CMPT 825: Natural Language Processing **Spring 2006**

Lecture 2 — Jan 12, 2006

Lecturer: Anoop Sarkar *Scribe: Dong Song*

2.1 Required Readings

- Read sections 1 - 22 of the workbook "A Statistical MT Tutorial Workbook", written by Kevin Knight.
- Read the paper "BLEU: A Method for Automatic Evaluation of Machine Translation".

2.2 Basic Probabilities and Bayes' Theorem

Machine translation(MT) is the computer application that translates *source language* (for example, French) into *target language* (for example, English). In MT, the statistical approach is widely used. Therefore, we first review certain basic probability concepts:

Let f represent a source language sentence, e a target language sentence.

- $P(e)$ is a prior probability: the chances that e happens;
- $P(f|e)$ is a conditional probability: the chance of f given e;
- $P(e, f)$ is a joint probability: the chance of e and f both happening. If e and f are independent of each other, then $P(e, f) = P(e) * P(f)$; otherwise, $P(e, f) = P(e) * P(f|e)$.

Theorem 2.1 (Bayes' Theorem). $P(e|f) = \frac{P(e)*P(f|e)}{P(f)}$.

Proof: According to the basic probabilities, $P(e, f) = P(e) * P(f|e)$, and also $P(f, e) = P(f) * P(e|f)$. Since $P(e, f) = P(f, e)$, we have:

$$P(e) * P(f|e) = P(e, f) = P(f, e) = P(f) * P(e|f).$$

Thus, $P(e|f) = \frac{P(e)*P(f|e)}{P(f)}$. □

For a sentence f in source language, we would like to find the sentence e in target language that maximizes $P(e|f)$. By Bayes' Theorem, we have:

$$argmax_e P(e|f) = argmax_e P(e) * P(f|e) \tag{2.1}$$

Thus, we can first compute $P(e)$ by constructing a language model, and compute $P(f|e)$ using a translation model. After that, we can find the optimal target sentence e for a given source sentence f.

2.3 Noisy Channel Model

Noisy channel model is developed by Shannon to describe optimal error correcting codes. We can imagine the following scenario: A person is planning to write the sentence e, but due to certain "noise", e is corrupted and is written as f. The challenge is to find out what the original sentence e was, given the observed text f and the noise distribution information. The model is shown in figure 2.1:

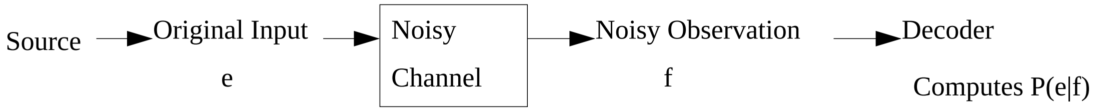

Figure 2.1. Noisy Channel Model

2.4 Story of Movie Production

The following story is taken from [1]. Suppose we have the image shown in figure 2.2, and we would like to produce that image with a physical scene.

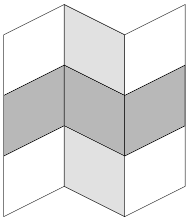

Figure 2.2. Image

Each possible operation and their related cost is as follows:

- Spray Painter Fees: Paint rectangle patch: \$5 each; Paint general polygon: \$5 per side.
- Sheet Metal Worker Fees: Right angle cuts \$2 each; Odd angle cuts \$5 each; Right angle bends \$2 each; Odd angle bends \$5 each.
- Lighting Designer Fees: Flood light \$5 each; Custom spot light \$30 each.
- Supervisor Fees: Consultation \$30 per job.

There will be many different ways of constructing scenes that produce the same image. For example,

- Painter's solution: Paint 9 general polygons: \$180; Setup 1 flood light: \$5; Cut 1 rectangle: \$8. Total Cost: \$193.
- Supervisor's solution: Cut 1 rectangle: \$8; Paint 3 rectangles: \$5; Bend 2 right angles: \$4; Supervisor's fee: \$30. Total Cost: \$47.

After thinking about this story, we can see that, inside one's mind, different models can be generated, and every model is quite complicated. Each model is ranked, and the optimal one is chosen. In addition, a person's mind can switch between models, but only one model can be kept in mind at one time.

2.5 Evaluation of Machine Translation

Kishore Papineni, et in [2] proposed the method *BLEU* to evaluate machine translation automatically.

First, let's review the meanings of *precision* and *recall*. In general, *precision* is the number of relevant records retrieved divided by the total number of records retrieved; *precision* increases by guessing less. *Recall* is the number of relevant records retrieved divided by the total number of relevant records, and it measures how well a system finds what you want. The relationship between *precision* and *recall* can be visualized in figure 2.3.

In *BLEU*, the modified n-gram precision is used as metric to measure translation closeness. Similar for any n, the modified n-gram precision is computed in the following steps:

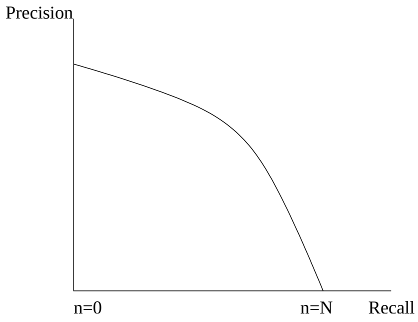

Figure 2.3. Precision v.s. Recall

- Step 1: collect all candidate n-gram counts and corresponding maximum reference counts;
- Step 2: clip the candidate counts by their corresponding maximum reference count;
- Step 3: The clipped counts are added together, and divided by the total number of candidate n-grams.

Adequacy and fluency, two of the aspects of translation is captured by this precision metric.

Why doesn't $BLEU$ use *recall* as measurement? Several reasons are mentioned in [2]. First, in $BLEU$, multiple reference translations are considered, and the same source word may be translated using a different word choice. Moreover, for a good candidate translation, only one of these possible choices is used.

2.6 References

[1] E.H. Adelson and A.P. Pentland. The perception of Shading and Reflectance. In D. Knill and W. Richards (eds.), *Perception as Baysian Inference*, (pp. 409-423). New York: Cambridge University Press (1996).

[2] Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu. 2002. BLEU: a Method of Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Lingusitics*, page 311-318, Philadelphia.