Discriminative Training and Maximum Entropy Models for Statistical Machine Translation

Franz Och and Hermanan Ney
Presented by Bill McNeill

Overview

- Four word summary
- Mathematical Motivation
- Och and Ney's Experiment
- Relevance for linguists

Four Word Summary

- *Do MT with MaxH*
- MT is "machine translation"
- MaxH is "maximum entropy"

Bayesian Throat-Clearing

- Source-Channel Approach
  - Decoding

$$\hat{e}_1^I = \underset{e_1^I}{\text{argmax}}\{Pr(e_1^I)Pr(f_1^J|e_1^I)\}$$

  - Training

$$\begin{aligned}\hat{\theta} &= \underset{\theta}{\text{argmax}} \prod_{s=1}^{S} p_\theta(\mathbf{f_s}|\mathbf{e_s}) \\ \hat{\gamma} &= \underset{\gamma}{\text{argmax}} \prod_{s=1}^{S} p_\gamma(\mathbf{e_s})\end{aligned}$$

- Och and Ney do *something else*

Direct Translation Model

- Normalized product of exponentials

$$p(e_1^I|f_1^J) = \frac{\exp\left[\sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J)\right]}{\sum_{e'^I_1} \exp\left[\sum_{m=1}^{M} \lambda_m h_m(e'^I_1, f_1^J)\right]}$$

- $h_m(e_1^I, f_1^J)$ are features

- λ_m are trainable parameters

- Where did this equation come from?

Generative Model

- Tell a (stochastic) story
- Model relates aligned corpora
- Identify relevant features
- Specify how they combine

Discriminative Model

- Directly relate data in aligned corpora
- The "model" is a generic data alignment
- Identify relevant features
- Be agnostic about how they combine

Maximum Entropy in General

- Say what we know and and nothing more

- There are many $p(x)$ such that

$$E_x\left[p(x)h_i(x)\right] = \alpha_i \text{ for } 1 \leq i \leq M$$

- The maximum entropy $p(x)$ has the form

$$p(x) = \exp\left[\lambda_0 + \sum_{i=1}^{M} \lambda_i h_i(x)\right]$$

  where λ_0^M is a function of α_0^M

- See Cover and Thomas *Elements of Information Theory* for details

Apply Maximum Entropy to MT

- Identify relevant $h_i(e_1^I, f_1^J)$ features
- We don't care how they interact
- $E_{\mathbf{e_S},\mathbf{f_S}}\left[h_i(e_1^I, f_1^J)\right]$ moments are defined
- We can use MaxH

Machine Learning Strategy

- Throw a bunch of features into the pot
- Generate λ_i^M weights that best align the training data
- ". . . maximizing the equivocation . . ."
- Hope these weights generalize to unseen data

Training Details

- Train the weights using

$$\widehat{\lambda}_1^M = \underset{\lambda_1^M}{\text{argmax}} \left\{ \sum_{s=1}^{S} \frac{1}{R_S} \sum_{r=1}^{R_S} \log p_{\lambda_1^M}(\mathbf{e_{s,r}}|\mathbf{f_s}) \right\}$$

- Easier said than done: λ_1^M is a vector

- Use Generalized Iterative Scaling (GIS)

Generative Shortcomings

- From the end of section 1.1

  1. Must specify model details
  2. Must specify how features interact
  3. Models must "make sense" stochastically

- Tradeoff: model freedom vs. optimization difficulty

Baseline Features

- λ_1 Trigram language model
- λ_2 Alignment template model
- λ_3 Lexicon model
- λ_4 Alignment model

Additional Features

- Word penalty: $h(f_1^J, e_1^I) = h(e_1^I) = I$
  (Note: we don't care how this interacts with alignment.)

- Class-based target 5-gram

- Lexicon co-occurrence

Proposed Features

- Lexical features of word pairs (f, e)
- Grammatical features, e.g. count verb groups
- Generative verb group counting would be hard

Experiment

- German-English VERBMOBIL task
- 58073 training sentences
- 251 test sentences
- Multiple error criteria
- Evaluate quality as a function of features

Required Training Effort

- SER plateaued after about 4000 GIS iterations
- How long does an iteration take?
- How about other error measures?
- "We do not observe significant overfitting" ?

Results

- Training the λ_1^M helped (unsurprisingly)
- Adding the target sentence length helped a lot
- Other features had less of an effect
- The lexicon co-occurrence did nothing or even hurt

Model Weights

- Magnitude of λ_1^M is a rough measure of feature importance

- Particularly important were alignment template and word penalty

- Not sure why they didn't normalize all the values

Other Things to Try

- Throw in the proposed features
- Could the lexical features give us phrase alignments?
- Different feature combinations (e.g. ME+CLM)

The Bad News for Linguists

- The simplest feature (target sentence length) helped the most

- The clever knowledge engineering feature (lexicon co-occurrence) did nothing or hurt

The Good News for Linguists

- We care about linguistically relevant features

- We don't care about probability distributions

- MaxH methods are mathematically sophisticated, but someone has done the hard part for us