Information Theory and Associative Word Learning

Brendan Burns
University of Massachusetts Amherst
bburns@cs.umass.edu

1 Introduction

Information theory has its roots in optimizing communication in the limited domain of a networks. However, over the years, it has been seen to have a wide range of applicability in areas like linguistics, artificial intelligence and cognitive science. ([**?**],[**?**]) More recently work has examined the application of information theory to finding episode boundaries in unsegmented time series [P. Cohen, 2002]. Much of this work is motivated by the belief that cognitive development is served by information theory. Specifically, that learning serves to reduce the entropy of the observed world.

Associative word learning is a natural place to examine this belief. Language learning begins relatively early on in cognitive development, and it is believed by many to lay the groundwork for much of the later development that takes place. Associative word learning can be seen as the categorization of the world into observation to word (and back) pairings. An agent learning to associate a word, or phrase, with an observation of the world, is learning a category of sensory observations which are successfully classified by that word or phrase. In this way word learning serves to carve up the world into areas which a word describes with high probability. The greatest success for the agent comes from finding the associations between words and its observations which result in the greatest improvement in its categorization ability. This improvement is easily measured by the mutual information between a word and an observation. The mutual information between an observation and a word is a measure of the reduction in entropy of the observations given the word.

In what follows we examine the existing state of the art in associative word learning and the use of mutual information for linguistic and other types of learning. We will then examine the learning algorithm we have chosen to apply to this problem and the differences between our approach and existing ideas. Finally an experimental domain and results are presented for the technique we have developed.

2 Related Work

2.1 Using Mutual Information in AI

Mutual information has been put to a wide variety of uses in the Artificial Intelligence community. In this work we will focus on efforts which are closely related to our own.

Pedersen [Pedersen, 2001] utilized both mutual information (in the form of the Dice coefficient) and the G statistic to select word bigrams to use as feature sets for learning decision trees to decide ambiguous word meanings. Mingers [**?**] also suggests the G statistic for decision tree learning.

Lowe [W.Lowe, 1997] utilized the G statistic, which he notes is "similar to mutual information" to build vectors of words which co-occur with high frequency from the British National Corpus of ten million words. He subsequently used these vectors to train self organizing maps, his model of lexical-semantic representation.

2.2 Existing Associative Word Learning

The problem of associative word learning has been examined a number of times since it was first proposed by Locke [**?**].

Oates et al [**?**] used mutual information to cluster words which had similar syntactic nature. Meaning was attributed to these clusters of words by estimating the probability that some cluster of words would co-occur with a sensory experience. These sensory experiences were time sequence observations of the state of the world which were clustered into prototypical experiences using the dynamic time warping algorithm.

This work was later expanded by Oates [**?**] in the PERUSE algorithm which segmented raw speech data, rather than using textual descriptions. The algorithm seg-

mented the speech stream in a manner similar to its segmentation of the sensor streams and learned the associations between the two representations.

Steels [**?**] uses reinforcement learning to learn associations between words and experiences. In his work the words are obtained from raw speech using off the shelf speech recognition software. Steels sensory perceptions where limited to simplified color histograms, clustered using nearest neighbor clustering. Additionally the robot in these experiments had no internal sensors so it was incapable of learning about words which relate to itself. In fact, the only words that Steels learns are the names of the three objects, only one of which is present in each of the robot's experiences.

2.3 Denotational versus Functional Meaning

In all of these cases the algorithms learned the denotational meanings of the words. That is, they learned the description of the sensor values pointed to by the word. Although Oates showed that these descriptions could be used for sensor recognition as well as word recognition, denotational meaning lacks some of the important aspects of meaning. Most importantly, denotational meaning does not allow an agent to (re)construct the thing which contains the meaning. The agent may "know it when it sees it", but it has no way of knowing how to build/enact what it knows. For example, the agent may be able to recognize, after the fact, that it has moved forward but it can not move forward based upon the denotational meaning it has mapped to "move forward".

In this work we would like build an associative learner which can learn functional meanings for words. We define a functional meaning to be a meaning which can be modeled by the agent. This model allows the agent to construct the word's context and situation in the agent's "imagination". For example, if the word is a verb the agent can use its meaning to enact the action described by the word, or model the effects of the action on other objects in the world. If the word is a noun, its meaning can be used by the agent to imagine its use. For example, the functional meaning of the word "cup" is such that the agent can model the knowledge that if liquid is poured into the cup, the liquid will be contained by the cup. The functional meaning of the verb "turn" contains the knowledge of an object's changing heading.

By and large functional meanings can be obtained by a transition from raw sensory representations to symbolic models. At this time we do not feel the need to weigh in on whether these symbolic models are learned or innate, but rather we feel that functional meanings (and thus symbolic models) are necessary for more complicated word learning to occur.

3 Associative Word Learning through Mutual Information

Since we would like to show that information theory is an adequate explanation for word learning in cognitive development, we need an algorithm for learning associations which utilizes information theory. We have used the Multi-Stream Dependency Detection algorithm (MSDD) developed by Oates et al [**?**], which is described below (3.1). The MSDD algorithm uses the G statistic to perform learning. By relating the G statistic for two distributions to the mutual information of the same distributions (3.2), we show that MSDD is an information theoretic learning system. The final sections (**??**, **??**) delineate the representations upon which learning takes place.

3.1 MSDD

The Multi-Stream Dependency Detection (MSDD) [**?**] algorithm finds dependencies between pairs of tokens. Each token is a representation of the state of an arbitrary number of streams at an instant in time. The token is a tuple with an item for each stream. The item may either be a wildcard "*" or a member of the set of possible values for that stream.

Given two tokens T_1, T_2, the algorithm operates by constructing the 2 x 2 contingency table shown in table 1.

Table 1: An example contingency table used by the MSDD algorithm

	T_1	$\neg T_1$
T_2	$O(T_1, T_2)$	$O(\neg T_1, T_2)$
$\neg T_2$	$O(T_1, \neg T_2)$	$O(\neg T_1, \neg T_2)$

If the G statistic for this contingency table is significant then we can dismiss the null hypothesis that T_1 and T_2 are conditionally independent, thus we can conclude that there is a statistical relationship between T_1 and T_2.

Obviously, the space of possible tokens is quite large. In order to obtain a solution in a reasonable amount of computation, the MSDD algorithm begins with T_1 and T_2 filled entirely with wild-cards. It then proceeds with a guided search down the tree which is built by expanding tokens. Tokens are expanded by changing a wildcard into its possible values. This search tree is pruned by a bounds on the G statistic.

3.2 Relating G to mutual information

Although the MSDD algorithm utilizes the G statistic to select word associations we would like to show that this

is actually an information theoretic measure.

Given a discrete distribution of pairs (x, y) drawn from the cartesian product of the sets X and Y, relate the G-statistic of the distribution of pairs to the mutual information between X and Y. Let $O(x, y)$ be the observed count for the pair (x, y), and $E(x, y)$ be the expected count under the hypothesis that X and Y are unrelated. Let $t = \sum_x \sum_y O(x, y)$

$$
\begin{aligned}
G &= 2\sum_x \sum_y O(x,y) \log \frac{O(x,y)}{E(x,y)} \\
G &= 2\sum_x \sum_y O(x,y) \log \frac{O(x,y)}{\frac{(\sum_y O(x,y)) \times (\sum_x O(x,y))}{t}} \\
G &= 2\sum_x \sum_y O(x,y) \log \left(\frac{O(x,y)}{\frac{(\sum_y O(x,y)) \times (\sum_x O(x,y))}{t}} \times 1\right) \\
G &= 2\sum_x \sum_y O(x,y) \log \left(\frac{O(x,y)}{\frac{(\sum_y O(x,y)) \times (\sum_x O(x,y))}{t}} \times \frac{\frac{1}{t}}{\frac{1}{t}}\right) \\
G &= 2\sum_x \sum_y O(x,y) \log \frac{\frac{O(x,y)}{t}}{\frac{(\sum_y O(x,y)) \times (\sum_x O(x,y))}{t^2}} \\
G &= 2\sum_x \sum_y O(x,y) \log \frac{\frac{O(x,y)}{t}}{\frac{\sum_y O(x,y)}{t} \times \frac{\sum_x O(x,y)}{t}} \\
G &= 2t\frac{1}{t}\sum_x \sum_y O(x,y) \log \frac{\frac{O(x,y)}{t}}{\frac{\sum_y O(x,y)}{t} \times \frac{\sum_x O(x,y)}{t}} \\
G &= 2t\sum_x \sum_y \frac{O(x,y)}{t} \log \frac{\frac{O(x,y)}{t}}{\frac{\sum_y O(x,y)}{t} \times \frac{\sum_x O(x,y)}{t}} \\
G &\approx 2t\sum_x \sum_y P(x,y) \log \frac{P(x,y)}{P(x) \times P(y)} \\
G &\approx 2tMI(x,y)
\end{aligned}
$$

Thus we can see that as long as t the total number of observations is kept constant, maximizing G is equivalent to maximizing mutual information. Fortunately for us, the total number of observations is constant for any pair of tokens associated by MSDD. This can be seen by observing that there is a fixed number of pairs of observations in a data-set, and each of these pairs of datum fall into one of the four table cells for any pair of tokens. Thus the sum of all of the cells of the contingency table is equal for any pair of associated tokens. From this we can conclude that any ranking of associations based upon the G statistic is equivalent to a ranking based upon mutual information.

3.3 Learning phrases rather than words

The previous word association work that has been done has focused upon learning the denotational meaning of individual words, however this framework has limitations. Consider an agent trying to learn the meaning of three simple actions; turning left ninety degrees and turning right ninety degrees and staying still. Suppose further that the agent's actions take place in an environment where to the left of the agent lies a sphere and to the right a cube. Figure **??** shows this hypothetical situation.

Suppose the robot's actions are equally divided between the three actions. When it chooses to turn left it receives the description "The robot turned left." When it chooses to turn right it receives the description "The robot turned right." When it stays still, it receives the description "The robot sits still with a cube to the right and a sphere to the left." Given this data, and an associative learner, the robot cannot learn the meaning of "left" or "right." There is no statistical correlation between turning left and the word "left" nor between the sphere present to the left. This occurs of course because the meanings are different depending on the verb used in the sentence, e.g. "turned left" and "turned right" versus "to the left" and "to the right."

The lesson of this example is that if we are interested in capturing this meaning, we must consider words in context, not just words individually. This is not a new point, many linguists have pointed out the contextual nature of meaning. In fact, Pedersen's work [**?**], discussed earlier, applies the G statistic to the solution of the problem of disambiguating context dependent meanings. Although Pedersen's work assumes that the set of meanings is already known.

Of course, considering whole sentences makes the associative learning task quite difficult. One need only compare Hemmingway and Faulkner to see that two sentences carrying essentially the same meaning may very greatly in size and structure. To simplify the problem, but maintain our ability to learn contextualized meanings, we have chosen to encode our phrases as $(subject, verb, object)$ triples. This representation is analogous to a simplified case frame. Although this representation is greatly simplified in comparison to natural language sentences, it contains enough context to greatly expand the meanings we can learn.

3.4 Learning verbs rather than descriptions

Previously we have argued that functional, rather than denotational meanings should be learned. At the time we mentioned that this shift was really a shift in representation of the robot's experience from sensory experi-

ences to symbolic models. The symbolic models we use are abstractions upon the robot's sensor data which encode learned information. For example instead of using Steels' histogram of the camera's raw image, or Oates' RGB time series as our visual input. We choose to utilize more symbolic representations such as concepts like *in-front(A,B)* or *on(A,B)*. These symbolic relations are programmed in to the robot's perception of the world. We feel justified in doing this since there is a great deal of evidence from the psychological literature that infants in the initial stages of language learning have complex spacial representations of the world [**?**], [**?**].

This enhanced perceptual ability allows the agent to learn more complex meanings, and it partially supports development of functional meanings. To fully support functional meanings the agent must also be aware of the effect its actions have upon the symbolic models of the world. To facilitate this we gave the robot the ability to perceive its current action. In the experiments detailed below, this action was simply a primitive action, however we envision the possibility of using more complex, planned actions as well.

Concretely, the agents perception of the world is a n-tuple, with $n-1$ symbolic perceptions of the world, and one percept containing its current action. These perceptions are the meaning the robot associates with the phrases previously described.

4 Emprical Examination

The development of these techniques would mean nothing without a real world examination of their abilities to learn word associations. To perform this examination we conducted a simple experiment with one of our mobile robots which is discussed in the following.

4.1 Experimental Set-up

The experiments took place on a Pioneer II mobile robot (figure 1). The Pioneer II was given five primitive actions which it could perform; moving forward, moving backward, turning left ninety degrees and turning right ninety degrees or sitting still. Nine digital movies (two of each action except sitting still) were recorded. While the robot was engaged in each action, its perceptual system recorded the following sensor vector *(rotational-velocity translational-velocity)*. $rotational-velocity$ and $translational-velocity$ were each one of the values: *negative, zero, positive*.

Each movie of the robot acting was shown to between eight and twelve people. Each person wrote a textual descriptions of the movies, for example: "The robot rotates ninety degrees and stops, facing away from the viewer." Each of these textual descriptions were manually processed into *(subject, verb, object)* tuples used as the phrases for the associative learner. This resulted in eighty one descriptions with a vocabulary of sixty different words; three words for subject (mostly "robot"), thirty nine verbs and nineteen objects (including the possibility of having no object).

MSDD was then run on the data to find associations between phrases and perception vectors. We used MSDD with k-best tree pruning, with k set to thirty. In k-best pruning, MSDD proceeds with a breadth first search through the space of associations until expanding a new level of the tree does not result in any changes to the k-best associations already found. Since all of the observations were actions, we also required that a verb be present in any of the associations which were learned.

4.2 Results

When MSDD was run with k-best value of thirty, it found the associations shown in table 2.

Table 2: Mapping of words associated with sensor values

Sensor Value	Phrases Associated
(zero positive)	"* moves *", "* moving forward", "* moves forward", "* moving *"
(zero negative)	"* * backward", "* backed *", "* backs *"
(zero zero)	"* idles *", "* resting *", "* stays *", "* sleeps *", "* motionless *", "* standing *"
(positive zero)	"* turning right", "* turning *", "* turning clockwise", "* turns clockwise"
(negative zero)	"* turning left", "* spinning left", "* turns counter-clockwise", "* turns left", "* turns *"

4.3 Discussion

The first thing that is interesting to note is that the subject is always wildcarded in the learned phrases. This is unsurprising given that seventy nine out of the eighty one descriptions collected used "robot" as the subject. As a result, the subject was never closely correlated with any particular action. Obviously this is a by product of the robot being the subject in all of the actions. In the future we will need to explore situations where the subject varies.

Although by and large the associations learned by MSDD are accurate meanings, three (* moves *) $\rightarrow$ (zero

Figure 1: The robot engaged in the left turn action

positive), (* turning *) → (positive zero) and (* turns *) → (negative zero). In each of these cases the meanings learned, while accurate, are overly specific. This is a coincidence of the descriptions resulting from the describers using "moves" much more often with "forward" than "backward" (they were more likely to use "backs", "backing", etc) Likewise, for some reason "turns" was used with "left", while "turning" was used with right. These results are symptomatic of the very real problem that accidental correlation can (and most likely will) occur. Any associative learner will learn correlations which exist in the data, but not in the real world. One potential solution to this problem is the addition of hypothesis testing, or a language game such as those proposed by Steels to provide a mediator to aid the learning system in correcting misinterpretations.

5 Conclusions/Future Work

We have sought to show that information theory is a plausible explanation for word learning in agents. To do this we have presented an associative learning system which selects associations based upon maximizing the mutual information in the association.

Additionally we have discussed the previous work in associative word learning and our perceptions of their shortcomings resulting from mapping individual words to meanings, and the use of denotational meanings. To remedy this we have proposed the use of functional meanings which allow the agent to utilize its meaning to model the action being described.

There are many ways in which this work can be expanded. Initially we have dealt only with primitive actions on the part of the robot. We would like to expand this to include more complex planned actions. Additionally we see language as a tool that allows an agent to plan its actions. E.G. a statement like "come here and pick this up" implicitly encodes a plan if you have a meaning for "come here" and "pick this up".

Like Steels we believe that a language game is important for successful language acquisition. A language game can introduce an active mediator to assist the agent's learning process. It can also necessitate the agent's use of functional meanings. A game can be designed which an agent with denotational meanings can not play.

Finally we would like to scale this work to truly large scale vocabularies. It is still an undecided problem if associative word learning can really learn large vocabularies, and this is a question we wish to explore.

References

[P. Cohen, 2002] N. Adams P. Cohen, B. Heeringa. An unsupervised algorithm for segmenting catergorical time series into episodes. In *2002 IEEE Conference on Data Mining*, 2002.

[Pedersen, 2001] T. Pedersen. A decision tree of bigrams is an accurate predictor of word sense. In *Proceedings of the second annual meeting of the North American chapter of the Association for Computational Linguistics*, pages 79–86, 2001.

[W.Lowe, 1997] W.Lowe. Semantic representation and priming in a self-organizing lexicon. In *Proceedings of the 4th Neural Computation and Psychology Workshop*, pages 227–239. Springer-Verlag, 1997.