Bayesian nets in probabilistic TTR

Staffan Larsson

*Joint work with Robin Cooper, Simon Dobnik, and Shalom Lappin,
University of Gothenburg

CLASP seminar 2016-04-06
(with some edits based on discussion at the seminar)

Outline

Probabilistic Semantics

Rich Type Theory and Probabilistic Types

Semantic Classification and Learning

Conclusions and Future Work

Outline

Probabilistic Semantics

Rich Type Theory and Probabilistic Types

Semantic Classification and Learning

Conclusions and Future Work

Classical Semantic Theories

- Classical semantic theories (Montague (1974)), as well as dynamic (Kamp and Reyle (1993)) and underspecified (Fox and Lappin (2010)) frameworks use categorical type systems.
- A type T identifies a set of possible denotations for expressions in T.
- The theory specifies combinatorial operations for deriving the denotation of an expression from the values of its constituents.
- These theories cannot represent the gradience of semantic properties that is pervasive in speakers' judgements concerning truth, predication, and meaning relations.

Semantic Classification and Learning

- There is a fair amount of evidence indicating that language acquisition in general crucially relies on probabilistic learning (Clark and Lappin (2011)).
- It is not clear how a reasonable account of semantic learning could be constructed on the basis of the categorical type systems that either classical or revised semantic theories assume.
- Such systems do not appear to be efficiently learnable from the primary linguistic data (with weak learning biases).
- There is little (or no) psychological data to suggest that classical categorical type systems provide biologically determined constraints on semantic learning.

Using Probability to Model Gradience and Learning

- A semantic theory that assigns probability rather than truth conditions to sentences is in a better position to deal with gradience and learning.
- Gradience is intrinsic to the theory by virtue of the fact that values are assigned to sentences in the continuum of real numbers [0,1], rather than Boolean values in $\{0,1\}$.
- A probabilistic account of semantic learning is facilitated if the target of learning is a probabilistic representation of meaning.
- Both semantic interpretation and semantic learning are characterised as reasoning under uncertainty.

Two Strategies

- On a top-down approach one sustains classical categorical type and model theories, and then specifies a function that assigns probability values to the possible worlds that the model provides.
- The probability value of a sentence relative to a model M is the sum of the probabilities of the worlds in which it is true.
- On a bottom-up approach one defines a probabilistic type theory.
- The probability value of a sentence is the output of a function that encodes probabilistic semantic type judgements associated with its predicative syntactic constituents.

A Top-Down Theory

- van Eijck and Lappin (2012) retain a classical type theory and the specification of intensions for each type as functions from worlds to extensions.
- They define a *probabilistic model* M as a tuple $\langle D, W, P\rangle$ with D a domain, W a set of worlds for that domain (predicate interpretations in that domain), and P a probability function over W, i.e., for all $w \in W$, $P(w) \in [0,1]$, and $\sum_{w\in W} P(w) = 1$.
- An interpretation of a language L in a model $M = \langle D, W, P\rangle$ is given in terms of the standard notion $w \models \phi$:

$$[\![\phi]\!]^M \quad := \quad \sum_{w_i \in W \wedge w_i \models \phi} P(w_i)$$

The Probability Calculus

- This definition of a model entails that $[\![\neg\phi]\!]^M = 1 - [\![\phi]\!]^M$.
- Also, if $\phi \models \neg\psi$, i.e., if $W_\phi \cap W_\psi = \emptyset$, then $[\![\phi \vee \psi]\!]^M = \sum_{w \in W_{\phi\vee\psi}} P(w) = \sum_{w \in W_\phi} P(w) + \sum_{w \in W_\psi} P(w) = [\![\phi]\!]^M + [\![\psi]\!]^M$.
- These equations satisfy the axioms of Kolmogorov's (1950) probability calculus.

Advantages of the Top-Down Approach

- This theory retains classical type and model theories to determine the value of a sentence in a world.
- Therefore, it uses well understood formal systems at both levels of representation.
- It applies a standard probability calculus for computing the probability value of a sentence.

Disadvantages of the Top-Down Approach

- It requires probabilities to be assigned to entire worlds in the model, with sentences receiving probability values derivatively from these assignments.
- Representing worlds (maximally consistent sets of propositions, or ultrafilters in a proof theoretic lattice of propositions) poses serious problems of tractability (Lappin (2014), Cooper et al. (2014)).
- The probability value of a sentence can only be computed relative to those of the other sentences of the language that specify the set of worlds (or possible situations).
- This holism seems to exclude the possibility of learning individual classifiers and type judgements independently of each other.

A Bottom-Up Approach

- A bottom-up approach avoids the representability problem by assigning probabilities to individual type judgements as classifier applications.
- The probability of a sentence is determined relative to a bounded set of situation types, which can be learned as classifiers for situations.
- A bottom-up probabilistic semantics requires a probabilistic type theory.
- This theory provides the basis for an account of semantic learning in which situation type classifiers are acquired probabilistically through sampling and observation driven Bayesian inference.

Austinian Propositions

- We take probability to be distributed over situation types (Barwise and Perry (1983)).
- An Austinian proposition is a judgement that a situation is of a particular type, and we treat it as probabilistic.
- It expresses a subjective probability in that it encodes the belief of an agent concerning the likelihood that a situation is of that type.
- The core of an Austinian proposition is a type judgement of the form $s : T$, which is expressed probabilistically as $p(s : T) = r$, where $r \in$ [0,1].

Outline

Probabilistic TTR: Basic Types and PTypes

Our type system is based on Cooper's (2012) Type Theory with Records (TTR), and it includes the following types.

- **Basic Types** are not constructed out of other objects introduced in the theory.
  - If T is a basic type, $p(a : T)$ for any object a is provided by an assignment of probabilities to judgements involving basic types.
- **PTypes** are constructed from a *predicate* and an appropriate sequence of arguments.
  - *man(john,18:10)* is the type of situation where John is a man at time 18:10.
  - A probability model provides probabilities $p(e : r(a_1, \ldots, a_n))$ for ptypes $r(a_1, \ldots, a_n)$.
  - We take both common nouns and verbs to provide the components out of which PTypes are constructed.

Meets and Joins

- **Meets and Joins** give, for T_1 and T_2, the meet, $T_1 \wedge T_2$ and the join $T_1 \vee T_2$, respectively.
- $a : T_1 \wedge T_2$ just in case $a : T_1$ and $a : T_2$.
- $a : T_1 \vee T_2$ just in case either $a : T_1$ or $a : T_2$ (possibly both).
- The probabilities for meet and join types are defined by the classical Kolmogorov (1950) equations.
  - $p(a : T_1 \wedge T_2) = p(a : T_1)p(a : T_2 \mid a : T_1)$ (equivalently, $p(a : T_1 \wedge T_2) = p(a : T_1, a : T_2)$)
  - $p(a : T_1 \vee T_2) = p(a : T_1) + p(a : T_2) - p(a : T_1 \wedge T_2)$

Subtypes

- **Subtypes**: A type T_1 is a subtype of type T_2, $T_1 \sqsubseteq T_2$, just in case $a : T_1$ implies $a : T_2$ no matter what we assign to the basic types.
- If $T_1 \sqsubseteq T_2$ then $a : T_1 \wedge T_2$ iff $a : T_1$, and $a : T_1 \vee T_2$ iff $a : T_2$.
- Similarly, if $T_2 \sqsubseteq T_1$ then $a : T_1 \wedge T_2$ iff $a : T_2$, and $a : T_1 \vee T_2$ iff $a : T_1$.
- If $T_2 \sqsubseteq T_1$, then $p(a : T_1 \wedge T_2) = p(a : T_2)$, and $p(a : T_1 \vee T_2) = p(a : T_1)$.
- If $T_1 \sqsubseteq T_2$, then $p(a : T_1) \leq p(a : T_2)$.
- These definitions also entail that $p(a : T_1 \wedge T_2) \leq p(a : T_1)$, and $p(a : T_1) \leq p(a : T_1 \vee T_2)$.

Generalized Probabilistic Meet

- Let $\bigwedge_p(a_0 : T_0, \ldots, a_n : T_n)$ be *the conjunctive probability of judgements* $a_0 : T_0, \ldots, a_n : T_n$.
- $\bigwedge_p(a_0 : T_0, \ldots, a_n : T_n) =$

  $\bigwedge_p(a_0 : T_0, \ldots, a_{n-1} : T_{n-1}) p(a_n : T_n \mid a_0 : T_0, \ldots, a_{n-1} : T_{n-1})$
- If $n = 0$, $\bigwedge_p(a_0 : T_0, \ldots, a_n : T_n) = 1$.
- Universal quantification is an unbounded conjunctive probability, which is true if it is vacuously satisfied ($n = 0$) (Paris (2010)).
- **Conditional Conjunctive Probabilities**: $\bigwedge_p(a_0 : T_0, \ldots, a_n : T_n \mid a : T) =$

  $\bigwedge_p(a_0 : T_0, \ldots, a_{n-1} : T_{n-1} \mid a : T) p(a_n : T_n \mid a_0 : T_0, \ldots, a_{n-1} : T_{n-1}, a : T))$.

  If $n = 0$, $\bigwedge_p(a_0 : T_0, \ldots, a_n : T_n \mid a : T) = 1$.

Generalized Probabilistic Join

- Let $\bigvee^{p}(a_0 : T_0, a_1 : T_1, \ldots, a_n : T_n)$ be the *disjunctive probability of judgements* $a_0 : T_0, a_1 : T_1, \ldots, a_n : T_n$.
- $\bigvee^{p}(a_0 : T_0, \ldots, a_n : T_n) =$

  $\bigvee^{p}(a_0 : T_0, \ldots, a_{n-1} : T_{n-1}) + p(a_n : T_n) - \bigwedge_{p}(a_0 : T_0, \ldots, a_{n-1} : T_{n-1})p(a_n : T_n \mid a_0 : T_0, \ldots, a_{n-1} : T_{n-1})$
- If $n = 0$, $\bigvee^{p}(a_0 : T_0, \ldots, a_n : T_n) = 0$.
- Existential quantification is an unbounded disjunctive probability, which is false if it lacks a single non-nil probability instance ($n = 0$).

Function Types

- **Function Types** give, for any types T_1 and T_2, the type $(T_1 \to T_2)$.
- This is the type of total functions with domain the set of all objects of type T_1 and range included in objects of type T_2.
- The probability that a function f is of type $(T_1 \to T_2)$ is the probability that everything in its domain is of type T_1, that everything in its range is of type T_2, and that everything not in its domain which has some probability of being of type T_1 is *not*, in fact, of type T_1
- $p(f : (T_1 \to T_2)) = \bigwedge^{p}_{a \in \mathrm{dom}(f)} (a : T_1, f(a) : T_2)(1 - \bigvee^{p}_{a \notin \mathrm{dom}(f)} (a : T_1))$

Function Types: Example 1

- Suppose that T_1 is the type of event where there is a flash of lightning, and T_2 is the type of event where there is a clap of thunder.
- Let f map lightning events to thunder events, and and let f have as its domain all events which have been judged to have probability greater than 0 of being lightning events.
- Assume all putative lightning events are clear examples of lightning and are associated by f with clear events of thunder.
- If there are four such pairs of events, then the probability of f being of type $(T_1 \rightarrow T_2)$ is $(1 \times 1)^4 = 1$.

Function Types: Example 2

- Alternatively, suppose that for for of the four events f associates a lightning event with a silent event.
- Then the probability of f being of type $(T_1 \rightarrow T_2)$ is $(1 \times 1)^3 \times (1 \times 0) = 0$.
- One clear counterexample is sufficient to show that the function is definitely not of the type $(T_1 \rightarrow T_2)$.

Increasing the Size of the Domain of a Function Type

- If the probabilities of the antecedent and the consequent type judgements are higher than 0, the probability of the entire judgement on the existence of a functional type f will decline in proportion to the size of $\mathrm{dom}(f)$.
- If, for example that there are k elements $a \in \mathrm{dom}(f)$, where for each such a, $p(a : T_1) = p(f(a) : T_2) \geq .5$.
- Every a_i that is added to $\mathrm{dom}(f)$ will reduce the value of $p(f : (T_1 \rightarrow T_2))$, even if it yields higher values for $p(a : T_1)$ and $p(f(a) : T_2)$.
- This is due to the fact that we are treating the probability of $p(f : (T_1 \rightarrow T_2))$ as the likelihood of there being a function that is satisfied by all objects in its domain.
- The larger the domain, the less probable that all elements in it fulfill the functional relation.

Function Type Judgements as Universally Quantified Assertions

- We are interpreting a functional type judgement of this kind as a universally quantified assertion over the pairing of objects in $\mathrm{dom}(f)$ and $\mathrm{range}(f)$.
- The probability of such an assertion is given by the conjunction of assertions corresponding to the co-occurrence of each element a in f's domain as an instance of T_1 with $f(a)$ as an instance of T_2.
- Functions which leave out some of the objects with lower likelihood of being of type T_1 should also have a probability of being of type $(T_1 \rightarrow T_2)$.
- This factor in the probability is represented by the second element of the product in the formula.

Negation and Instantiation of Types

- **Negation**: $\neg T$, of type T, is the function type $(T \rightarrow \bot)$, where $\bot$ is a necessarily empty type and $p(\bot) = 0$.
- It follows from our rules for function types that $p(f : \neg T) = 1$ if $\mathrm{dom}(f) = \emptyset$, ($T$ is empty, and 0 otherwise).
- We also assign probabilities to judgements concerning the (non-)emptiness of a type, $p(T)$.
- Our account of negation entails that $p(T \vee \neg T) = 1$, and (ii) $p(\neg\neg T) = p(T)$.
- Therefore, we sustain classical Boolean negation and disjunction, in contrast to Martin-Löf's (1984) intuitionistic type theory.

Dependent Types

- **Dependent Types** are functions from objects to types.
- Given appropriate arguments as functions they will return a type.
- Therefore, the account of probabilities associated with functions above applies to dependent types.

Record Types

- **Record Types** are sets of ordered pairs (*fields*) whose first member is a label and whose second member is an object of some type, possibly itself a record, where records are functional on labels (each label in a record can only occur once in the record's left projection).
- If T is a record type, ℓ is a label not occuring in T, $\mathcal{T}$ is a dependent type requiring n arguments, and $\langle \pi_1, \ldots, \pi_n \rangle$ is an n-place sequence of paths in T, then $T \cup \{\langle \ell, \langle \mathcal{T}, \langle \pi_1, \ldots, \pi_n \rangle \rangle \rangle\}$ is a record type.
- $r : T \cup \{\langle \ell, \langle \mathcal{T}, \langle \pi_1, \ldots, \pi_n \rangle \rangle \rangle\}$ just in case $r : T$, $r.\ell$ is defined, and $r.\ell : \mathcal{T}(r.\pi_1, \ldots, r.\pi_n)$.

Record Types

The probability that an object r is of a record type T:

1. $p(r : Rec) = 1$ if r is a record, 0 otherwise
2. $p(r : T_1 \cup \{\langle \ell, T_2 \rangle\}) = \bigwedge_p (r : T_1, r.\ell : T_2)$
3. If $\mathcal{T} : (T_1 \rightarrow (\ldots \rightarrow (T_n \rightarrow Type)\ldots))$, then
   $p(r : T \cup \{\langle \ell, \langle \mathcal{T}, \langle \pi_1, \ldots, \pi_n \rangle\rangle\rangle\}) =$
   $\bigwedge_p (r : T, r.\ell : \mathcal{T}(r.\pi_1, \ldots, r.\pi_n) \mid r.\pi_1 : T_1, \ldots, r.\pi_n : T_n)$

Probabilistic Austinian Propositions

- *Probabilistic Austinian propositions* are records of type

$$\begin{bmatrix} \text{sit} & : & \mathit{Sit} \\ \text{sit-type} & : & \mathit{Type} \\ \text{prob} & : & [0,1] \end{bmatrix}$$

- They assert that the probability that a situation s is of type *Type* with the value of *prob*.
- The definition of $[\![\cdot]\!]_p$ specifies a compositional procedure for generating an Austinian proposition (record) of this type from the meanings of the syntactic constituents of a sentence.

Outline

Observations as Type Judgements

- We assume that agents track observed situations and their types, modelled as probabilistic Austinian propositions.
- An observation of a red object might yield the following probabilistic Austinian proposition for some a:*Ind*, s_1:red(a)

$$\begin{bmatrix} \text{sit} & = & \begin{bmatrix} \text{ref} & = & a \\ \text{c}_{\text{red}} & = & s_1 \end{bmatrix} \\ \text{sit-type} & = & \begin{bmatrix} \text{ref} & : & \mathit{Ind} \\ \text{c}_{\text{red}} & : & \text{red(ref)} \end{bmatrix} \\ \text{prob} & = & 0.7 \end{bmatrix}$$

Computing the Probability of a Type Judgement

- When an agent A encounters a new situation s and wants to know if it is of type T or not, he/she uses probabilistic reasoning to determine the value of $p_{A,\mathfrak{J}}(s : T)$.
- This denotes the probability that agent A assigns with respect to prior judgements $\mathfrak{J}$ to s being of type T.

Summing probabilities of type judgements

- An agent makes judgements based on a finite string of probabilistic Austinian propositions, $\mathfrak{J}$.
- For a type, T, $\mathfrak{J}_T$ represents that set of probabilistic Austinian propositions j such that j.sit-type $\sqsubseteq T$.

$$\mathfrak{J}_T = \{j \mid j \in \mathfrak{J}, j.\text{sit-type} \sqsubseteq T\}$$

- If T is a type and $\mathfrak{J}$ a finite string of probabilistic Austinian propositions, then $||T||_{\mathfrak{J}}$ represents the sum of all probabilities associated with T in $\mathfrak{J}$

$$||T||_{\mathfrak{J}} = \sum_{j \in \mathfrak{J}_T} j.prob$$

- $\mathcal{P}(\mathfrak{J})$ is the sum of all probabilities in $\mathfrak{J}$

$$\mathcal{P}(\mathfrak{J}) = \sum_{j \in \mathfrak{J}} j.prob$$

Priors on Type Judgements

- $\text{prior}_{\mathfrak{J}}(T)$ represents the prior probability that anything is of type T given $\mathfrak{J}$.

$$\text{prior}_{\mathfrak{J}}(T) = \frac{\|\ T\ \|_{\mathfrak{J}}}{\mathcal{P}(\mathfrak{J})} \text{ if } \mathcal{P}(\mathfrak{J}) > 0, \text{ and } 0 \text{ otherwise.}$$

A Type Theoretic Bayesian Rule for Conditional Probability

- $p_{\mathfrak{J}}(T_1||T_2)$ is the probability that agent A assigns with respect to prior judgements $\mathfrak{J}$ to some situation s being of type T_1, given that A judges s to be of type T_2.
- A computes these conditional probabilities with the equation

$$\mathrm{p}_{\mathfrak{J}}(T_1||T_2) = \frac{||\ T_1 \wedge T_2\ ||_{\mathfrak{J}}}{||\ T_2\ ||_{\mathfrak{J}}},\ \text{if}\ ||\ T_2\ ||_{\mathfrak{J}} \neq 0$$

Otherwise,

$$\mathrm{p}_{\mathfrak{J}}(T_1||T_2) = 0$$

$$p_{\mathfrak{J}}(T_1||T_2) = \frac{|| \ T_1 \wedge T_2 \ ||_{\mathfrak{J}}}{|| \ T_2 \ ||_{\mathfrak{J}}}$$

where

$$||T||_{\mathfrak{J}} = \sum_{j \in \mathfrak{J}_T} j.prob$$

is equivalent to

$$p_{\mathfrak{J}}(T_1||T_2) = \frac{\sum_{j \in \mathfrak{J}_{T_1 \wedge T_2}} j.prob}{\sum_{j \in \mathfrak{J}_{T_2}} j.prob}$$

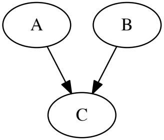
Note that we may not store probabilities for conjunctive types such as $T_1 \wedge T_2$ directly. Instead, we will need to compute them from probabilities of the conjucts T_1 and T_2.

- This is a TTR variant of the standard Bayesian formula for computing conditional probabilities:

$$\mathrm{p}(A \mid B) = \frac{\mid A\&B \mid}{\mid B \mid}$$

- Instead of counting categorical judgements, we are summing probabilities of judgements.
  - Our "training data" is not limited to categorical observations
  - Instead we assume that it consists of probabilistic observations of situations being of certain types
- Still, we keep the assumption from Naive Bayes classifiers that the evidence are independent of each other
- Later, we will remove this restriction

A simple Bayes Net



- A and B are the evidence variables, each with a probilitity distribution over their respective possible values (the probabilities of the values for each variable summing to 1)
  - For a binary variable X, the values are X and not-X
  - For a discrete non-binary variable X, there are more than 2 values
  - (Variables may also be non-discrete)
- C is the conclusion variable

Variable types in TTR

- To do Bayesian inference in probabilistic TTR, we need a notion corresponding to a variable in Bayesian inference
- Assume a single (discrete) variable with a range of possible (mutually exclusive) values
- We introduce an *variable type* V whose range is a set of *value types* $\mathfrak{R}(V) = \{T_1^V, \ldots, T_n^V\}$ such that

$$T_j^V \sqsubseteq V \text{ for } 1 \leq j \leq n$$

$$T_j^V \perp T_i^V \text{ for all } i, j \text{ such that } 1 \leq i \neq j \leq n$$

$$\mathrm{p}(r : V) \in \{0, 1\} = \sum_{T \in \mathfrak{R}(V)} \mathrm{p}(r : T)$$

- V is an *variable type*
- $\mathfrak{R}(V) = \{T_1^V, \ldots, T_n^V\}$ is the set of n possible *value types* such that

$$(1)\, T_j^V \sqsubseteq V \text{ for } 1 \leq j \leq n$$

$$(2)\, T_j^V \perp T_i^V \text{ for all } i, j \text{ such that } 1 \leq i \neq j \leq n$$

$$(3)\ \mathrm{p}(r : V) \in \{0, 1\} = \sum_{T \in \mathfrak{R}(V)} \mathrm{p}(r : T)$$

- (1) says that all value types for a variable type V are subtypes of V.
- (2) says that all value types for a given variable type V are mutually exclusive, i.e. there are no objects that are of two value types for V.
- (3) says that unless the probability of a situation being of a variable type V is 0 (i.e., the variable has no value for the situation), the probabilites of the situation being of each of the value types for V sum to one

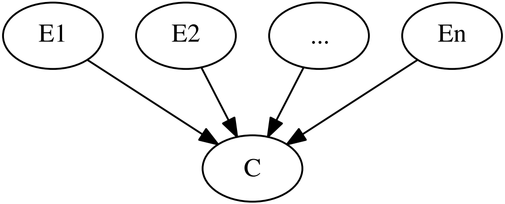
(Aside)

- (3) encodes a conceptual difference between the probability that something has a property (such as colour, p(r:Colour)), and the probability that it has a certain value of a variable (e.g. p(r:Green)).
- If the probability distribution over different values (colours) sums to 1, the probability that the object in question has a colour is 1.
- However, the probability that an object has colour is either 0 or 1.
- We thus assume that there are categorical ontological/conceptual inferences of the type "physical objects have colour" that always have probability 1.

Variable types for evidence and conclusion

Associated with a classifier κ is

- a collection of m evidence variable types $E_1^\kappa, \ldots, E_m^\kappa$
- associated sets of evidence value types $\mathfrak{R}(E_1^\kappa), \ldots, \mathfrak{R}(E_m^\kappa)$
- a conclusion variable type C^κ
- an associated set of conclusion value types $\mathfrak{R}(C^\kappa)$



- To classify a situation r using a classifier κ, the evidence is acquired by observing and classifying r with respect to the evidence types.
- This can be done using another layer of probabilistic classification based on yet another layer of evidence types, or by probabilistic or non-probabilistic classification of low-level sensory readings resulting directly from observations

The TTR Bayes classifier

- We define a TTR Bayes classifier κ as a function from a situation s to a set of probabilistic Austinian propositions defining a probability distribution over the values of the conclusion variable type C^κ, given probability distributions over the values of each evidence variable type $E_1^\kappa, \ldots, E_m^\kappa$.
- If the classifier is a function

$$\kappa : \text{Sit} \rightarrow \text{Set}(\begin{bmatrix} \text{sit} & : & \text{Sit} \\ \text{sit-type} & : & \text{Type} \\ \text{prob} & : & [0,1] \end{bmatrix})$$

such that if r:Sit, then

$$\kappa(r) = \{\begin{bmatrix} \text{sit} & = & r \\ \text{sit-type} & = & T^c \\ \text{prob} & = & \text{p}_{\mathfrak{J}}^{\kappa}(r : T^c) \end{bmatrix} \mid T^c \in \mathfrak{R}(C^\kappa)\}$$

- For our classifier κ, we are interested in the *marginal* probability $\mathrm{p}^{\kappa}_{\mathfrak{J}}(r : T^c)$ of the situation being of a conclusion value type T^c in light of the evidence.
- We obtain the marginal probabilities of the different possible conclusions by factoring, for each evidence variable type, the conditional probabilities of any situation being of the conclusion type given that it is of the various associated evidence value types with the probabilities of r being of the respective evidence types

$$p^{\kappa}_{\mathfrak{J}}(r : T^c) =$$

$$\sum_{\substack{T^{E_1} \in \mathfrak{R}(E^{\kappa}_1) \\ \ldots \\ T^{E_m} \in \mathfrak{R}(E^{\kappa}_m)}} \mathrm{p}_{\mathfrak{J}}(T^c || T^{E_1} \wedge \ldots \wedge T^{E_m}) \mathrm{p}_{\mathfrak{J}}(r : T^{E_1}) \ldots \mathrm{p}_{\mathfrak{J}}(r : T^{E_m})$$

- Note that $\mathrm{p}(T_0 || T_1 \wedge \ldots \wedge T_n)$ is equivalent to $\mathrm{p}(T_0 || T_1, \ldots, T_n)$.

$$p^{\kappa}_{\mathfrak{J}}(r : T^c) = \sum_{\substack{T^{E_1} \in \Re(E_1^{\kappa}) \\ \ldots \\ T^{E_m} \in \Re(E_m^{\kappa})}} \mathrm{p}_{\mathfrak{J}}(T^c \| T^{E_1} \wedge \ldots \wedge T^{E_m}) \mathrm{p}_{\mathfrak{J}}(r : T^{E_1}) \ldots \mathrm{p}_{\mathfrak{J}}(r : T^{E_m})$$

- Note that we are summing across all combinations of all evidence value types for all the evidence variable types associated with κ.
- Note also that we do not here single out one conclusion type as the winner; this is of course possible also (using argmax).
- Instead, the classifier as defined only returns a probability disitribution over the evidence types.
- Of course, an agent sooner or later has to prune some low-probability possibilities and normalise to maintain a Bernouilli distribution over the remaining possibilities.

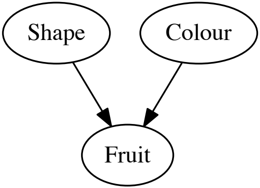
Classification

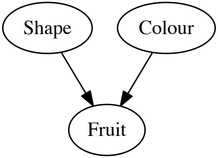
$$p^{\kappa}_{\mathfrak{J}}(r : T^c) = \sum_{\substack{T^{E_1} \in \mathfrak{R}(E^{\kappa}_1) \\ \ldots \\ T^{E_m} \in \mathfrak{R}(E^{\kappa}_m)}} \mathrm{p}_{\mathfrak{J}}(T^c || T^{E_1} \wedge \ldots \wedge T^{E_m}) \mathrm{p}_{\mathfrak{J}}(r : T^{E_1}) \ldots \mathrm{p}_{\mathfrak{J}}(r : T^{E_m})$$

- Our agent gets $\mathrm{p}_{\mathfrak{J}}(r : T^{E_1}) \ldots \mathrm{p}_{\mathfrak{J}}(r : T^{E_m})$ by observing the situation r the she wants to classify
- $\mathrm{p}_{\mathfrak{J}}(T^c || T^{E_1} \wedge \ldots \wedge T^{E_m})$ can be computed in two ways:
  - From a joint probability distribution
  - Using Bayes rule for conditional probabilities
- Obtaining the probabilities required for computing $\mathrm{p}_{\mathfrak{J}}(T^c || T^{E_1} \wedge \ldots \wedge T^{E_m})$ constitutes the learning component

An example: the Apple game

- Teacher shows fruits, agent makes a guess, teacher provides correct answer, agent learns
- Conclusion variable type Fruit with value types: Apple, Pear
- Evidence variable types
  - Col(our), with value types $\mathfrak{R}(\text{Col}) = \{\text{Green}, \text{Red}\}$
  - Shape, with value types $\mathfrak{R}(\text{Shape}) = \{\text{Ashape}, \text{Pshape}\}$





$$FruitC(r) = \left\{ \left[\begin{array}{lll} \text{sit} & = & r \\ \text{sit-type} & = & T^c \\ \text{prob} & = & \mathrm{p}^{\kappa}_{\mathfrak{J}}(r : T^c) \end{array} \right] \mid T^c \in \mathfrak{R}(C^{FruitC}) \right\}$$

$$p^{FruitC}_{\mathfrak{J}}(r : T^{Fruit}) =$$

$$\sum_{\substack{T^{Col} \in \mathfrak{R}(Col) \\ T^{Shape} \in \mathfrak{R}(Shape)}} \mathrm{p}_{\mathfrak{J}}(T^{Fruit} || T^{Col} \wedge T^{Shape}) \mathrm{p}_{\mathfrak{J}}(r : T^{Col}) \mathrm{p}_{\mathfrak{J}}(r : T^{Shape})$$

$$p_{\mathfrak{J}}^{FruitC}(r : Apple) =$$

$$\sum_{\substack{T^{Col} \in \mathfrak{R}(Col) \\ T^{Shape} \in \mathfrak{R}(Shape)}} \mathrm{p}_{\mathfrak{J}}(Apple || T^{Col} \wedge T^{Shape}) \mathrm{p}_{\mathfrak{J}}(r : T^{Col}) \mathrm{p}_{\mathfrak{J}}(r : T^{Shape}) =$$

$$\mathrm{p}_{\mathfrak{J}}(Apple || Green \wedge Ashape) \mathrm{p}_{\mathfrak{J}}(r : Green) \mathrm{p}_{\mathfrak{J}}(r : Ashape) +$$

$$\mathrm{p}_{\mathfrak{J}}(Apple || Green \wedge Pshape) \mathrm{p}_{\mathfrak{J}}(r : Green) \mathrm{p}_{\mathfrak{J}}(r : Pshape) +$$

$$\mathrm{p}_{\mathfrak{J}}(Apple || Red \wedge Ashape) \mathrm{p}_{\mathfrak{J}}(r : Red) \mathrm{p}_{\mathfrak{J}}(r : Ashape) +$$

$$\mathrm{p}_{\mathfrak{J}}(Apple || Red \wedge Pshape) \mathrm{p}_{\mathfrak{J}}(r : Red) \mathrm{p}_{\mathfrak{J}}(r : Pshape)$$

Learning

- Where does $p_{\mathfrak{J}}(T^c || T^{E_1} \wedge \ldots \wedge T^{E_m})$ come from?
  - From previous experience, i.e. $\mathfrak{J}$
- Can be computed in different ways
  - Using Bayes rule
  - Using a joint probability distribution
- What kinds of probabilities are stored in $\mathfrak{J}$?
  - For nodes with no parents, a probability distribution over the values
  - For nodes with parents, conditional probabilities for all combinations of evidence variable value types and conclusion value types

Learning using Bayes rule for conditional probabilities

- Bayes rule:

$$p(C|E_1,\ldots,E_n) = \frac{p(E_1,\ldots,E_n|C)\text{prior}_{\mathfrak{J}}(C)}{\text{prior}_{\mathfrak{J}}(E_1,\ldots,E_n)}$$

- Chain rule for conditional probabilities:

$$p(E_1,\ldots,E_n|C) = p(E_1|C)p(E_2|E_1,C)\ldots p(E_n|E_1,\ldots,E_{n-1},C)$$

- Given conditional indepence of evidence (not true in general for Bayes nets)

$$p(E_1,\ldots,E_n|C) = p(E_1|C)p(E_2|C)\ldots p(E_n|C)$$

$$prior\, E_1,\ldots,E_n = \text{prior}_{\mathfrak{J}}(E_1)\ldots\text{prior}_{\mathfrak{J}}(E_n)$$

- This is what will change when we move to full Bayes Nets

Learning using Bayes rule for conditional probabilities (TTR version)

- Bayes rule:

$$\mathrm{p}_{\mathfrak{J}}(C||E_1 \wedge \ldots \wedge E_n) = \frac{\mathrm{p}_{\mathfrak{J}}(E_1 \wedge \ldots \wedge E_n||C)\mathrm{prior}_{\mathfrak{J}}(C)}{\mathrm{prior}_{\mathfrak{J}}(E_1 \wedge \ldots \wedge E_n)}$$

- Chain rule for conditional probabilities:

$$\mathrm{p}_{\mathfrak{J}}(E_1, \ldots, E_n||C) = \mathrm{p}_{\mathfrak{J}}(E_1||C)\mathrm{p}_{\mathfrak{J}}(E_2||E_1, C)\ldots\mathrm{p}_{\mathfrak{J}}(E_n||E_1, \ldots, E_{n-1}, C)$$

- Given conditional indepence of evidence

$$\mathrm{p}_{\mathfrak{J}}(E_1 \wedge \ldots \wedge E_n||C) = \mathrm{p}_{\mathfrak{J}}(E_1||C)\mathrm{p}_{\mathfrak{J}}(E_2||C)\ldots\mathrm{p}_{\mathfrak{J}}(E_n||C)$$

$$\mathrm{prior}_{\mathfrak{J}}(E_1 \wedge \ldots \wedge E_n) = \mathrm{prior}_{\mathfrak{J}}(E_1)\ldots\mathrm{prior}_{\mathfrak{J}}(E_n)$$

Learning (Bayes rule) in the Apple game

$$p_{\mathfrak{I}}^{FruitC}(r : T^{Fruit}) =$$

$$\sum_{\substack{T^{Col} \in \mathfrak{R}(Col) \\ T^{Shape} \in \mathfrak{R}(Shape)}} \mathrm{p}_{\mathfrak{I}}(T^{Fruit} || T^{Col} \wedge T^{Shape}) \mathrm{p}_{\mathfrak{I}}(r : T^{Col}) \mathrm{p}_{\mathfrak{I}}(r : T^{Shape})$$

$$\mathrm{p}_{\mathfrak{I}}(T^{Fruit} || T^{Col} \wedge T^{Shp}) = \frac{\mathrm{p}_{\mathfrak{I}}(T^{Col} \wedge T^{Shape} || T^{Fruit}) \mathrm{prior}_{\mathfrak{I}}(T^{Fruit})}{\mathrm{prior}_{\mathfrak{I}}(T^{Col} \wedge T^{Shape})} =$$

$$\frac{\mathrm{p}_{\mathfrak{I}}(T^{Col} || T^{Fruit}) \mathrm{p}_{\mathfrak{I}}(T^{Shape} || T^{Fruit}) \mathrm{prior}_{\mathfrak{I}}(T^{Fruit})}{\mathrm{prior}_{\mathfrak{I}}(T^{Col}) \mathrm{prior}_{\mathfrak{I}}(T^{Shape})}$$

So where do *these* conditional probabilities come from?

Learning using a joint probability distribution

- Summing over alternatives

$$p_{\mathfrak{J}}(T^c || T^{E_1} \wedge \ldots \wedge T^{E_m}) = \frac{p_{\mathfrak{J}}(T^c \wedge T^{E_1} \wedge \ldots \wedge T^{E_m})}{p_{\mathfrak{J}}(T^{E_1} \wedge \ldots \wedge T^{E_m})} =$$

$$\frac{p_{\mathfrak{J}}(T^c \wedge T^{E_1} \wedge \ldots \wedge T^{E_m})}{\sum_{T \in \mathfrak{R}(C)} p_{\mathfrak{J}}(T \wedge T^{E_1} \wedge \ldots \wedge T^{E_m})}$$

- This requires either storing probabilities for a large number of conjunctive types, or a way of computing them from simpler types
- Storing them is computationally complex
- How could they be computed?

A problem

- Conditional probabilities in Bayes nets are computed in the same way as in a Naive Bayes model, i.e. by counting instances of different values of variables (e.g. the number of situations where A and B both hold)
- This assumes a model where judgements are binary
- We want to instead sum together probabilities of previous judgements, including situations being of conjunctive types ($A \wedge B$)

$$\mathrm{p}_{\mathfrak{J}}(T_1||T_2) = \frac{||T_1 \wedge T_2||_{\mathfrak{J}}}{||T_2||_{\mathfrak{J}}} = \frac{\sum_{j \in \mathfrak{J}_{T_1 \wedge T_2}} j.prob}{\sum_{j \in \mathfrak{J}_{T_2}} j.prob}$$

- We want to compute conditional probabilities of the form

$$p_{\mathfrak{J}}(T_1||T_2) = \frac{||T_1 \wedge T_2||_{\mathfrak{J}}}{||T_2||_{\mathfrak{J}}} = \frac{\sum_{j \in \mathfrak{J}_{T_1 \wedge T_2}} j.prob}{\sum_{j \in \mathfrak{J}_{T_2}} j.prob}$$

- More specifically, for example (for some $T^{Col} \in \mathfrak{R}(Col)$, $T^{Shape} \in \mathfrak{R}(Shape)$)

$$p_{\mathfrak{J}}(T^{Col}||T^{Fruit}) = \frac{\sum_{j \in \mathfrak{J}_{T^{Col} \wedge T^{Fruit}}} j.prob}{\sum_{j \in \mathfrak{J}_{T^{Fruit}}} j.prob}$$

- So we need to have probabilities for judgements involving conjunctive types $T_1 \wedge T_2$, e.g. $\begin{bmatrix} \text{sit} & = & \text{s} \\ \text{sit-type} & = & \text{Red}\wedge\text{Ashape} \\ \text{prob} & = & 0.7 \end{bmatrix}$

- We need to have probabilities for judgements involving conjunctive types $T_1 \wedge T_2$, e.g. $\begin{bmatrix} \text{sit} & = & \text{s} \\ \text{sit-type} & = & \text{Red}\wedge\text{Ashape} \\ \text{prob} & = & 0.7 \end{bmatrix}$
- Where do we get such a probability?
  - Our model will only output non-conjunctive probabilities $r : T^c$ where $T^c \in \mathfrak{R}(C)$
  - It also has access to observational probabilities of the evidence of the form $r : T^E$ where $T^E \in \mathfrak{R}(E)$ for some evidence variable type E
- Can the conjunctive probability be computed from these non-conjunctive probabilities?
- Recall that unless T_1 and T_2 are independent,

$$p_{\mathfrak{J}}(a : T_1 \wedge T_2) = p_{\mathfrak{J}}(a : T_1)p_{\mathfrak{J}}(a : T_2 \mid a : T_1)$$

- T^{Col} and T^{Fruit} do not seem to be independent

Solution?

- In standard Bayesian probability theory, probabilities of conjuncts $A\&B$ are computed by counting the number of instances (cases, situations) where A and B both hold, regardless of whether they are dependent or not
- Estimation of probability from the model is not the same as computing the probability using the probability calculus (even if the model consists of probabilities rather than counts)
- Hence, we can do the same in probabilistic TTR
- For meet types $T_1 \wedge T_2$, we want the probability of a situation s being of T_1 and T_2, based both on the probabilities stored in $\mathfrak{J}$ for s being of the meet type *and* on the probabilities stored in $\mathfrak{J}$ for $s : T_1$ and $s : T_2$

Solution: revise how probabilities are estimated

- If $\mathfrak{J}$ is a finite string of probabilistic Austinian proposition and T is a type, then $||T||_{\mathfrak{J}}$ represents the sum of all probabilities associated with T in $\mathfrak{J}$
- If T is a meet type $T_1 \wedge T_2$, then

$$||T_1 \wedge T_2||_{\mathfrak{J}} = \sum_{j \in \mathfrak{J}_{T_1 \wedge T_2}} j.prob + \sum_{\substack{j_1 \in \mathfrak{J}_{T_1} \\ j_2 \in \mathfrak{J}_{T_2} \\ j_1.\text{sit} = j_2.\text{sit}}} j_1.\text{prob} \cdot j_2\text{prob}$$

  Otherwise,

$$||T||_{\mathfrak{J}} = \sum_{j \in \mathfrak{J}_T} j.prob$$

- Note that the judgements involving T_1 and T_2 must concern the same situation

Remaining issue

- What are the consequences of including in $\mathfrak{J}_T$ probabilities of subtypes of T?
  - The connection between the probability of a situation being of a type T and the same situation being of a subtype of T is not straightforward
  - If we include subtypes, we get a lower bound of the summed probability of T (since the probability of a situation s being of a subtype of T can never be higher than the probability s being of type T)
  - Alternatively, we can say

$$\mathfrak{J}_T = \{j | j \in \mathfrak{J}, j.\text{sit-type} = T\}$$

instead of

$$\mathfrak{J}_T = \{j | j \in \mathfrak{J}, j.\text{sit-type} \sqsubseteq T\}$$

Continued: learning using a joint probability distribution

- Summing over alternatives

$$p_{\mathfrak{J}}(T^c || T^{E_1} \wedge \ldots \wedge T^{E_m}) = \frac{p_{\mathfrak{J}}(T^c \wedge T^{E_1} \wedge \ldots \wedge T^{E_m})}{p_{\mathfrak{J}}(T^{E_1} \wedge \ldots \wedge T^{E_m})} =$$

$$\frac{p_{\mathfrak{J}}(T^c \wedge T^{E_1} \wedge \ldots \wedge T^{E_m})}{\sum_{T \in \mathfrak{R}(C)} p_{\mathfrak{J}}(T \wedge T^{E_1} \wedge \ldots \wedge T^{E_m})} = \frac{||T^c \wedge T^{E_1} \wedge \ldots \wedge T^{E_m}||_{\mathfrak{J}}}{\sum_{T \in \mathfrak{R}(C)} ||T \wedge T^{E_1} \wedge \ldots \wedge T^{E_m}||_{\mathfrak{J}}}$$

- In case no relevant meet types are in $\mathfrak{J}$, this reduces to

$$\frac{||T^c||_{\mathfrak{J}} \cdot ||T^{E_1}||_{\mathfrak{J}} \cdot \ldots \cdot ||T^{E_m}||_{\mathfrak{J}}}{\sum_{T \in \mathfrak{R}(C)} ||T||_{\mathfrak{J}} \cdot ||T^{E_1}||_{\mathfrak{J}} \cdot \ldots \cdot ||T^{E_m}||_{\mathfrak{J}}}$$

- If we compute the conditional probabilities when needed, it is sufficient to store the probabilistic Austinian propositions for judgements j where j.sit is an evidence value type or a conclusion value type

Towards full Bayes nets

- Bayes rule:

$$p_{\mathfrak{J}}(C||E_1 \wedge \ldots \wedge E_n) = \frac{p_{\mathfrak{J}}(E_1 \wedge \ldots \wedge E_n||C)\text{prior}_{\mathfrak{J}}(C)}{\text{prior}_{\mathfrak{J}}(E_1 \wedge \ldots \wedge E_n)}$$

- Chain rule for conditional probabilities:

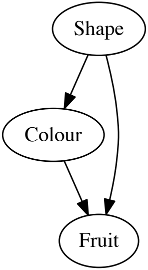
$$p_{\mathfrak{J}}(E_1, \ldots, E_n||C) = p_{\mathfrak{J}}(E_1||C)p_{\mathfrak{J}}(E_2||E_1, C)\ldots p_{\mathfrak{J}}(E_n||E_1, \ldots, E_{n-1}, C)$$

- Cannot assume conditional indepence of evidence, instead need to include dependencies as given by arcs in net

~~$$p_{\mathfrak{J}}(E_1 \wedge \ldots \wedge E_n||C) = p_{\mathfrak{J}}(E_1||C)p_{\mathfrak{J}}(E_2||C)\ldots p_{\mathfrak{J}}(E_n||C)$$~~

~~$$\text{prior}_{\mathfrak{J}}(E_1 \wedge \ldots \wedge E_n) = \text{prior}_{\mathfrak{J}}(E_1)\ldots\text{prior}_{\mathfrak{J}}(E_n)$$~~

Bayes net in Apple game



Learning in the Apple game with full Bayes net

$$p_{\mathfrak{J}}^{FruitC}(r : T^{Fruit}) =$$

$$\sum_{\substack{T^{Col} \in \mathfrak{R}(Col) \\ T^{Shape} \in \mathfrak{R}(Shape)}} \mathrm{p}_{\mathfrak{J}}(T^{Fruit} || T^{Col} \wedge T^{Shape})\mathrm{p}_{\mathfrak{J}}(r : T^{Col})\mathrm{p}_{\mathfrak{J}}(r : T^{Shape})$$

If we are learning using Bayes rule:

$$\mathrm{p}_{\mathfrak{J}}(T^{Fruit} || T^{Col} \wedge T^{Shp}) = \frac{\mathrm{p}_{\mathfrak{J}}(T^{Col} \wedge T^{Shape} || T^{Fruit})\mathrm{prior}_{\mathfrak{J}}(T^{Fruit})}{\mathrm{prior}_{\mathfrak{J}}(T^{Col} \wedge T^{Shape})} =$$

$$\frac{\mathrm{p}_{\mathfrak{J}}(T^{Col} || T^{Shape} \wedge T^{Fruit})\mathrm{p}_{\mathfrak{J}}(T^{Shape} || T^{Fruit})\mathrm{prior}_{\mathfrak{J}}(T^{Fruit})}{\mathrm{prior}_{\mathfrak{J}}(T^{Col} \wedge T^{Shape})}$$

Outline

Conclusions

- Our probabilistic formulation of a rich type theory with records provides the basis for a compositional semantics in which functions apply to categorical semantic objects in order to return functions from categorical interpretations to probabilistic judgements.
- For sentences, the rules generate probabilistic Austinian propositions.
- This framework differs from classical model theoretic semantics, *inter alia*, in that the basic types and type judgements at the foundation of the type system correspond to perceptual judgements concerning objects and events in the world, rather than to entities in a model and set theoretic constructions defined on them.

Conclusions

- We have offered a schematic view of semantic learning in which observations of situations in the world support the acquisition of Bayes Classifiers.
- The basic probabilistic types of our type theoretical semantics are extracted from these classifiers.
- The proposed type theory specifies the interface between observation-based learning of classifiers for objects and situations, and the computation of complex semantic values for the expressions of a natural language.
- Our general model of interpretation achieves a highly integrated bottom-up treatment of linguistic meaning and perceptually-based cognition.
- It situates meaning in learning how to make observational judgements concerning the likelihood of situations obtaining in the world.

Future Work

- Bayesian reasoning from observation provides the incremental basis for learning and refining predicative types.
- In future work we will explore implementations of our learning theory in order to study the viability of our probabilistic type theory as an interface between perceptual judgement and compositional semantics.
- We hope to show that, in addition to its cognitive and theoretical interest, our proposed framework will yield results in robotic language learning, and dialogue modelling.