Contents

1 Introduction and Recap

1.1 Talk from Larry Abbott

Last week, we started with an excellent talk from Larry Abbott which brought together many aspects of what we've talked about this semester.

1.2 Reinforcement Learning

The Rescola-Wagner model is a model of classical conditioning, with plasticity:

$$w \to w + \epsilon \cdot (R - x) \cdot w \tag{1}$$

And used gradient descent with error

$$\frac{1}{2}(\hat{r} - r)^2, \tag{2}$$

where $\hat{r}$ is the predicted reward and r is the true reward. This model asserts associations do not increase by same increment. Yet Rescorla-Wagner plasticity has an issue: it only models reward at the current timestep, which is to say that it does not model foresight.

Sutton and Barto subsequently gave us the framework of Reinforcement Learning, in which an agent predicts the discounted value of states instead of per-timestep rewards:

$$v = \sum_t \gamma^t r_t \tag{3}$$

Rescorla-Wagner plasticity corresponds precisely to reinforcement learning with $\gamma = 0$. Something like reinforcement learning may occur in the brain, and there is a precise region which we believe predicts value, with dopamine representing the prediction error $\hat{r} - r$, in the notation of the above.

We then discussed the multi-armed bandits problem, an important example problem for reinforcement learning. In the problem, you are given $a = 1 \dots m$ actions, where each has an unknown reward distribution. You want to maximize total reward obtained in the long run. One machine M^* has the greatest expected value, but you don't know which it is. Each machine has an unknown gap $G_a = M^* - M_a$ and you want to minimize regret:

$$\text{Regret}(T) = \sum_{t=1}^{T} G_a(t) \tag{4}$$

Here we assume an undiscounted model with $\gamma = 1$.

There are several popular algorithmic approaches to the bandits problem. A greedy algorithm always takes the action with the highest mean expectation so far, perhaps randomly exploring some small fraction of the time.

As an alternative approach, one might want to add a punishment term to discourage staying in a small set of actions with high mean expectation. The UCB1 algorithm implements this idea, additively boosting the mean reward of each arm by $\sqrt{\frac{log(t)}{n}}$, where t is the current timestep and n is the number of times that action has been sampled previously.=

Thompson sampling is another algorithm, involving the maintenance of a parametrized model of the reward distributions. Given the current parameters, you sample rewards and pick best action. You then update the internal parameters depending on the result. Thompson sampling achieves the lower bound of regret, making it an optimal solution to the multi-armed bandits problem.

A more advanced model considers the case where each bandit has a Markov chain inside of it. You are given the choice of a Markov chain M and a terminal one state Markov chain R. On termination, the reward is the sum of expected rewards from M until the final state R. The Gittens index is the smallest R for which you would not touch M. According to

Gittins's Theorem, an optimal policy is one which chooses the machine with the highest Gittins index.

Afterwards, we talked about Markov Decision Processes (MDPs), in which there is a single MDP shared across all arms (i.e. actions). It might seem that MDPs are an exponential time problem, but it turns out that there are at least two ways to solve it with linear programming. One way is to write the Bellman equation where the value of a state is given by

$$V(x) = \max_a R(s,a) + \gamma \mathbb{E}_a(V(s'), \tag{5}$$

with s' the successor of s. There is, however, a problem: many practical problems have exponentially many states. Chess, for example, has on the order of 10^{50} states. Deep Reinforcement Learning is one approach to deal with these exponentially many states, and it is very similar to traditional (tabular) reinforcement learning. The main difference is that we parametrize the policy by a deep neural network, and optimize it by SGD, estimating the gradient with Monte Carlo samples from the Markov chain (plus lots of tricks).

2 Talk from Dan Mitropolsky: The Brain through Language

Languages that exist in human brains are a consequence of or a reflection of the human mind. That said, there are many ways to answer the question "what is language?" The psychological viewpoint sees it as something internal, in the brain, whereas the anthropological viewpoint is more external: Hockett's design features are twelve features that distinguish human language from animal languages. For example, human language is specialized to communication (unlike dogs panting, which is also for temperature regulation), and it is arbitrary (in the sense that not all words are onomotopoetic). The linguistic answer involves the intersection of 5 different subfields of linguistics: phonology, morphology, syntax, semantics, and pragmatics. These roughly correspond to 5 subprocesses that occur in the brain, each of which is extremely complex. Children, however, learn all of this intuitively in an unsupervised way through language exposure alone.

This talk will begin with a brief overview of these subfields of linguistics, with the exception of pragmatics.

2.1 Phonology

Phonology is the structure of phonemes in a language, the smallest units of sound which compose spoken language. Phonology is one of the most challenging aspects of English to foreign speakers. Most languages have relatively simple vowel tables (a vowel table covers the phonetic map for a given language). The vowel table in Spanish, for example, can be represented by a 3 by 2 grid with 6 pronunciations. English, however, has a very complex one. As another example, Swedish has an even more complex vowel table.

2.2 Morphology

Morphology is the study of combinations of phonemes (not necessarily words). Expletive infixation is an amusing phenomenon in which there are complex rules governing where exactly expletives can be placed within words. For example, the word fucking is a morpheme, and it should be placed *between* other morphemes. "Fan-fucking-tastic" somehow sounds better than "fanta-fucking-stic." As an example of the importance of morphology, Arabic has a complicated system based on templates for combining morphemes into words.

2.3 Syntax

Syntax is the next level in the hierarchy, and it operates at the level of a sentence. Some say that syntax is the most varied aspect of linguistics; others say it is the most shared. At the surface level, all syntaxes seem quite different, but underneath they are all some version of a context free grammar. A syntactic fun fact is that adjective orders are usually fixed within languages, including in English.

Japanese is an example of a language that has almost exactly the reverse syntax relative to English. It is referred to a head-final language, as opposed to English which is head-initial. Both languages still place the topic or subject at the begninning of the sentence. Some sentences in Japanese are ambiguously parseable. Slavic apparently doesn't have word order, so morphology takes the place of syntax in some way. You change words instead of changing word order to change meaning.

2.4 Semantics

Semantics is the study of meaning, and this is where math comes in. Words can be accessed semantically and phonetically: people can list words that relate to a semantic group, like animals, but we can also list words that rhyme with a prompt. There is some way in which your brain generates the syntax of a sentence and then sends it through a phonology pipeline. But the different systems in the brain are intertwined, since phonology can also be used to modify meaning and access words.

2.5 Variety vs Unity

There are certain similarities between human languages, but also often exceptions. One consistency is that no language in the world depends on counting the position of a word in the sentence, and it would be interesting to see if human children are capable of learning such a language. Another consistency is that all grammar can be viewed as tree-based. However there are debates on what accounts for diversity in language–can we view languages as having principles and parameters that change whether, say, the language is head-final or not? Furthermore, Greenberg's linguistic universals attempt to describe what languages all have in common, but there are always exceptions.

2.6 Sapir-Whorf Theory

There was a famous anthropologist, Edward Sapir, who was a student of the father of anthropology. He was known for creating a field of research which involved living with a community and studying their language. In his case, he lived with Native Americans on the West Coast. Benjamin Lee Whorf was Sapir's student/protege who lived with the Hopi. He came up with the hypothesis that language has something to do with the way humans think. In the strongest version of this hypothesis, language *is* the way we think, and in a weaker version it simply influences it in some way.

2.6.1 Weak Evidence for Sapir-Whorf

Some studies have shown that Chinese students are better at arithmetic than Indo-European students, and argued that it is because Chinese numbers are mono-syllabic and very structured. Interesting fact: people almost always think of numbers in their native language. Gender is also interesting, as in languages with it you always have to think of gender. It's very difficult for Chinese natives, for example, to learn the distinction between "he" and "she" in English. Colors are interesting too: Hungarian has two deep words for red that are truly different, and there have been some studies testing whether Hungarians are better at recognizing shades of red, to varying degrees of success. As an aside, there is a fixed order in which colors are added to languages, and there are weird remnants of this even in modern languages. In Japanese, for example, green street lights are called blue because the word for green was added to the language afterwards. Inuit has more root words for snow than English, but it's unclear what this says about the ways in which they think relative to English speakers. Using these examples of evidence as proof is often circular, and it is unclear whether the choices made in language represent differences in structuring or actual thought.

2.6.2 Stronger Evidence for Sapir-Whorf

Sign languages are fully independent natural languages, and are not functions applied to existing languages. ASL has syntax very similar to Japanese, for example, and Japanese sign language is the only East Asian language with gender. Some studies have shown that speakers of sign languages are slightly better at spatial tasks.

In the language Guugu Yimithir, they don't use ego-centric locations, for the most part. They never use left and right, they use compass directions instead. This certainly changes the way they think, as they always have to keep in mind their orientations.

2.6.3 Research of Daniel Everett

Researcher Daniel Everett studied Piraha, a language in the Amazonian region, that was spoken by a group of people in complete isolation. Piraha only has one word for parent, no numbers (only one, few, and lots), and most importantly no recursion (we think). For recursion, Everett claimed that they could not generate sentences such as "I told mom that

dad said I'm sick", but only "I told mom that I'm sick." However the claim is the most dubious for reasons described below.

2.6.4 Evidence Against Sapir-Whorf

Chomsky believes in a deeply universal grammar that is shared across humans, and believes that this language in Brazil isn't a problem. To Chomsky, the lack of recursion just means that they have to break things up into separate sentences, and doesn't mean that they think in a non-recursive way.

2.7 Alinguality

Another question you could ask is: does having a complex, recursive language enable us to have complex thought? There is an agreed-upon critical period during which people have to be exposed to natural language in order to ever acquire language during their lifetimes. Feral children, for example, are unfortunately sometimes not exposed to language during this critical period. Such alingual people appear not to have the same cognitive abilities as others, but this may just be because of a lack of socialization. Some people acquired language extremely late, so they remember what life was like pre-language, and it's interesting to ask such people what life was like before they spoke any language. One well-known case is that of Ildefonso, though unfortunately he never gave a clear answer when asked what life was like before he learned to speak. What is relatively well-established is that bilingualism helps the development of cognition. Interestingly, a child will learn language perfectly without any interaction with adults (i.e. in a totally unsupervised way).

3 Origin of Language

How did language come about? Some 3 million years ago, the homo group separated from the chimps, but only homo sapiens seem to have had language. Our main evidence is the lack of trappings of symbolic behavior such as figurative art. 80 thousand years ago the first figurative art in Africa was made, and it is believed that language came about at about the same time. In 1866, the French Academy banned discussion of the origin of language. Chomsky once said that studying language has to be scientific, and Lewontin says that studying language is tricky because we don't have an example of it in another species. Stephen Pinker, Canadian-American cognitive psychologist, has likened our fascination with humans with elephants admiring their trunk. An important question is whether language came about in a cognitive Big Bang or gradually. Corballis has a theory that language started gesturally. Fun fact: English and Japanese are separated by the changes effected by about four thousand mothers teaching language to their children.

4 Talk by Michael Collins: Into to NLP

NLP is about getting computers to understand language, or to generate language. Various problems include machine translation, information extraction (structuring unstructured data), text summarization, dialogue systems. One of the most basic problems is part of speech tagging, as is named entity recognition. Parsing is also a fundamental question showing the grammatical structure of a sentence, and there are linguistic theories for the rules that go into these kinds of systems. Language is incredibly ambiguous, and there are lots of semantically incoherent but grammatically correct parsings for sentences. Rule-based systems date back to the 80s, but these problems with ambiguity became insurmountable. There was a shift in the early 90s to statistical methods, where instaed of trying to explicitly write down rules for languages, we try to form problems in terms of supervised machine learning problems. There are lots of naturally occurring translation data, including the Canadian parliament which was in both English and French, and the European parliament which is in about 25 languages. There have also been efforts to manually parse tens of thousands of sentences as training data for machine learning-based parsers.

4.1 Language Modeling

The language modeling problem is as follows: given a training sample of example sentences in a language, we are tasked with inducing a probability distribution over sentences. This is a practical problem for speech recognition, in which the acoustics give a distribution over phonemes, and a language model gives a distribution over words. This is common in information theory, in which you have a predictive distribution of the channel, as well as a distribution over what is being communicated given the contents of the channel. Language modeling can also often recover interesting latent structure in text. Shannon estimated peoples' ability to model language by asking them to predict the next character or the next word. Chomsky said that grammar is distinct from likelihood, giving examples of two sentences, one which is ungrammatical, and one which is grammatical but semantically incoherent. In a language model, we wish to estimate the distribution

$$p(w_i|w_1, w_2, \ldots, w_{i-1}) \tag{6}$$

4.2 Log-Linear Models

There is a general machine problem in which we have an input domain $\mathcal{X}$, a finite label set $\mathcal{Y}$, and wish to provide a conditional probability $p(x|y)$, for $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. For language modeling, x is the "history" $w_1, \ldots, w_{i-1}$, and y is w_i.

Feature vector representations use features, functions $f_k(x, y) \in \mathbb{R}$ (often features are binary), and if we have m features, then we have a feature vector representing (x, y). For example, we might have f_1 represent whether y is the word "model", f_2 represent whether y is "model" and w_{i-1} is "statistical". 5-gram features are more or less the limit, and for many years 3-gram features were the best we could do.

Log-linear models use these m features, and learn a parameter vector $v \in \mathbb{R}^m$, and represent $p(y|x;v)$ as a softmax over $v \cdot f(x,y)$. Then we take a loss function

$$L(v) = \sum_i \log p(y_i|x_i, v) - \frac{\lambda}{2}\|v\|^2 \tag{7}$$

Log-linear models were defined by Physicists at IBM to be entropy based. The feature vectors are not learned, such that a trained neural net can be applied.

4.3 Dependency Parsing

Dependency parsing involves recovering a rooted, planar tree of word dependencies. Labeled dependency parsing is very similar, except dependencies are labeled by their linguistic functions. One difficulty is that grammatically correct parsings are in general not unique. An example of an ambiguous sentence is "Jim drove down the street in his car". In one parsing, the car is on the street, and in the other the street is inside his car. Shift reduce models are one way to do dependency parsing. A configuration consists of a stack consisting of a sequence of words (potentially including the root), e.g. $\sigma = [\text{root}_0, \text{I}_1, \text{live}_2]$, a buffer consisting of a sequence of words, e.g. $\beta = [\text{in}_3, \text{New}_4, \text{York}_5, \text{city}_6, .{}_7]$, and a set of dependencies α, e.g. $\alpha = \{\text{Live}_2 \to I_1\}$. We start with the stack just being the root. One operation is shift, which takes the top word off of the buffer and puts it onto the stack. Another operation is the left-arc operation, which takes top two words on stack, adds dependency between them in left direction, and the right-arc does the same but for dependencies from the second to last to last word in the buffer. One can match a sentence using both the right and left-arcs. A parse tree can now be represented as a sequence of actions, and we can use a log-linear model to estimate the probabilities of such a sequence. By the chain rule we have that

$$p(a_1 \dots a_m | w_1 \dots w_n) = \prod_i p(a_i | a_1 \dots a_{i-1}, w_1 \dots w_n) \tag{8}$$

Then we would build feature extractors (each feature extractor is a simple function which looks at a word and pulls out a speech tag or dependency label) for the configuration that a sequence of actions corresponds to (e.g. what the top few words on the stack or buffer are, certain dependency relationships between words), often resulting in about 50 features. The features produced can then be used as input into a neural network and as an application of the log-linear model.

4.4 Neural Models

In the early 2000s, Chan and Manning used simple neural models to outperform the log-linear models that had long been the state of the art. This involved greedy, or beam, search, neural networks, and global training. In a common model of the time, one would first embed each of the features in some Euclidean space, then concatenate these representations and pass

them through a simple feed-forward neural network to get a representation of a configuration. Then one could use the same softmax as with the log-linear model to estimate the probability of various actions given this configuration. Embeddings can either be initialized randomly, or learned on language modeling tasks and either fixed or fine-tuned for the parsing task.

4.5 Recurrent Neural Models

Often in NLP one wants to map a sequence $x_1 \dots x_n$ to a label y or a distribution $p(y|x_1 \dots x_n)$. One way to do this is through recurrent neural nets, whereby we have some representation $h_t = g(h_{t-1}, x_t)$, and output $R_t = f(h_n)$. The simplest example has g as a linear function of x and h passed through a simple nonlinearity, and f linear in h as well. Simple RNNs run into issues of vanishing or exploding gradients, intuitively because over many timesteps we multiply many matrices together when computing gradients, and LSTMs help this by replacing the multiplication with addition.

Perplexity is a measure of the performance of a language model. Having a perplexity of P means roughly that the information theoretic uncertainty in the model is equivalent to a uniform distribution over P words.

RNNs can also be used for dependency parsing as well as language modeling. One very effective way to do this is to first run a Bi-directional LSTM (Bi-LSTMs) over the words, and since each word has a single parent, take the resulting concatenated bi-directional representations for each words, and pass pairs of representations into a feedforward neural net to estimate the likelihood that one is the parent of the other.

In machine translation, one way to use LSTMs is to map a sequence of words in one language first to a sequence of abstract representations with BiLSTMs, then to generate words in the other language recurrently with an LSTM based on the current hidden state of the generating network and on an attention-based weighted combination of the abstract representations of the input words. As with language modeling, in machine translation, you can use greedy search or beam search to output sentences. Neural models are dramatically (60-80%) better than the best pre-neural models, phrase-based machine translation systems.

Moving forward, linguists are excited about joining neural techniques with more explicit representations of linguistic objects like parse trees.

5 How Language Came About

No human species other than homo sapiens created advanced works of art. 80,000 years ago we see the first figurative art in Africa, which corresponds to one of the times language is thought to have evolved (although other theories suggest it may have evolved 500,000 years ago). There is no concrete evidence to prove when the genes for language and figurative art first developed, although verbal abstraction was certainly important in cognitive development. The capacity for language may have benefited early humans without necessarily giving them aural language, instead offering the ability to internally plan and create associations.