The Noisy Channel Model

11-761: Language and Statistics

The entropy of a source is a lower bound on its compressibility. This is one of the fundamental results of Information Theory, and the tools and techniques of data compression are extremely useful in statistical language processing. The channel coding theorem is concerned with the fundamental limits on data *transmission*. This result is useful in language processing for a different reason. Rather than providing a set of computational tools for manipulating language, it provides an extremely powerful *conceptual framework* for building language processing systems.[1] In this section we will outline the use of the noisy channel paradigm in statistical language processing by first presenting the basic result from communication theory, and then several examples of the source-channel paradigm in action.

In most of what follows, we'll be following the presentation of Cover and Thomas [7] closely. There are fundamentally two aspects of Information Theory–the theory of data compression and data transmission. As we'll recall below, the joint source channel coding theorem establishes that they can be treated as separate problems (in practice, it may not be so simple, and some researchers are now considering joint source-channel coding issues.) Roughly speaking, a communication channel is a system in which the output depends statistically on the input. It is characterized by a conditional distribution $p(y \mid x)$ that y emerges from the channel given that x was input. It is remarkable that the maximum rate at which we can send information over the channel with arbitrarily small probability of error can be characterized by a single number, depending only on the physical characteristics of the channel, formalized as the conditional distribution $p(y \mid x)$. This is the *channel capacity*

$$C = \max_{p(x)} I(X;Y) \tag{1}$$

where $I(X;Y)$ is the mutual information of the joint distribution $p(x,y) \equiv p(x)p(y \mid x)$. The basic idea behind coding to achieve high transmission rate is to encode messages using redundant codewords that are mutually far apart in the sense that their corrupted versions are easily distinguishable. This is formally equivalent to the sphere packing problem in high dimensions. The basic idea behind showing any rate below C can be achieved with small probability of error is to show that a randomly chosen code will do the job with high probability. This does not show how to obtain an efficient decoding algorithm, however, and in practice we have to introduce a lot of structure into the codes in order to get practical algorithms.

The mathematical idealization of a communication system is shown in Figure 1. The output of the channel is random, with a distribution that depends on the input. From the output sequence we attempt to decode the original message.

[1] Perhaps the fundamental reason that the techniques associated with channel coding are not of use in language processing is that there is no freedom of choice for the codes when working with language — the languages we speak are the result of a complex evolution of social and historical factors over which we have little control. Similarly, although the genetic code is the result of a complex evolution, statistical and information-theoretic methods of compression are now proving to be of great use in computational genetics.

$$\textit{Message} \xrightarrow{W} \boxed{\textit{Encoder}} \xrightarrow{X} \boxed{\begin{array}{c}\textit{Noisy Channel}\\ p(y\,|\,x)\end{array}} \xrightarrow{Y} \boxed{\textit{Decoder}} \xrightarrow{\widehat{W}} \textit{Decoded message}$$

Figure 1: The Noisy Channel Model

Example - *Binary Symmetric Channel.* In this channel a bit is corrupted with probability p, and so it is transmitted accurately with probability $1-p$. Then we can easily compute the capacity since

$$\begin{aligned} I(X;Y) &= H(Y) - H(Y\,|\,X) && (2)\\ &= H(Y) - \sum_x p(x)\, H(Y\,|\,X=x) && (3)\\ &= H(Y) - \sum_x p(x)\, H(p) && (4)\\ &= H(Y) - H(p) && (5)\\ &\le 1 - H(p) && (6) \end{aligned}$$

Thus, the capacity is $1 - H(p)$ bits.

Example - *Noisy Typewriter.* In this channel a letter `A-Z` is sent uncorrupted with probability $\frac{1}{2}$ or is sent as the next letter (modulo 26) with probability $\frac{1}{2}$. In this case

$$\begin{aligned} I(X;Y) &= H(Y) - H(Y\,|\,X) && (7)\\ &\le \log 26 - 1 && (8)\\ &= \log 13 && (9) \end{aligned}$$

It's clear that we can transmit 13 characters with zero probability of error – just use the "even" letters.

The channel capacity has nice properties: it is non-negative (since mutual information is), it is no more than one (in the binary case), it is a continuous and concave function of $p(x)$.

Now, to roughly state the channel coding theorem, we need another definition. We break up our message and send it as a *block* of n symbols. If there are M possible codewords in the code, then the *rate* of the code is defined to be

$$R = \frac{\log M}{n} \qquad (10)$$

bits per transmission.

The Channel Coding Theorem. If R is any rate strictly less than the channel capacity C, then there is a sequence of codes of block length $n \to \infty$ and size 2^{nR} with probability of error approaching zero. Conversely, any such sequence of codes must satisfy $R \le C$.

So, the definition of entropy leads to the result that the "compression rate" R satisfies $H(X) < R$, and the channel coding theorem defines an intrinsic capacity C so that the "transmission rate" R satisfies $R < C$. The joint source channel coding theorem says that $H(X) < C$ is a necessary and sufficient condition for transmitting a source X over a noisy channel.

In sending text or speech over a noisy channel, we could design a code which uses the "raw data" directly, or we could first compress the text or speech and then code the compressed data for transmission over the channel. It is not immediately evident that first compressing doesn't make the job of coding intrinsically more difficult. The method of compression, after all, need not depend on the channel at all. Nevertheless, in theory nothing can be lost from first compressing, so the problems of source coding (compression) and channel coding can be considered separately.

The Source-Channel Coding Theorem. A stationary and ergodic stochastic process $\mathcal{X}$ can be sent across a noisy channel with probability of error approaching zero if and only if $H(\mathcal{X}) < C$.

We refer to [7] for a precise statement and all of the details on this.

In practice it might be that the channel is well suited to the source, and it would be advantageous to encode the source directly. For example, in sending English text over an erasure channel (in which letters are randomly deleted), it may be easy for a human receiver to decode the original message (a good fraction of the vowels can be dropped, for example), but removing the errors from a version that is first compressed might be difficult. Similarly, we can easily decode speech under a variety of noise conditions. Thus, the natural redundancy in the source is appropriate for the channel; in fact, the codes that are the languages we speak and write have been "chosen," in part, for precisely this reason! For applications such as image transmission over the Internet, researchers are now beginning to consider joint source and channel coding techniques which prioritize the data and apply channel coding where it is most needed to minimize a loss function [1, 8].

1.2 Applications of the Noisy Channel Paradigm

The use of the noisy channel paradigm is one of the "big ideas" in statistical natural language processing. Sometimes the appropriateness of the approach is obvious and natural; other times it is subtle and hidden. We'll now try to give some examples of the channel model at work as a *conceptual framework* for attacking complex speech and language processing tasks.

1.2.1 Speech recognition

The most widespread use of the noisy channel paradigm is probably in speech recognition. This was first formulated and exploited by the IBM speech recognition group [2]. The basic system consists of an *acoustic processor* and a *linguistic decoder*. The acoustic processor produces a string of labels which characterize the speech waveform locally, typically over a period of several to several hundred milliseconds. The linguistic decoder translates the labels into a hypothesis word sequence.

From the point of view of information theory, the channel consists of the speaker together with the acoustic processor. We are free to chose the parameters of the acoustic processor, but unlike the traditional model, the code is fixed by the language itself.

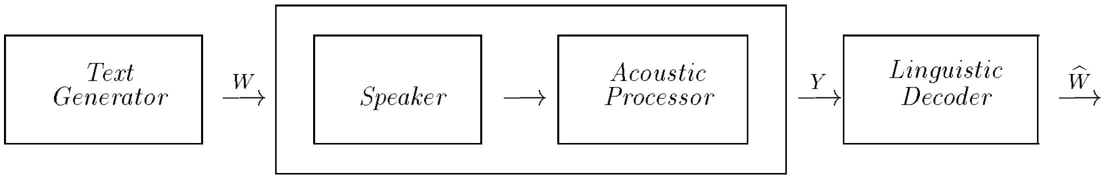

Figure 2: The communication theory view of speech recognition

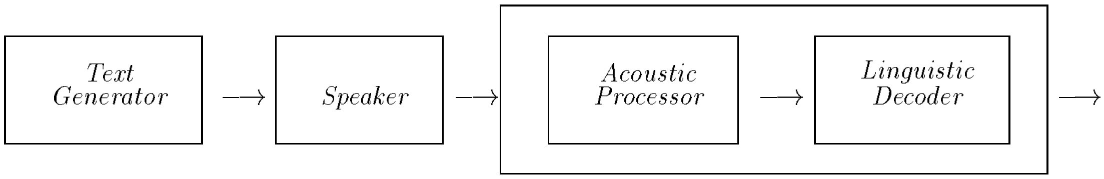

Figure 3: A continuous speech recognition system

The acoustic processor produces a string of labels A. From this, the decoder makes an estimate $\widehat{W}$ of the word string that the text generator output. To minimize the probability of error, $\widehat{W}$ should be chosen so that

$$p(\widehat{W} \mid A) = \max_W p(W \mid A) \tag{11}$$

By Bayes' theorem, this is equivalent to setting

$$\widehat{W} = \arg\max_W \, p(W) \, p(A \mid W) \tag{12}$$

where $p(W)$ is the *prior* probability of W, called the *language model*. The *acoustic model* $p(A \mid W)$ is currently typically constructed using hidden Markov models.

1.3 Treebank Mapping

Many statistical parsing algorithms are carried out by *supervised learning* algorithms on a corpus that has been hand-labeled with parses. It is, in general, very time consuming and expensive to construct such corpora. Suppose that we are interested in building a parser for a grammatical formalism Π, and we have a corpus with Π-labeled text. There are techniques, which we will discuss later on, for building a distribution $p(\pi \mid x)$ for parses of a given input string x. Using this distribution, a maximum likelihood parser is constructed by choosing the parse

$$\widehat{\pi} = \arg\max_\pi \, p(\pi \mid x) \tag{13}$$

The parser will, in general, significantly improve as the amount of training data increases.

Now suppose that we also have another labeled corpus, but this corpus is marked up in terms of another grammar or formalism—call this the Γ-labeled text. How can be make best use of both corpora? There is a natural answer using the noisy channel paradigm, if we have some of the data labeled in *both* formalisms.

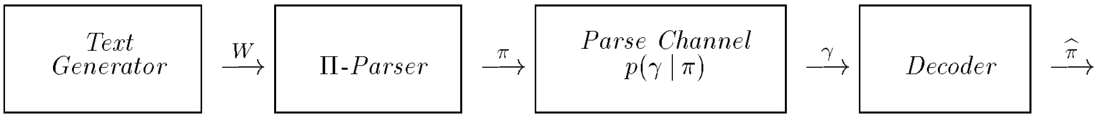

Figure 4: Channel model for treebank mapping problem

The idea of *treebank mapping* is to apply the channel model to transform a Γ-labeled sentence into a Π-labeled sentence. Using a statistical parsing model $p(\pi \mid x)$ constructed from the Π-labeled corpus as a prior, and a *translation model* $p(\gamma \mid \pi, x)$ constructed from the portion of the data that is labeled in both formalisms, Bayes theorem is applied as in all of the other examples:

$$\begin{aligned} \widehat{\pi} &= \arg\max_{\pi} p(\pi \mid \gamma, x) && (14) \\ &= \arg\max_{\pi} p(\gamma \mid \pi, x)\, p(\pi \mid x) && (15) \end{aligned}$$

This results in a decoder that can label the Γ-corpus in the Π formalism. The effect is to use the Γ-parses as a strong "hint" as to the correct Π-parse. This can then be used to relabel the second corpus in the formalism of the first, and results in a larger set of data for training new parsing models $p(\pi \mid x)$. (By the data processing inequality, we haven't gained any new information, but it is significantly more convenient to carry out the mapping!) In this way a statistical parser can be *bootstrapped* as a series of increasingly powerful parsers.

1.4 Other applications

In class we will discuss several other examples including statistical machine translation [4, 5], translation-dictation [3, 6], spelling correction, optical character recognition and document image decoding [9], speech decoding with "auxiliary information" [10].

References

[1] A. Albanese, J. Bloemer, J. Edmonds, M. Luby and M. Sudan. Priority encoding transmission. In *Proc. Foundations of Computer Science*, Santa Fe, NM 1994, pages 604–612.

[2] L. R. Bahl, F. Jelinek, and R. L. Mercer. A maximum likelihood approach to continuous speech recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-5(2):179–190, 1983.

[3] J. Brousseau, G. Foster, P. Isabelle, R. Kuhn, Y. Normandin, and P. Plamondon. French speech recognition in an automatic dictation system for translators: the TransTalk project. In *Proceedings of Eurospeech*, pages 193–196, Madrid, 1995.

[4] P.F. Brown, J. Cocke, S.A. Della Pietra, V.J. Della Pietra, F. Jelinek, J.D. Lafferty, R.L. Mercer, and P.S. Roossin. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85, June 1990.

[5] P.F. Brown, S.A. Della Pietra, V.J. Della Pietra, J.D. Lafferty, and R.L. Mercer. Analysis, statistical transfer, and synthesis in machine translation. In *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 83–100, 1992.

[6] P. Brown, S. Chen, S. Della Pietra, V. Della Pietra, S. Kehler, and R. Mercer. Automatic speech recognition in machine aided translation. *Computer Speech and Language*, 8:177–187, 1994.

[7] T.M. Cover and J.A. Thomas. *Elements of Information Theory.* John Wiley, 1991.

[8] G. Davis and J. Danskin. Joint source and channel coding for image transmission over lossy packet networks, preprint, 1996.

[9] G. Kopec and P. Chou, Document image decoding using Markov source models. *IEEE. Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 6, June, 1994.

[10] P. Placeway and J. Lafferty. Cheating with imperfect transcripts. In *Proceedings of the ICSLP*, Philadelphia, 1996.