Statistical Machine Translation and its Challenges

Hermann Ney

Lehrstuhl für Informatik VI
Human Language Technology and Pattern Recognition
RWTH Aachen – University of Technology
D-52056 Aachen, Germany
ney@informatik.rwth-aachen.de

Abstract

In addition to speech recognition and syntactic parsing, during the last 10 years, the statistical approach has found widespread use in machine translation of both written language and spoken language. In many comparative evaluations, the statistical approach was found to be competitive or superior to the existing conventional approaches. Since the first statistical approach was proposed at the end of the 80s, many attempts have been made to improve the state of the art. Like other natural language processing tasks, machine translation requires four major components: a decision rule, a set of probability models, a training criterion and an efficient generation of the target sentence. We will consider each of these four components in more detail and point out promising research directions.

1. Introduction

This paper will address the machine translation of both written and spoken language. Spoken language translation has been and is being investigated in a number of joint projects at some national levels, the European level and the international level (C-Star, ATR, Verbmobil, EuTrans, Nespoli!, Fame, LC-Star, PF-Star, TC-Star, ...). The best performing translation systems are based on various types of statistical approaches [14] including example-based methods [21], finite-state transducers [6] and other data driven approaches. This is the characteristic and most striking result of the various projects. The principles on which the statistical approach is based were worked out only around 1990 [5]. Considerable progress has been made since then due to improvements in the underlying models and algorithms and to the availability of bilingual parallel corpora and greater processing power. Recently, in the US Tides project for written language translation with large vocabularies (about 50,000 words), it was also found that, as a result of this progress, the statistical approach is able to produce competitive results in comparison with conventional translation systems that had been optimized over decades, e.g. for Chinese-English translation.

2. Statistical Approach to Machine Translation

The concept of the statistical approach to machine translation and other natural language processing (NLP) tasks is illustrated in Fig. 1. The most crucial role is taken by the probability model, which provides the link between the input data and the output data that have to be produced by the NLP system. The probability model has free parameters that are learned using a suitable training criterion from training examples that are representative of the NLP task to be performed. In addition, in Fig. 1, we have the decision rule that is used to select the most suitable output from the many possible outputs. When building an automatic system for NLP, we should try to use as much prior knowledge as possible about the task under consideration. This knowledge is used to guide the modelling process and to enable improved generalization with respect to unseen data. Therefore, in a good statistical modelling approach, we try to identify the common patterns underlying the observations, i.e. to capture dependencies between the data in order to avoid the pure ‘black box’ concept.

In summary, every statistical approach to machine translation and any other NLP task requires four essential ingredients:

• the decision rule based on the true but unknown probabilistic dependencies between input and output;
• the probability models that replace the true but unknown probability distributions;
• the training criterion that is used to learn the unknown model parameters from training data;
• the generation, i.e. efficient implementation of the decision rule, which as in speech recognition, is also referred to as search or decoding.

Each of these four ingredients will be considered in more detail in the following.

3. Decision Rule

For the statistical approach, we consider the probabilistic dependencies between a target string $e^1 = e_1 \ldots e_t$ (e=’English’) and a source string $f^t = f_1 \ldots f_j \ldots f_s$ (f=’foreign’) and the associated joint posterior probability \( p_r(I,e^1 | f^t) \). The starting point for any Bayes decision rule in machine translation is the definition of the posterior risk:

$$ R(I,e^1 | f^t) := \sum_{I,e^1} p_r(I,e^1 | f^t) \cdot L[e^1, e^1 \tilde{I}] $$

with a suitable error measure (or loss function) \( L[\cdot, \cdot] \) between two symbol strings $e^1$ and $e^1 \tilde{I}$. The Bayes decision rule selects that sequence (of unknown length) that, over all possible target
strings $e_i^I$ of any length $I$, minimizes the posterior risk for the observed source string $f_i^I$:

$$f_i^I \rightarrow (\hat{I}, \hat{e}_i^I) = \arg \min_{I, e_i^I} \{ R(I, e_i^I | f_i^I) \}$$

The question of how to carry out this minimization will be considered later in the section on generation.

### 3.1. String Error Measure

The traditional Bayes decision rule (i.e., maximizing the joint posterior probability over the unknown target strings) is obtained only when we apply the following string error measure:

$$L[e_i^I, \hat{e}_i^I] = 1 - \delta(I, \hat{I}) \cdot \prod_{i=1}^I \delta(e_i, \hat{e}_i)$$

with the Kronecker delta function $\delta(x, y)$. This error measure counts string or sentence errors, i.e., two strings are considered to involve no loss only when the symbols $e_i$ and $\hat{e}_i$ are identical in each position $i = 1, ..., I (= \hat{I})$. If the two strings are different, the loss is one no matter how many of the positions $i$ are different. In other words, this error measure counts the sentence errors. For the posterior risk, we obtain:

$$R(I, e_i^I | f_i^I) = 1 - p_r(I, e_i^I | f_i^I)$$

The key quantity in the posterior risk is the joint probability $p_r(I, e_i^I | f_i^I)$. This is the starting point for (virtually) all statistical approaches to machine translation and other NLP tasks. In practice however, the errors are counted at the symbol level. This inconsistency between the error measure and the specific form of the Bayes decision rule is rarely addressed in the literature [7, 10, 13].

### 3.2. Symbol Error Measure

A number of different error measures at the symbol level are used in machine translation [11, 20]. To simplify the presentation, we will use the following error measure that computes a normalized count of the symbol errors (here word errors) for strings of identical length $I = \hat{I}$:

$$L[e_i^I, \hat{e}_i^I] = 1 - \frac{\delta(I, \hat{I})}{I} \cdot \sum_{i=1}^I \delta(e_i, \hat{e}_i)$$

As opposed to the string error measure, this symbol error measure allows for ‘soft’ loss values for string errors, not simply zero and one. Note that, like the string error measure, this symbol error is symmetric with respect to the two strings to be compared. For the resulting posterior risk, we obtain (omitting the details of the calculations):

$$R(I, e_i^I | f_i^I) = 1 - \frac{1}{I} \cdot \sum_{i=1}^I p_r(I, e_i^I | f_i^I)$$

The key quantity here is the marginal probability $p_r(I, e_i^I | f_i^I)$ that depends on the target positions $i = 1, ..., I$. This marginal probability is obtained from the joint probability $p_r(I, e_i^I | f_i^I)$ by marginalization for each position $i = 1, ..., I$. A similar type of marginal probabilities is used in the context of confidence measures [24].

### 4. Probability Models

In the derivation of the Bayes decision rule, we have used the true probability distributions, which however are unknown and have to be learned from data. In addition, we are faced with the problem of handling probability distributions over symbol strings like $e_i^I$ and $f_i^I$ so that storing the probabilities directly in ‘big tables’ is prohibitive. Therefore the usual approach is to introduce probability models that have suitable structures along with unknown parameters to be estimated from data. It is here in the definition and selection of the model structures where task specific linguistic knowledge comes in. However so far, remarkably enough, widely used models like the alignment models use very little specific linguistic knowledge. For the purpose of modelling, it is convenient to consider the joint probability $p_r(I, e_i^I, f_i^I)$ and decompose it into a language model and a translation model:

$$p_r(I, e_i^I, f_i^I) = p_r(I, e_i^I) \cdot p_r(f_i^I | e_i^I)$$

#### 4.1. Alignment-based Translation Models

We want to find suitable structures for the translation probability $p_r(f_i^I | e_i^I)$. A key problem in modelling the string translation probability $p_r(f_i^I | e_i^I)$ is the question of how we define the correspondence between the words of the target sentence and the words of the source sentence. We assume a sort of pairwise dependence by considering all word pairs $(f_j^I, e_i^I)$ for a given sentence pair $(f_i^I, e_i^I)$. A family of such alignment models (IBM-1,...IBM-5) was developed in [5] and was extended by Hidden Markov models (HMM) similar to those developed for speech recognition [17]. We re-write the translation probability by introducing the hidden alignments $A$ for each sentence pair $(f_i^I, e_i^I)$:

$$p_r(f_i^I | e_i^I) = \sum_A p(f_i^I, A | e_i^I)$$

Here and in the following, we use the notation $p(\cdot | \cdot)$ for model distributions as opposed to $p_r(\cdot | \cdot)$ for true but unknown
distributions. For the generation of the target sentence, it is convenient to use the concept of *inverted alignments* which perform a mapping from a target position \(i\) to a set of source positions \(j\), i.e. we consider mappings \(B\) of the form:

\[
B : i \rightarrow B_i \subset \{1, ..., j, ..., J\}
\]

with the constraint that each source position \(j\) is covered exactly once. Using such an alignment \(A = B_i = B_i \ldots B_1\), we re-write the probability \(p(f_j^i, A | e^i)\):

\[
p(f_j^i, B_i^i | e^i) = p(J | I) \cdot \prod_{i=1}^{I} \left[ p(B_i | B_i^{i-1}, e_i) \cdot \prod_{j \in B_i} p(f_j | e_i) \right]
\]

with the length model \(p(J | I)\), the alignment model \(p(B_i | B_i^{i-1}, e_i)\) and the lexicon model \(p(f_j | e_i)\). In addition, we make further suitable modelling assumptions such as first-order dependencies for the inverted alignment model \(p(B_i | B_i^{i-1}, e_i)\) to obtain what is more or less equivalent to the alignment models IBM-3, 4 and 5 [17].

**Phrase-based Models.** The above approach does not take into account the context in which both the source and the target words appear. There is an evident need to introduce more context dependencies into these models, e.g. by handling word groups and phrases rather than single words. There have been a number of extensions to move away from single words and handle word groups in both the source and target language [18, 28]. Apart from [12], these extensions seem to be limited to the extraction of bilingual phrases after the IBM/HMM parameters have been trained. In other words, the phrase-based models are not yet incorporated into the training procedure.

The past experience with speech and language processing has shown that a substantial amount of progress was always achieved by the improvement of the more or less purely algorithmic concepts of how we model the dependencies of the data and how the system better learns from the data. We expect that future work along these lines will result in significant improvements.

### 4.2. Syntax-based Translation Models

Whereas the IBM/HMM approaches do not seem to make use of any syntactic concepts, there were attempts at introducing explicit syntactic structures into the modelling of the translation probability \(p_r(I, e^1_1 | f_1^I)\) [1, 26, 27]. In such a way, it is expected that the difference in the word order between target and source sentences can be better taken into account. At present, these syntax-based extensions do not seem to pay off in terms of performance. In addition, the syntactic approach could also include a morphological analysis [15] so that the statistical approach could go beyond the usual full forms of words.

### 4.3. Language Models

In language modelling, we want to find suitable structures for \(p_r(I, e^1_1)\), i.e. we want to model the redundancy of the target sentence \(e^1_1\) at all levels (lexical, syntactic, semantic,...). For machine translation so far, we use what has been found successful in speech recognition, namely the trigram or general \(n\)-gram approach, which can be extended by word classes, cache models and maximum entropy models.

To improve the syntactic structure of the target sentences, it seems to be promising to exploit probabilistic grammar-based language models. Recently in syntactic parsing, there have been significant advances by probabilistic lexicalized context free grammars.

### 5. Training Criterion

The unknown parameters of the models are estimated from training data. For the alignment and lexicon models, we use a corpus of bilingual sentence pairs. For the language model, the corpus is simply a monolingual collection of target sentences. For a formal specification of the training task, an optimization criterion is used as training criterion.

Traditionally, this formal criterion has been the so-called maximum likelihood criterion. In particular, in training the alignment and lexicon models, we are faced with a difficult mathematical optimization problem because at the same time both the word-to-word alignments and the model parameters are unknown [5]. For this type of problem, the EM algorithm provides a constructive and efficient procedure for maximizing the likelihood criterion. This may be the reason for the popularity of the maximum likelihood criterion and the negligence of other training criteria.

In speech recognition, other training criteria have been proposed that are more *discriminative* and thus are better tailored to the decision rule. Examples are the class posterior probability criterion (maximum mutual information) [2] and the minimum classification error, i.e. the (suitably smoothed) number of decision errors on the training data [8]. These discriminative criteria so far are more popular in speech recognition than in machine translation. In [16], discriminative training was used for machine translation, but primarily to estimate the interpolation parameters of log-linear models and not for the alignment or lexicon models themselves.

In addition, for the training criterion, the question comes up which type of probability to consider, i.e. the joint probability \(p_r(I, e^1_1 | f_1^I)\) or the marginal probability \(p_r(I, e^1_1)\). So far, only the joint probability has been used.

### 6. Generation

The task of the generation algorithm is to generate the minimum-risk target sentence \(e^1_1\) of unknown length \(I\) for a source sentence \(f_1^I\). The generation must make use of all three knowledge sources: the alignment model, the (bilingual) lexicon model and the language model. All three of them must contribute in the final decision about the words in the target language.

This generation is a combinatorial problem, in particular when the vocabulary of the target language is large. Thus the real challenge is to find an efficient implementation of the Bayes decision rule. Naturally, the details of the generation will depend very much on the specific structure of the chosen probability models.
6.1. String-based Decision Rule

Like speech recognition, machine translation so far has focused on the string error measure for the Bayes decision rule. To obtain the final generation criterion, we replace the sum over all alignments by the best alignment, which is referred to as maximum approximation in speech recognition. Using a trigram language model \( p(e_i | e_{i-1}) \) and dropping the length model \( p(J | I) \), we obtain the following search criterion:

\[
\max_{l, b} \left\{ \prod_{i=1}^{n} \left[ p(e_i | e_{i-1}) \cdot p(B_i | B_{i-1}, e_i) \cdot \prod_{j \in B_i} p(f_j | e_i) \right] \right\}
\]

An important constraint for the alignment is that each position of the source sentence should be covered exactly once. This constraint is similar to that of the travelling salesman problem where each city has to be visited exactly once [9]. Details on various search strategies can be found in [4, 19, 23, 25].

6.2. Symbol-based Decision Rule

For the symbol-based decision rule, the generation task involves two steps:

- The efficient calculation of the marginal probability \( pr^r(I, e_j | f_i^j) \), which is defined by summing out the joint probability \( pr(I, e_j | f_i^j) \) over all target strings \( e_j \) with \( e_i = e \);
- The efficient minimization of the posterior risk \( R(I, e_j | f_i^j) \) in order to generate the target string \( e_j \).

In the context of POS tagging, Merialdo [13] used a forward-backward algorithm to compute the marginal probability efficiently. Part of this work goes back to [3], where a related problem in coding theory was studied. For machine translation itself, this symbol-based decision rule seems to have been used only in the context of confidence measures [24].

Acknowledgment. The work reported here was supported partly by the Verbmobil project (German BMBF), by the German Science Foundation (DfG) and by various European projects (Eutrans, LC-Star, PF-Star and TC-Star-P).

7. References