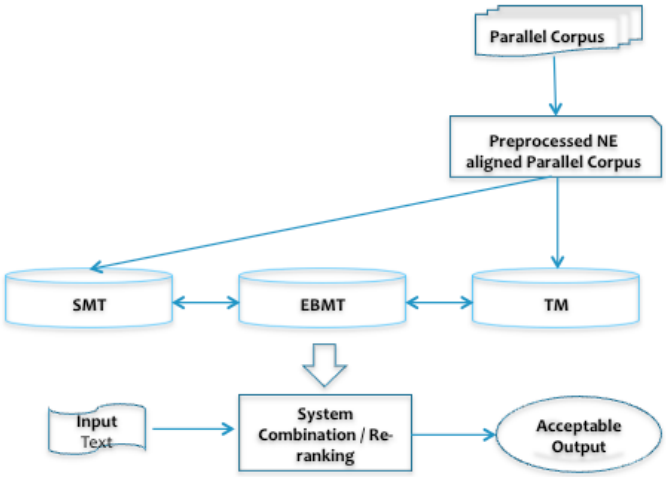

Figure 5: System Architecture

2.3 Towards Optimal Human-Machine Interactive System

Despite the efforts by MT developers and users to achieve good quality MT output, by populating dictionaries with company-specific and industry-specific terms, employing controlled language (CL) rules; machine translation outputs do not always produce satisfactory results. In almost all cases where publishing quality output is required, human intervention is required on MT output in terms of post-editing (PE), to achieve an acceptable quality level (Allen 2001, Schäfer 2003, Hutchins 2003). PE is generally defined as the act of correcting and editing of the text translated by an MT system. Various automatic or semi-automatic post-processing techniques to implement corrections for repetitive errors or fully automate PE have been developed, although MT output still needs to be post-edited by humans in order to produce publishing quality translation (Roturier 2009, TAUS 2010). Even though MT output needs human PE, it is often faster and cheaper to post-edit MT output than to perform human translation from scratch. In some cases, recent studies have even shown that the quality of MT plus PE can exceed the quality of human translation (Fiederer & O'Brien 2009, Koehn 2009, DePalma & Kelly, 2009) as well as productivity (Zampieri and Vela, 2014). Aimed at cost-effective and timesaving use of MT, the PE process needs to be further optimized (TAUS 2010).

There have also been many studies regarding the impacts of various factors and methods; those were examined against the volume of PE effort. However, those studies have not been conducted to observe PE effort in a commercial work environment. The overall purpose of the present study is to answer two fundamental questions 'What would be the optimal design of a PE system?' which is ultimately determined by the quality of MT output. And "How can human involvement be optimized in a PE system to reduce post editing effort?"

2.3.1 Post Editing by using human in the loop

In order for the post-editors to design an optimal human-machine interactive system, they should ideally take the following aspects into consideration.

- The post-edited output quality should be the same as that achieved by the traditional manual translation process.
- Do minor changes, if the MT translation is acceptable.
- If the MT translation has problems but could be understood, do necessary editing.
- If the MT translation is totally unacceptable, use the key terminology from the translation and re-translate from scratch.
- Absolutely spend no time on evaluating MT output.
- If the word order is correct but is slightly different, there is no need to post-edit.
- Avoid replacing a word with a synonym if the original word is correct.

Using linguistic phenomena

SMT systems are considered as one of the most popular approaches to machine translation (MT). However, SMT can suffer from grammatically incorrect output with erroneous syntactic and semantic structure for the language pair on which it is being applied. It is observed that the grammatical errors not only weaken the fluency, but in some cases it may even completely change the meaning of a sentence. In morphologically rich languages, grammatical accuracy is of particular importance, as the interpretation of syntactic relations depends heavily on the morphological agreement within sentences. Morphological errors create serious problems in the context of translating the sentiment related components from source to target language. In the research reported in (Pal et al., 2014e), we reduce these errors by focusing on the roles of sentiment-holder, sentiment expression, corresponding objects and their relations with each other at the clause level. State-of-the-art Machine Translation (MT) does not perform well while translating sentiment components from source to target language. The components such as the sentiment holders, sentiment expressions and their corresponding objects and relations are not always maintained well during translation. In this section, we describe how sentiment analysis can improve the translation quality by incorporating the roles of such components to augment translation tables and to support automatic post-editing. We demonstrate how a baseline PB-SMT system based on the sentiment components can achieve 33.88% relative improvement in BLEU for the under-resourced language pair English-Bengali.

A common error that occurs during translation using SMT is that the relations among the holders, associated sentiment expressions and their corresponding objects in a sentence (in case of complex and compound sentences) may interchange. In the following example, the position of the sentiment expression has been changed in target language during the translation. Similar instances are found if any interchange occurs in case of other sentiment components such as holder or object.

Example 10:

Source: In 1905, <*holder*>Calcutta</*holder*> <*expression_1*>protested</*expression_1*> <*object_1*>the partition of Bengal</*object_1*> and <*expression_2*>boycotted </*expression_2*> <*object_2*>all the British Goods </*object_2*>.

Target: 1905 sale, <*holder*>Calcutta </*holder*> <*object_1*>bongo vongo-r</*object_1*> <*expression_2*>boykot korechilo </*expression_2* > ebong <*object_2*>ssmosto british samogri </*object_2*> <*expression_1*> protibad janiyechilo </e*xpression_1*>.

As a result the entire semantics of the sentence has been changed even though the sentence is considered as grammatically correct. Another major challenge is to develop a sentiment phrase aligned system between a resource-rich language English and a resource-constrained language Bengali.

In our approach, sentiment expressions, sentiment holder and the corresponding objects of the holders are used to improve the phrase alignment of the SMT system during the training stage. Sentiment information is also used in the automatic post-editing of the SMT output after the decoding phase. SMT is based on a mathematical model, is reliable and cost effective in many applications. This is one of the main reasons to choose SMT for our English-Bengali translation task. For automatic post-editing, we marked the phrases that contain sentiment expression, holders and their corresponding object. After translating the marked-up sentences, we then restructure the output according to the sentiment relations between the sentiment holder and the sentiment expression. Our approach involves the following steps:

- We first identify phrases, which contain sentiment holder, sentiment expressions and their corresponding objects.
- We aligned these phrases using word alignment provided by GIZA++.
- The aligned phrases are incorporated with the PB-SMT phrase table.
- Finally, the automatic post-editing is carried out using the positional information of sentiment components.

2.3.1.1.1 System Description

Initially, we identify the sentiment expressions, holders and objects from English-Bengali parallel sentences. A sentiment phrase alignment model has been developed using our existing baseline table provided by GIZA++. These aligned sentiment phrases are integrated with the state-of-the-art PB-SMT phrase table as supplementary information. Finally, an automatic post editing system has been developed to correct the translation output using the textual clues identified from the sentiment components.

Sentiment expression, holder and object identification from Parallel corpus

Sentiment: Initially, sentiment expressions in our translation data were not tagged with sentiment polarity. Therefore, we developed a bootstrapping method to tag the words with sentiment polarity. We have tagged the English sentiment words using the SentiWordNet 3.0 (Baccianella et al., 2010). The raw English sentences were parsed and the stems of the words were extracted using the Stanford parser[7]. SentiWordNet examines stemmed words along with their part of speech and provides a sentiment score

[7] http://nlp.stanford.edu/software/lex-parser.shtml

for each stemmed word. The sentiment of the word is judged positive, negative or neutral according to its sentiment scores. We have manually created a stop word list of around 300 stop words that helps us to remove the stop words from the sentences. But the words 'not', 'neither' etc. are not removed as they are valence shifters and can change the sentiment of the whole sentence. We identified 76924 and 36125 positive and negative word tokens in our data respectively.

Holder (Subject Based): Sentiment analysis involves identifying its associated holder and the event or topic. A sentiment holder is the person or organization that expresses the positive or negative sentiment towards a specific event or topic. English input sentences are parsed by the Stanford Parser to extract the dependency relations. The output is checked to identify the predicates (i.e., "*nsubj*" and "*xsubj*"), so that the *subject* related information in the "*nsubj*" and "*xsubj*" predicates are considered as probable candidates of sentiment holders.

We correlate our sentiment words with the holder using the dependency tree. For example, the sentence "*I hate chocolate but he loves it.*" has two sentiment expressions, "hate" and "love". Here the root word and the sentiment expression is the same, i.e. "hate". We identify that the sentiment expression, "*hate*" and subject "*I*" are related with "*nsubj*" relation. We conclude that "*I*" is the sentiment holder of the word "*hate*". Similarly, we identify that "*he*" is the sentiment holder of word "*loves*".

Example 11: **nsubj**(hate-2, **I**-1), root(ROOT-0, hate-2), **dobj**(hate-2, **chocolate**-3), nsubj(loves-6, **he**-5), conj_but(hate-2, loves-6), **dobj**(loves-6, **it**-7).

In our English source training data we have identified only 22992 sentiment holders, in comparison to a total of 113049 sentiment expressions.

Object: The parsed data were analyzed to identify the object of a sentence. It is found that the relations, "*dobj*" and "*obj*" are considered as the probable candidates for the object. In the above example sentence along with the parsed output and dependency relations (example 11), the "*dobj*" dependency relation includes the object. Here, "*chocolate*" and "*it*" are identified and tagged as the "*object*".

Sentiment Phrase Alignment

In case of low-resource languages, chunking the parallel sentences (both source and target) adds more complexity in building any system. POS taggers or Chunkers might not be available for some low-resource languages. In such cases, the methodology we present below can help chunk sentences. In this paper, we propose a simple but effective chunking technique. The resulting sentence fragments are very similar with grammatical phrases or chunks. We collected the stop word lists for English as well as Bengali to implement this method (Groves and Way, 2005). We chop a sentence into several fragments whenever a stop word is encountered.

Example 12: English sentence fragmentation
"In 1905, <holder>Calcutta</holder> <expression_1>protested</expression_1> the <object_1>partition</object_1> of Bengal and <expression_2>boycotted</expression_2> all the <object_2>British Goods</object_2>."
1. (In 1905) (, Calcutta protested) 2. (the partition) 3. (of Bengal) 4. (and boycotted) 5. (all) 6. (the British Goods)

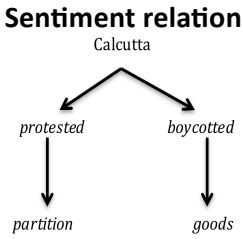

Figure: 6

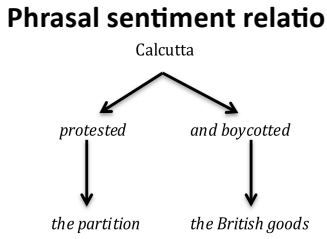

Figure 7

Bengali sentence fragmentation:

"1905 sale, Kolkata bongo vongo-r protibad janiyechilo ebong ssmosto british samogri boykot korechilo."

Pre-processing: *1905 sale , Kolkata bongo vongo -r protibad janiyechilo ebong ssmosto british samogri boykot korechilo.*
1. (1905 sale) 2. (, Kolkata bongo vongo) 3. (-r protibad janiyechilo) 4. (ebong ssmosto british samogri boykot korechilo.)

Initially, we built an English-Bengali word alignment model, which was trained with the same EILMT tourism domain parallel corpus of 22,492 sentences. Using this word alignment knowledge we aligned bilingual sentiment phrases. For establishing the alignment, we use the same phrase alignment algorithm which is used in the existing state-of-the-art PB-SMT system Moses. The rest of the processes, such as scoring and phrase table creation also follow the state-of-the-art system.

Automatic Post Editing using Sentiment Knowledge

The decoding process is carried out with the Moses decoder and the PB-SMT model is computed with Moses. Recall our previous example, and that after translation; the sentiment relation may interchange, so that the semantic meaning of the sentence may be the opposite of what was stated in the source. For example:

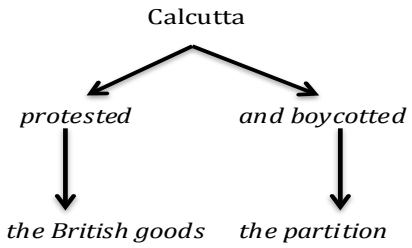

Figure 8

To correct this error in translation, we repositioned the words using our source sentiment relation knowledge obtained on the English side of the data. First, we marked the input source sentence with sentiment holder, sentiment word and their corresponding objects and measured the distance between them. The distance is then also measured on the translated sentence. If they maintain similar distance this ensures that they did not exchange their positions. Otherwise, we have to exchange the position of sentiment expression with their corresponding holder and object according to the distance measure. In this way, we automatically post-edited the entire translated sentences, if required.

Automatic Post Editing (APE)

The MT output is post-edited (PE) automatically using both cross-lingual and monolingual APE systems.

Cross-lingual Automatic Post Editing (APE)

The cross-lingual APE operates on the following:

- Input source file
- Machine translated output file
- The corresponding cross lingual word alignment knowledge between Input Source and MT output.
- Post edited (PE) output

This module is designed as a purely statistical one. It estimates post edited output given the source input sentence and corresponding machine translation output.
The APE model will be designed in such a way that it estimates an n-gram based PE table (like the phrase table in PB-SMT) where each phrase will be ranked and stored as *P(post-editing phrase | source phrase, target phrase)*. The decoder operates on this PE model and a language model. The decoder provides APE output using Statistical cross lingual APE (SC-APE) model.

The SAPE model can be enhanced with the following factors:

- Lemma
- Root
- POS
- Morphology

Monolingual word alignment feature can also be use as one of the features, which will be based on

- Berkeley / GIZA++ word aligner, METEOR, TER word alignment
- Other TER features

The module can also be trained on some error based input features, e.g., P (post-editing phrase | source phrase, target phrase, ***errors*** **(see next section on error classifier)**).

Monolingual Automatic Post Editing (APE)

Post editing using monolingual MT

The monolingual APE model is designed in such a way that it estimates an n-gram based PE table (like the phrase table in PB-SMT) where each phrase is ranked and stored as *P(post-editing phrase | Machine*

Translation phrase). The decoder is operated on this PE model and a language model with post edited data. The decoder provides APE output using a statistical monolingual APE (SM-APE) model. The SM-APE model is estimated on hybrid word alignment and parallel monolingual training data. Here, monolingual parallel training data is basically defined as Machine Translation output data and their corresponding manually post-edited data. Hybrid word alignment is a union of three different word alignment tables (Berkeley word alignment, Meteor word alignment and TER word alignment table) and it is trained on monolingual parallel data. The monolingual word alignment model has been prepared with hybrid word alignment

Learning from Errors

This module is mainly designed with two modules: Error detection of MT output and automatic error correction based on error types.
The Error Types cover:

- Lexical Error
- Reordering error
- Morphological error

The following predictive features are extracted to feed the error classifier:

Language Independent

- class based feature extraction
- error position detection
- insertion/deletion/substitution
- other probabilistic features
- context

Language Specific

- lexical and syntactic: word dependency, chunk, POS, lemma (Super Tag or CCG, TAG feature)
- Morph
- Semantics
- coreference
- context

Combined

combining both language independent and language specific features.

Automatic Error correction

Design a model which provides possible solution to given error(s). This model can be purely statistical, supervised or unsupervised.
Both tasks (cross-lingual and monolingual APE) work within the system in a hand-shaking way. They are further categorised into two different methods: language independent and language dependent. Each

module consists of an error detection classifier, an error correction module and a decoder. Finally we combine all the different modules to test the entire pipeline.

2.3.2 Human in the loop and Feedback System

The output provided by the entire system is to be used as future training data in a bootstrapping way. This will enhance the entire system during iteration.

3 Ideal Hybrid MT Workflow

3.1 Evaluation by using plug and play of various components

The system will be evaluated based on a plug and play architecture combining different components of the component technology as described before (Section 2). Ideally, users have complete freedom to configure their workflow by selecting different components that they wish to use.

3.2 Propose ideal translation workflow with respect to User requirement Analysis

By evaluating various settings as described in Section 3.1, the optimal setting will be provided to the users as default settings. An ideal workflow using the Component Technologies is designed and presented in Figure 9. The first step of the workflow is preprocessing of the parallel training data. This step also involves knowledge acquisition from external resources such as extracted parallel texts from comparable corpora which are typically added to the training corpus as additional training material. Data preprocessing also serves as a crucial role as identification and special treatment of the linguistic units, e.g., MWEs, NEs and compound verbs are expected to improve the performance of SMT systems. This answers the first research question (RQ1): How can existing resources and data be optimally used?

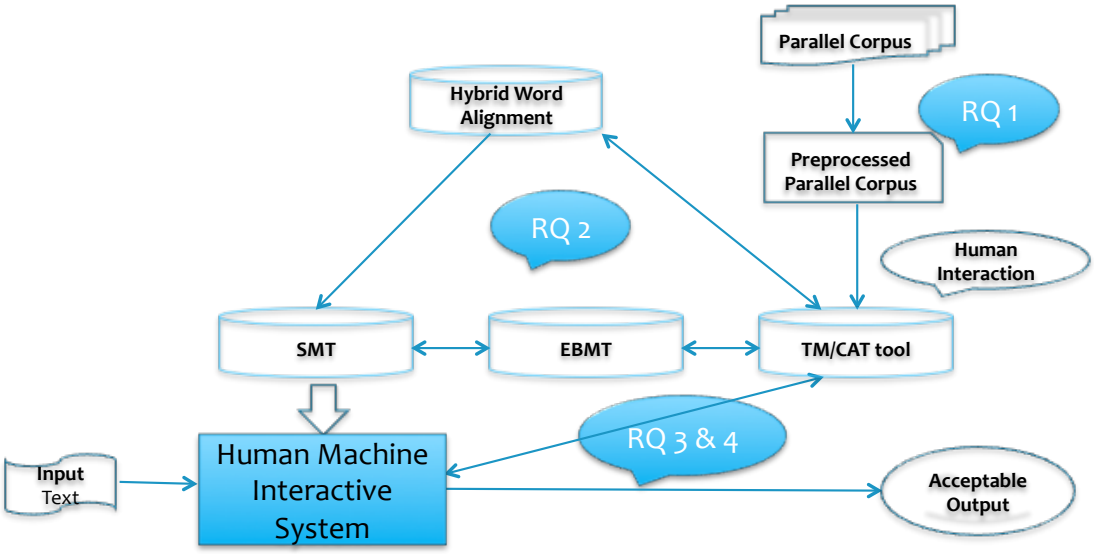

Figure 9: Ideal Hybrid MT workflow

The second step of the workflow deals with how to find an ideal hybrid implementation of MT, which provides the answer of RQ2: What would the ideal hybrid implementation of MT be? Different MT technologies (e.g. SMT, EBMT, TM etc.) and their different components (such as word alignment, phrase extraction in SMT, example-based phrase extraction in EBMT, incorporation of linguistic phrases in SMT) are combined together in terms of "plug-in and play" methodology and present a default ideal hybrid MT system to the translation technology users. The translation technology users have the complete freedom to use the hybrid technology in their own terms, i.e., they can choose whichever components they intend to use, and decide on whichever components provide the best solution as per their requirements. This facilitates the user attention as well as their consideration to use this new technology. The user can edit or make corrections to the MT output as a post-editor. The hybrid system will learn automatically in accordance with human corrections. This answers the following research questions:

- RQ3: How can human interaction be implemented in existing MT workflows?
- RQ4: How could human involvements be optimized?

3.2.1 Market Study and user requirements

The most popular type of technology that is widely used in today's translation market is translation memory (TM) systems. The acceptance of these tools appearing in the market is based on the fact that they have the ability to reduce the translator's effort, increase their productivity and reduce cost. TMs provide support to translators by retrieving segments of text that were already translated. This can be done by simple string matching or it can also be improved by using semantic/syntactic information or paraphrasing, as in the work carried out by EXPERT ESR4 (Gupta and Orasan, 2014; Gupta et al., 2015). Due to technological advancement, not only segment matching became more accurate but developers of these tools have added more features and functionalities to them as well as translation-related resources (e.g. term banks and translation memory repositories). Therefore, tools often become too complex and require more time and effort for the translation to learn and to used.

Some features such as terminology extractors, corpora compilation tools and automatic translation systems and some translation-related resources are really beneficial and already integrated in some translation software (SDL Multiterm[8] in SDL Trados Studio[9], LiveDocs in MemoQ[10], MyMemory[11], Web-based applications MateCAT[12] and Wordfast[13] TM as an add-on to Microsoft Word through macros etc.). It would be of interest to find out why translators prefer to work with these tools and how and to what degree of flexibility should developers satisfy users with different preferences (source: EXPERT deliverables D2.1: User Requirement Analysis).

Collecting live user Requirements to enhance the tool: This ensures that data are collected to provide live user support to users after using the tool. The tool needs to save the following logs:

User attitude log

[8] http://www.sdl.com/products/sdl-multiterm/desktop.html
[9] http://www.sdl.com/products/sdl-trados-studio/
[10] https://www.memoq.com/
[11] http://mymemory.translated.net/
[12] https://www.matecat.com/
[13] http://www.wordfast.net/

- Binary quality scores for each output. (Good/bad) [Per sentence]
- Fine grained quality score (percentage of change during Post Editing) [per sentence]
- Track cursor position [per sentence]
- Which words are deleted/inserted/ substitution [per sentence]
- Repositioning words [per sentence]
- Correction time [per sentence]
- Percentage of satisfaction to use this tool

Component choice log by user

- TM [check box]
- EBMT [check box]
- SMT [check box]
- External Terminology [check box]
- External Data/ resources [check box]

Language pair Component choice log by user

- User expertise bilingual [expert/medium/preliminary] [check box]
- User expertise monolingual [expert/medium/preliminary] [check box]
- User profile
- Translation Domain
- Domain Expertise

3.2.2 Needs or problem encountered by real life users to use of TM and related tools

According to a survey (D2.1: User Requirement Analysis) based on popularity of various translation technologies, TM systems appear to be the only type of tools that was used regularly by the majority of translators. However, there are still a considerable number of translators who have never heard of such technologies at all. Some tools are not fully adopted. Only one type of technology, concordance system, is unknown to the majority of translators. Tools for compiling or managing corpora are less commonly used on a regular basis. One reason is possibly that some of the technologies are newly integrated in CAT tools but translators are completely unaware of them or that they may not provide satisfactory result or that they are slow. Some technologies may not be considered appropriate for everyday use (e.g. compiling and managing corpora). The usage of MT services is also similar; some users are still using it, few are planning to use it in future and some users abandoned MT due to poor quality. In comparison to automatic translation, translation memories turned out to be much more popular compared to other systems. But still translators prefer to use the MT system integrated in their CAT tool. Terminology management tools integrated in the translation software is also helpful as per translator preferences.

Tool comparison: In 2010, half the respondents cited the use of some version of "SDL Trados" (more details on this was provided in a later survey); the next highest responses at just under 20% were for Déjà Vu and memoQ. Three and a half years later, Atril's share of users appears to have declined considerably, and the use of memoQ appears to be on par with SDL Trados Studio. OmegaT, an excellent free and Open Source translation support tool capable of working with translation formats from the leading tools, appears to be doing better than many of the commercial tools in the survey[14].

Translation Tools	*Selections*	*% of respondent*
OmegaT	93	8.19%
Déjá Vu	117	10.31%
SDL Trados 2007	354	31.19%
SDL Trados Studio	**365**	**32.16%**
Wordfast Classic	158	13.92%
Wordfast Pro	83	7.31%
MemoQ	352	31.01%
SDLX	50	4.41%
STAR Transit	36	3.17%
Across	47	4.14%
Cafetran	59	5.20%
MemoSource	21	1.85%
Fluency	7	0.62%
Other	123	10.84%
None	47	4.14%
Total	1912	
Respondent	1135	

Table 3: Tool usage statistics

ESR2 Research Publications

Book Chapter [to be published]

- **Santanu Pal**, Sudip Kumar Naskar. "*Hybrid Word Alignment Model.*"

2015

- **Santanu Pal,** Partha Pakray, Alexander Gelbukh and Josef Van Genabith. 2015. *Mining Parallel Resources from Comparable Corpora to improve performance of Machine Translation*. **Springer publication of Lecture Notes on Computer Science** Proceedings of the 16th International

[14] http://www.translationtribulations.com/2014/01/the-2013-translation-environment-tools.html

Conference on Intelligent Text Processing and Computational Linguistics (CICLING - 2015), Cairo, Egypt.

2014

- **Santanu Pal**, Ankit Srivastava, Sandipan Dandapat, Josef van Genabith, Qun Liu and Andy Way. 2014d. *USAAR-DCU Hybrid Machine Translation System for ICON 2014.* In Proceedings of the 11th International Conference on Natural Language Processing, ICON-2014, Goa, India.
- **Santanu Pal**, Braja Gopal Patra, Dipankar Das, Sudip Kumar Naskar, Sivaji Bandyopadhyay and Josef van Genabith. 2014e. *How Sentiment Analysis Can help Machine Translation.* In Proceedings of the 11th International Conference on Natural Language Processing, ICON-2014, Goa, India.
- Debasis Ganguly, **Santanu Pal** and Gareth J.F Jones. (2014). *DCU@FIRE-2014: fuzzy queries with rule-based normalization for mixed script information retrieval.* In: Forum for Information Retrieval Evaluation (FIRE 2014) workshop, 5-7 Dec 2014, Bangalore, India.
- Lapshinova-Koltunski, E. and **Santanu Pal**. 2014. *Comparability of Corpora in Human and Machine Translation.* In Proceedings of BUCC, 7th Workshop on Building and Using Comparable Corpora. Building Resources for Machine Translation Research, (BUCC-2014), Reykjavik, May 27, 2014
- Liling Tan and **Santanu Pal**. 2014. *Manawi: using multi-word expressions and named entities to improve machine translation.* In Proceedings of Ninth Workshop on Statistical Machine Translation. Baltimore, USA.
- **Santanu Pal**, Partha Pakray and Sudip Kumar Naskar. 2014a. "*Automatic Building and Using Parallel Resources for SMT from Comparable Corpora*", In Hybrid Approaches to Translation (HyTra-2014) Workshop in 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014), Gothenburg, Sweden, 26–30 April 2014.
- **Santanu Pal,** Pintu Lohar and Sudip Kumar Naskar. 2014b. *Role of Paraphrases in PB-SMT*. Published in the **Springer LNCS** Proceedings of the 15th International Conference on Intelligent Text Processing and Computational Linguistics (CICLING - 2014), Kathmandu, Nepal.
- Pintu Lohar, Pinaki Bhaskar, **Santanu Pal** and Sivaji Bandyopadhyay. 2014. *Cross Lingual Snippet Generation using Snippet Translation System*. Published in the **Springer LNCS** Proceedings of the 15th International Conference on Intelligent Text Processing and Computational Linguistics (CICLING - 2014), Kathmandu, Nepal.
- **Santanu Pal**, Sudip Kumar Naskar and Sivaji Bandyopadhyay. 2014c. *Word Alignment-Based Reordering of Source Chunks in PB-SMT.* Published in the Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC), Reykjavik, Iceland.

2013

- **Santanu Pal**, Mahammed Hasanuzzaman, Sudip Kumar Naskar and Sivaji Bandyopadhyay. 2013b. *Impact of Linguistically Motivated Shallow Phrases in PB-SMT.* In the Proceedings of In the 10th International Conference on Natural Language Processing (ICON-2013), Noida, India, pp. 272-277.
- **Santanu Pal**, Sudip Kumar Naskar and Sivaji Bandyopadhyay. 2013a. *A Hybrid Word Alignment Model for Phrase-Based Statistical Machine Translation.* 2nd Workshop of Hybrid Approaches to Machine Translation (HyTra 2013), ACL-2013, Sofia, Bulgaria, pp. 94-101

References

Allen, J. (2003). Post-editing. In Somers, H. L. (Ed.). Computers and Translation: A Translator's Guide, pp. 297-318. Amsterdam: John Benjamins.
Baccianella, S., Esuli, A. and Sebastiani, F. (2010). Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In Proceedings of the 7th conference on International Language Resources and Evaluation (LREC'10), Valletta, Malta.

Banerjee, S., and Lavie, A. (2005). An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In proceedings of the ACL-2005 Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization, pp. 65-72. Ann Arbor, Michigan., pp. 65-72.

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. (1993). The mathematics of statistical machine translation: parameter estimation. Computational Linguistics, 19(2):263-311.

Carl, M. and Way, A. (2006). Recent Advances in Example-Based Machine Translation, Kluwer Academic Cicekli, I. and Guvenir, H. A. (2001). Learning Translation Templates from Bilingual Translation Examples, Applied Intelligence, 15(1):57–76.

Du, J., He, Y., Penkale, S., and Way, A. (2009). MaTrEx: The DCU MT System for WMT 2009, Proceedings of the Fourth Workshop on Statistical Machine Translation, European Association for Computational Linguistics, :95–99.

Eck, M., Vogel, S., and Waibel, A. (2004). Improving statistical machine translation in the medical domain using the Unified Medical Language System. In Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004), Geneva, Switzerland, pp. 792-798.

Ekbal, A. Naskar, S. K. and Bandyopadhyay., S. (2006) A Modified Joint Source-Channel Model for Transliteration. Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, pages 191–198,Sydney, July 2006.

Ekbal, A., and Bandyopadhyay, S. (2008). Maximum Entropy Approach for Named Entity Recognition in Indian Languages. International Journal for Computer Processing of Languages (IJCPOL), Vol. 21 (3), 205-237.

Ekbal, A., and Bandyopadhyay, S. (2009). Voted NER system using appropriate unlabeled data. In proceedings of the ACL-IJCNLP-2009 Named Entities Workshop (NEWS 2009), Suntec, Singapore, pp.202-210.

Groves, D. and Way, A. (2005). Hybrid data-driven models of machine translation. Machine Translation 19(3-4): 301-323.

Gupta, R. and Orasan, C. (2014). Incorporating Paraphrasing in Translation Memory Matching and Retrival. In Proceedings of the European Association of Machine Translation (EAMT-2014), Dubrovnik, Croatia.

Gupta, R., Orasan, C., Zampieri, M. Vela, M. and van Genabith, J. (2015). Can Translation Memories Afford Not to Use Paraphrasing? Proceedings of the 18th Annual Conference of the European Association for Machine Translation (EAMT). Antalya, Turkey.

Hansen-Schirra, S. (2002). The Nature of Translated Text - An Interdisciplinary Methodology for the Investigation of the Specific Properties of Translation. Dissertation. Saarbrücken, Germany: Saarland University.

Hutchins, W. J.; Somers, H. L. (1992). An introduction to machine translation. London: Academic Press.
Kay, M. (1980). The proper place of men and machines in language translation, Xerox PARC, :1–21.

Kneser R. and Ney H. (1995). Improved backing-off for m-gram language modeling, In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Detroit, MI vol. 1, : 181-184. .

Koehn, P. (2010). Statistical Machine Translation, Cambridge University Press.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A. and Herbst, E. (2007). Moses: open source toolkit for statistical machine translation, In Proc. of the 45th Annual meeting of the Association for Computational Linguistics (ACL 2007): Proc. of demo and poster sessions, Prague, Czech Republic, : 177-180.

Koehn, P., Och, F. J. and Marcu, D. (2003). Statistical phrase-based translation, In Proc. of HLT-NAACL 2003, Edmonton, Canada, : 48-54.

Kumar S. and Byrne, W. (2004). Minimum Bayes Risk Decoding for Statistical Machine Translation, In Proceedings of the North American Association for Computational Linguistics (NAACL), Boston,MA: 169-176.

Liang, P., Taskar, B. and Klein D. (2006). 6th Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, HLT-NAACL-2006, Pages 104-11

Marcu, D. (2001). Towards a Unified Approach to Memory and Statistical-Based Machine Translation. In Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL 2001), Toulouse, France, pp 386-393.

Matusov, E., Ueffing, N. and Ney, H. (2006). Computing Consensus Translation from Multiple Machine Translation Systems Using Enhanced Hypotheses Alignment, In Proceedings of the European Association for Computational Linguistics (EACL 2006), Trento, Italy, : 33-40.

O'Brien, S., Roturier, J., De Almeida, G. (2009). Researching and Teaching Post-Editing. In Post-Editing MT Output - Views from the researcher, trainer, practitioner. Download at: http://mt-archive.info/MTS-2009-OBrien-ppt.pdf.

Och, F. J. (2003). Minimum error rate training in statistical machine translation, In Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL-2003), Sapporo, Japan, : 160-167.

Pal, S. and Bandyopadhyay, S. (2012). Bootstrapping Method for Chunk Alignment in Phrase Based SMT. In the Proceedings of the Joint workshop on exploiting Synergies between Information Retrieval and

Machine Translation (ESIRMT) and Hybrid Approaches to Machine Translation (HyTra), EACL-2012, Avignon France, pp.93-100

Pal, S., Chakraborty T., and Bandyopadhyay, S. (2011). Handling Multiword Expressions in Phrase-Based Statistical Machine Translation. In the Proceedings of the Machine Translation Summit XIII, Xiamen, China. pp. 215-224.

Pal, S., Naskar, S. K. and Bandyopadhyay S. (2013). MWE Alignment in Phrase Based Statistical Ma- chine Translation, In the Proceedings of the Ma- chine Translation Summit XIV, Nice, France, : 61- 68

Pal, S., Naskar, S. K., Pecina, P., Bandyopadhyay, S. and Way, A. (2010). Handling Named Entities and Compound Verbs in Phrase-Based Statistical Ma- chine Translation, In Proc. of Multiword Expression Workshop (MWE-2010) The 23rd International conference of computational linguistics (Coling 2010), Beijing, Chaina,: 46–54

Papineni, K., Roukos, S., Ward, T. and Zhu W. (2002). BLEU: a method for automatic evaluation of ma- chine translation, In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002), Philadelphia, PA, : 311-318

Ramaswamy, S. (1993). EnGendering Language: The Poetics of Tamil Identity, In Comparative Studies in Society and History, vol. 35:4, : 683-725.

Sag, I. A., Baldwin, T. Bond, F., Copestake, A. and Flickinger, D. 2002. Multiword expressions: A pain in the neck for NLP. In Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2002), Mexico City, Mexico, pp. 1–15.

Smith, J. R., Quirk, C., and Toutanova, K. (2010, June).Extracting parallel sentences from comparable corpora using document level alignment. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (pp. 403-411). Association for Computational Linguistics.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L. and Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In Proceedings of the 7th Conference of the Association for Machine Translation in the Americas (AMTA 2006), Cambridge, MA, pp. 223-231.

Srivastava, A., Haque, R., Naskar, S., and Way A. (2008). MaTrEx: The DCU Machine Translation System for ICON 2008, Proceedings of the NLP Tools Contest: SMT (English to Hindi), ICON 2008.

Stolcke, A. (2002). SRILM: An Extensible Language Modeling Toolkit, In Proceedings of International Conference on Spoken Language Processing, Denver vol. 2, : 901-904.

Tan L. and Pal, S. (2014). Manawi: Using Multi-Word Expressions and Named Entities to Improve Machine Translation, Proceedings of the Ninth Workshop on Statistical Machine Translation, ACL, :201–206.

TAUS Report. (2010, March). Postediting in Practice, p. 6. Download at: http://www.translationautomation.com/reports/postediting-in-practice

TAUS/CNGL. (2010). Maschine Translation Post-Editing Guidelines Published. Online article: http://www.cngl.ie/tauscngl-machine-translation-post-editing-guidelines- published/

Vogel, S., Ney, H. and Tillmann, C. (1996). HMM-based word alignment in statistical translation. In Proceedings of the 16th International Conference on Computational Linguistics (COLING 1996), Copenhagen, pp. 836-841.

Wu, H., Wang, H. and Chengqing Zong. (2008). Domain adaptation for statistical machine translation with domain dictionary and monolingual corpora. In Proceedings of the 22nd International Conference on Computational Linguistics (COLING 2008), Manchester, UK, pp. 993-1000.

Zampieri, M. Vela, M. (2014). Quantifying the Influence of MT Output in the Translators' Performance: A Case Study in Technical Translation. Proceedings of the EACL Workshop on Humans and Computer-assisted Translation (HaCat). Gothenburg, Sweden. p. 93-98.

Zhixiang R., Lü, Y., Cao, J., Liu, Q. and Huang, Y. (2009). Improving statistical machine translation using domain bilingual multiword expressions. In Proceedings of the 2009 Workshop on Multiword Expressions, ACL-IJCNLP 2009, Suntec, Singapore, pp. 47-54.