MAXIMUM LIKELIHOOD AND DISCRIMINATIVE TRAINING OF DIRECT TRANSLATION MODELS

K. A. Papineni *S. Roukos* *R. T. Ward*
IBM T. J. Watson Research Center
Yorktown Heights, NY 10598, USA

ABSTRACT

We consider translating natural language sentences into a formal language using direct translation models built automatically from training data. Direct translation models have three components: an arbitrary prior conditional probability distribution, features that capture correlations between automatically determined key phrases or sets of words in both languages, and weights associated with these features. The features and the weights are selected using a training corpus of matched pairs of source and target language sentences to maximize the entropy or a new discrimination measure of the resulting conditional probability model. We report results in Air Travel Information System (ATIS) domain and compare the two methods of training.

1. INTRODUCTION

The crux of an automatic translation task is to capture correlations between groups of words in one language and groups of words in the other. Better translation performance can be expected when the words in these groups are not forced to be contiguous. Our interest is in developing a statistical translation system that is fully data-driven and can be built automatically from the training data.

For this paper, the source language is a natural language in a restricted domain and the target language is an artificial (formal) language. Such situations arise in building natural language interfaces to applications such as email and data-bases. The formal language expresses operations that the target applications can perform.

We apply our techniques to Air Travel Information System (ATIS) domain. Here we translate English queries on air travel information into a formal language that can then be translated deterministically into a database query. The data for ATIS was collected in an ARPA-sponsored program [3].

The starting point for building a model is several thousands of English queries and their man-made formal language translations. These pairs of English and formal sentences form the *training corpus*. Some examples from the training corpus follow:

S_1: show me all nonstop flights from airport-1 to city-1 leaving airport-1 before time-1 on day-1 .
T_1: LIST FLIGHTS NONSTOP DEPARTING BEFORE TIME-1 FLYING-ON DAY-1 FROM:AIRPORT AIRPORT-1 TO:CITY CITY-1

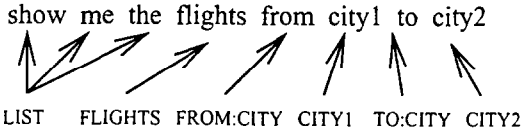
S_2: what are the available flights on air-1 from city-1 to city-2 the evening of date-1 .
T_2: LIST FLIGHTS AIR-1 EVENING FLYING-ON DATE-1 FROM:CITY CITY-1 TO:CITY CITY-2

Statistical translation models are used to translate a new (unseen in the training corpus) source sentence S in the following natural way: Evaluate conditional probability $P(T|S)$ for all T in the target language space and select the best scoring T as the translation. Parameters of these models are "trained" from the training corpus.

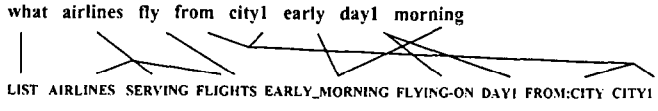
Brown, et al [1] introduced statistical translation models in the context of French to English translation. These models are based on the familiar source-channel paradigm that uses two component models: 1. $P(S|T)$ called the channel model, and 2. $P(T)$ called the language model (or source model). Since $P(T|S) = P(S|T)P(T)/P(S)$, we choose the T that maximizes the product

$P(S|T)P(T)$ as the translation of S. The channel model can also be thought of a translation model, but *in the wrong direction* (from target to source). As in speech recognition task, the language model is built separately without regard to its end-purpose (translation task).

In the channel model, roughly speaking, the target sentence words can be seen as states in an HMM and source sentence words can be seen as observation vector. Being a generative model, the channel model allows only *one* target language word to be associated with a *contiguous* group of source language words, but not vice versa. That is, the source-channel model is *constrained* to generate one group of English words for each Formal word, as below.



However, the central task in translation is to determine correlations between *groups* of words in one language and *groups* of words in the other. An example of such a general correlation is below.



The source-channel model fails in capturing such general correlations.

2. GOODNESS OF A MODEL

We consider two objective functions of *any* probabilistic model $P(T|S)$: One is the familiar log-likelihood on training data, given by

$$\sum_{h,f} \tilde{P}(h,f) \log P(f|h),$$

where $\tilde{P}$ is the *empirical* distribution. The second is a measure of model's discrimination described below.

Let the model's best guess of future $\hat{f}_P(h)$ be

$$\arg\max_f P(f|h)$$

where the maximum is taken over an *a priori* fixed *finite* set of futures.

Suppose we are given a collection of training pairs (h_i, f_i), $i = 1, \ldots, T$. Treating training data as truth, we ideally want a model P such that $\hat{f}_P(h_i) = f_i$ for each i. A measure of goodness of the model P is the *discrimination* defined by

$$D(P) := \sum_{i=1}^{T} \log \frac{P(f_i|h_i)}{P(\hat{f}_P(h_i)|h_i)}.$$

This can be compared to the loss function in [2] with $\eta \to \infty$ and without the smoothing. The motivation for the current measure is as follows. When the model incorrectly predicts the future, there is a penalty that is proportional to how far away truth is to prediction. The ratio is smaller than 1 and hence the cost function is negative. A variation of the measure is obtained by measuring how far away truth is to the best guess or to the nearest competitor when best guess is indeed the truth. From a theoritical point of view, this variation is inessential.

3. THE DIRECT TRANSLATION MODEL

As in [5], here we use a powerful alternative to the source-channel model for translation: we build a *direct* model of the *a posteriori* conditional distribution $P(T|S)$. The direct translation model has three components: features that capture translation effects and language model effects in a unified framework, weights associated with the features, and a *prior* conditional probability distribution $P_0(T|S)$. The prior could be uniform, or could be derived from a decision tree, or any arbitrary probabilistic model. The model can be seen as a correction to the prior relative to feature functions. We consider a variety of features involving phrases, set of words, parses, and long-distance relations in the source and target sentences.

The selection of features and weights is fully data-driven and can incorporate *a variety of objective functions*. Feature selection and training with respect to log-likelihood of training data was described in [5].

3.1. Features

For simplicity, we consider only binary-valued feature functions here. So a feature maps the product set of source and target language sentences to 0 or 1. Some concrete examples of features that we considered follow. First consider some sample English

and Formal sentences. The formal sentences are not translations of the English sentences.

E_1: what are least expensive flights from city-1 to city-2.
E_2: what flights do you have from city-1 to city-2.

F_1: LIST FLIGHTS MORNING EARLIEST-ARRIVING FROM:CITY CITY-1 TO:CITY CITY-2
F_2: LIST FLIGHTS CHEAPEST FROM:CITY CITY-1 TO:CITY CITY-2

Certain phrases in source language sentences tend to co-occur with certain phrases in target language translations. To model this fact, we considered phrase-features of the form

$$\phi_{s,t}(S,T) = \begin{cases} 1 & \text{if } s \in S, t \in T \\ 0 & \text{else.} \end{cases}$$

The feature

$$\phi_{\text{least expensive, CHEAPEST}}$$

fires on (E_1, F_2), but not on (E_2, F_2) or (E_1, F_1).

A special case with a null s-phrase results in features that induce target language modeling. With such features, there is no need to estimate target language model separately. A variation of a phrase feature is one which ignores the order of words in s-phrase and t-phrase. Another is a long-distance bigram feature.

Sometimes we know that certain words will not occur in the translation except when certain words occur in the source sentence. A feature that looks for existence of words in the target sentence that do not have an "informant" in the source sentence models this fact. An example feature is one that looks for the word "CHEAPEST" in Formal in the absense of "lowest" and "cheapest" in English. This feature fires on (E_2, F_2) but not on (E_1, F_2). Such features almost never fire on the training data.

In summary, features query presence *or absence* of n-grams, long-distance bigrams, or unordered (possibly empty) sets of words in both source and target language sentences.

3.2. Feature Selection and Optimization

We described a variety of features so far. Let $\phi(S,T)$ be a vector feature of dimension n. We consider models P of the form

$$P_{\phi,\lambda}(T|S) := \frac{P_0(T|S)e^{\lambda\phi(S,T)}}{Z(S)}$$

with the normalization factor

$$Z(S) := \sum_T P_0(T|S)e^{\lambda\phi(S,T)}.$$

With $\alpha_i := e^{\lambda_i}$, we can rewrite the above as

$$P(T|S) = \frac{P_0(T|S)\Pi\alpha_i^{\phi_i(S,T)}}{Z(S)}.$$

In this formulation, we see that each feature that is true (i.e. takes the value 1) gets a multiplicative "vote" α_i to modify the prior score $P_0(T|S)$.

Exponential models of the above form arise naturally in maximum entropy framework. In that framework, we start with linear constraints on feature functions and look for any probability distribution that satisfies the constraints and is as close to a prior distribution as possible. From Lagrange multiplier theory, it turns out that the optimal solution is an exponential model of the above form. So we need only search in the family of exponential models for the optimum solution. Here, we take an exponential family of models as the starting point and allow the possibility of using objective functions other than log-likelihood of training data in choosing a model. We consider maximum likelihood and maximum discrimination problems here. Once we start with the exponential family, we do not impose any additional linear (equality) constraints on the model.

We now describe feature selection. First, we assume that a set of n features ϕ have already been selected somehow. We then solve the maximum entropy (or maximum discrimination) problem described above. The solution is standard for the maximum entropy problem [4]. For the maximum discrimination probelm, the solution involves solving a series of linear programming problems and will be described elsewhere [6]. Then, D_*, the minimum achieved by λ_*, is a figure of merit of the feature set $\{\phi_1, \cdots, \phi_n\}$. Once the set $\{\phi_1, \cdots, \phi_n\}$

and $\{\lambda_1, \cdots, \lambda_n\}$ are selected, we compute D_* for $\{\phi_1, \phi_2, \cdots, \phi_n, f\}$ for all features f in the remaining pool and rank the features by the new D_*. We can then add the top-ranking feature as ϕ_{n+1} to the set of features already selected and find the optimal weights $\lambda_1, \lambda_2, \cdots, \lambda_{n+1}$. Thus, in principle, we can start with $n = 0$ and build a good feature set by increasing the set by 1 in each batch. In practice, we add top k-ranking new features in each batch to the features already selected. The figure of merit D_* increases monotonically with the size of the feature set. We stop feature selection when the increment is marginal.

4. EXPERIMENTAL RESULTS

We built a model to translate context-independent English queries. We used 5627 pairs of context-independent sentences from the ATIS training data. Examples of features that our system selected are shown below along with their near-optimal weights.

Source Phrase	Target Phrase	α
arrive	FLIGHTS ARRIVING-ON	39
about	AROUND	2900
late-afternoon	LATE-AFTERNOON	56000
cheapest round-trip	FARES CHEAPEST ROUND-TRIP	43
including	ALONG-WITH	280

We compare maximum discriminative training and maximum likelihood training. Both methods share the same prior, which consists of about 300 features. Since our purpose is in comparing training of features by two objective functions, we may assume that the features in question have already been selected somehow. However, if they are selected by one criterion and trained by another, the results can be biased against the second critetion. In an attempt to reduce this bias, a large pool of features is first filtered by using log-likelihood criterion. From this small filtered pool (about 500 features), discrimination measure is used to select 25 features at a time. Each of these batches of features were then trained by log-likelihood criterion in one experiment and by discrimination criterion in another experiment.

We report results on context-independent queries from ATIS DEV94 test set, which is outside the training corpus. Translation performance is measured by Common Answer Specification, a metric defined by ARPA in terms of response from air travel database. First column shows the number of features in the model, the second column shows the performance when these features are trained by optimizing discrimination and the third the performance when log-likelihood is optimized.

Size	Discr	LL
75	84.39	82.92
150	84.63	84.14
225	84.63	83.17
300	84.14	83.41

REFERENCES

[1] P. F. Brown et al. "The mathematics of statistical machine translation: Parameter estimation", *Computational Lingustics*, 19 (2), 263-311, June 1993.

[2] W. Chou et al, "Segmental GPD training of HMM based speech recognizer," Proceedings of the IEEE ICASSP-92, Vol. I, pp. 473-476

[3] L. Hirschman, "Multi-Site Data Collection for a Spoken Language Corpus", Proceedings of the DARPA Speech and Natural Language Workshop, pp 7-14, Harriman, NY, Feb 1992.

[4] S. Della Pietra et al, "Inducing features of random fields," CMU Technical Report CMU-CS-95-144, 1995.

[5] K. Papineni et al, "Feature-based language understanding," Proceedings of EUROSPEECH'97, vol 3, pp. 1435-1438.

[6] K. Papineni, "Efficient discriminative training of exponential probability models," Manuscript in preparation.