
Machine Translation

Andy Way*

National Centre for Language Technology & Centre for Next Generation Localisation,
School of Computing,
Dublin City University,
Dublin 9, Ireland
{away}@computing.dcu.ie

Abstract This chapter has two main aims: (i) to present the state-of-the-art in Machine Translation (MT), namely Phrase-Based Statistical MT, together with the major competing paradigms used in MT research and development today; and (ii) to provide an overview of the MT research carried out by my team here at *DCU*, characterised here in terms of ‘hybrid MT’. In addition, we provide our views on the directions that MT research might take in the near future, and conclude the chapter with lists of further reading for the interested reader.

* Thanks to Jinhua Du, Hany Hassan, Patrik Lambert, Yanjun Ma, Sara Morrissey, Sudip Naskar, Sylwia Ozdowska, John Tinsley, and Ventsislav Zhechev, for their considerable help in putting this Chapter together. Special thanks are due to Mikel Forcada and Felipe Sánchez-Martínez for helping with the section on RBMT. A draft chapter for the Blackwell *Computational Linguistics and Natural Language Processing Handbook*, edited by Alex Clark, Chris Fox and Shalom Lappin. This draft formatted on 9th June 2009.

1 Introduction

There are many other overviews of Machine Translation (MT) available, e.g. (Somers, 2000; Hutchins, 2003; Somers, 2003a; Jurafsky & Martin, 2008). In this chapter, we plan to inform the reader as to the state-of-the-art in MT *now*, rather than giving a detailed history of the field, much of which has been written before.

It is clear to all who are active in the area of MT today that the leading paradigm, especially in the research field, is Phrase-based Statistical Machine Translation (PB-SMT) (Marcu & Wong, 2002; Koehn *et al.*, 2003). Until such papers appeared, SMT models of translation were based on the simple word alignment models of Brown *et al.* (1990, 1993). Now that SMT systems learn phrasal as well as lexical alignments, this has led to an unsurprising increase in translation quality compared to the IBM word-based models. In addition, it has become harder to describe the differences between statistical models of translation and Example-Based MT (EBMT), though the latter still accesses the corpus of source-to-target examples at runtime.²

When it comes to which commercial systems are available, however, the balance is tipped in completely the opposite direction, for the vast majority of such models are Rule-Based MT (RBMT) systems. Research systems such as Apertium (Armentano-Oller *et al.*, 2006) are also prominent, and we give some attention to such models later in the chapter.

The remainder of the chapter is organised as follows. In Section 2, we present a thorough overview of the leading paradigm in MT today, namely PB-SMT. We give an end-to-end description of all tasks involved, from pre-processing, to decoding, and thence to post-processing and evaluation. In Section 3, we describe alternative approaches to this mainstream model, each of which has attracted a strong following. These include Hierarchical and Tree-Based models of MT, EBMT, RBMT, and hybrid combinations of these approaches. In Section 4, we describe a number of MT applications, including online MT, undoubtedly *the* biggest growth area for MT in the last few years. In addition, we describe Translation Memories, Spoken Language Translation, and MT for non-spoken languages. Section 5 then focuses on our own MT research and development at *DCU*, presented in the form of hybrid systems. In Section 6 we summarize the state of affairs in MT today, and provide our view on the directions that MT research might take in the next few years. Finally, we provide a list of further reading for the interested reader to follow up on any of the core sections.

² Note, however, that (Lopez, 2008) describes an SMT system which uses pattern matching to avoid problem of computing infeasibly large statistical models. His approach directly accesses the training corpus at runtime, but his model is by any measure an EBMT system, despite the steps taken to avoid the term.

E1: Often, in the textile industry, businesses close their plant in Montreal to move to the Eastern townships.
F1: Dans le domaine du textile souvent, dans Montréal, on ferme et on va s'installer dans les Cantons de l'Est.

E2: There is no legislation to prevent them from doing so, for it is a matter of internal economy.
F2: Il n'ya aucune loi pour empêcher cela, c'est de la régie interne.

E3: That is serious.
F3: C'est grave.

Figure 1. A sentence-aligned corpus

2 The State-of-the-Art: Phrase-Based Statistical MT

Phrase-based Statistical Machine Translation (PB-SMT) (Marcu & Wong, 2002; Koehn *et al.*, 2003) is clearly the dominant paradigm in MT today. In this section, we take the reader through all the steps involved in developing a PB-SMT system, from gathering training resources, through pre-processing, runtime application and post-processing.

2.1 Pre-Processing

Notwithstanding the particulars of the approach taken, the developer of any corpus-based system will be confronted with the following stages of development prior to running the system: corpus collection and clean-up, and system training (i.e. word- and phrase-alignment, and parameter tuning). We describe each of these steps in the following sections.

Data

A prerequisite for the training of a data-driven MT system is a parallel corpus of sentences and their translations aligned at sentence level. In the simplest case, the 'source' side of the bitext are the original sentences, and the 'target' side consists of the translations of those sentences. However, it is quite often the case that either some texts may have been translated from language A to language B and others the other way round, or more than two languages are involved and both parts were translated from one or several other languages (cf. (Ozdowska & Way, 2009) for an interesting investigation of the effect on translation quality of training SMT systems with such more or less appropriate sets of training data).

Of course, even in the simplest scenario above, the bitext can be used just as easily for translation from 'target' into the 'source' language; the system itself doesn't care. Given a text in language A, its translated counterpart version B and an SMT system translating from A to B, SMT training assumes A to be the source language and B to be the target language irrespective of the original translation direction or languages involved.

E1: Hon. members opposite scoff at the freeze suggested by this party; to them it is laughable.
F1: Les députés d'en face se moquent du gel que a propose notre parti.
F2: Pour eux, c'est une mesure risible.

Figure 2. A non-exact alignment

Moreover, given that the parallel corpus is assumed to be aligned at sentence level, sentence alignment is usually performed automatically prior to training. Examples of 1:1 and 1:2 alignments from the Canadian Hansards³ are given in Figures 1 and 2 (adapted from Arnold *et al.* (1994), p.203).

Creating and promoting resources (corpora and tools) is now a well-established tradition in the area of NLP in general, and in SMT in particular. This is done through linguistic data centres such as the Linguistic Data Consortium (LDC)⁴ or the Evaluations and Language resources Distribution Agency (ELDA),⁵ which allow broad access to resources of various kinds (parallel and monolingual corpora, tokenisers, segmentation tools, aligners, etc.) for a wide range of languages, in some cases in return for a licence. For example, the LDC provides data for two of the major MT evaluation shared tasks (cf. Section 2.5): NIST⁶ and IWSLT.⁷ On the other hand, some resources are also made freely available within MT-related projects such as EuroMatrix,⁸ or certain MT shared tasks such as WMT.⁹ WMT makes available to all participants a complete set of resources for state-of-the-art as well as advanced experiments in MT allowing for comparable results within a common framework.

SMT quality is strongly conditioned by the size of the training corpora, and further by the type and amount of resources used (linguistic tools, dictionaries, etc.). Systems are usually trained on several million words of data in order to achieve good translation quality. In this respect, the availability of corpora suitable for SMT mainly depends on two criteria: language pair, and domain (or genre) of texts. Large parallel corpora exist only for a limited number of language pairs. The richest languages in terms of corpora are those in which international institutions or governments are required to produce translations. Texts coming from such organisations are amongst the largest and most widely used corpora in MT, especially for European languages; this is the case for the Europarl corpus (Koehn, 2005),¹⁰ the JRC-Acquis,¹¹ and Canadian Hansards as far as number of covered

³ <http://www.isi.edu/natural-language/download/hansard/index.html>

⁴ <http://www ldc.upenn.edu/>

⁵ <http://www.elda.org/>

⁶ National Institute of Standards and Technology: <http://www.nist.gov/speech/tests/mt/>

⁷ International Workshop on Spoken Language Translation. For the 2008 edition see <http://www.slc.atr.jp/IWSLT2008/>.

⁸ <http://www.euromatrix.net/>

⁹ Workshop on Statistical Machine Translation. For the 2009 edition see <http://www.statmt.org/wmt09/>.

¹⁰ <http://www.iccs.inf.ed.ac.uk/~pkoehn/publications/europarl/>

¹¹ <http://langtech.jrc.it/JRC-Acquis.html>

languages and size are concerned. Parallel and monolingual corpora of variable yet sufficient size for MT also exist for languages of a particular political/economic interest such as Chinese, Arabic or Indian languages in combination with English, mostly consisting of news agency material.

Although the number and/or size of available parallel corpora is increasing, the scope remains somewhat limited in terms of languages and domains covered. Apart from the languages mentioned above, recent MT-related shared tasks featured language pairs with less abundant resources such as Japanese-to-English,¹² English-to-Inuktitut¹³ or Romanian-to-English.¹⁴ As these corpora mainly come from governments, international institutions or news agencies, they are rather open/general in terms of domain, even for Europarl, which is often considered to be a ‘sublanguage’, but is in fact extremely heterogeneous. By contrast, large specialised corpora suitable for MT remain rare.

Corpus Clean-Up, Segmentation and Tokenization

Corpora are usually not created with MT in mind, and so a number of issues need to be borne in mind before using them ‘as is’ for MT training.

The first thing to check is whether a special character encoding (e.g. UTF-8, the Unicode (Unicode Consortium, 2006) attempt to encode characters from *all* languages, as opposed to those supported only in ASCII (Institute, 1986)) is required for the translator output or by the linguistic tools used. In this case, if the encoding does not match that used in a particular corpus, an encoding conversion solves the problem (assuming the corpus is correctly encoded). Some characters reserved by the tools used must be protected. For example, the Moses decoder (Koehn *et al.*, 2007) stumbles over vertical bars (“|”) in the input. Filtering multiple and initial or ending white spaces makes the corpus cleaner and avoids processing errors at later stages.

The main issue of corpus pre-processing—*tokenization*—is the division of the sentences into tokens separated by a white space. In some languages (Latin script languages, Arabic, etc.) this division exists naturally in the form of words. In others, like Chinese or Japanese, word boundaries are not orthographically marked and the tokenization problem is distinct and more difficult (it is often called ‘segmentation’). When word boundaries are orthographically marked, the problem is reduced to determining when special signs such as punctuation marks should be considered as part of the word or not. This is the case, for example, in abbreviations or acronyms, but

¹² <http://iwslt07.itc.it/>

¹³ <http://www.cse.unt.edu/~rada/wpt05/>

¹⁴ <http://www.cse.unt.edu/~rada/wpt/>

not when acting as a punctuation mark (“Mr. Obama was elected president of the U.S.A.”). Most tokenizers are based on machine learning approaches, or use dictionaries of abbreviations and acronyms, for example.

Because the execution time of the training algorithms used in MT grows very fast as the input increases, very long sentences are often removed from the corpora. Sentence pairs having a very different number of source and target tokens usually correspond to an incorrect source-to-target mapping and may also be filtered.

Finally, for some languages, special pre-processing is appropriate. Examples include the separation of clitics in Spanish, and prefixes and suffixes in Arabic, which allows for a reduction in data sparseness. Grouping compound words (such as the head verb and its particle with German compound verbs) can help to make source and target language word order more similar, which facilitates subsequent processing.

Word Alignment

Word alignment, which determines the translational correspondences at word level given a bilingual corpus such as those just described, is a fundamental component in all SMT variants. A set of high-quality word alignments is essential for phrase-based SMT systems since the phrase extraction normally relies on word alignment.

The most common approach to word alignment is *generative models*, which view the translation (or alignment) process as the generation of a sentence (or word) in one language from another. Here we assume the generation of a target language sentence t_1^I from a source sentence s_1^J .¹⁵ The transformation from source to target language in the generative model may include word insertion or deletion, word reordering (‘distortion’), 1-to- n alignments (‘fertility’), and so on (cf. the ‘IBM models’ of Brown *et al.* (1993)). Depending on whether fertility is explicitly modelled or not, these generative models can be broadly classified into fertility-based versus non-fertility models.

The most widely used non-fertility models are HMM-based models. IBM model 1 and 2 are zero-order HMM models where a source position is firstly selected for each position in the target sentence, and a target

¹⁵ Newcomers to the field may be somewhat confused at differences between the notation used in this chapter and some of the primary sources noted here and in Section 7. It is much more common to use f_1^J (to be read as ‘foreign’) to indicate the source sentence, and e_1^I (‘English’) to represent the target sentence. At first sight, the use of such terms might be upsetting for non-English speakers, and betray to an extent the Anglocentric nature of our field, given that most translation in MT is into English. Instead, it might be more fruitful perhaps to think of them as simple mnemonics for the terms in the various equations used to describe (especially) statistical MT systems, cf. (1) and (3) below. In any case, here and in the rest of this Chapter, we will stick to the less-widely used (yet less emotive) terms s and t to indicate source and target respectively.

word is produced as the translation of the selected source word. In IBM model 1, the source position is selected uniformly, while in IBM model 2 the selection depends on the *target* position in question. The first-order HMM model of Vogel *et al.* (1996) refines the generative story by further assuming that the selection of a source position depends on the previously selected *source* position. In the context of SMT, the search for the best target translation t_1^I given a source sentence s_1^J is achieved in the *noisy-channel model* by maximising the conditional probability $P(t_1^I | s_1^J)$. Using a Bayesian transformation, this maximisation criterion can be reformulated as in (1):

$$P(s_1^J | t_1^I) P(t_1^I) \quad (1)$$

where $P(s_1^J | t_1^I)$ is the *translation model* and $P(t_1^I)$ is the *language model*.

The alignment a_1^J , which describes the mapping from a source word position j to a target position a_j , is introduced as a hidden variable in modelling the translation probability, as in (2):

$$P(s_1^J | t_1^I) = \sum_{a_1^J} P(s_1^J, a_1^J | t_1^I) \quad (2)$$

where the *alignment model* $P(s_1^J, a_1^J | t_1^I)$ can be decomposed in different ways to model the transformation from the source to the target language. However, non-fertility models are generally considered to be relatively weak models, mainly because of the simplicity of the generation process.

Fertility-based alignment models, most notably IBM models 3 and 4, are much more complicated by introducing fertility into the alignment model. These models first determine the source word fertility, i.e. how many target words each source word should generate, e.g. *not* \rightarrow *ne ... pas* would mean that *not* has a fertility of 2 (French words). For each source word, that many target words will be preferred as the translation of the source word. The model then arranges the hypothesised target words to produce a target string according to the *distortion models*. IBM model 3 utilises a zero-order distortion model, i.e. each target position is chosen independently for the target words generated by each source word, whereas IBM model 4 utilises a simplified first-order dependence (i.e. a context of the neighbouring previous word) in positioning the target words. However, both distortion models assign some probability to invalid target strings in order to achieve a more simplified approximation, resulting in the problem of ‘deficiency’, which is resolved in IBM model 5.

In next para, cross-ref with EM Chapter ...

The generative models described above consist of a large number of parameters which are normally estimated in an unsupervised manner (given that annotated data is difficult to obtain) using the Expectation-Maximisation (EM) algorithm (Dempster *et al.*, 1977) (cf. also (Manning & Schütze, 1999, p. 518f.)) on a

large bilingual corpus. There exist efficient training and searching algorithms for HMM models; however, we are unaware of any efficient algorithm for fertility-based IBM models. Consequently, such an approach can only be implemented by approximate hill-climbing methods, and parameter estimation can be very slow, memory-intensive and difficult to parallelise. Given this, Deng & Byrne (2005) proposed an HMM-based word-to-phrase alignment model which explores the desirable features in IBM fertility-based models while keeping the parameter estimation step tractable. Furthermore, previous generative models have also faced the criticism that they make unreasonable assumptions about word alignment structure, i.e. the 1-to- n assumption, meaning that each target word can be aligned to zero or more source words, but not vice versa. Such an asymmetric alignment structure cannot capture the pervasive m -to- n alignments in real world alignment tasks. Consequently, heuristics are needed to derive alignments from bidirectional word alignments in order to produce high-quality phrase pairs for phrase-based SMT (cf. Section 2.1) or translation rules for syntax-based SMT (cf. Section 3.1). Fraser & Marcu (2007a) attempted to address such a problem by proposing a new generative model capturing m -to- n alignment structures. In general, generative models have been shown to have powerful modelling capabilities and can produce high-quality alignments with successful application to various types of statistical (and other data-driven) MT systems. The most often used implementation of HMM models and IBM models 3, 4 and 5 is GIZA++¹⁶ (Och, 2003), and the MTTK¹⁷ (Deng & Byrne, 2006) implementation models HMM word-to-phrase alignments.

Discriminative word alignment models were developed with the specific intention of overcoming the shortcomings faced by generative models. Firstly, such models can incorporate various features encoded in the input data. Secondly, these models require only a relatively small amount of annotated word alignment data for training. Formally, an estimate \hat{a} of the optimal ('arg max' in (3), i.e. the highest score) alignment a is searched for by maximising a log-linear combination of a set of i features h_i , as in (3):

$$\hat{a} = \arg \max_a \sum_i \lambda_i h_i(s, a, t) \quad (3)$$

The parameters (or 'weights') λ_i can be learned in a supervised manner using various machine learning techniques, including perceptron (Moore, 2005), maximum entropy (Liu *et al.*, 2005; Ittycheriah & Roukos, 2005), Support Vector Machines (Taskar *et al.*, 2005; Cherry & Lin, 2006), and Conditional Random Fields (Blunsom & Cohn, 2006). Despite having the flexibility to incorporate various features, the need for a certain amount of annotated word alignment data is often put forward as a criticism of such approaches, given that the annotation

¹⁶ <http://www.fjoch.com/GIZA++.html>

¹⁷ <http://mi.eng.cam.ac.uk/~wjb31/distrib/mttkv1/>

of word alignments is a highly subjective task. Moreover, parameters optimised on manually annotated data are not necessarily optimal for MT tasks. (Fraser & Marcu, 2007b) showed that Alignment Error Rate (AER) (Och & Ney, 2000), the widely used metric to measure word alignment quality against manually annotated data, has a weak correlation with MT quality in terms of BLEU (Papineni *et al.*, 2002) in a PB-SMT system. Therefore, some approaches have been proposed to optimise the parameters according to the MT task rather than on annotated data (Lambert *et al.*, 2007). Some semi-supervised approaches have also been used to take advantages of both generative and discriminative approaches (Wu *et al.*, 2006; Fraser & Marcu, 2006). However, we have not yet seen a consistent discriminative word alignment model that can outperform generative models when used for SMT.

Another class of approaches to word alignment are *heuristics-based methods*, which obtain word alignment using similarity functions (Smadja *et al.*, 1996; Ker & Chang, 1997; Melamed, 2000). Such approaches are extremely simple compared to both generative and discriminative models. However, the use of similarity functions can be somewhat arbitrary and the performance of such methods is inferior compared to the above-mentioned statistical approaches (Och & Ney, 2003).

Phrase Alignment and Translation Models

Motivation for phrase-based models

Word-based SMT systems (e.g. (Germann, 2003)) learn lexical translation models describing one-to-one mappings between a given language pair. However, words are not the best atomic units of translation because we can have one-to-many mappings between languages. Furthermore, by translating word for word, no contextual information is made use of during the translation process. To attempt to overcome some of these issues, sequences of words can be translated together. By using these sequences of words, so-called ‘phrases’ (but not in the linguistic, ‘constituent’ sense of the word; a ‘phrase’ in SMT is any sequence of length n of contiguous words, hence ‘ n -grams’), it is possible to avoid many cases of translational ambiguity and better capture instances of local reordering. An example of this is illustrated in Figure 3.

The set of phrase pairs extracted from the bilingual parallel corpus constitutes the core translation model (*phrase table*, or *t(translation)-table*) of the phrase-based SMT system.



Figure 3. In the word-based translation on the left we see that the noun-adjective reordering into English is missed. On the right, the noun and adjective are translated as a single phrase and the correct ordering is modelled in the phrase-based translation.

Learning phrase-based translation models

There are a number of ways to extract a phrase table from a parallel corpus. We will describe the most common method here and refer the reader to Section 7 for alternative approaches. To learn the phrase translation model we first induce a word alignment between the sentence pairs in the parallel corpus, as described in Section 2.1. Then for each word-aligned sentence pair we extract the set of phrase pairs consistent with the word alignment.

A more formal definition of *consistency* is as follows: a phrase pair $(\tilde{s}|\tilde{t})$ is consistent with an alignment A , if all words s_1, \dots, s_n in \tilde{s} that have alignment points in A have these with words t_1, \dots, t_n in \tilde{t} and vice versa (Koehn, 2009).

We then estimate a probability distribution over the set of phrase pairs where the probability of a phrase pair $P(\tilde{s}|\tilde{t})$ is its relative frequency in the entire set of phrase pairs:

$$P(\tilde{s}|\tilde{t}) = \frac{\text{count}(\tilde{t}, \tilde{s})}{\sum_{\tilde{s}_i} \text{count}(\tilde{t}, \tilde{s}_i)} \quad (4)$$

This model is then included as a core factor in the log-linear model (cf. (3) and (10)).

Refined word alignments for phrase extraction

Both the quality and the quantity of the word alignments have a significant effect on the extracted phrase translation model. One might think that the better the word alignments the better the subsequently extracted phrases should be, but many studies have shown that an expected correlation between an intrinsic improvement in word and phrase alignment quality (as measured by AER, or precision, recall, and F-score) and an increase in performance on the extrinsic MT task (as calculated by BLEU, say) is by no means guaranteed (Liang *et al.*, 2006; Ma *et al.*, 2008). Vilar *et al.* (2006) show similar findings by optimising word alignment on BLEU, and reporting MT scores using F-score (i.e. the other way round, compared to Liang *et al.* (2006); Ma *et al.* (2008)). Zhang

et al. (2008) and Ma *et al.* (2009) also show that the correlation is weak when the intrinsic quality is measured with F-score.

As mentioned in Section 2.1, word alignment is a directional task, so when we align a source sentence to a target sentence, each target word can be aligned to one source word at most. This is undesirable as it may be correct in many instances to have a target word map to multiple source words. In order to overcome this problem we carry out *symmetrization* of the word alignments.

This process involves running the word alignment in both directions: source-to-target and target-to-source. We can then merge the the two sets of alignments by taking their union or the intersection. This process is illustrated in Figure 4. These alignments can be further refined by ‘growing’ additional alignment points (Och & Ney, 2003). For SMT a higher recall word alignment is preferred as it leads to fewer spurious additions to the phrase translation model. For this reason, the union of the two sets of alignments along with additional refinements is generally preferred. For other precision-based tasks, however, this may not be the case, and the union of word alignments will be chosen instead.

2.2 Reordering Models

Another important feature of phrase-based systems that we only mention briefly here is the *reordering model*. The problems posed by differences in the word order of languages naturally depends on the language pair at hand. For instance, between English and French, modelling short local movements (adjective-noun reordering, say) may suffice. However, for English and German, where long-range movement of verbs is common, such a model would be inadequate.

Many state-of-the-art systems (e.g. (Tillmann, 2004; Koehn *et al.*, 2007)) employ lexicalised reordering models in which the reorderings are conditioned directly on the phrases (or ‘blocks’). These models are learned synchronously with the phrase translation model. Each phrase pair in the lexicalised reordering model is assigned one of three orientations: monotone (*m*), swap (*s*) or discontinuous (*d*). The orientation is assigned based on the position of the phrase relative to other word alignments for the sentence pair. For example, in Figure 4, the phrase pair ⟨he,er⟩ has an alignment pointing to the top left, i.e. to the phrase pair ⟨that,dass⟩. Accordingly, this means that the orientation type of the phrase pair ⟨he,er⟩ is monotone, as the preceding English word aligns to the preceding German word. For a French-to-English phrase pair ⟨wine,vin⟩ in a translation *white wine* → *vin blanc*, there would be an alignment pointing to the top right, i.e. to the phrase pair ⟨white,blanc⟩. This indicates that there is evidence for a swap with the previous pair, indicating that by and large English adjective-noun sequences like *white wine* are mapped to noun-adjective sequences like *vin blanc* in French.

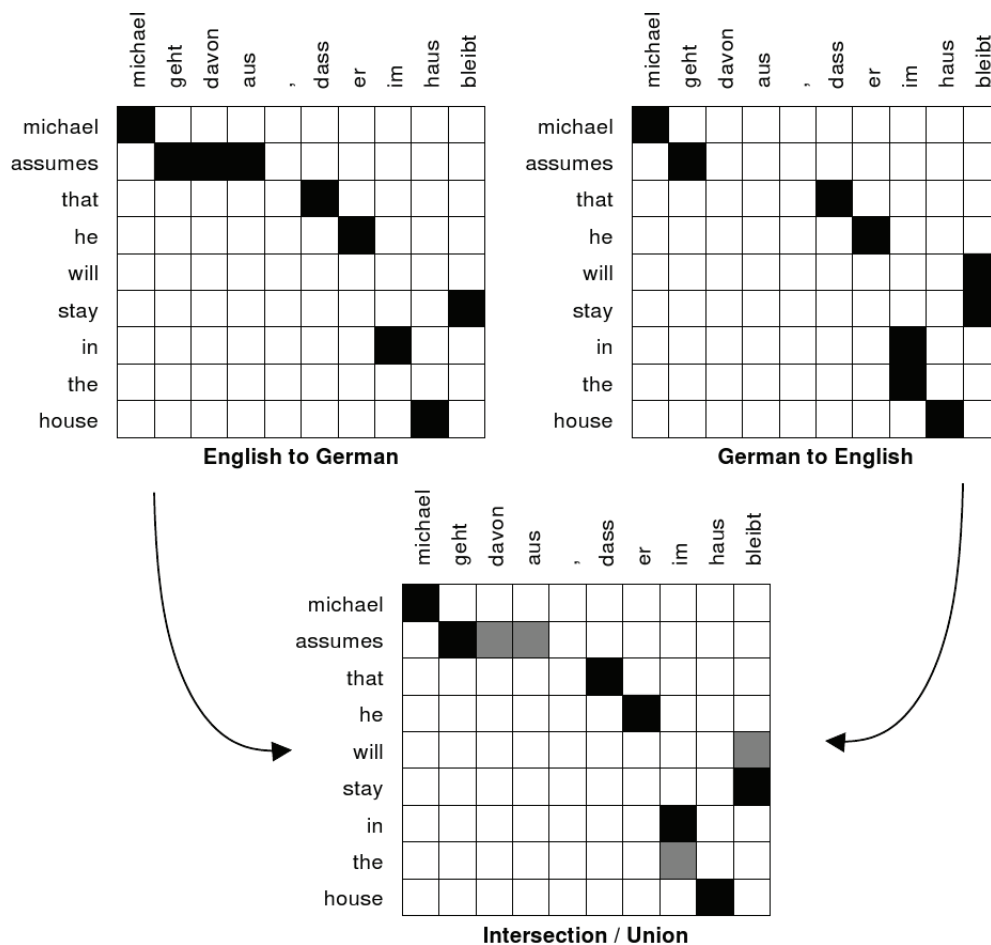


Figure 4. Merging source-to-target and target-to-source alignments (from Koehn (2009))

When a phrase pair is extracted for the translation model, its orientation for the reordering model is also extracted. A probability distribution p_o for the reordering model is then estimated based on the counts of how often specific phrase pairs occur with each of the three orientation types using the maximum likelihood (ML) principle (Manning & Schütze, 1999, p. 197), as in (5):

$$p_o(orientation|\tilde{f}, \tilde{e}) = \frac{count(orientation|\tilde{e}, \tilde{f})}{\sum_o count(o, \tilde{e}, \tilde{f})} \quad (5)$$

where an $orientation \in \{m, s, d\}$ is predicted for each source-to-target phrase pair for all possible orientations o .

Language Models

In the noisy-channel model of SMT (cf. (1)), $P(t)$ refers to the language model (LM), which is a probability distribution over target strings t that attempts to reflect the frequency with which each string t occurs as a sentence in text or speech. Especially in SMT, it can smooth and adjust the word orders to some extent by providing contextual information. In this section, we mainly focus on the n -gram LM which is used in most state-of-the-art SMT systems, as well as other data-driven models.

n -gram Language Model

In an n -gram LM, the probability $P(t)$ of a string t is expressed as the product of the probabilities of the words or tokens in t , with each word probability conditioned on a number of previous words. That is, if $t = \{w_1, w_2, \dots, w_l\}$ we have (6):

$$P(t) = P(w_1)P(w_2 | w_1)P(w_3 | w_1, w_2) \cdots P(w_l | w_1, \dots, w_{l-1}) \quad (6)$$

In typical usage, given the string t , the LM estimation using the above chain rule and an order-3 (i.e. trigram) or higher-order Markov assumption leads to (7):

$$P(t) = \prod_{i=1}^l P(w_i | w_1^{i-1}) \approx \prod_{i=1}^l P(w_i | w_{i-n+1}^{i-1}) \quad (7)$$

where w_i^j denotes the words w_i, \dots, w_j .

Consider the case $n = 3$. To estimate the probabilities $P(w_i | w_{i-2}, w_{i-1})$ in (7), a simple ML algorithm, as in (9), can estimate the approximate probabilities from the training data:

$$P(w_i | w_{i-2}, w_{i-1}) = \frac{P(w_{i-2}, w_{i-1}, w_i)}{P(w_{i-2}, w_{i-1})} \quad (8)$$

$$= \frac{\text{count}(w_{i-2}, w_{i-1}, w_i)}{(w_{i-2}, w_{i-1})} \quad (9)$$

Language Model Smoothing

Given the training data, it is easy to build an n -gram LM, because all we need to do is count the occurrences of the word n -gram events from the training data. However, the ML estimate does not perform well when the amount of training data is small or sparse compared to the size of the model being built. From the statistical point of view, if the training data cannot cover the test data (i.e. if a string α does not occur in the training data, but α

occurs in the test data), then a problem arises that a zero probability is generated, which is clearly inaccurate as this probability should be larger than zero. Accordingly, we need to estimate or predict the probability of events which were not seen in the training data.

ML estimates are based on the observations from the training data, so according to (9), unseen word n -grams will obtain a zero probability. Furthermore, according to (7), the sentence t will also receive a zero probability because of the products, which indicates that the sentence is not possible at all. Therefore, every sentence which contains n -grams which do not occur in the training data will be deemed impossible. As we pointed out in Section 2.1, in practice, the amount of training data available is limited, so data sparseness is often a real issue. Thus if we are unable to estimate the unseen n -gram sequences and give them an appropriate probability, it will have a fatal influence on many practical applications. Improving the model in (9) so that no word sequence receives zero probability is called *smoothing* (Jelinek, 1977). This process involves techniques for adjusting the ML estimate to hopefully produce more accurate probabilities.

The basic idea of smoothing techniques is to reserve some small probability mass from the relative frequency estimates (cf. (9)) of the probabilities of seen events, and to redistribute this probability to unseen events. There are several smoothing techniques which work fairly well for SMT and other applications. The main differences relate to how much probability mass is subtracted out ('discounting') and how it is redistributed ('back-off'). The most popular method is Kneser-Ney smoothing (Kneser & Ney, 1995).

2.3 Log-Linear Representation

As described in the previous sections, PB-SMT consists of three probabilistic components: a phrase translation model (TM), a reordering (distortion) model and the language model (LM). Och & Ney (2002) represent these probabilistic components as a log-linear model interpolating a set of feature functions as in (10):

$$t^* = \arg \max_t \prod_{f \in F} H_f(s, t)^{\lambda_f} \quad (10)$$

The set F is a finite set of features and λ_f are the interpolation weights over feature functions H_f of the aligned source-to-target sentence pairs s and t . The set of different features consists of the following:

- (1) An n -gram LM over target sequences,
- (2) A source-to-target t-table,
- (3) A target-to-source t-table (the reverse of the previous table),

- (4) Lexical translation probabilities in both directions,
- (5) A phrase reordering model,
- (6) The standard word/phrase penalty which allows for control over the length of the target sentence.

Minimum Error Rate Training

The parameters of each component of the log-linear model components are estimated independently. For example, the phrase translation probabilities are estimated from a bilingual corpus while the language model probabilities are estimated usually from a much larger monolingual corpus. The various components are interpolated in the log-linear framework by a set of parameters following the Maximum Entropy (MaxEnt) approach as shown in (10).

In the MaxEnt framework, each feature is associated with a weight. These weights can be estimated using iterative search methods to find a single optimal solution under the MaxEnt principle, but this is a computationally expensive process. Therefore, Och (2003) proposed an approximation technique called Minimum Error Rate Training (MERT) to estimate the model parameters for a small number of features, which will be discussed in Section 2.4. An error function that corresponds to the translation accuracy (Section 2.5) is defined and MERT estimates the log-linear model parameters such that this error function is minimized using the n -best output of the MT system. MERT proceeds as follows:

- (1) Initialize all parameters with random values.
- (2) Produce the n -best translations using the current parameter set.
- (3) Compute the error function using the reference translations.
- (4) Optimize each parameter to minimize the error function while fixing all other parameters.
- (5) Iterate over all parameters.

MERT provides a simple and efficient method to estimate the model parameters; however, it can only handle a small number of parameters, and when the number of parameters increases there is no guarantee that MERT will find the most suitable combination (Chiang *et al.*, 2008).

2.4 Decoding

At present, the state-of-the-art implementation of decoding for PB-SMT is a beam-search decoder (Koehn *et al.*, 2003). The decoder uses a log-linear model which is a MaxEnt (Jelinek, 1977) direct translation model. The

decoding process includes (i) the selection of translation options, (ii) future cost estimation, (iii) beam-search, and (iv) n -best list generation, all of which are explained in the following sections.

Translation Options Selection

Given an input string of words and a phrase table, only a certain number of phrases in the table are related to the input string, so we just need to collect these related phrases before decoding. This not only lowers the amount of memory required, but also increases decoding speed. During the selection, typically the following information is stored:

- (1) First and last source word covered,
- (2) Corresponding target phrase translation,
- (3) Phrase translation probability.

Given an input string of source words, all possible phrases with a limited span are found which are in accordance with the maximum length of the extracted phrase table. Then for each source phrase, the phrase table is searched and the matching target phrases stored.

Future Cost Estimation

In the decoding process, the target output sentence is generated left-to-right in the form of hypotheses which store the target phrase, translation cost and other related information. Each hypothesis is then stored in a stack which has the same source words covered. As shown in Figure 5, many possible segmentations for the source sentence along with many possible translations are available from the phrase table.

In order to reduce the search space (cf. Section 2.4 below), a breadth-first beam-search is used in decoding so that pruning is applied in a stack. In the pruning phase, not only the current translation cost but also the future cost is considered. The future cost is tied to the source words that have not yet been translated. Thus, we are looking for the cheapest cost (or the maximum probability) for the source words that are not yet covered. This future cost estimation should favour hypotheses that have already covered difficult parts of the sentence and have only easy parts left, while discounting hypotheses that have covered the easy parts first.¹⁸

¹⁸ The ‘ease’ or ‘difficulty’ associated with translating certain parts of a sentence is usually expressed in terms of weighted log probabilities which take into account (at least) language model, translation model and reordering costs. As you might

Maria	no	dio una bofetada	a	la	bruje	verde
-------	----	------------------	---	----	-------	-------

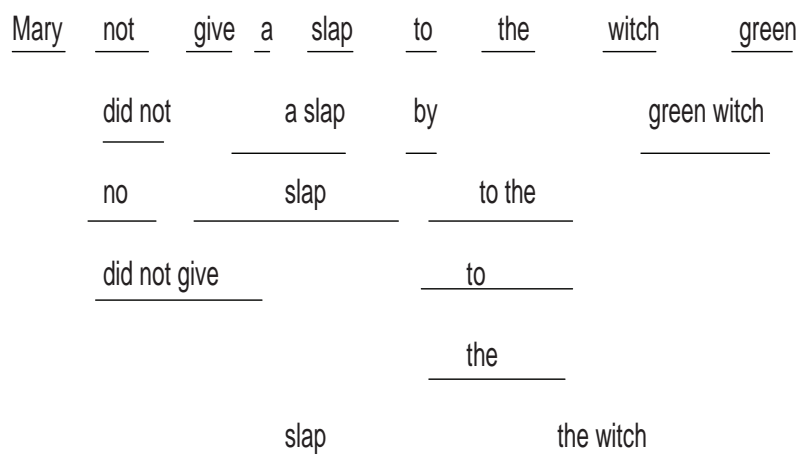


Figure 5. All possible source segmentations with all possible target translations (from (Koehn, 2004))

For the Translation Options in Section 2.4, each source phrase \tilde{s}_i^j has one or more target phrase candidates \tilde{t} , so the maximum probability for a source phrase \tilde{s}_i^j consisting of words i to j can be obtained by (11):

$$P(\hat{t} | \tilde{s}_i^j) = \arg \max \sum_m \lambda_m \log(p_m(\tilde{t}, \tilde{s})) \quad (11)$$

where $p_m(\tilde{t}, \tilde{s})$ is a product of the bidirectional phrase probabilities, bidirectional lexicalized probabilities, phrase length penalty and LM probability. Since we do not know the preceding target words for a translation operation, we approximate the LM cost by computing the LM score for the generated target words alone.

The future cost score for a source phrase can be efficiently estimated *a priori* by Dynamic Programming (Koehn, 2009), and simply looking up the score for this hypothesis in the cache. The lowest cost for any particular phrase will be the cheapest cost of a particular translation option, or the cheapest sum of costs from two smaller phrases that completely cover the phrase.

Beam Search

Typical phrase-based decoders like Moses (Koehn *et al.*, 2007) employ a beam-search algorithm. Starting from the initial hypothesis where no source input words have yet been translated, source words are then expanded in a monotone or non-monotone manner, i.e. following the source word/phrase order or not. New hypotheses

expect, common words are ‘easier’ to translate in this model than less frequent words, despite these being among the ‘hardest’ words to get right for humans.

can be generated from the expanded hypotheses with a phrasal translation that covers some of the source input words which have not yet been translated.

Each hypothesis is added into a beam stack as a new node, which is represented by:

- (1) a link back to the best previous state (needed for tracing the best translation of the sentence by backtracking through the search states),
- (2) the source words covered so far,
- (3) the last $n-1$ target words generated (if an n -gram-based LM is used),
- (4) the end of the last source phrase covered (needed for computing future distortion costs),
- (5) the most recently added target phrase,
- (6) the cost so far,
- (7) an estimate of the future cost,
- (8) feature functions (cf. Section 2.3),
- (9) additional arcs (needed for generating the n -best list).

The final states in the search are hypotheses that cover all source words. Among these hypotheses, the one with the lowest cost (highest probability) is selected as the best translation. If we want to output an n -best list, we can generate the translations with a ranked cost during the backtracking process. The hypothesis expansion process in a beam-search decoder is illustrated in Figure 6.

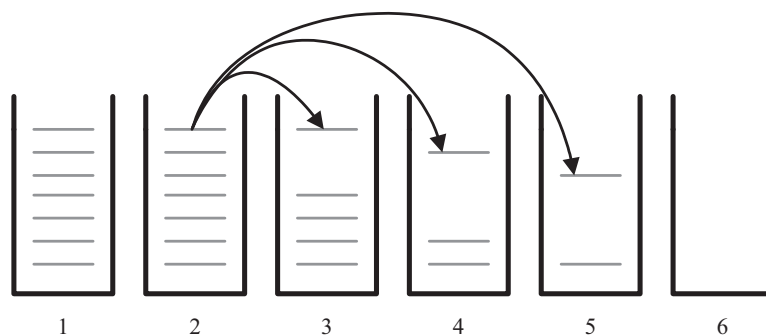


Figure 6. Hypothesis Expansion via Stack Decoding

In Figure 6, each stack is marked by the covered source words during expansion. A newly created hypothesis will be placed in a new stack further down, e.g. the top phrase in stack 2 (comprising 2 words, *the man*, say) is

linked to various hypotheses in stacks 3 (*goes*, i.e. 3 words are now covered), 4 (*does go*, 4 words), and 5 (*might be going*, 5 words).

In order to improve decoding speed and to reduce the search space, pruning techniques (such as recombining hypotheses, or histogram pruning (Koehn, 2009)) are employed to optimize the search by discarding hypotheses that cannot be part of the path to the best translation (i.e. they have a low score).

***n*-best List Generation**

After the expansion process, the final translation can be generated by backtracking. Generally, we just need one translation with the maximum highest probability as the final output, but in some cases such as MERT (Och, 2003) (cf. Section 2.3) or reranking (cf. Section 2.6), the *n*-best list will be needed. In typical approaches to phrase-based decoding, the A* algorithm is used to generate *n*-best lists (Koehn, 2009).

2.5 MT Evaluation

The constant development of MT systems using test sets of hundreds or thousands of sentences has meant that automatic MT evaluation metrics have become indispensable for quickly and cost-effectively rating candidate translations, and by extension the MT engines themselves. Some of the more widely used metrics include:

BLEU (Papineni *et al.*, 2002): a precision-based metric that compares a system's translation output against reference translations by summing over the 4-grams, trigrams, bigrams and unigram matches found, divided by the sum of those found in the reference translation set. It produces a score for the output translation of between 0 and 1. A higher score indicates a more accurate translation.

Sentence Error Rate (SER) : computes the percentage of incorrect full sentence matches by comparing the system's candidate translations against the reference translations. With all error rates, a lower percentage score indicates better candidate translations.

Word Error Rate (WER) (Levenshtein, 1966): computes the distance between the reference and candidate translations based on the number of insertions, substitutions and deletions in the words of the candidate translations divided by the number of correct reference words.

Position-independent Word Error Rate (PER) (Tillmann *et al.*, 1997): computes the same distance as the WER but without taking word order into account.

METEOR (Banerjee & Lavie, 2005): performs two stages of comparative matching for candidate and reference translations: (i) exact matching of unigrams, and (ii) stemmed matching, where remaining unmatched words

are decomposed into stems using the Porter stemmer and subsequently form matches. Stem matching and synonym matching are based on WordNet models (Miller *et al.*, 1990). Scores are obtained by calculating the sum of n -gram matches.

General Text Matcher (GTM) (Turian *et al.*, 2003): bases its evaluations on accuracy measures such as recall, precision, and F-score.

Dependency-based evaluation (Owczarzak *et al.*, 2007b): employs Lexical-Functional Grammar (LFG) (Kaplan & Bresnan, 1982; Bresnan, 2001) dependency triples using paraphrases derived from the test set through word/phrase alignment with BLEU and NIST (Doddington, 2002). It evaluates translations on a structural rather than string level and allows for lexical variance.

Automatic evaluation metrics are designed to assess linear text output, requiring the provision of at least one ‘gold standard’ version of the testing data as a reference for comparison. The majority, including BLEU, are string-based matching algorithms that do not take syntactic or lexical variation into account and penalise any divergence from the reference sentence(s). This can mean that candidate sentences which translate the source sentence both fluently and accurately, but have different lexical or syntactic choices to the reference sentence(s), may be given a low score. More recent developments, such as dependency-based evaluation, do allow for variance in lexical items (such as paraphrasing or synonyms), increasing the likelihood of a candidate sentence getting a good score.

While automatic evaluation best facilitates MT in terms of speed, human evaluation is often used as well. A panel of human evaluators with native knowledge of the target language can be asked to assess the output translations based on a prescribed set of criteria noting scales of fidelity and intelligibility, such as those outlined by Pierce *et al.* (1966).

In summary, both methodologies have their advantages, depending on whether the aim is speed of evaluation or a broader assessment of intelligibility and fidelity.

2.6 Reranking

SMT decoders may not find the best translation from the large number of candidate translation hypotheses. Reranking MT output is performed by obtaining the n -best translation candidates for each sentence using a baseline translation system. The candidates are reranked using features extracted from these n -best candidates to obtain a better translation than the one proposed by the decoder.

Generally, SMT rerankers train a discriminative model that can use features from the proposed n -best candidates to discriminate between the different translation candidates.

Och *et al.* (2004) used a large number of POS tags and syntactic features for reranking the n -best output of the baseline system using the log-linear model. Shen & Joshi (2005) used the best features from Och *et al.* (2004) to train a perceptron classifier for reranking the n -best list of candidate translations. Unlike these last two approaches, Yamada & Muslea (2006) trained the reranker on the entire corpus, not only on the test set.

In general, the improvements provided by reranking the SMT output are modest due to the fact that the number of translation candidates variations, even with a very large n -best list, is not enough to guarantee that a better translation will be obtained.

3 Other Approaches to MT

3.1 Hierarchical Models

In contrast to Koehn *et al.* (2003), who demonstrated that using syntax to constrain their phrase-based system actually harmed its quality, a number of researchers have, to different degrees, reported improvements when grammatical information is incorporated into their models of translation. We focus in the next few sections on perhaps the most popular alternative to the pure phrase-based approach, namely the hierarchical phrase-based model proposed by Chiang (2005).

Model

In general, given a source sentence s , a *synchronous CFG*¹⁹ will have many source-side derivations that yield (i.e. produce the sentence) s , and therefore many possible translations t on the target side. In hierarchical phrase-based MT, the model over derivations D (of the form $X \rightarrow \langle \textit{gamma}, \textit{alpha}, \textit{tilde} \rangle$, with X a non-terminal, γ strings of terminals, and α strings of non-terminals) is also defined as a log-linear model, as in (12):

$$P(D) \propto \prod_i \phi_i(D)^{\lambda_i} \quad (12)$$

where ϕ_i are features defined on derivations and λ_i are feature weights. In Chiang (2005), typical features used are $P(\gamma | \alpha)$, $P(\alpha | \gamma)$, lexical weights $P_w(\gamma | \alpha)$ and $P_w(\alpha | \gamma)$ (derived via word alignments), and a phrase penalty $\exp(1)$, where the system can learn preferences for longer or shorter derivations (cf. the phrase penalty in PB-SMT of Koehn *et al.* (2003) in Section 2.3).

For hierarchical phrase-based decoding, the integration of the LM is quite different compared to phrase-based decoding (cf. Section 3.1), so the LM is regarded as a special feature $P_{LM}(t)$ in the log-linear model, while the remainder of the features are defined as products of functions on the rules used in the derivation, as in (13):

¹⁹ Originally known as ‘syntax-directed transduction grammars’ (Lewis & Stearns, 1968) or ‘syntax-directed translation schemata’ (Aho & Ullman, 1969), ‘inversion transduction grammars’ (Wu, 1997) are a special case of synchronous CFGs, while a more recent terminological introduction is ‘2-multitext grammars’ (Melamed, 2003).

$$\phi_i(D) = \prod_{(X \rightarrow \langle \gamma, \alpha \rangle) \in D} \phi_i(X \rightarrow \langle \gamma, \alpha \rangle) \quad (13)$$

By merging (12) and (13), we end up with (14) as the model:

$$P(D) \propto P_{LM}(e)^{\lambda_{LM}} \times \prod_{i \neq LM} \prod_{(X \rightarrow \langle \gamma, \alpha \rangle) \in D} \phi_i(X \rightarrow \langle \gamma, \alpha \rangle) \quad (14)$$

That is, the weight of D is the product of the weights of the rules used in translation $(X \rightarrow \langle \gamma, \alpha \rangle) \in D$, the language model $P_{LM}(t)^{\lambda_{LM}}$, and any other functions ϕ_i such as the phrase penalty.

As Chiang (2005) notes, it is perhaps more convenient from a notational point of view to factor out the LM and word penalty probability models, although it is cleaner (and ensures polynomial-time complexity in decoding) to integrate them into the rule weights, in order to maintain the whole model as a weighted synchronous CFG.

Features

The basic features used in hierarchical phrase-based system are analogous to the default feature set of Pharaoh (Koehn, 2004) (cf. Section 2.3). The rules extracted from the training bitext have the following features:

- (1) $P(\gamma \mid \alpha)$ and $P(\alpha \mid \gamma)$, the bi-directional phrase/rule probabilities which are estimated by counting the frequency of rules;
- (2) the lexical weights $P_w(\gamma \mid \alpha)$ and $P_w(\alpha \mid \gamma)$, which estimate how well the words in α translate the words in γ (Koehn *et al.*, 2003);
- (3) a penalty \exp^{-1} for hierarchical rules, similar to the phrase penalty of (Koehn, 2003), which allows the model to learn a preference for longer or shorter derivations.
- (4) \exp^{-1} for the ‘glue rule’, so that the model can learn a preference for hierarchical phrases over serial combination of phrases;
- (5) \exp^{-1} for each of the four types of rules (numbers, dates, names, bylines);
- (6) a word penalty $\exp_{-count(T(\alpha))}$, where $count(T)$ is a count of terminals in the target sentence t .

Decoding

The decoder is a CKY parser (Younger, 1967) with beam search together with a postprocessor for mapping source derivations to target derivations. The parsing process starts with the axioms, and proceeds by applying inference rules to prove more items until a goal is proven. We refer the interested reader to (Chiang, 2007) for more details.

Incorporating the Language Model For hierarchical phrase-based MT, incorporating the LM is a challenging problem. Chiang proposed three solutions: first, using the above-mentioned parser to obtain an n -best list of translations and rescoring it with the LM; second, incorporating the LM directly into the grammar in a construction reminiscent of intersection of a CFG with a finite-state automaton; third, a hybrid method called *cube pruning*. In his experiments, the third method proved to be the most practical one which is a compromise and balances speed and accuracy. Again, we invite the reader to consult the primary sources for more on these possible solutions.

3.2 Tree-Based Models

Recently in the MT community many researchers have come to the realisation that in order to build good quality MT systems, new translation models need to be developed that are capable of handling complex source language syntactic and semantic representations, as well as their correspondences in the target language. This has led to the emergence of several models that employ syntactically parsed data to varying extents. In this section we will outline the most prominent developments.

Tree to String Models

Yamada & Knight (2001) present a tree-to-string model that adheres largely to the standard noisy-channel model of MT; the target-language sentence is produced after applying certain operations to the source-language sentence. Its main difference to the standard PB-SMT models is that it uses parsed data on the source-language side. The operations that this model encodes are the following:

Reorder , where the children of a node in the source-side parse tree may be reordered arbitrarily;

Insert , where a target-language word may be inserted at any position in the source-side tree; and

Translate , where the surface string of the source-side tree is translated word-for-word to obtain the target-language sentence. The tree structure is discarded after the translate operation.

The parameters of this model are the channel operations that can be performed and their probabilities for all available contexts. The values for these parameters are estimated automatically using the EM algorithm (Dempster *et al.*, 1977). Due to the vast number of possible contexts, the computation of all possible combinations of parameters is very expensive. Nevertheless, Yamada & Knight (2001) present an efficient algorithm that estimates the probabilities in polynomial time. Evaluation results are presented on automatic word alignments in which improvements in alignment average score are seen over a baseline IBM Model 5 system.

Unsupervised tree-to-tree models

Nesson *et al.* (2006) strive to develop an expressive and flexible formalism for MT that at the same time allows for efficient parsing. Thus they introduce Probabilistic Synchronous Tree-Insertion Grammar, which is an unsupervised tree-to-tree translation model.

In next two paras, cross-ref with Complexity Chapter ...

The basis for their formalism lies with Tree-Insertion Grammars (TIG) (Schabes & Waters, 1995). TIGs are a computationally attractive alternative to Tree-Adjoining Grammars (TAG) (Joshi, 1985) while continuing to use the same operations of substitution and adjunction. The main difference lies in additional restrictions on the form of elementary trees that TIG imposes. The restrictions limit the formalism to context-free expressivity and $O(n^3)$ parsability.

Synchronous TIG (STIG) extends the TIG formalism by using elementary structures consisting of pairs of TIG trees with links between particular nodes in those trees. Derivation for STIG proceeds as for TIG with the requirement that all operations have to be paired. An STIG can express lexically-based dependencies and can generally be parsed in $O(n^6)$ time.

Translation is performed using slightly modified inference rules that account for not having the target sentence during parsing. Having produced the possible derivation trees in this way it is trivial to generate the target-language sentences.

In next para, cross-ref with Complexity Chapter ...

The full model presented in (Nesson *et al.*, 2006) learns a probability for every combination of tree pairs in the training corpus. Thus, in a corpus with high word co-occurrence the number of free parameters will be of the order of $O(n^4)$, where n is the size of the largest monolingual vocabulary. This slows the model and may lead to overfitting of the training data. Therefore the authors propose to pre-process the word-co-occurrence data to eliminate word pairs that are unlikely to encode true relationships. This introduces another possible problem,

however, where too many word pairs could be pruned, thus rendering the model unable to parse some training sentence pairs.

By evaluating the model on a translation task, (Nesson *et al.*, 2006) show an improvement in BLEU and fluency scores over Pharaoh (Koehn, 2004) and GIZA++ (Och, 2003) systems trained on the same data, while achieving comparable adequacy scores.

Supervised tree-to-tree models

Data-Oriented Translation (DOT) is a hybrid model of translation which combines examples, linguistic information and a statistical translation model. The DOT model is specified in terms of (i) the type of representation expected in the example base; (ii) how fragments are to be extracted from these representations; (iii) how extracted fragments are to be recombined when analysing and translating input sentences; and (iv) how the resulting translations are to be ranked.

Tree-DOT (Hearne, 2005; Hearne & Way, 2006) (cf. also Section 5.2) was designed to utilise parallel treebanks, i.e. bilingual corpora annotated with syntactic structures for both the source and target side and with links between corresponding constituents in corresponding sentence pairs. From such a parallel treebank, linked subtree pairs can be extracted with associated probabilities. These subtree pairs can be used to analyse source-side sentences and construct compositionally corresponding target-side translations. An example is given in Figure 7.

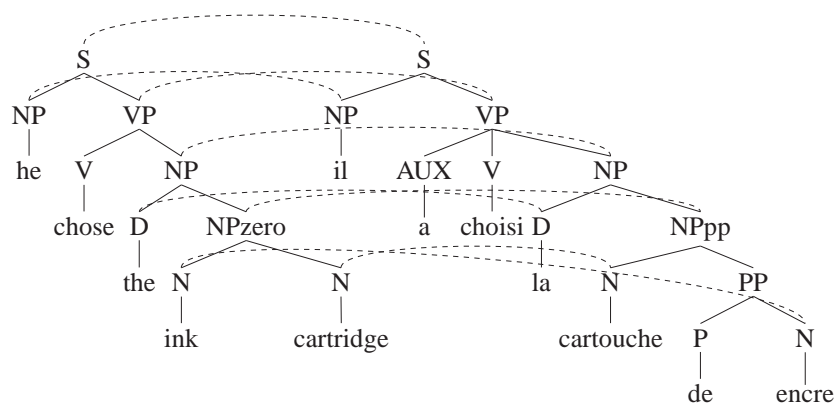


Figure 7. An aligned tree-pair in DOT for the sentence pair *he chose the ink cartridge, il a choisi la cartouche d'encre*

Tree-DOT standardly uses phrase structure trees as training data. Links between the constituents of two trees represent semantic/translational equivalence between these constituents. The translational equivalence relation

is reflexive, symmetric and transitive. For training, from all tree pairs in a parallel treebank a bag of all possible linked subtree pairs is created, where linked subtree pairs occur exactly as often as they can be identified in the parallel treebank. These subtree pairs can be composed together to produce analyses of complete sentence pairs.

For translation, the source-language sentence is analysed, whereby all possible derivations for the sentence are generated using linked subtree pairs. The correspondences in the subtree pair fragments can be used to generate target-language translations.

Supervised tree-to-tree and tree-to-string model

Hanneman *et al.* (2008) present a general framework for the development of search-based syntax-driven machine translation systems: Stat-XFER. This framework uses a declarative formalism for symbolic transfer grammars which consist of synchronous context-free rules that can additionally be augmented by unification-style feature constraints. These transfer rules specify the correspondences between phrase structures in the source and target languages.

The transfer formalism was designed considering the fact that the rules have to be simple enough so that they can be learned automatically, but also expressive enough to allow for manually-crafted rule additions and changes. The rules incorporate the following components ((Hanneman *et al.*, 2008) use ‘*x*-side’ to refer to the source language, and ‘*y*-side’ for the target language):

Type information identifies the type of transfer rule and generally corresponding to a syntactic constituent type. The formalism allows for the *x*- and *y*-side type information to be different.

POS/constituent information represents a linear sequence of components that constitute an instance of the rule type. These correspond logically to the right-hand sides of CFG rules for the *x*- and *y*-sides.

Alignments explicitly describe how the set of source-language components in a rule align and transfer to the set of target-language components. The formalism allows for both no and many-to-many alignments.

***x*-side constraints** apply to the source language and determine at run-time whether a transfer rule applies to a given sentence.

***y*-side constraints** apply to the target language and guide the generation of the target-language sentence.

***xy*-constraints** provide information about the feature values that transfer from the source to the target language.

`Cross-ref. to Parsing chapter in next para ...`

The transfer engine uses lexical transfer rules from a bilingual lexicon, while the higher-level structural rules can either be manually developed or automatically acquired. This engine fully integrates parsing, transfer and generation in a bottom-up “parse-and-transfer” algorithm that is essentially an extended chart parser (Kaplan, 1973; Kay, 1973). Parsing is performed using the source grammar, where x -side constraints are applied. Then the transfer rules are used to generate the target language side, constrained by the target grammar (where y -side and xy constraints are enforced).

3.3 Example-Based Machine Translation

Especially since the introduction of PB-SMT (Marcu & Wong, 2002; Koehn *et al.*, 2003), there has been a strong convergence between the leading corpus-based approaches to MT. As we stated in Way & Gough (2005a), before PB-SMT was introduced, describing the differences used to be easy, as since its inception (Nagao, 1984) EBMT has sought to translate new texts by means of a range of sub-sentential data—both phrasal and lexical—stored in the system’s memory. Until quite recently, by contrast, SMT models of translation were based on the simple word alignment models of (Brown *et al.*, 1993). Now that SMT systems learn phrasal as well as lexical alignments, this has led to an unsurprising increase in translation quality compared to the IBM word-based models (Brown *et al.*, 1993) (cf. Section 2.1 above).

A very wide array of techniques are used in EBMT today (cf. (Carl & Way, 2003)). Nonetheless, it is widely accepted that there are three main stages in translating with an example-based model, namely:

Matching : searching for fragments of the source text in the reference corpus;

Alignment : identifying the corresponding translation fragments;

Recombination : composing these translation fragments into the appropriate target text.

Just like PB-SMT, EBMT is a dynamic, fully automatic translation process. All three of the above stages depend very heavily on the nature of the training examples in the system’s database. The initial *matching* process uses a distance-based metric to compare the input string against examples from the source side of the reference corpus. In EBMT, the ‘classical’ similarity measure is the use of a thesaurus to compute word similarity on the basis of meaning or usage (Nagao, 1984; Sato & Nagao, 1990; Sumita *et al.*, 1990; Furuse & Iida, 1992; Nomiya, 1992; Matsumoto & Kitamura, 2005). Other approaches calculate similarity based on the relative length and content of strings (Way & Gough, 2003). ‘Similar’ examples are searched for, and a cost is calculated

taking into account deletions, insertions and substitutions, e.g. a missing comma would be penalised less than a missing adjective.²⁰

Probably the biggest divergence in approach among different types of EBMT system can be seen in the second *alignment* (or *adaptation*) phase, which again depends largely on the nature of the examples used in the EBMT system. A rich diversity of models can be seen, for example:

- (1) pure string-pairs with no additional information (e.g. (Nagao, 1984; Somers *et al.*, 1994; Lepage & Denoual, 2005));
- (2) annotated constituency tree (from context-free phrase-structure grammars, (Chomsky, 1957)) pairs (e.g. (Hearne, 2005; Hearne & Way, 2006), cf. Sections 3.2 and 5.2);
- (3) dependency tree-pairs (e.g. (Watanabe, 1992; Menezes & Richardson, 2003));
- (4) LFG f-structure pairs (e.g. (Way, 2003));
- (5) tree-to-string systems (e.g. (Langlais & Gotti, 2006; Liu *et al.*, 2006));
- (6) generalized examples (e.g. (Brown, 1999; Cicekli & Güvenir, 2003; Way & Gough, 2003)).

Particularly in relation to generalized examples, EBMT has successfully integrated translation templates into their models, in a similar manner to rule-based approaches. It is fair to state that the use of generalized templates has not caught on anywhere near as much in PB-SMT as it has in EBMT, despite the well-received “Alignment Template” approach in in PB-SMT (Och & Ney, 2004), which mirrors quite closely the method of generalisation most widely used in EBMT.

While the third *recombination* stage also differs according to the nature of the examples used in the appropriate EBMT model, it is broadly similar to the *decoding* stage in SMT (cf. (Germann, 2003) for word-based models, and (Koehn, 2004; Koehn *et al.*, 2007) for phrase-based approaches, cf. Section 2.4). Indeed, many

²⁰ Although it is not described until Section 4.2, a quick comparison between EBMT and Translation Memory is apposite here. Although the latter is a translation tool as opposed to an MT system *per se*, the initial *matching* process is extremely similar in nature in both approaches. Where the examples in the EBMT system consist of (unannotated) text pairs, the matching process is identical. In Translation Memory systems such as *Trados* (<http://www.trados.com>), ‘fuzzy’ (i.e. non-exact) matches have an associated measure of similarity which can be put to good use by the translator in honing the search for higher precision (imposing a high threshold of fuzziness) or recall (lowering the threshold). Note that the second and third EBMT phases do not form part of any Translation Memory system; rather, the end-user (usually a qualified translator) selects the appropriate parts of each fuzzy match for manual combination into the appropriate target-language sentence.

systems which are called ‘example-based’ currently use Moses as their decoder, and more and more the term ‘recombination’ is being replaced by the PB-SMT term ‘decoding’.

3.4 Rule-Based Machine Translation

As mentioned in the introduction, the leading paradigm in published MT research is PB-SMT; however, most available commercial systems are rule-based MT (RBMT) systems. The main reason why RBMT systems are still being developed is that the vast bilingual and monolingual training corpora needed to build PB-SMT systems are not available for all language pairs. Furthermore, the translation errors produced by RBMT systems tend to have a more repetitive nature than those of a PB-SMT system,²¹ which may render RBMT systems more predictable and easier to post-edit by human translators.

It may be useful to offer a contrast between RBMT and corpus-based systems such as PB-SMT and EBMT. RBMT systems are deductive: they use rules, dictionaries, etc. explicitly coded in a computer-readable form by experts using knowledge *deduced* or derived from their linguistic knowledge. This process may involve *elicitation*, that is, making explicit the implicit knowledge of translators and linguists. In contrast, PB-SMT and EBMT systems are *inductive*; they use information *inferred* from sentence-aligned parallel texts.

However, this deductive RBMT knowledge is somewhat hidden in commercial products. As we said earlier, commercial MT is overwhelmingly dominated by the rule-based paradigm. Most commercial MT companies tend to withhold information about the inner workings of their products, to avoid compromising their competitiveness in a licence-based closed-software business model; therefore, papers describing real RBMT systems are somewhat scarce (cf. Section 7 for some examples). However, a moderate effort of *reverse engineering* (Forcada, 2001) using carefully prepared test sets may be easily used to reveal the strategies and rules used by these systems, with “incorrect” translations playing an important role in the extraction of this information.

While it may take some effort to see what ‘rules’ might be underpinning existing commercial systems, it is important to note that not all RBMT systems are closed. For example, the Logos system has been released

²¹ This can be easily demonstrated by trying some simple examples through Google Translate. For instance, the December 4, 2008 Spanish-to-English version gave the translation of the sentence *Me los regaló tu hermanastro* (lit. ‘To-me them gave-as-a-present your half-brother’, i.e. ‘Your half-brother gave them to me as a present’) as *I gave you the half-brother*, while *Me los regaló tu madre* is translated as *Your mother gave me*, and *Me los regaló tu hermano* is translated as *I am your brother the gift*; note that the three Spanish sentences only differ with respect to the noun acting as subject (*hermanastro*, *madre*, *hermano*). Similar examples can be found at <http://www.euromatrix.net/deliverables/deliverable61.pdf>.

as free/open-source software as “OpenLogos”,²² and there is also very active development around a free/open-source MT platform called Apertium (Armentano-Oller *et al.*, 2006),²³ mainly by private companies.

Despite such shifts, it remains the case that these open-source systems use technologies that have been around for decades: Apertium uses a classical partial syntactic-transfer architecture (also known as a ‘transformer’ architecture (Arnold *et al.*, 1994, ch.4)). The indirect strategy used by Logos is harder to characterize in terms of a standard architecture (Scott, 2003).

With respect to closed-source systems, one of the leaders is the Barcelona-based Translendum,²⁴ which may be seen as a modern version of Siemens-Nixdorf’s full syntactic-transfer METAL system (White, 1985). Systems such as Softissimo’s Reverso²⁵ uses a partial syntactic-transfer strategy, able to translate correctly *The senior expert’s large desk* or *The computer expert’s desk* but failing to translate a slightly more complex phrase such as *The senior computer expert’s large desk* because of lack of a suitable pattern to detect and translate it, revealing the application of shorter patterns.

RBMT systems (of the transformer and transfer kind) were designed in the seventies and eighties to run on mainframe computers. They were then ported to become slow desktop applications for personal computers in the nineties, and subsequently they have been run on high-performance web-based systems without changes in their basic design. The commercial nature of these products and the apparent lack of innovation may explain why it is hard to find papers describing new developments in RBMT, as compared to those in corpus-based MT.

3.5 Hybrid Methods

While we feel it is appropriate here to feature systems which espouse to exhibit some degree of hybridity, we should perhaps begin with a word of caution:

“Much current research in MT is neither based purely on linguistic knowledge nor on statistics, but includes some degree of hybridization. At AMTA 2004 and MT Summit 2005 just about all commercial MT developers also claimed to have hybrid systems. But is this mostly a good way to allow painting oneself into whatever paradigm that current ‘fashion’ suggests one should be?” [Cavalli-Sforza & Lavie, 2006], AMTA-06 Hybrid MT Panel Session)

²² <http://logos-os.dfki.de/>

²³ <http://www.apertium.org>

²⁴ <http://www.translendum.com>

²⁵ <http://www.reverso.net>

Accordingly, we make a distinction in what follows between serial system combination (or ‘multi-engine MT’) and truly integrated systems. In what follows, we assume that only the latter qualify for the label ‘hybrid’. Nonetheless, ROVER-like system combinations (Fiscus, 1997) are increasingly to be seen, especially in large-scale open MT evaluations, and we feature some examples below. In Section 5, we discuss the contributions of our own work in the context of hybridity in translation, so the interested reader should also look there for comparisons with the work cited in the current section.

Multi-Engine MT

The term ‘Multi-engine machine translation’ (MEMT) was first introduced by (Frederking & Nirenburg, 1994) in their Pangloss system. Broadly speaking, MEMT systems try to select the best output from a number of MT hypotheses generated by different systems, while leaving the individual hypotheses intact.

Alegria *et al.* (2008) report a hierarchical strategy to select the best output from three MT engines for Spanish-to-Basque translation. First they apply EBMT (if it covers the input), then SMT (if the confidence score is higher than a given threshold), and then RBMT. The best results were obtained by the combination of EBMT and SMT.

Mellebeek *et al.* (2006) report a technique in which they recursively decompose the input sentence into smaller chunks and produces a consensus translation by combining the best chunk translations, selected through majority voting, a trigram LM score and a confidence score assigned to each MT engine. This is a quite different approach to all the other methods presented here, which operate on the MT outputs for complete sentences.

van Zaanen & Somers (2005) report a language-independent “plug-and-play” MEMT system that constructs a consensus translation from the outputs of off-the-shelf MT systems, relying solely on a simple edit distance-based alignment of the translation hypotheses, with no training required.

The work of Paul *et al.* (2005a,b) presents a multi-engine hybrid approach to MT, making use of statistical models to generate the best possible output from various MT systems. When using an SMT model to select the best output from multiple initial hypotheses produced by a number of SMT and EBMT systems, Paul *et al.* (2005a) found that a PB-SMT system modelled on HMMs provided the best results.

Integrated Systems

Rosti *et al.* (2007) look at sentence-, phrase- and word-level system combinations exploiting information from *n*-best lists, system scores and target-to-source phrase alignments. Accordingly, it could be described as either MEMT or Integrated, but we choose to discuss it here rather than in the previous section.

Chen *et al.* (2007) describe an architecture that allows combining SMT with (one or more) RBMT system(s) in a multi-engine setup. It uses a variant of standard SMT technology to align translations from RBMT systems with the source text and incorporates phrases extracted from these alignments into the phrase table of the SMT system. In related work, Eisele *et al.* (2008) report on two hybrid architectures combining RBMT with SMT. In the first architecture, several existing RBMT engines are used in a multi-engine set-up to enrich the lexical resources (phrase table) available to the SMT decoder, which combines the best expressions proposed by different engines. The modified phrase table combines statistically extracted phrase pairs with phrase pairs generated by linguistic rules. The second architecture uses lexical entries found using a combination of SMT technology together with shallow linguistic processing and manual validation, to extend the lexicon of the RBMT engine.

Seneff *et al.* (2006) exploit techniques to combine an interlingual MT system with phrase-based statistical methods, for translation from Chinese into English.

Bangalore *et al.* (2001) also use insights from post-editing to compute a consensus translation via majority voting from several translation hypotheses encoded in a confusion network. However, since edit-distance only focuses on insertions, deletions and substitutions, the model is unable to handle translation hypotheses with significantly different word orders. Jayaraman & Lavie (2005) try to overcome this problem by allowing non-monotone alignments of words in different translation hypotheses for the same sentence. They use a basic edit-distance (Levenshtein, 1966) that ignores case and which uses a stemmer to increase the number of matches.

Matusov *et al.* (2006) compute the consensus translation by voting on a confusion network (Mangu *et al.*, 2000; Hakkani-Tür & Riccardi, 2003) constructed from pair-wise word alignments of the multiple hypotheses to explicitly capture word reordering.

4 MT Applications

Advances in MT have meant that translation quality is now good enough to facilitate the needs of the general public with online MT systems (Section 4.1), assist human translators through the development of translation memory systems (Section 4.2), and help address specific problems such as inter-cultural communication (Section 4.4). It can also be combined with other NLP technologies (Section 4.3).

4.1 Online MT Systems

Consistent development of MT technology and the increasing need for translation at great speed with little cost has fuelled the proliferation of online MT systems such as Systran,²⁶ Google Translate,²⁷ Babelfish²⁸ and Windows Live Translator.²⁹ These systems predominantly offer their services free-of-charge as part of a web-based platform. They provide real-time translation to the general public through web-based platforms that allow users to type sentences, paragraphs of text or URLs for almost instantaneous translation into their chosen language. Although online MT systems may not be the best choice for highly accurate, large-scale, domain-specific translation, they adequately serve the small-scale, open-domain translation needs of the general public—as can be seen by the millions of hits per day that such sites receive—where the need for *gisting* (i.e. access to the basic information contained in the document) is greater than a perfect translation.

4.2 Translation Memory Tools

Translation Memories (Garcia, 2007; Biçici & Dymetman, 2008) comprise bilingual corpora of previously translated phrases usually within a particular domain. Translation Memory tools are used to assist human translators, and as well as the memories themselves, contain glossary and terminology management components, alignment technology, pre-translate functions, etc. Input phrases, or phrases selected using a computer-assisted translation tool, are compared against the corpus and a set of relevant target language sentences are produced for the translator to select appropriate parts from each to combine together to produce the output translation (cf. Section 3.3 for a comparison with EBMT).

²⁶ <http://www.systran.co.uk>

²⁷ <http://translate.google.com>

²⁸ <http://babelfish.yahoo.com>

²⁹ <http://www.windowslivetranslator.com>

4.3 Spoken Language Translation

As MT technology has developed, the range of use scenarios has increased particularly with respect to combining approaches with other NLP technologies. Coupling MT and speech technology, for example, particularly facilitates communication when text input is not convenient or where literacy skills impede such usage. For instance, the “Phraselator”³⁰ used by the US military is a handheld speech-to-speech translation system that aids communication where one party does not speak English, without the need for an interpreter or literacy skills. Such technology also bypasses the need for both parties to be able to operate the device, which may speed up the language exchange in time-critical situations. A further example of this is the role of MT in healthcare for patients with limited English (Somers, 2007). MT combined with speech recognition and synthesis can play an important role in safety-critical situations such as doctor-patient communication where patients are vulnerable, and may have little English or literacy skills.

4.4 Sign Languages

MT can also be a valuable tool to bridge the cross-modal communication gap between spoken and signed languages. Although research in this area is still relatively novel compared to mainstream spoken language MT, it has gained ground over the decade of its development with work in both rule-based (e.g. (Veale *et al.*, 1998)) and more recently data-driven approaches ((Morrissey *et al.*, 2007)). Where language barriers exist, person-to-person communication usually requires one or the other party to break from using their native language, something which may not be possible for either party in the context of Deaf-hearing communication. In this context MT can act as a useful substitute, and help maintain confidentiality in situations such as doctor-patient scenarios which are currently compromised by the use of teletype phones and human interpreters.

³⁰ <http://www.voxtec.com/phraselator>

5 Machine Translation at DCU

The MT group³¹ at DCU initially carried out research on EBMT (Carl & Way, 2003), and especially Marker-Based approaches (Way & Gough, 2003; Gough & Way, 2004; Way & Gough, 2005a; Gough, 2005). However, in the intervening period, we have worked on a very wide range of other areas of MT research and development, including:

- (1) Syntax-Driven Statistical Machine Translation (Hassan *et al.*, 2006, 2007b, 2008; van den Bosch *et al.*, 2007; Stroppa *et al.*, 2007; Haque *et al.*, 2009)
- (2) Hybrid Statistical & Example-Based Machine Translation (Way & Gough, 2005a; Groves & Way, 2005a,b; Groves, 2007)
- (3) Tree-Based Machine Translation (Hearne & Way, 2003; Hearne, 2005; Hearne & Way, 2006)
- (4) Word Alignment (Ma *et al.*, 2007a,b, 2008, 2009)
- (5) Sub-sentential Alignment for Machine Translation (Tinsley *et al.*, 2007a,b; Hearne *et al.*, 2008; Zhechev & Way, 2008)
- (6) Improvement of Rule-Based Machine Translation (Mellebeek *et al.*, 2006)
- (7) Evaluation in Machine Translation (Owczarzak *et al.*, 2007a,b; He & Way, 2009)
- (8) Controlled Language & Machine Translation (Way & Gough, 2004, 2005b)
- (9) Human Factors in Machine Translation (Morrissey *et al.*, 2007)

We will outline some of this work in the following sections.

5.1 Hybridity on the Source Side

Adding source-language context into PB-SMT

The DCU MATREX system (Stroppa & Way, 2006; Hassan *et al.*, 2007a; Tinsley *et al.*, 2008) uses Moses (Koehn *et al.*, 2007) as a backbone. In a different strand of work, a novel (albeit uncompetitive) decoder based on a memory-based classifier smoothed with a trigram LM is presented in van den Bosch *et al.* (2007). Contrast this with the work of Carpuat & Wu (2007), who use a pre-existing word-sense disambiguation tool to demonstrate improvements over an SMT baseline. Later work (Stroppa *et al.*, 2007) improves on the method of van den

³¹ <http://www.nclt.dcu.ie/mt/>

Bosch *et al.* (2007) by integrating a memory-based classifier as a kind of ‘pre-decoder’. It is demonstrated that a PB-SMT system using Moses improves significantly when context-informed features from the source language are used. We are able to (i) introduce context-informed features *directly* in the original log-linear framework (cf. (10) above), and (ii) still benefit from the existing training and optimization procedures of standard PB-SMT.

Essentially, we use two sets of context-informed features: Word-based features, and Class-based features. As far as the former are concerned, we can use a feature that includes the direct left- (s_{b_k-1}) and right-context (s_{j_k+1}) words of a given source phrase $\tilde{s}_k = s_{b_k} \dots s_{j_k}$ derived from a particular sentence pair s_1^K (consisting of words $1 \dots K$), as in (15):

$$h_m(s_1^J, t_1^I, s_1^K) = \sum_{k=1}^K \tilde{h}_m(\tilde{s}_k, s_{b_k-1}, s_{j_k+1}, \tilde{t}_k, s_k). \quad (15)$$

Here, the context is a window of size 3 (focus phrase + left context word + right context word), centred on the source phrase \tilde{f}_k . As in (10), \tilde{h}_m are the weights of the various features. Larger contexts may also be considered, so more generally, we have (16):

$$h_m(s_1^J, t_1^I, s_1^K) = \sum_{k=1}^K \tilde{h}_m(\tilde{s}_k, CI(\tilde{s}_k), \tilde{t}_k, s_k), \quad (16)$$

where $CI(\tilde{s}_k)$ denotes some contextual information (neighbouring words, phrases, Part-Of-Speech (POS)-tags etc.) about \tilde{s}_k .

In addition to the context words themselves, it is possible to exploit several knowledge sources characterizing the context. For example, we can consider the POS of the focus phrase and of the context words. In our model, the POS of a multi-word focus phrase is the concatenation of the POS tags of the words composing that phrase. Here, the context for a window of size 3 looks as in (17):

$$CI(\tilde{s}_k) = \langle POS(\tilde{s}_k), POS(s_{b_k-1}), POS(s_{j_k+1}) \rangle. \quad (17)$$

We can, of course, combine the class-based and the word-based information together if it leads to further improvements.

Essentially, the source context (words and/or POS-tag sequences) suggest target-language sequences for incorporation into the log-linear PB-SMT model. When testing on the Italian-to-English and Chinese-to-English IWSLT 06 data (Stroppa *et al.*, 2007), we found a consistent improvement for all metrics, for each type of contextual information: words-only, POS-only, and (for one of the language pairs) words+POS. Compared to

the baseline PB-SMT system, the significance of the improvements depended on the metric. Interestingly, the words+POS combination leads to a slight improvement for Italian-to-English, but not for Chinese-to-English (due to the poor quality of the Chinese POS-tagging).

This work is extended in Haque *et al.* (2009) to include supertags (cf. Section 5.3 below) as an additional, beneficial source-language contextual feature.

5.2 Hybridity in the Translation Phase

Comparing EBMT & Word-Based SMT

Rather surprisingly, until our work in Way & Gough (2005a), there had been *no* published comparative research between the respective merits of SMT and EBMT, largely due to (i) the relative unavailability of EBMT systems; (ii) the lack of participation of EBMT researchers in competitive evaluations; and (iii) the clear dominance of SMT.

In Way & Gough (2005a), on a 203K sentence-pair Translation Memory from *Sun Microsystems*, and on a 4K testset (average sentence length 13.1 words for English, 15.2 words for French) taken from the same collection, our EBMT system in Gough & Way (2004) outperformed a baseline word-based SMT system (Giza++ (Och, 2003), CMU-Cambridge statistical toolkit (Clarkson & Rosenfeld, 1997), ISI ReWrite Decoder (Germann *et al.*, 2001; Germann, 2003)) for both French-to-English and especially English-to-French, according to BLEU (Papineni *et al.*, 2002).

Combining EBMT & PB-SMT chunks

However, as PB-SMT had already been developed in Marcu & Wong (2002), it was clear that despite being of interest, the research in Way & Gough (2005a) was not an entirely fair comparison. Accordingly, in a range of papers, we conducted a variety of experiments to compare EBMT and PB-SMT, including:

- (1) Comparing EBMT and PB-SMT on *Sun Microsystems* Translation Memory data (Groves & Way, 2005a,b);
- (2) Combining EBMT and PB-SMT chunks (Groves & Way, 2005a,b);
- (3) Changing domain to Europarl (322K sent.) (Groves & Way, 2005a,b);
- (4) Different Language Pairs (Spanish-to-English) and more data (958K sent.) (Armstrong *et al.*, 2006);
- (5) Quite different language pairs (Basque-to-English, 273K sent.) (Stroppa *et al.*, 2006).

On the *Sun Microsystems* Translation Memory, our EBMT system outperformed the PB-SMT system. However, one interesting finding was that the PB-SMT system seeded in the usual way with Giza-data (cf. Section 2.1) outperforms a PB-SMT system built with EBMT-data. We also built a ‘semi-hybrid’ system consisting of EBMT phrases and Giza++ words, as well as a ‘fully hybrid’ system comprising Giza++ words and phrases and EBMT words and phrases.

Using the *Sun Microsystems* Translation Memory, we observed that the ‘semi-hybrid’ system (with a total of 430K entries in the t-table) performed significantly better than the same system seeded with EBMT data (403K entries) alone. This showed us that the Giza++ word lexicon was much better than the EBMT system’s, and henceforth we abandoned our EBMT word-level lexicon. Using *all* (i.e. Giza++ words and phrases and EBMT words and phrases) data (2.05M entries) improves the PB-SMT system, i.e. EBMT data improves the PB-SMT system, and for French-to-English, the fully hybrid ‘example-based PB-SMT’ system improves over the EBMT system, i.e. combining chunks from both systems improves over both the SMT and EBMT baselines.

On the *Europarl* data (Koehn, 2005), we observed, unsurprisingly, that doubling training data (78K, 156K, 322K) improves both EBMT and PB-SMT systems. This time, however, the PB-SMT system significantly outperforms our EBMT system. We put this down to the relative homogeneity (i.e. consistency of domain) of the *Sun Microsystems* Translation Memory compared to the heterogeneity of *Europarl*. Adding the Giza++ word lexicon improves the EBMT system a little, and the hybrid ‘statistical EBMT’ system seeded with all PB-SMT and EBMT data improves over the EBMT baseline. Adding the EBMT data to the hybrid ‘example-based PB-SMT’ system beats the baseline PB-SMT system, even when trained using only half the amount of data (156K vs. 322K) for French-to-English. For English-to-French, the hybrid PB-SMT system using 78K sentences of training data has almost the same performance as the baseline PB-SMT system trained on four times as much data (322K).

On other language pairs and corpora, we found that adding EBMT chunks to a baseline Pharaoh system (Koehn, 2004) adds 4 BLEU points for Spanish-to-English (Armstrong *et al.*, 2006) trained on nearly 1 million sentences of *Europarl* data. Furthermore, we showed that adding EBMT chunks to a baseline Pharaoh system adds 5 BLEU points for Basque-to-English (Stroppa *et al.*, 2006).

Adding statistical language models to EBMT

Groves & Way (2005a,b) showed that adding a statistical LM to their EBMT helps improve translation performance. However, unlike in PB-SMT, we did not integrate the target LM (cf. Section 2.2) *directly* into the EBMT

system, but rather used it only for EBMT reranking (cf. 2.6). Adding the target LM improves both the baseline and the hybrid ‘statistical EBMT’ systems (by 10% and 6–7% relative improvement in BLEU, respectively).

Tree-based translation

We have already described in Section 3.2 the basic system architecture of our DOT tree-to-tree MT system. One might be able to claim with some conviction that Tree-to-Tree translation (e.g. (Hearne, 2005; Hearne & Way, 2006) *is* hybrid MT, seeing as the DOT model includes examples (trees, in tree-DOT), source and target syntax (in the trees), rules (how the trees relate), and statistics (in the probability model) (cf. Figure 7).

There are two fragmentation operations in DOT which allow smaller, more general aligned tree pairs to be extracted from larger aligned tree pairs. The *root* operation selects a linked node pair to be root nodes and deletes all except these nodes, the subtrees they dominate and the links between them. The *frontier* operation selects a set of linked node pairs to be frontier nodes and deletes the subtrees they dominate.

The Tree-DOT composition operation (\circ) requires that tree fragments be composed at the leftmost site on the fragment’s source side, and at the target site *linked to* the leftmost source site. This ensures that each derivation is unique, and that translational equivalences encoded in the example base are respected (Way, 2003). An example derivation is given in Figure 8.

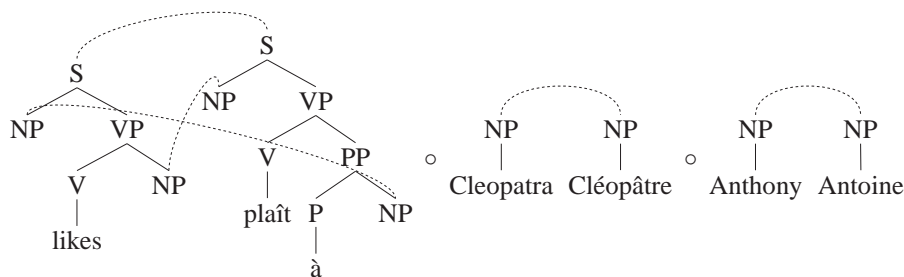


Figure 8. Composition in Tree-DOT

The probability model in DOT is a sum-of-products model, consisting of the probability of a fragment $\langle s_x, t_x \rangle$ (comprising a source fragment s_x and its translation t_x), the probability of a derivation D_x , the probability of a parse $\langle S_x, T_x \rangle$, and the probability of a source-to-target sentence pair s, t . Combined together, we derive the probability model in (18):

$$\sum_{\langle S_x, T_x \rangle \text{ yields } s, t} \sum_{D_x \text{ yields } \langle S_x, T_x \rangle} \prod_{\langle s_x, t_x \rangle \in D_x} \frac{|\langle s_x, t_x \rangle|}{\sum_{\text{root}(s)=\text{root}(s_x) \wedge \text{root}(t)=\text{root}(t_x)} |\langle s, t \rangle|} \quad (18)$$

As for disambiguation strategies, in Hearne & Way (2006) we compared a range of different techniques, including:

Most Probable Translation (MPT): the *most probable sequence of target terminals* given the input string;

Most Probable Parse (MPP): the sequence of target terminals read from the *most probable bilingual representation* for the input string;

Most Probable Derivation (MPD): the sequence of target terminals read from the *most probable derivation of a bilingual representation* for the input string;

Shortest Derivation (SDER): the sequence of target terminals read from the *shortest derivation of a bilingual representation* for the input string.

The first two of these were computed using *Monte Carlo Sampling* (Bod, 1998), while the latter two were calculated using the *Viterbi* algorithm (Viterbi, 1967).

Using the English-to-French section of the HomeCentre corpus, we split 810 parsed, sub-sententially aligned translation pairs into 12 training/test sets, 6 for English-to-French, and 6 for French-to-English. The splits were randomly produced such that all test words occurred in the training set, i.e. there were no OOV items.

One problematic issue with DOT models is grammar size. For our experiments, the grammar sizes are given in Table 1 (using the notion of “link depth” from (Hearne & Way, 2003)).

	link depth=1	depth≤2	depth≤3	depth≤4
English-to-French:	6,140	29,081	148,165	1,956,786
French-to-English:	6,197	29,355	150,460	2,012,632

Table 1. Number of fragments for English-to-French and French-to-English HomeCentre experiments

The full results for English-to-French and French-to-English in terms of exact match, BLEU and F-score, averaged over the splits, are given in in (Hearne & Way, 2006). In sum, the DOP Hypothesis (Bod, 1998) is confirmed for both language directions, i.e. as fragment depth increases, accuracy increases. For English-to-French, for all metrics and depths bar MPP at link depth 2, either MPD or SDER is preferred. Interestingly, MPT does not achieve highest accuracy at any depth for any metric, and overall, the highest performance is at

link depth 4 using MPD or SDER. For French-to-English, except for the BLEU score at link depth 3, the MPT scores best for both BLEU and F-score, whereas for exact match there are no significant trends to report.

As might be expected, execution time increases as link depth increases. However, the extra time required is spent building the translation space rather than disambiguating, and we note that translating from French takes longer because the average sentence length is longer. For English-to-French, we see that $SDER = MPD < MPP < MPT$, while for French-to-English, $MPT < SDER = MPD < MPP$. Interestingly, ranking with Monte Carlo sampling does not take longer than ranking with the Viterbi algorithm for this dataset.

One of the major remaining issues for us is scaling DOT to training sizes of at least two orders of magnitude larger than those used to date. Data acquisition has been a problem, which resulted in our building an automatic subtree aligner, described in Tinsley *et al.* (2007b). See also (Galron *et al.*, 2009) for a novel method of rescoring the DOT fragments with the evaluation metrics (cf. Section 2.5 above) used to measure the performance of the MT end-task in mind.

Augmenting PB-SMT with subtree pairs

Once we had developed our automatic subtree aligner (Tinsley *et al.*, 2007b), we incorporated subtree alignments into PB-SMT systems (Tinsley *et al.*, 2007a; Hearne *et al.*, 2008). The motivation for this work was the observation that most state-of-the-art MT systems (i) are not syntax-aware, (ii) use models which are based on n -grams, and (iii) incorporate only a limited amount of linguistic information.

Parallel treebanks are not widely used in MT, if at all. However, we believe that the data encoded within parallel treebanks could be useful in MT.³² In order to confirm this view, we built large parallel treebanks automatically, using off-the-shelf parsers and our subtree aligner, and then used these parallel treebanks to train a range of PB-SMT systems.

In (Tinsley *et al.*, 2007a), we used two data sets for two different language pairs. For English-to-German we used a small subset of Europarl data (Koehn, 2005), with a 9000:1000 sentence split for training and testing. The monolingual parsers used were (Bikel, 2002) for English, and BitPar (Schmid, 2004) for German (trained on the Tiger treebank). For English-to-Spanish we used a 4500:500 sentence split of Europarl data for training and testing. The parser of Bikel (2002) was again used for English, with a version of the same parser adapted by Chrupała & van Genabith (2006) (trained on the Cast3LB treebank (Civit & Martí, 2004)) used for Spanish.

³² Consult (Zhechev & Way, 2008) for how our subtree aligner can be used to automatically generate parallel treebanks, for any language for which constituency- or dependency-based parsers exist.

There were three main findings: (i) the parallel treebank word and phrase pairs improve translation quality when combined with traditional corpus-based extraction; (ii) the parallel treebank word pairs are better for translation than those given by traditional word alignment; but also (iii) that the parallel treebank phrase pairs are too few in number to be used alone for translation.

Nonetheless, just like the work of Groves & Way (2005a,b), this strand of work clearly demonstrates that retraining word- and phrase-extraction to one particular method will lead to sub-optimal performance.

In Hearne *et al.* (2008), the authors demonstrate that the subtree aligner of Tinsley *et al.* (2007b) can also be used to extract word- and phrase-pairs from dependency parses. In brief, the authors demonstrate that while both constituency- and dependency-based sets of alignments improved a baseline PB-SMT system, the combination caused system performance to deteriorate. Working out precisely why this is the case is the subject of ongoing work.

5.3 Hybridity on the Target Side

Incorporating supertags into PB-SMT

In Hassan *et al.* (2006, 2007b, 2008), we have shown that supertags (both CCG and LTAG) improve the performance of a state-of-the-art PB-SMT system on large data sets: for Arabic-to-English, on the NIST'05 data,³³ and for German-to-English, on the ACL 2007 MT Workshop shared task (WMT 2007) (Callison-Burch *et al.*, 2007).

Our approach can be described with respect to both the noisy-channel model (cf. (1)) as well as the log-linear model (cf. (3)). The noisy-channel formulation would extend equation (1) as in (19):

$$\begin{aligned}
 \arg \max_t \sum_{ST} P(s | t, ST) P_{ST}(t, ST) &\approx \\
 \arg \max_{t, ST} P(s | t, ST) P_{ST}(t, ST) &\approx \\
 \arg \max_{\sigma, t, ST} P(\phi_s | \phi_{t, ST}) P(O_s | O_t)^{\lambda_o} P_{ST}(t, ST) &\quad (19)
 \end{aligned}$$

where $P(\phi_s | \phi_{t, ST})$ is the translation model containing supertags on the target side, $P(O_s | O_t)^{\lambda_o}$ is the distortion model, and $P_{ST}(t, ST)$ is the target language model containing supertags. ST is the supertag sequence

³³ <http://www.nist.gov/speech/tests/mt/>

for the target string t . We use σ to indicate a segmentation into supertagged phrase pairs, just as in the baseline model.

We can also formalize our approach in terms of the log-linear model, as in (20):

$$t^* = \arg \max_{t, \sigma, ST} \prod_{f \in F'} H_f(s, t, \sigma, ST)^{\lambda_f} \quad (20)$$

Our model interpolates (log-linearly) a novel set of *supertagged features* f with the features of the baseline model F' . These include $H_{lm.st}(s, t, \sigma, ST) = P(ST)$, a Markov supertagging language model (hence *lm*) over sequences of supertags (hence *st*), as in (21):

$$P(ST) = \prod_{i=1}^n p(st_i | st_{i-4}^{i-1}) \quad (21)$$

We also use two weight functions $H_{\phi.st}(s, t, \sigma, ST) = P(\phi_s | \phi_{t,ST})$ and its reverse $H_{r_{\phi.st}}(s, t, \sigma, ST) = P(\phi_{t,ST} | \phi_s)$. The supertagged phrase translation probability is approximated in the usual (i.e. bidirectional) way:

$$P(\phi_s | \phi_{t,ST}) \approx \prod_{\langle s_i, t_i, ST_i \rangle \in (\phi_s \times \phi_{t,ST})} p(s_i | t_i, ST_i) \quad (22)$$

$$P(\phi_{t,ST} | \phi_s) \approx \prod_{\langle s_i, t_i, ST_i \rangle \in (\phi_s \times \phi_{t,ST})} p(t_i, ST_i | s_i) \quad (23)$$

In both (22) and (23), $\langle s_i, t_i, ST_i \rangle$ is a supertagged phrase pair consisting of the phrases $\langle s_i, t_i \rangle$ where t_i is supertagged with ST_i . As usual, the parameters $p(s | t, ST)$ and $p(t, ST | s)$ are estimated with the relative frequency in the multiset of all supertagged phrase pairs extracted from the parallel corpus, as in (24):

$$\begin{aligned} P(s | t, ST) &= \frac{\text{count}(s, t, ST)}{\sum_s \text{count}(s, t, ST)} \\ P(t, ST | s) &= \frac{\text{count}(s, t, ST)}{\sum_{t, ST} \text{count}(s, t, ST)} \end{aligned} \quad (24)$$

Finally, we employ two more feature functions ($x.\phi.st$ and $x.r\phi.st$) capturing the statistics $p(s_i | ST_i)$ and $P(ST_i | s_i)$, which in effect smooth the feature functions $\phi.st$ and $r\phi.st$.

In sum, incorporating supertags into PB-SMT demonstrates clearly that lexical syntax helps, for a number of reasons: (i) supertags fit seamlessly with PB-SMT as they are lexical, linguistically rich and can be used in efficient HMMs; (ii) supertags do not admit (much) redundant ambiguity into the phrase translation tables; (iii) the huge amount of baseline PB-SMT phrases are constrained using *bona fide* syntactic constraints; (iv) more informed decisions regarding the best candidate can be taken; and (v) there is no need for full parsing or treebanking.

If the reader needs any further persuasion that adding lexical syntax really helps, our Arabic-to-English system (Hassan *et al.*, 2007a) was ranked first at IWSLT-07 (Fordyce, 2007) according to human judges.

5.4 What works?

Given all the above, it might be useful to summarize what we have found to work well in practice.

As far as Incorporating Hybridity into EBMT is concerned, adding Giza++ lexical and phrasal chunks, and using target LMs for reranking have proven very effective.

Regarding the incorporation of hybridity into PB-SMT, adding EBMT lexical and phrasal chunks improves translation quality, reduces the t-table size for the hybrid system while continuing to compare favourably with much larger baseline PB-SMT systems. This may be important for language pairs with scarce resources, as well as situations where systems with a much smaller footprint are required. In addition, factoring in parallel treebank word and phrase pairs improves translation quality, as does incorporating supertags into the target LMs and target side of TM. Finally, adding source-language features directly into the log-linear model improves translation quality quite considerably.

5.5 Future Research Directions

Much of the above research is work in progress, and the intention is to continue to improve on the steps taken so far. Some of the issues to be tackled include:

- (1) combining the content-word generalized templates (*CMU*, in (25)) of Brown (1999) with our own marker-based generalized templates (*DCU*, in (26)):

$$CMU : Flights from < PLACE > to < PLACE > \quad (25)$$

DCU : *Flights* < PREP > *New York* < PREP > *Denver* (26)

- (2) incorporating a target LM *directly* into our EBMT system;
- (3) combining all source, target and translational improvements in *one* system.

In the context of the Centre for Next Generation Localisation (CNGL),³⁴ there are a number of open research avenues, including many of the issues raised here. However, other work packages address the development of probabilistic transfer engines, the tuning of MT systems to text type and genre, the development of general alignment models capable of inducing sub-sentential alignments for any type of annotated data, the incorporation of controlled language guidelines into the range of MT systems being developed in our team, and the development of intelligent engines for speech-to-speech translation. We continue to extend the range of language pairs that our systems can cope with (cf. English-to-Hindi (Srivastava *et al.*, 2008)), as well as participate in large-scale MT evaluation competitions.

³⁴ <http://www.cngl.ie>. The CNGL is a large 5-year project funded by the Irish Government involving 4 academic and 9 industrial partners.

6 Concluding Remarks and Future Directions

For a number of reasons, it can be said with some conviction that the field of MT currently finds itself in a quite good state of health:

- (1) There is evidence of increased levels of funding (especially in the US, Europe, and Asia);
- (2) MT is being used more widely than ever before;
- (3) More free and open-source tools are available to MT developers;
- (4) Large-scale MT evaluation competitions are attracting more and more systems, for an ever widening array of language pairs.

There exists, therefore, a real opportunity for our community to drive forward MT research and development to demonstrate clearly that good quality output can be achieved, which is useful to a wide array of potential users, both in industry and to the wider public.

Failure to do so may result in a return to the post-ALPAC report³⁵ (Pierce *et al.*, 1966) state of affairs where funding is cut—especially given the current economic environment—in favour of more fundamental requirements. Despite the wide variety of tools and techniques featured in this chapter, it remains the case that most MT research and development today is rather monolithic in the approaches taken, largely due to the availability of tools for PB-SMT. When it comes to purchasing MT systems, customers do not know what to buy. While MT evaluation metrics such as BLEU are well-understood by the research community, they do not provide any insight to potential users as to the effectiveness of such solutions, and bear little relation to the TM notion of “fuzzy match score” widely used in industry. When BLEU appeared in 2002, it was clear that it was more than capable of informing developers whether their systems had improved incrementally. Now, however, today’s research systems have overtaken the ability of the available MT evaluation metrics to discern the quality of the output translations. Accordingly, better MT evaluation metrics are needed, not just for MT developers, but also for potential users of our systems.

As well as improvements in MT evaluation, it is widely agreed that more linguistic knowledge can indeed play a role in improving today’s statistical systems, in all phases of the process. Syntax *is* of use in PB-SMT in the source, translation and target phases, as has been acknowledged for some time in RBMT and EBMT. Furthermore, it is recognised in the tree-to-string and string-to-tree models that having structure on one side helps, and in the near future we can expect to see large-scale, robust systems with trees on both sides.

³⁵ <http://www.nap.edu/books/ARC000005/html>

While there has clearly been a movement away from RBMT to statistical methods, now the pendulum is swinging back (slowly) in the opposite direction. We predict that, just like in the old rule-based times, the community will move further up the “Vauquois Pyramid” (Vauquois, 1968) and avail of more diverse sources of linguistic information; while syntax is useful, a new ceiling will be approached where further improvements will only be brought about by the use of *semantic* knowledge. As a final remark, note that this is not at all contrary to the original IBM models (Brown *et al.*, 1993), a fact that most of the MT community seems to have overlooked, if not forgotten entirely.

7 Further Reading

For *Sentential Alignment* (cf. Section 2.1), consult (Brown *et al.*, 1991; Gale & Church, 1993) for length-based algorithms (words and characters, respectively) and (Kay & Röscheisen, 1993) for a dictionary-based solution using ‘anchors’.

The primary sources on *Word Alignment* (cf. Section 2.1) are (Brown *et al.*, 1993) and (Och, 2003). For improvements to IBM model 1, consult (Moore, 2004), and (Toutanova *et al.*, 2002; Lopez & Resnik, 2005; Liang *et al.*, 2006) for extensions to the first-order HMM models. Other approaches include Inversion Transduction Grammar (Wu, 1997), which performs synchronous parsing on bilingual sentence pairs to establish translational correspondences, and the tree-to-string alignment model of Yamada & Knight (2001), which aligns a source tree to a target string. For an approach which bootstraps word alignments via optimising word segmentations, consult (Ma *et al.*, 2007b). With respect to investigations into the effect of balancing precision and recall on MT performance, (Mariño *et al.*, 2006) observed that an alignment with higher recall improved the performance of an *n*-gram-based SMT system, while (Ayan & Dorr, 2006) observed that higher precision alignments are more useful in phrase-based SMT systems, although this finding is not confirmed by Fraser & Marcu (2007b).

Regarding other methods of *Phrase Extraction* (cf. Section 2.1), (Marcu & Wong, 2002) describe a joint phrase model by which phrase pairs are estimated directly from the parallel corpus using the Expectation-Maximisation (EM) algorithm (Dempster *et al.*, 1977). Other proposed methods can be found in Tillmann & Xia (2003), (Ortiz-Martínez *et al.*, 2005) and (Zhang & Vogel, 2005), amongst others.

As for *Reordering* (cf. Section 2.2), the method of Galley & Manning (2008) differs from those of Tillmann (2004); Koehn *et al.* (2007) by estimating sequences of orientations directly from data, and by dynamically updating the segmentation of the source and target sentences with hierarchical phrases.

With respect to *Language Modelling* (cf. Section 2.2), the main sources are (Kneser & Ney, 1995; Jelinek, 1977), with more details to be found in Chen & Goodman (1998), especially for ‘modified’ Kneser-Ney smoothing, and Kim *et al.* (2001), on lowering the perplexity of the structured language model of Chelba & Jelinek (2000).

As far as *Minimum Error Rate Training* (MERT) is concerned (cf. Section 2.3), two novel papers which will benefit the reader are those of Moore & Quirk (2008), where trade-offs in terms of decoding and MERT time are considered, and Chiang *et al.* (2008), where alternative models are given in which a much larger number of features can be integrated.

For *Decoding* (cf. Section 2.4), the reader is directed towards the primary sources, namely (Koehn, 2004) and (Koehn *et al.*, 2007).

With respect to *Reranking* (cf. Section 2.6), useful sources include (Och *et al.*, 2004), (Shen & Joshi, 2005) (who use the best subset of features tested by (Och *et al.*, 2004)), and (Yamada & Muslea, 2006), who train their reranker on the whole training corpus, as opposed to just reranking on the test set.

If interested in *MT Evaluation* (cf. Section 2.5), consult the primary sources given in Section 2.5. A nice recent paper which we recommend is that of Hwa & Albrecht (2008).

For two quite different overview papers on *Statistical MT* (SMT), we recommend (Way, 2009a) for a critique of the paradigm, and (Hearne & Way, 2009), which explains *Phrase-Based SMT* (PB-SMT) for the non-expert.

The primary sources on *hierarchical phrase-based models* (cf. Section 3.1) are (Chiang, 2005, 2007). (Huang & Chiang, 2005) provides a valuable explanation of *cube pruning* (cf. Section 3.1).

For good summaries on *Example-Based MT* (EBMT) (cf. Section 3.3), we encourage the reader to consult (Somers, 1999, 2003b) and (Way, 2009b). The monograph by (Carl & Way, 2003) provides a representative sample of the myriad array of techniques used in EBMT.

Some examples of current research in *Rule-Based MT* (RBMT) (cf. Section 3.4) include (Probst *et al.*, 2002) and (Lavie *et al.*, 2004) on knowledge elicitation for under-resourced languages. (Font-Llitjós *et al.*, 2007) addresses the issue of rule refinement, while (Zhu & Wang, 2005) investigates the relationship between the number of rules and the performance of RBMT systems. (Menezes & Richardson, 2003; Caseli *et al.*, 2006; Sánchez-Martínez & Forcada, 2007) all focus on automatically obtaining some of the resources required for RBMT.

Good papers on *Hybrid Models* (cf. Section 3.5) include those of Tidhar & Küssner (2000); Callison-Burch & Flounoy (2001); Akiba *et al.* (2002). For a novel view on hybridity in MT, we encourage the reader to consult (Wu, 2005), where a 3-D space of hybrid models of translation is presented. Systems are categorised according to the extent to which they may be described as statistical vs. logical, example-based vs. schema-based, and compositional vs. lexical. Another novel paper is that of Simard *et al.* (2007), who present a combination of MT systems based on a post-editing strategy, in which the PB-SMT system Portage corrects the output of the Systran RBMT system.

Two good papers on *Translation Memory* (cf. Section 4.2) are those of (Planas & Furuse, 2003; Garcia, 2007), while the papers of Vogel & Ney (2000) and Marcu (2001) demonstrate how Translation Memories can be automatically extracted. (Carl & Hansen, 1999) show how Translation Memories can be integrated with

EBMT. A nice recent paper that shows how PB-SMT can upgrade Translation Memory *fuzzy matches* to classes that require less post-editing is that of Biçici & Dymetman (2008).

A recent paper on *Spoken Language Translation* (cf. Section 4.3) emanating from the TC-STAR project³⁶ is that of Fügen *et al.* (2007). One notable finding in TC-STAR was that today's leading PB-SMT systems are robust in the face of errors coming from the automatic speech recognition phase.

As regards our own work described in Section 5, the primary sources listed will provide the reader with further information on any of the topics of interest.

³⁶ <http://www.tc-star.org/>

Acknowledgements

The work described in this Chapter is partially funded by a number of Science Foundation Ireland (<http://www.sfi.ie>) awards, namely: Principal Investigator Award 05/IN/1732, Basic Research Award 05/RF/CMS064, and CSET Award 07/CE/I1142.

References

- Aho, Alfred & Jeffrey Ullman (1969), Syntax directed translations and the pushdown assembler, *Journal of Computer and System Sciences* 3:37–56.
- Akiba, Yasuhiro, Taro Watanabe, & Eiichiro Sumita (2002), Using language and translation models to select the best among outputs from multiple MT systems, in *Proceedings of 19th International Conference on Computational Linguistics (COLING 2002)*, Taipei, Taiwan, (1079–1085).
- Alegria, Iñaki, Arantza Casillas, Arantza Díaz de Ilarraza, Jon Igartua, Gorka Labaka, Mikel Lersundi, Aingeru Mayor, Kepa Sarasola, Xabier Saralegi, & Victor Laskurain (2008), Mixing approaches to MT for Basque: selecting the best output from RBMT, EBMT and SMT, in *MATMT 2008: Mixing Approaches to Machine Translation*, Donostia-San Sebastian, Spain, (27–34).
- Armentano-Oller, Carmen, Rafael Carrasco, Antonio Corbí-Bellot, Mikel Forcada, Mireia Ginestí-Rosell, Sergio Ortiz-Rojas, Juan Antonio Pérez-Ortiz, Gema Ramírez-Sánchez, Felipe Sánchez-Martínez, & Miriam Scalco (2006), Open-source Portuguese–Spanish machine translation, in *Computational Processing of the Portuguese Language, Proceedings of the 7th International Workshop on Computational Processing of Written and Spoken Portuguese, PROPOR 2006*, Springer-Verlag, Itatiaia, Brazil, volume 3960 of *Lecture Notes in Computer Science*, ISBN 3-540-34045-9, (50–59).
- Armstrong, Stephen, Marian Flanagan, Yvette Graham, Declan Groves, Bart Mellebeek, Sara Morrissey, Nicolas Stroppa, & Andy Way (2006), MaTrEx: Machine translation using examples, in *TC-STAR OpenLab on Speech Translation*, Trento, Italy, (not numbered).
- Arnold, Doug, Lorna Balkan, Lee Humphreys, Siety Meijer, & Louisa Sadler (1994), *Machine Translation: An Introductory Guide*, NCC Blackwell, Oxford, UK.
- Ayan, Necip Fazil & Bonnie Dorr (2006), Going beyond AER: An extensive analysis of word alignments and their impact on MT, in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia, (9–16).
- Banerjee, Satanjeev & Alon Lavie (2005), METEOR: An automatic metric for MT evaluation with improved correlation with human judgments, in *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, Ann Arbor, Michigan, (65–72).
- Bangalore, Srinivas, German Bordel, & Giuseppe Riccardi (2001), Computing consensus translation from multiple machine translation systems, in *Workshop on Automatic Speech Recognition and Understanding*, Madonna di Campiglio, Italy, (351–354).
- Biçici, Ergun & Marc Dymetman (2008), Dynamic translation memory: : Using statistical machine translation to improve translation memory fuzzy matches, in *Computational Linguistics and Intelligent Text Processing*, Springer Verlag, Berlin, Germany, volume 4919 of *Lecture Notes in Computer Science*, (454–465).
- Bikel, Daniel (2002), Design of a multi-lingual, parallel-processing statistical parsing engine, in *Proceedings of HLT 2002, Second International Conference on Human Language Technology Conference*, San Diego, CA, (178–182).

- Blunsom, Phil & Trevor Cohn (2006), Discriminative word alignment with conditional random fields, in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia, (65–72).
- Bod, Rens (1998), *Beyond Grammar: An Experience-Based Theory of Language*, CSLI, Stanford, CA.
- van den Bosch, Antal, Nicolas Stroppa, & Andy Way (2007), A memory-based classification approach to marker-based EBMT, in *Proceedings of the METIS-II Workshop on New Approaches to Machine Translation*, Leuven, Belgium, (63–72).
- Bresnan, Joan (2001), *Lexical-Functional Syntax*, Blackwell, Oxford.
- Brown, Peter, John Cocke, Stephen Della Pietra, Vincent Della Pietra, Fred Jelinek, John Lafferty, Robert Mercer, & Paul Roosin (1990), A statistical approach to machine translation, *Computational Linguistics* 16(2):79–85.
- Brown, Peter, Jennifer Lai, & Robert Mercer (1991), Aligning sentences in parallel corpora, in *29th Annual Meeting of the Association for Computational Linguistics*, Berkeley, CA, (169–176).
- Brown, Peter, Stephen Della Pietra, Vincent Della Pietra, & Robert Mercer (1993), The mathematics of statistical machine translation: Parameter estimation, *Computational Linguistics* 19(2):263–311.
- Brown, Ralf (1999), Adding linguistic knowledge to a lexical example-based translation system, in *Proceedings of the 8th International Conference on Theoretical and Methodological Issues in Machine Translation*, Chester, England, (22–32).
- Callison-Burch, Chris & Raymond Flounoy (2001), A program for automatically selecting the best output from multiple machine translation engines, in *MT Summit VIII: Machine Translation in the Information Age, Proceedings*, Santiago de Compostela, Spain, (63–66).
- Callison-Burch, Chris, Cameron Fordyce, Philipp Koehn, Christof Monz, & Josh Schroeder (2007), (Meta-)evaluation of machine translation, in *Proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic, (136–158).
- Carl, Michael & Silvia Hansen (1999), Linking translation memories with example-based machine translation, in *MT Summit VII “MT in the great translation era”*, Singapore, (617–624).
- Carl, Michael & Andy Way (eds.) (2003), *Recent Advances in Example-Based Machine Translation*, volume 21 of *Text, Speech and Language Technology*, Kluwer Academic Publishers, Dordrecht, The Netherlands.
- Carpuat, Marine & Dekai Wu (2007), Improving statistical machine translation using word sense disambiguation, in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, Prague, Czech Republic, (61–72).
- Caseli, Helena, Maria das Graças Nunes, & Mikel Forcada (2006), Automatic induction of bilingual resources from aligned parallel corpora: application to shallow-transfer machine translation, *Machine Translation* 20(4):227–245.
- Chelba, Ciprian & Fred Jelinek (2000), Structured language modeling, *Computer Speech and Language* 14(4):283–332.
- Chen, Stanley & Joshua Goodman (1998), An empirical study of smoothing techniques for language modeling, Technical Report TR-10-98, Harvard University.

- Chen, Yu, Andreas Eisele, Christian Federmann, Eva Hasler, Michael Jellinghaus, & Silke Theison (2007), Multi-engine machine translation with an open-source decoder for statistical machine translation, in *ACL 2007: Proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic, (193–196).
- Cherry, Colin & Dekang Lin (2006), Soft syntactic constraints for word alignment through discriminative training, in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia, (105–112).
- Chiang, David (2005), A Hierarchical Phrase-Based Model for Statistical Machine Translation, in *43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, Ann Arbor, MI, (263–270).
- Chiang, David (2007), Hierarchical Phrase-Based Translation, *Computational Linguistics* 33(2):201–228.
- Chiang, David, Yuval Marton, & Philip Resnik (2008), Online large-margin training of syntactic and structural translation features, in *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, Honolulu, HI, (224–233).
- Chomsky, Noam (1957), *Syntactic Structures*, Mouton, The Hague.
- Chrupała, Grzegorz & Josef van Genabith (2006), Using machine-learning to assign function labels to parser output for Spanish, in *44th Annual Meeting of the Association for Computational Linguistics (ACL'06)*, Sydney, Australia, (136–143).
- Cicekli, Ilyas & Altay Güvenir (2003), Learning translation templates from bilingual translation examples, in Michael Carl & Andy Way (eds.), *Recent Advances in Example-Based Machine Translation*, Kluwer Academic Publishers, Dordrecht, The Netherlands, (255–286).
- Civit, Montserrat & Maria Antònia Martí (2004), Building Cast3LB: A Spanish treebank, *Research on Language and Computation* 2(4):549–574.
- Clarkson, Philip & Roni Rosenfeld (1997), Statistical language modeling using the CMU-Cambridge toolkit, in *Proceedings of ESCA Eurospeech 1997*, Rhodes, Greece, (2707–2710).
- Dempster, Arthur, Nan Laird, & Donald Rubin (1977), Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society Series B* 39:1–38.
- Deng, Yonggang & William Byrne (2005), HMM word and phrase alignment for statistical machine translation, in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, Vancouver, Canada, (169–176).
- Deng, Yonggang & William Byrne (2006), MTTK: An alignment toolkit for statistical machine translation, in *Proceedings of the Human Language Technology Conference of the NAACL*, New York City, NY, (265–268).
- Doddington, George (2002), Automatic evaluation of MT quality using n-gram co-occurrence statistics, in *Proceedings of Human Language Technology Conference 2002*, San Diego, CA, (138–145).
- Eisele, Andreas, Christian Federmann, Hans Uszkoreit, Hervé Saint-Amand, Martin Kay, Michael Jellinghaus, Sabine Hunsicker, Teresa Herrmann, & Yu Chen (2008), Hybrid machine translation architectures within and beyond the euromatrix

- project, in *EAMT 2008: 12th annual conference of the European Association for Machine Translation*, Hamburg, Germany, (27–34).
- Fiscus, Jonathan (1997), A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER), in *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU'1997)*, Santa Barbara, CA., (347–354).
- Font-Llitjós, Ariadna, Jaime Carbonell, & Alon Lavie (2007), Improving transfer-based MT systems with automatic refinements, in *Proceedings of MT Summit XI*, Copenhagen, Denmark, (183–190).
- Forcada, Mikel L. (2001), Learning machine translation strategies using commercial systems: discovering word reordering rules, *Machine Translation Review* 12:13–18.
- Fordyce, Cameron (2007), Overview of the IWSLT07 evaluation campaign, in *Proceedings of the International Workshop on Spoken Language Translation*, Trento, Italy, (1–12).
- Fraser, Alexander & Daniel Marcu (2006), Semi-supervised training for statistical word alignment, in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Sydney, Australia, (769–776).
- Fraser, Alexander & Daniel Marcu (2007a), Getting the structure right for word alignment: LEAF, in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, Prague, Czech Republic, (51–60).
- Fraser, Alexander & Daniel Marcu (2007b), Measuring word alignment quality for statistical machine translation, *Computational Linguistics* 33(3):293–303.
- Frederking, Robert & Sergei Nirenburg (1994), Three heads are better than one, in *Proceedings of the Fourth International Conference on Applied Natural Language Processing (ANLP4)*, Stuttgart, Germany, (95–100).
- Fügen, Christian, Alex Waibel, & Muntsin Kolss (2007), Simultaneous translation of lectures and speeches, *Machine Translation* 21(4):209–252.
- Furuse, Osamu & Hitoshi Iida (1992), An example-based method for transfer-driven machine translation, in *Fourth International Conference on Theoretical and Methodological Issues in Machine Translation: Empiricist vs. Rationalist Methods in MT, TMI-92*, Montréal, Canada, (139–150).
- Gale, William & Ken Church (1993), A program for aligning sentences in bilingual corpora, *Computational Linguistics* 19(1):75–102.
- Galley, Michel & Christopher D. Manning (2008), A simple and effective hierarchical phrase reordering model, in *Proceedings of EMNLP 2008, Conference on Empirical Methods in Natural Language Processing*, Waikiki, HI., (848–856).
- Galron, Daniel, Sergio Penkale, & Andy Way (2009), Accuracy-based scoring for DOT: A step towards evaluation measure-based MT training, in *Proceedings of EMNLP 2009*, Singapore, (to appear).
- García, Ignacio (2007), Power shifts in web-based translation memory, *Machine Translation* 21(1):55–68.

- Germann, Ulrich (2003), Greedy decoding for statistical machine translation in almost linear time, in *HLT-NAACL: Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, Edmonton, AL, Canada, (72–79).
- Germann, Ulrich, Michael Jahr, Kevin Knight, Daniel Marcu, & Kenji Yamada (2001), Fast decoding and optimal decoding for machine translation, in *Association for Computational Linguistics, 39th Annual Meeting and 10th Conference of the European Chapter, Proceedings*, Toulouse, France, (228–235).
- Gough, Nano (2005), *Data-Oriented Models of Parsing and Translation*, Ph.D. thesis, Dublin City University, Dublin, Ireland.
- Gough, Nano & Andy Way (2004), Robust large-scale EBMT with marker-based segmentation, in *Proceedings of the 10th International Conference on Theoretical and Methodological Issues in Machine Translation*, Baltimore, MD, (95–104).
- Groves, Declan (2007), *Hybrid Data-Driven Models of Machine Translation*, Ph.D. thesis, Dublin City University, Dublin, Ireland.
- Groves, Declan & Andy Way (2005a), Hybrid data-driven models of machine translation, *Machine Translation* 19(3–4):301–323.
- Groves, Declan & Andy Way (2005b), Hybrid example-based SMT: the best of both worlds?, in *Proceedings of the Workshop on Building and Using Parallel Texts: Data-Driven Machine Translation and Beyond, ACL 2005*, Ann Arbor, MI, (183–190).
- Hakkani-Tür, Dilek & Giuseppe Riccardi (2003), A general algorithm for word graph matrix decomposition, in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Hong Kong, (596–599).
- Hanneman, Greg, Edmund Huber, Abhaya Agarwal, Vamshi Ambati, Alok Parlikar, Erik Peterson, & Alon Lavie (2008), Statistical transfer systems for French–English and German–English machine translation, in *Proceedings of the Third Workshop on Statistical Machine Translation at the 46th Meeting of the Association for Computational Linguistics (ACL-2008)*, Columbus, OH, (163–166).
- Haque, Rejwanul, Sudip Naskar, Yanjun Ma, & Andy Way (2009), Using supertags as source language context in SMT, in *Proceedings of EAMT-09, the 13th Annual Meeting of the European Association for Machine Translation*, Barcelona, Spain, (234–241).
- Hassan, Hany, Mary Hearne, Andy Way, & Khalil Sima'an (2006), Syntactic phrase-based statistical machine translation, in *Proceedings of the IEEE 2006 Workshop on Spoken Language Translation*, Palm Beach, Aruba, (no page numbers).
- Hassan, Hany, Yanjun Ma, & Andy Way (2007a), MATREX: the DCU machine translation system for IWSLT 2007, in *IWSLT 2007, Proceedings of the 4th International Workshop on Spoken Language Translation*, Trento, Italy, (69–75).
- Hassan, Hany, Khalil Sima'an, & Andy Way (2007b), Supertagged phrase-based statistical machine translation, in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, Prague, Czech Republic, (288–295).
- Hassan, Hany, Khalil Sima'an, & Andy Way (2008), Syntactically lexicalized phrase-based SMT, *IEEE Transactions on Audio, Speech and Language Processing* 16(7):1260–1273.

- He, Yifan & Andy Way (2009), Learning labelled dependencies in machine translation evaluation, in *Proceedings of EAMT-09, the 13th Annual Meeting of the European Association for Machine Translation*, Barcelona, Spain, (44–51).
- Hearne, Mary (2005), *Example-Based Machine Translation using the Marker Hypothesis*, Ph.D. thesis, Dublin City University, Dublin, Ireland.
- Hearne, Mary, Sylwia Ozdowska, & John Tinsley (2008), Comparing constituency and dependency representations for SMT phrase-extraction, in *15ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN '08)*, Avignon, France, (no page numbers).
- Hearne, Mary & Andy Way (2003), Seeing the wood for the trees: Data-oriented translation, in *Machine Translation Summit IX*, New Orleans, LA, (165–172).
- Hearne, Mary & Andy Way (2006), Disambiguation strategies for data-oriented translation, in *11th Annual Conference of the European Association for Machine Translation, Proceedings*, Oslo, Norway, (59–68).
- Hearne, Mary & Andy Way (2009), On the role of translations in state-of-the-art statistical machine translation, *COMPASS* :in press.
- Huang, Liang & David Chiang (2005), Better k-best parsing, in *Proceedings of the Ninth International Workshop on Parsing Technologies (IWPT)*, Vancouver, BC, Canada, (53–64).
- Hutchins, John (2003), Machine translation: General overview, in Ruslan Mitkov (ed.), *The Oxford Handbook of Computational Linguistics*, Oxford University Press, Oxford, UK, (501–511).
- Hwa, Rebecca & Josh Albrecht (2008), Regression for machine translation evaluation at the sentence level, *Machine Translation* 22(1–2):1–27.
- Institute, American National Standards (1986), *ANSI X3.4-1986. American National Standard for Information Systems — Coded Character Sets — 7-bit American National Standard Code for Information Interchange (7-bit ASCII)*, ANSI, New York.
- Ittycheriah, Abraham & Salim Roukos (2005), A maximum entropy word aligner for Arabic–English machine translation, in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, Vancouver, British Columbia, Canada, (89–96).
- Jayaraman, Shyamsundar & Alon Lavie (2005), Multi-engine machine translation guided by explicit word matching, in *Proceedings of the Tenth Conference of the European Association for Machine Translation (EAMT-05)*, Budapest, Hungary, (143–152).
- Jelinek, Fred (1977), *Statistical Methods for Speech Recognition*, MIT Press, Cambridge, MA.
- Joshi, Aravind (1985), *How Much Context-Sensitivity is Necessary for Characterizing Structural Descriptions—Tree-Adjoining Grammar*, Cambridge University Press, Cambridge, UK.
- Jurafsky, Daniel & James Martin (2008), *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Series in Artificial Intelligence, Prentice Hall, Upper Saddle River, NJ, 2nd edition.

- Kaplan, Ronald (1973), A general syntactic processor, in Randall Rustin (ed.), *Natural Language Processing*, Algorithmics Press, New York, NY., (193–241).
- Kaplan, Ronald & Joan Bresnan (1982), Lexical-functional grammar: A formal system for grammatical representation, in *The Mental Representation of Grammatical Relations*, MIT Press, Cambridge, MA, (173–281).
- Kay, Martin (1973), The MIND system, in Randall Rustin (ed.), *Natural Language Processing*, Algorithmics Press, New York, NY., (155–188).
- Kay, Martin & M. Röscheisen (1993), Text-translation alignment, *Computational Linguistics* 19(1):121–142.
- Ker, Sue & Jason Chang (1997), A class-based approach to word alignment, *Computational Linguistics* 23(2):313–343.
- Kim, Woosung, Sanjeev Khudanpur, & Jun Wu (2001), Smoothing issues in the structured language model, in *Proceedings of EuroSpeech 2001*, Aalborg, Denmark, (717–720).
- Kneser, Reinhard & Hermann Ney (1995), Improved backing-off for m-gram language modeling, in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Detroit, MI, volume 1, (181–184).
- Koehn, Philipp (2003), *Noun Phrase Translation*, Ph.D. thesis, University of Southern California, Los Angeles, CA.
- Koehn, Philipp (2004), Pharaoh: A beam search decoder for phrase-based statistical machine translation models, in *Proceedings of the 6th biennial conference of the Association for Machine Translation in the Americas*, Washington, DC, (115–124).
- Koehn, Philipp (2005), Europarl: A parallel corpus for statistical machine translation, in *Machine Translation Summit X*, Phuket, Thailand, (79–86).
- Koehn, Philipp (2009), *Statistical Machine Translation*, Cambridge University Press, Cambridge, UK.
- Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, & Evan Herbst (2007), Moses: Open source toolkit for statistical machine translation, in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, Prague, Czech Republic, (177–180).
- Koehn, Philipp, Franz Och, & Daniel Marcu (2003), Statistical phrase-based translation, in *HLT-NAACL: Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, Edmonton, AL, Canada, (127–133).
- Lambert, Patrik, Rafael Banchs, & Josep Crego (2007), Discriminative alignment training without annotated data for machine translation, in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, Rochester, NY, (85–88).
- Langlais, Philippe & Fabrizio Gotti (2006), EBMT by tree-phrasing, *Machine Translation* 20(1):1–23.
- Lavie, Alon, Katharina Probst, Erik Peterson, Stephan Vogel, Lori Levin, Ariadna Font-Llitjós, & Jaime Carbonell (2004), A trainable transfer-based machine translation approach for languages with limited resources, in *Proceedings of the Ninth EAMT Workshop, Broadening Horizons of Machine Translation and its Applications*, Valetta, Malta, (116–123).

- Lepage, Yves & Etienne Denoual (2005), Purest ever example-based machine translation: Detailed presentation and assessment, *Machine Translation* 19(3–4):251–282.
- Levenshtein, Vladimir (1966), Binary codes capable of correcting deletions, insertions, and reversals, *Soviet Physics Doklady* 10:707–710.
- Lewis, Philip & Richard Stearns (1968), Syntax-directed transduction, *Journal of the ACM* 15:465–488.
- Liang, Percy, Ben Taskar, & Dan Klein (2006), Alignment by agreement, in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, New York, NY, (104–111).
- Liu, Yang, Qun Liu, & Shouxun Lin (2005), Log-linear models for word alignment, in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, Ann Arbor, MI, (459–466).
- Liu, Zhanli, Haifeng Wang, & Hua Wu (2006), Example-based machine translation based on tree-string correspondence and statistical generation, *Machine Translation* 20(1):25–41.
- Lopez, Adam (2008), Tera-scale translation models via pattern matching, in *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, Manchester, UK, (505–512).
- Lopez, Adam & Philip Resnik (2005), Improved HMM alignment models for languages with scarce resources, in *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, Ann Arbor, MI, (83–86).
- Ma, Yanjun, Patrik Lambert, & Andy Way (2009), Tuning syntactically enhanced word alignment for statistical machine translation, in *Proceedings of the 13th Annual Meeting of the European Association for Machine Translation (EAMT 2009)*, Barcelona, Spain, (250–257).
- Ma, Yanjun, Sylwia Ozdowska, Yanli Sun, & Andy Way (2008), Improving word alignment using syntactic dependencies, in *Proceedings of the ACL-08: HLT Second Workshop on Syntax and Structure in Statistical Translation (SSST-2)*, Columbus, OH, (69–77).
- Ma, Yanjun, Nicolas Stroppa, & Andy Way (2007a), Alignment-guided chunking, in *Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation*, Skövde, Sweden, (114–121).
- Ma, Yanjun, Nicolas Stroppa, & Andy Way (2007b), Bootstrapping word alignment via word packing, in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, Prague, Czech Republic, (304–311).
- Mangu, Lidia, Eric Brill, & Andreas Stolcke (2000), Finding consensus in speech recognition: word error minimization and other applications of confusion networks, *Computer Speech and Language* 14(4):373–400.
- Manning, Christopher & Hinrich Schütze (1999), *Foundations of Statistical Natural Language Processing*, MIT Press, Cambridge, MA.
- Marcu, Daniel (2001), Towards a unified approach to memory- and statistical-based machine translation, in *Association for Computational Linguistics, 39th Annual Meeting and 10th Conference of the European Chapter, Proceedings*, Toulouse, France, (378–385).
- Marcu, Daniel & William Wong (2002), A phrase-based, joint probability model for statistical machine translation, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-02)*, Philadelphia, PA., (133–139).

- Mariño, José, Rafael Banchs, Josep Crego, Adrià de Gispert, Patrik Lambert, José Fonollosa, & Marta Costa-jussà (2006), N-gram-based machine translation, *Computational Linguistics* 32(4):527–549.
- Matsumoto, Yuji & Mihoko Kitamura (2005), Acquisition of translation rules from parallel corpora, in R. Mitkov & N. Nicolov (eds.), *Recent Advances in Natural Language Processing: Selected Papers from the Conference*, John Benjamins, Amsterdam, The Netherlands, (405–416).
- Matusov, Evgeny, Nicola Ueffing, & Hermann Ney (2006), Computing consensus translation from multiple machine translation systems using enhanced hypotheses alignment, in *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, Trento, Italy, (33–40).
- Melamed, Dan (2000), Models of translational equivalence among words, *Computational Linguistics* 26(2):221–249.
- Melamed, Dan (2003), Multitext grammars and synchronous parsers, in *HLT-NAACL: Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, Edmonton, AL, Canada, (79–86).
- Mellebeek, Bart, Karolina Owczarzak, Josef Van Genabith, & Andy Way (2006), Multi-engine machine translation by recursive sentence decomposition, in *AMTA 2006, Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, Visions for the Future of Machine Translation*, Cambridge, MA, (110–118).
- Menezes, Arul & Steve Richardson (2003), A best-first alignment algorithm for automatic extraction of transfer mappings from bilingual corpora, in Michael Carl & Andy Way (eds.), *Recent Advances in Example-Based Machine Translation*, Kluwer Academic Publishers, Dordrecht, The Netherlands, (421–442).
- Miller, George, Richard Beckwith, Christiane Fellbaum, Derek Gross, & Katherine Miller (1990), Introduction to wordnet: an on-line lexical database, *International Journal of Lexicography* 3(4):235–244.
- Moore, Robert (2004), Improving IBM Word Alignment Model 1, in *42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, Barcelona, Spain, (518–525).
- Moore, Robert (2005), A discriminative framework for bilingual word alignment, in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, Vancouver, BC, Canada, (81–88).
- Moore, Robert & Chris Quirk (2008), Random restarts in minimum error rate training for statistical machine translation, in *Coling 2008, The 22nd International Conference on Computational Linguistics, Proceedings*, Manchester, UK, (585–592).
- Morrissey, Sara, Andy Way, Daniel Stein, Jan Bungeroth, & Hermann Ney (2007), Combining Data-Driven MT Systems for Improved Sign Language Translation, in *Proceedings of Machine Translation Summit XI*, Copenhagen, Denmark, (329–336).
- Nagao, Makoto (1984), A framework of a mechanical translation between Japanese and English by analogy principle, in Alick Elithorn & Ranan Banerji (eds.), *Artificial and Human Intelligence*, North-Holland, Amsterdam, The Netherlands, (173–180).

- Nesson, Rebecca, Stuart Shieber, & Alexander Rush (2006), Induction of probabilistic synchronous tree-insertion grammars for machine translation, in *AMTA 2006, Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, Visions for the Future of Machine Translation*, Cambridge, MA, (128–137).
- Nomiyama, Hiroshi (1992), Machine translation by case generalization, in *Proceedings of the fifteenth [sic] International Conference on Computational Linguistics, COLING-92*, Nantes, France, (714–720).
- Och, Franz (2003), Minimum error rate training in statistical machine translation, in *41st Annual Meeting of the Association for Computational Linguistics*, Sapporo, Japan, (160–167).
- Och, Franz, Dan Gildea, Sanjeev Khudanpur, Anoop Sarkar, Kenji Yamada, Alex Fraser, Shankar Kumar, Libin Shen, David Smith, Katherine Eng, Viren Jain, Zhen Jin, & Dragomir Radev (2004), A smorgasbord of features for statistical machine translation, in *Proceedings Human-Language Technology and North American Association of Computational Linguistics (HLT-NAACL)*, Boston, MA, (161–168).
- Och, Franz & Hermann Ney (2000), A comparison of alignment models for statistical machine translation, in *Coling 2000 in Europe: the 18th International Conference on Computational Linguistics. Proceedings*, Saarbrücken, Germany, volume 2, (1086–1090).
- Och, Franz & Hermann Ney (2002), Discriminative training and maximum entropy models for statistical machine translation, in *40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA, (295–302).
- Och, Franz & Hermann Ney (2003), A systematic comparison of various statistical alignment models, *Computational Linguistics* 29(1):19–51.
- Och, Franz & Hermann Ney (2004), The alignment template approach to statistical machine translation, *Computational Linguistics* 30(4):417–449.
- Ortiz-Martínez, Daniel, Ismael Garcia-Varea, & Francisco Casacuberta (2005), Thot: A toolkit to train phrase-based models for statistical machine translation, in *Proceedings of Machine Translation Summit X*, Phuket, Thailand, (141–148).
- Owczarzak, Karolina, Josef van Genabith, & Andy Way (2007a), Evaluating Machine Translation with LFG Dependencies, *Machine Translation* 21(2):95–119.
- Owczarzak, Karolina, Josef van Genabith, & Andy Way (2007b), Labelled Dependencies in Machine Translation Evaluation, in *Proceedings of the 2nd Workshop on Statistical Machine Translation at the 45th Annual Meeting of the Association for Computational Linguistics (ACL-07)*, Prague, Czech Republic, (104–111).
- Ozdowska, Sylwia & Andy Way (2009), Optimal bilingual data for French-English PB-SMT, in *Proceedings of EAMT-09, the 13th Annual Meeting of the European Association for Machine Translation*, Barcelona, Spain, (96–103).
- Papineni, Kishore, Salim Roukos, Todd Ward, & Wei-Jing Zhu (2002), BLEU: a method for automatic evaluation of machine translation, in *40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA, (311–318).
- Paul, Michael, Takao Doi, Youngsook Hwang, Kenji Imamura, Hideo Okuma, & Eiichiro Sumita (2005a), Nobody is perfect: ATR's hybrid approach to spoken language translation, in *Proceedings of the International Workshop on Spoken Language Translation*, Pittsburgh, PA, (55–62).

- Paul, Michael, Eiichiro Sumita, & Seichi Yamamoto (2005b), A machine learning approach to hypotheses selection of greedy decoding for SMT, in *Proceedings of Second Workshop on Example-Based Machine Translation, MT Summit X*, Phuket, Thailand, (117–124).
- Pierce, John, John Carroll, Eric Hamp, David Hays, Charles Hockett, Anthony Oettinger, & Alan Perlis (1966), *Language and Machines: Computers in Translation and Linguistics*, Technical Report: Automatic Language Processing Committee, National Academy of Sciences, National Research Council, Washington, DC.
- Planas, Emmanuel & Osamu Furuse (2003), Formalizing translation memory, in Michael Carl & Andy Way (eds.), *Recent Advances in Example-Based Machine Translation*, Kluwer Academic Publishers, Dordrecht, The Netherlands, (157–188).
- Probst, Katharina, Lori Levin, Erik Peterson, Alon Lavie, & Jaime Carbonell (2002), MT for minority languages using elicitation-based learning of syntactic transfer rules, *Machine Translation* 17(4):245–270, ISSN 0922-6567.
- Rosti, Antti-Veikko, Bing Xiang, Spyros Matsoukas, Richard Schwartz, Necip Fazil Ayan, & Bonnie Dorr (2007), Combining outputs from multiple machine translation systems, in *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, Rochester, NY, (228–235).
- Sánchez-Martínez, Felipe & Mikel Forcada (2007), Automatic induction of shallow-transfer rules for open-source machine translation, in *Proceedings of the 11th Conference on Theoretical and Methodological Issues in Machine Translation (TMI 2007)*, Skövde, Sweden, (181–190).
- Sato, Satoshi & Makoto Nagao (1990), Toward memory-based translation, in *COLING-90, Papers Presented to the 13th International Conference on Computational Linguistics*, Helsinki, Finland, volume 3, (247–252).
- Schabes, Yves & Richard Waters (1995), Tree insertion grammar—a cubic-time, parsable formalism that lexicalizes context-free grammar without changing the trees produced, *Computational Linguistics* 21:479–513.
- Schmid, Helmut (2004), Efficient parsing of highly ambiguous context-free grammars with bit vectors, in *Coling, 20th International Conference on Computational Linguistics, Proceedings*, Geneva, Switzerland, (162–168).
- Scott, Bernard (2003), The Logos model: An historical perspective, *Machine Translation* 18(1):1–72.
- Seneff, Stephanie, Chao Wang, & John Lee (2006), Combining linguistic and statistical methods for bi-directional English Chinese translation in the flight domain, in *AMTA 2006, Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, Visions for the Future of Machine Translation*, Cambridge, MA, (213–222).
- Shen, Libin & Aravind Joshi (2005), Ranking and reranking with perceptron, *Machine Learning* 60(1–3):73–96.
- Simard, Michel, Nicola Ueffing, Pierre Isabelle, & Roland Kuhn (2007), Rule-based translation with statistical phrase-based post-editing, in *ACL 2007: Proceedings of the Second Workshop on Statistical Machine Translation*, Prague, Czech Republic, (203–206).
- Smadja, Frank, Kathleen McKeown, & Vasileios Hatzivassiloglou (1996), Translating collocations for bilingual lexicons: A statistical approach, *Computational Linguistics* 22(1):1–38.
- Somers, Harold (1999), Review article: Example-based machine translation, *Machine Translation* 14:113–157.

- Somers, Harold (2000), Machine translation, in Robert Dale, Hermann Moisl, & Harold Somers (eds.), *A Handbook of Natural Language Processing*, Marcel Dekker, New York, NY, (329–346).
- Somers, Harold (2003a), Machine translation: Latest developments, in Ruslan Mitkov (ed.), *The Oxford Handbook of Computational Linguistics*, Oxford University Press, Oxford, UK, (512–528).
- Somers, Harold (2003b), An overview of EBMT, in Michael Carl & Andy Way (eds.), *Recent Advances in Example-Based Machine Translation*, Kluwer Academic Publishers, Dordrecht, The Netherlands, (3–57).
- Somers, Harold (2007), Theoretical and Methodological Issues Regarding the Use of Language Technologies for Patients with Limited English Proficiency, in *Proceedings of the Eleventh Conference on Theoretical and Methodological Issues in Machine Translation (TMI-07)*, Skövde, Sweden, (206–213).
- Somers, Harold, Ian McLean, & Danny Jones (1994), Experiments in Multilingual Example-Based Generation, in *CSNLP 1994: 3rd International Conference on the Cognitive Science of Natural Language Processing*, Dublin, Ireland, (pages not numbered).
- Srivastava, Ankit, Rejwanul Haque, Sudip Naskar, & Andy Way (2008), MaTrEx: the DCU MT system for ICON 2008, in *Proceedings of the NLP Tools Contest: Statistical Machine Translation (English to Hindi), 6th International Conference on Natural Language Processing*, Pune, India, (no page numbers).
- Stroppa, Nicolas, Antal van den Bosch, & Andy Way (2007), Exploiting source similarity for SMT using context-informed features, in *Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation*, Skövde, Sweden, (231–240).
- Stroppa, Nicolas, Declan Groves, Andy Way, & Kepa Sarasola (2006), Example-based machine translation of the Basque language, in *Proceedings of the 7th biennial conference of the Association for Machine Translation in the Americas*, Cambridge, Massachusetts, (232–241).
- Stroppa, Nicolas & Andy Way (2006), MaTrEx: the DCU machine translation system for IWSLT 2006, in *Proceedings of IWSLT 2006 Workshop*, Kyoto, Japan, (31–36).
- Sumita, Eiichiro, Hitoshi Iida, & Hideo Kohyama (1990), Translating with examples: A new approach to machine translation, in *Third International Conference on Theoretical and Methodological Issues in Machine Translation of Natural Language*, Austin, TX, (203–21).
- Taskar, Ben, Simon Lacoste-Julien, & Dan Klein (2005), A discriminative matching approach to word alignment, in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Vancouver, BC, Canada, (73–80).
- Tidhar, Dan & Uwe Küssner (2000), Learning to select a good translation, in *Coling 2000 in Europe: the 18th International Conference on Computational Linguistics. Proceedings*, Saarbrücken, Germany, volume 2, (843–849).
- Tillmann, Christoph (2004), A unigram orientation model for statistical machine translation, in *Proceedings Human-Language Technology and North American Association of Computational Linguistics (HLT-NAACL)*, Boston, MA, (101–104).

- Tillmann, Christoph, Stephan Vogel, Hermann Ney, Hassan Sawaf, & Alex Zubiaga (1997), Accelerated DP-based search for statistical translation, in *Proceedings of the 5th European Conference on Speech Communication and Technology (EuroSpeech '97)*, Rhodes, Greece, (2667–2670).
- Tillmann, Christoph & Fei Xia (2003), A phrase-based unigram model for statistical machine translation, in *Proceedings of the Joint Meeting of the Human Language Technology Conference and the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2003)*, Edmonton, Alberta, Canada, (106–108).
- Tinsley, John, Mary Hearne, & Andy Way (2007a), Exploiting parallel treebanks to improve phrase-based statistical machine translation, in *Proceedings of the Sixth International Workshop on Treebanks and Linguistic Theories (TLT-07)*, Bergen, Norway, (175–187).
- Tinsley, John, Yanjun Ma, Sylwia Ozdowska, & Andy Way (2008), MaTrEx: The DCU MT system for WMT 2008, in *Proceedings of the Third Workshop on Statistical Machine Translation*, Columbus, OH, (171–174).
- Tinsley, John, Venstislav Zhechev, Mary Hearne, & Andy Way (2007b), Robust language-pair independent sub-tree alignment, in *Machine Translation Summit XI*, Copenhagen, Denmark, (467–474).
- Toutanova, Kristina, Tolga Ilhan, & Christopher Manning (2002), Extensions to HMM-based statistical word alignment models, in *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, Philadelphia, PA, (87–94).
- Turian, Joseph, Luke Shen, & Dan Melamed (2003), Evaluation of machine translation and its evaluation, in *Machine Translation Summit IX*, New Orleans, LA, (386–393).
- Unicode Consortium (2006), *The Unicode Standard, Version 5.0*, Addison-Wesley Professional, Boston, MA.
- Vauquois, Bernard (1968), A survey of formal grammars and algorithms for recognition and transformation in machine translation, in *IFIP Congress-68*, Edinburgh, UK, (254–260).
- Veale, Tony, Alan Conway, & Brona Collins (1998), The Challenges of Cross-Modal Translation: English to Sign Language Translation in the Zardoz System, *Machine Translation* 13(1):81–106.
- Vilar, David, Maja Popovic, & Hermann Ney (2006), AER: Do we need to “improve” our alignments?, in *Proceedings of the International Workshop on Spoken Language Translation*, Kyoto, Japan, (205–212).
- Viterbi, Andrew (1967), Error bounds for convolutional codes and an asymptotically optimum decoding algorithm, *IEEE Transactions on Information Theory* 13:1260–1269.
- Vogel, Stefan, Hermann Ney, & Christoph Tillmann (1996), HMM-based word alignment in statistical translation, in *Proceedings of the 16th International Conference on Computational Linguistics*, Copenhagen, Denmark, (836–841).
- Vogel, Stephan & Hermann Ney (2000), Construction of a hierarchical translation memory, in *Coling 2000 in Europe: the 18th International Conference on Computational Linguistics. Proceedings*, Saarbrücken, Germany, volume 2, (1131–1135).
- Watanabe, Hideo (1992), A similarity-driven transfer system, in *Proceedings of the fifteenth [sic] International Conference on Computational Linguistics, COLING-92*, Nantes, France, (770–776).

- Way, Andy (2003), Machine translation using LFG-DOP, in Rens Bod, Remko Scha, & K. Sima'an (eds.), *Data-Oriented Parsing*, CSLI, Stanford, CA, (359–384).
- Way, Andy (2009a), A critique of statistical machine translation, in Walter Daelemans & Véronique Hoste (eds.), *Journal of translation and interpreting studies: Special Issue on Evaluation of Translation Technology*, Linguistica Antverpiensia, Antwerp, Belgium, (in press).
- Way, Andy (2009b), Panning for EBMT gold, or “remembering not to forget”, *Machine Translation* 23:in press.
- Way, Andy & Nano Gough (2003), wEBMT: Developing and validating an EBMT system using the World Wide Web, *Computational Linguistics* 29(3):421–457.
- Way, Andy & Nano Gough (2004), Example-based controlled translation, in *Proceedings of the Ninth EAMT Workshop, Broadening Horizons of Machine Translation and its Applications*, Valetta, Malta, (73–81).
- Way, Andy & Nano Gough (2005a), Comparing example-based and statistical machine translation, *Natural Language Engineering* 11(3):295–309.
- Way, Andy & Nano Gough (2005b), Controlled translation in an example-based environment, *Machine Translation* 19(1):1–36.
- White, John (1985), Characteristics of the METAL machine translation system at production stage, in *Proceedings of the Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages*, New York, NY, (359–369).
- Wu, Dekai (1997), Stochastic Inversion Transduction Grammars and bilingual parsing of parallel corpora, *Computational Linguistics* 23(3):377–403.
- Wu, Dekai (2005), MT model space: Statistical vs. compositional vs. example-based machine translation, *Machine Translation* 19(3–4):213–227.
- Wu, Hua, Haifeng Wang, & Zhanyi Liu (2006), Boosting statistical word alignment using labeled and unlabeled data, in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Sydney, Australia, (913–920).
- Yamada, Kenji & Kevin Knight (2001), A syntax-based statistical translation model, in *Association for Computational Linguistics, 39th Annual Meeting and 10th Conference of the European Chapter, Proceedings*, Toulouse, France, (523–530).
- Yamada, Kenji & Ion Muslea (2006), Re-ranking for large-scale statistical machine translation, in *Proceedings of NIPS-06 Workshop on Machine Learning for Multi-lingual Information Access*, Whistler, BC, Canada.
- Younger, Daniel (1967), Recognition and parsing of context-free languages in time n^3 , *Information Control* 10(2):189–208.
- van Zaanen, Menno & Harold Somers (2005), Democrat: deciding between multiple outputs created by automatic translation, in *MT Summit X, The Tenth Machine Translation Summit*, Phuket, Thailand, (173–180).
- Zhang, Ruiqiang, Keiji Yasuda, & Eiichiro Sumita (2008), Improved statistical machine translation by multiple Chinese word segmentation, in *Proceedings of the Third Workshop on Statistical Machine Translation*, Columbus, OH, (216–223).

- Zhang, Ying & Stephan Vogel (2005), An efficient phrase-to-phrase alignment model for arbitrarily long phrase and large corpora, in *Proceedings of the Tenth Conference of the European Association for Machine Translation (EAMT-05)*, Budapest, Hungary, (294–301).
- Zhechev, Ventsislav & Andy Way (2008), Automatic generation of parallel treebanks, in *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, Manchester, UK, (1105–1112).
- Zhu, Jiang & Haifeng Wang (2005), The effect of adding rules into the rule-based MT system, in *MT Summit X, The Tenth Machine Translation Summit*, Phuket, Thailand, (298–304).