The role of definitions in coordinating on perceptual meanings

Staffan Larsson
Centre for Linguistic Theory and Studies in Probabillity (CLASP)
Dept. of Philosophy, Linguistics and Theory of Science
University of Gothenburg, Box 200, SE 40530 Sweden
staffan.larsson@ling.gu.se

Abstract

This paper provides an account of how learning of (and coordination on) perceptual meaning can be initialised by partial definitions given in interaction, assuming that the words used in the definition themselves have perceptual meanings. In brief, the idea is that definitions provide a structure (a Naive Bayes classifier) connecting the defined concept with the concepts used in the definition. We formalise this account in Probabilistic Type Theory with Records, ProbTTR.

1 Introduction

Human first language learners typically learn from demonstrations, where a word becomes associated with a perceptual stimuli. This kind of semantic learning can be modeled as training a perceptual classifier on new perceptually available examples (Larsson, 2015, 2020). However, it also seems clear that at least adult humans can learn tentative new meanings, including perceptual meanings, from verbal descriptions.

This paper explores how learning of perceptual meaning can be initialised by partial definitions given in interaction, provided that the words used in the definition themselves have known perceptual meanings. In brief, the idea is that definitions provide hints on a structure (here, a Naive Bayes classifier) connecting the defined concept with the concepts used in the definition. The defined concept is an unobserved variable (for a classifier, the class variable), and the concepts used in the definition are evidence variables.

Semantic coordination, the process of interactively agreeing on the meanings of words and expressions, can be regarded as a process of reciprocal learning, where agents learn from each other. Semantic coordination can happen tacitly as a side-effect of dialogue interaction, or through more or less explicit discussion and negotiation of meanings of words and expressions – sometimes referred to as Word Meaning Negotiations (WMNs) (Myrendal, 2015; Noble et al., 2021).

An account of probabilistic inference and classification in ProbTTR is introduced in Larsson and Cooper (2021), where it is also demonstrated how probabilistic classification of perceptual evidence can be combined with probabilistic reasoning. Building on Larsson and Cooper (2021), Larsson et al. (2021) propose a probabilistic account of semantic learning from interaction formulated in terms of a Probabilistic Type Theory with Records (ProbTTR) (Cooper et al., 2014, 2015). Starting from a probabilistic type theoretic formulations of naive Bayes classifiers, the account of semantic learning is illustrated with a simple language game (the fruit recognition game).

In the following, we will connect these strands of work in an attempt to provide a formal account of the role of definitions in semantic coordination, and in particular for perceptual meanings. We first provide a brief overview of TTR and ProbTTR. We go on to review earlier work on probabilistic classification and learning from interaction using ProbTTR. Section 3 follows Larsson and Myrendal (2017) in relating dialogue acts involved in WMNs to semantic updates on an abstract level. The main contribution of this paper is Section 4, which explores the idea that the the dependency structure of Bayesian classifiers can be derived (learned) from definitions, and that one effect of a definition can be to update the structure of a Bayesian classifier. We provide examples of several ways in which this can happen in the context of a simple language game, the *fruit fetching game*. We end the paper with conclusions and future work.

Proceedings of the 25th Workshop on the Semantics and Pragmatics of Dialogue, September 20–22, 2021, Potsdam / The Internet.

$$\begin{bmatrix} \ell_1 & = & a_1 \\ \ell_2 & = & a_2 \\ \dots & & \\ \ell_n & = & a_n \\ \dots & & \end{bmatrix} : \begin{bmatrix} \ell_1 & : & T_1 \\ \ell_2 & : & T_2(l_1) \\ \dots & & \\ \ell_n & : & T_n(\ell_1, l_2, \dots, l_{n-1}) \end{bmatrix}$$

Figure 1: Schema of record and record type

$$\begin{bmatrix} \text{ref} & = & \text{obj}_{123} \\ \text{c}_{\text{man}} & = & \text{prf}_{\text{man}} \\ \text{c}_{\text{run}} & = & \text{prf}_{\text{run}} \end{bmatrix} : \begin{bmatrix} \text{ref} & : & \text{Ind} \\ \text{c}_{\text{man}} & : & \text{man(ref)} \\ \text{c}_{\text{run}} & : & \text{run(ref)} \end{bmatrix}$$

Figure 2: Sample record and record type

2 Background

This section reviews the background needed to follow the rest of the paper: TTR, Probabilistic TTR fundamentals, and Bayes nets and Naive Bayes classifiers.

2.1 TTR: A brief introduction

We will be formulating our account in a Type Theory with Records (TTR). We can here only give a brief and partial introduction to TTR; see also Cooper (2005) and Cooper (2012). To begin with, $s : T$ is a judgment that some s is of type T. One *basic type* in TTR is Ind, the type of an individual; another basic type is Real, the type of real numbers.

Next, we introduce *records* and *record types*. If $a_1 : T_1, a_2 : T_2(a_1), \dots, a_n : T_n(a_1, a_2, \dots, a_{n-1})$, where $T(a_1, \dots, a_n)$ represents a type T which depends on the objects $a_1, \dots, a_n$, the record to the left in Figure 1 is of the record type to the right.

In Figure 1, $\ell_1, \dots \ell_n$ are *labels* which can be used elsewhere to refer to the values associated with them. A sample record and record type is shown in Figure 2.

Types constructed with predicates may be *dependent*. This is represented by the fact that arguments to the predicate may be represented by labels used on the left of the ':' elsewhere in the record type. In Figure 2, the type of c_{man} is dependent on ref (as is c_{run}).

If r is a record and ℓ is a label in r, we can use a *path* $r.\ell$ to refer to the value of ℓ in r. Similarly, if T is a record type and ℓ is a label in T, $T.\ell$ refers to the type of ℓ in T. Records (and record types) can be nested, so that the value of a label is itself a record (or record type). As can be seen in Figure 2, types can be constructed from predicates, e.g., "run" or "man". Such types are called *ptypes* and correspond roughly to propositions in first order logic.

2.2 Probabilistic TTR fundamentals

In ProbTTR (as in TTR generally), situations are understood in a sense similar to that of Barwise and Perry (1983). It is also assumed that agents can individuate situations, and that they have a way of judging situations to be of situation types.

The core of ProbTTR is the notion of a probabilistic judgement, where a situation s is judged to be of a type T with some probability.

(1) $p(s : T) = r$, where $r \in [0,1]$

Such a judgement expresses a subjective probability in that it encodes an agent's take on the likelihood that a situation is of that type.

A *probabilistic Austinian proposition* is an object (a record) that corresponds to, or encodes, a probabilistic judgement. Probabilistic Austinian propositions are records of the type in (2).

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

A probabilistic Austinian proposition φ of this type corresponds to the judgement that φ.sit is of type φ.sit-type with probability φ.prob.

(3) $p(\varphi.\text{sit}{:}\varphi.\text{sit-type}) = \varphi.\text{prob}$

We assume that agents track observed situations and their types, modelled as probabilistic Austinian propositions.

We use $p(T_1||T_2)$ to represent the probability that an agent assigns to some situation s being of type T_1, given that s is of type T_2. Note that $p(T_1||T_2)$, the conditional probability for some s of $s : T_1$ given that $s : T_2$, is different from $p(T_1|T_2)$, the probability of there being something of type T_1 given that there is something of type T_2. We refer to the former as the *bound variable* conditional probability, and the latter as the *existential* conditional probability.

Larsson and Cooper (2021) introduce a type theoretic counterpart of a random variable in Bayesian inference. To represent a single (discrete) random variable with a range of possible (mutually exclusive) values, ProbTTR uses a *variable type* V whose range is a set of *value types* $\mathfrak{R}(V) = \{A_1, \dots, A_n\}$ which are all (mutually disjoint) subtypes of V ($A_j \sqsubseteq V$ for $1 \leq j \leq n$).

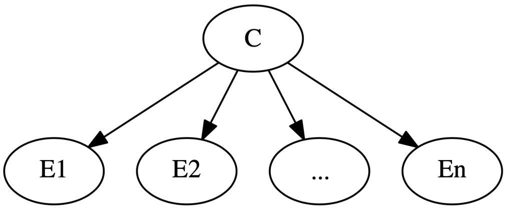

Figure 3: Evidence and Class in a Naive Bayes classifier

2.3 Bayesian nets and the Naive Bayes classifier

A Bayesian Network is a Directed Acyclic Graph (DAG). The nodes of the DAG are random variables, each of whose values is the probability of one of the set of possible states that the variable denotes. Its directed edges express dependency relations among the variables. When the values of all the variables are specified, the graph describes a complete joint probability distribution (JPD) for its random variables. Bayesian Networks provide graphical models for probabilistic learning and inference (Pearl (1990); Halpern (2003)).

A standard Naive Bayes model is a special case of a Bayesian network. More precisely, it is a Bayesian network with a single class variable C that influences a set of evidence variables $E_1, \ldots, E_n$ (the evidence), which do not depend on each other. Figure 2 illustrates the relation between evidence types and class types in a Naive Bayes classifier.

A Naive Bayes classifier computes the marginal probability of a class, given the evidence:

(4)

$$p(c) = \sum_{e_1,\ldots,e_n} p(c \mid e_1, \ldots, e_n)p(e_1)\ldots p(e_n)$$

where c is the value of C, e_i is the value of E_i $(1 \leq i \leq n)$ and

(5) $p(c \mid e_1, \ldots, e_n) =$

$$\frac{p(c)p(e_1 \mid c)\ldots p(e_n \mid c)}{\sum_{C=c'} p(c')p(e_1 \mid c')\ldots p(e_n \mid c')}$$

2.4 A ProbTTR Naive Bayes classifier

Corresponding to the evidence, class variables, and their value types, we associate with a ProbTTR Naive Bayes classifier κ:

(6) a. a collection of n evidence variable types $\mathbb{E}_1^\kappa, \ldots, \mathbb{E}_n^\kappa$

b. n associated sets of evidence value types $\mathfrak{R}(\mathbb{E}_1^\kappa), \ldots, \mathfrak{R}(\mathbb{E}_n^\kappa)$

c. a class variable type $\mathbb{C}^\kappa$, e.g. *Fruit*, and

d. an associated set of class value types $\mathfrak{R}(\mathbb{C}^\kappa)$

We can encode this as a TTR record as seen in Figure 4. (The function lbl takes a type T and returns a label unique to T.)

To classify a situation s using a classifier κ, the evidence is acquired by observing and classifying s with respect to the evidence types. Larsson and Cooper (2021) define a ProbTTR Bayes classifier κ as a function from a situation s (of the meet type[1] of the evidence variable types $\mathbb{E}_1^\kappa, \ldots, \mathbb{E}_n^\kappa$) to a set of probabilistic Austinian propositions that define a probability distribution over the values of the class variable type $\mathbb{C}^\kappa$, given probability distributions over the values of each evidence variable type $\mathbb{E}_1^\kappa, \ldots, \mathbb{E}_n^\kappa$. Formally, a ProbTTR Naive-Bayes classifier is a function

(7) $\kappa : \mathbb{E}_1^\kappa \wedge \ldots \wedge \mathbb{E}_n^\kappa \rightarrow$

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

such that if $s : \mathbb{E}_1^\kappa \wedge \ldots \wedge \mathbb{E}_n^\kappa$, then

(8) $\kappa(s)=\{\begin{bmatrix} \text{sit} = s \\ \text{sit-type} = C \\ \text{prob} = p^\kappa(s : C) \end{bmatrix} \mid C \in \mathfrak{R}(\mathbb{C}^\kappa)\}$

2.5 Semantic classification in the fruit recognition game

Larsson and Cooper (2021) illustrate semantic classification using a Naive Bayes classifier in ProbTTR using the *fruit recognition game*. In this game a teacher shows fruits to a learning agent. The agent makes a guess, the teacher provides the correct answer, and the agent learns from these observations.

We use short-hands *Apple* and *Pear* for the types corresponding to an object being an apple or a pear, respectively[2]. Furthermore, we will assume that

[1] An object a is of the meet type of T_1 and T_2, $a : T_1 \wedge T_2$, iff $a : T_1$ and $a : T_2$.

[2] For details, see Larsson and Cooper (2021).

$$\upsilon(\kappa) = \left[\begin{array}{lll} \text{cvar} & = & \mathbb{C}^\kappa \\ \text{cvals} & = & \mathfrak{R}(\mathbb{C}^\kappa) \\ \text{evars} & = & \{\mathbb{E}_1^\kappa, \ldots, \mathbb{E}_n^\kappa\} \\ \text{evals} & = & \{\left[\begin{array}{lll} \text{lbl}(\mathbb{E}_1^\kappa) & = & \mathfrak{R}(\mathbb{E}_1^\kappa) \\ \ldots & & \\ \text{lbl}(\mathbb{E}_n^\kappa) & = & \mathfrak{R}(\mathbb{E}_n^\kappa) \end{array}\right] \end{array}\right]$$

Figure 4: Variables and values associated with a Naive Bayes classifier κ

$$\upsilon(\text{FruitC}) = \left[\begin{array}{lll} \text{cvar} & = & \textit{Fruit} \\ \text{cval} & = & \{\textit{Apple}, \textit{Pear}\} \\ \text{evars} & = & \{\textit{Col}, \textit{Shp}\} \\ \text{evals} & = & \{\left[\begin{array}{lll} \text{lbl}(\textit{Col}) & = & \{\textit{Red}, \textit{Green}\} \\ \text{lbl}(\textit{Shp}) & = & \{\textit{AShape}, \textit{PShape}\} \end{array}\right] \end{array}\right]$$

Figure 5: Variables and values associated with a Naive Bayes fruit classifier FruitC

the objects in the Apple Recognition Game have one of two shapes (a-shape or p-shape, corresponding to types *Ashape* and *Pshape*) and one of two colours (green or red, corresponding to types *Green* and *Red*).

The class variable type is *Fruit*, with value types $\mathfrak{R}(\textit{Fruit}) = \{\textit{Apple}, \textit{Pear}\}$. The evidence variable types are (i) *Col*(our), with value types $\mathfrak{R}(\textit{Col}) = \{\textit{Green}, \textit{Red}\}$, and (ii) Shape, with value types $\mathfrak{R}(\textit{Shape}) = \{\textit{Ashape}, \textit{Pshape}\}$.

For a situation s the classifier FruitC(s) returns a probability distribution over the value types in $\mathfrak{R}(\textit{Fruit})$.

(9) FruitC(s) =

$$\{\left[\begin{array}{l} \text{sit} = s \\ \text{sit-type} = F \\ \text{prob} = p^{\text{FruitC}}(s : F) \end{array}\right] \mid F \in \mathfrak{R}(\textit{Fruit})\}$$

We follow Larsson and Cooper (2021) in showing how semantic classification (i.e., estimating a probability distribution over class value types) works under the assumption that we can compute conditional probabilities $p(C_j||E_1 \ldots E_n)$ of a class value types C_j given evidence value types $E_1 \ldots E_n$.

In general, for $C_j \in \mathfrak{R}(\mathbb{C}^\kappa)$, we have

(10) $p^\kappa(s : C_j) =$

$$\sum_{\substack{E_1 \in \mathfrak{R}(\mathbb{E}_1^\kappa) \\ \ldots \\ E_n \in \mathfrak{R}(\mathbb{E}_n^\kappa)}} \hat{p}^\kappa(C_j||E_1 \ldots E_n) p(s : E_1) \ldots p(s : E_n)$$

Correspondingly, in the fruit recognition game, for each $F \in \mathfrak{R}(\textit{Fruit})$ we have

(11) $\hat{p}^{\text{FruitC}}(s : F) =$

$$\sum_{\substack{L \in \mathfrak{R}(\textit{Col}) \\ S \in \mathfrak{R}(\textit{Shape})}} p^{\text{FruitC}}(F||L \wedge S) p(s : L) p(s : S)$$

Larsson (2015) shows how perceptual classification can be modelled in TTR, and Larsson (2020) reformulates and extends this formalisation to probabilistic classification. Larsson and Cooper (2021) suggests regarding the non-conditional probabilities (e.g. $p(s : L)$ and $p(s : S)$ above) as resulting from probabilistic classification of real-valued (non-symbolic) visual input, where a classifier assigns to each image a probability that the image shows a situation of the respective type. Such a classifier can be implemented in a number of different ways, e.g. as a neural network, as long as it outputs a probability distribution. The training of perceptual classifiers are outside the scope of this paper, but see Larsson (2013); Fernández and Larsson (2014).

2.6 Semantic learning

For the model of semantic classification that uses conditional probabilities, a central question is of course how to estimate conditional probabilities, of the form $p(C||E_1 \wedge \ldots \wedge E_n)$ (where $C \in \mathfrak{R}(\mathbb{C}), E_i \in \mathfrak{R}(\mathbb{E}_i), 1 \leq i \leq n$). Using Bayes rule and marginalising over the class value types, we get for a Naive Bayes classifier:

(12) $\hat{p}^\kappa(C||E_1 \wedge \ldots \wedge E_n) =$

$$\frac{p(C)p(E_1||C) \ldots p(E_n||C)}{\sum_{C' \in \mathfrak{R}(\mathbb{C}^\kappa)} p(C')p(E_1||C') \ldots p(E_n||C')}$$

For all combinations of evidence value types $E_1, \ldots, E_n$ and class value types C, we need (a) the conditional probability of the evidence value types given the class value type, $p(E_i||C)$, and (b) the prior of the class value type, $p(C')$.

We compute likelihoods and probabilities as ratio of the frequencies of occurrences, summed over all judgements in the history:

(13)

$$p(E_i||C) = \frac{\sum_{j \in \mathfrak{J}, j.\text{sit}=s} p(s:C)p(s:E_i)}{\sum_{j \in \mathfrak{J}, j.\text{sit}=s} p(s:C)}$$

The formula (13) tells us that we can consider probabilities in the history of judgements as fractions of events; and this is justified by interpreting them as fractions of language-community speakers making the corresponding categorical judgement. In this sense, we are providing a frequentist interpretation of epistemic probability. (For the full account and motivation, see [paper under review].)

In addition to conditional probabilities, (12) requires the prior probabilities of the class value types $C \in \mathfrak{R}(\mathbb{C})$. We use $\mathfrak{p}_{\mathfrak{J}}(T)$ to denote the prior probability that an arbitrary situation is of type T given $\mathfrak{J}$.

(14)

$$\mathfrak{p}_{\mathfrak{J}}(T) = \frac{\sum_{j \in \mathfrak{J}_T} j.\text{prob}}{\text{P}(\mathfrak{J})} \text{ if P}(\mathfrak{J}) > 0\text{, otherwise } 0$$

where $\text{P}(\mathfrak{J})$ is the cardinality of situations in $\mathfrak{J}$, i.e. the total number of situations in $\mathfrak{J}$.

(15) $\text{P}(\mathfrak{J}) = |\{s | \exists j \in \mathfrak{J}, j.\text{sit} = s\}|$

We can encode the relevant conditional probabilities and priors as a TTR record $\pi(\kappa)$, as seen in Figure 6. Accordingly, we replace (12) with (16):

(16) $\hat{p}^{\kappa}(C||E_1 \wedge \ldots \wedge E_n) =$

$$\frac{\mathfrak{p}^{\kappa}_{\mathfrak{J}}(C)p^{\kappa}(E_1||C)\ldots p^{\kappa}(E_n||C)}{\sum_{C' \in \mathfrak{R}(\mathbb{C}^{\kappa})} \mathfrak{p}^{\kappa}_{\mathfrak{J}}(C')p^{\kappa}(E_1||C')\ldots p^{\kappa}(E_n||C')}$$

where

$p^{\kappa}(E||C) = \pi(\kappa).\text{condps.lbl}(C).\text{lbl}(E)$

$\mathfrak{p}^{\kappa}_{\mathfrak{J}}(C) = \pi(\kappa).\text{priors.lbl}(C)$

What this buys us is the possibility of updating classifiers by manipulating records encoding them. In Section 4, we will exploit this in formulating semantic updates resulting from word meaning negotiations.

3 Word Meaning Negotiation and semantic updates

In Myrendal (2015, 2019), a taxonomy for dialogue acts involved in WMNs of so-called *trigger words* T in online discussion forum communication is presented. Two central dialogue acts are:

- Explicification: Provides an explicit (partial or complete) definition of T. We will here refer to this as simply *definition*.
- Exemplification: Providing examples of what the trigger word can mean, or usually means.

To describe the effects of these dialogue acts (once they are grounded), Larsson and Myrendal (2017) propose an abstract formalism for conceptual updates, where we assume that a definition D of a word (or expression) T has been provided, or an example situation E. D or E is then used for updating the meaning in question.

- $\delta(T, D)$: T updated with D as a partial definition of T
- $\epsilon(T, E)$: T updated with E as an example of a situation described by T

The abstract meaning update functions[3] serve as a sort of API between dialogue acts and their consequent meaning updates. We can see the learning from examples described above in Section 2.6 as part of the specification of ϵ(T, E). While we leave the exact formulation for future work, updating with an example E in the frequentist learning paradigm amounts to (1) adding example E to $\mathfrak{J}$, (2) recomputing the conditional probabilities and priors based on the updated $\mathfrak{J}$, and (3) updating the probabilities and priors in the classifier record. If we assume that P' is a record like the one shown in Figure 7 but with updated values based on $\mathfrak{J}$ updated with E, step (3) could be formalised thus (taking FruitR to be the union[4] of the records in Figures 5 and 7, so FruitR=υ(FruitC)$\cup\pi$(FruitC)):

(17) FruitR$'$=FruitR[P$'$]

Simplifying somewhat, if r_1 and r_2 are records, then $r_1[r_2]$ is the union of r_1 and r_2 except that if a label ℓ occurs in both r_1 and r_2, the value of ℓ in $r_1[r_2]$ will be $r_2.\ell$. See Cooper (in prep) for details.

[3] We ignore the polarity of the updates here; in general, definitions and examples can be positive or negative.

[4] Records are labelled sets.

$$\pi(\kappa)=\left[\begin{array}{lll} \text{condps} & = & \left[\begin{array}{lll} \text{lbl}(C_1) & = & \left[\begin{array}{lll} \text{lbl}(E_1) & = & p(E_1||C_1) \\ \ldots & & \\ \text{lbl}(E_v) & = & p(E_v||C_1) \end{array}\right] \\ \ldots & & \\ \text{lbl}(C_w) & = & \left[\begin{array}{lll} \text{lbl}(E_1) & = & p(E_1||C_w) \\ \ldots & & \\ \text{lbl}(E_v) & = & p(E_v||C_w) \end{array}\right] \end{array}\right] \\ \text{priors} & = & \left[\begin{array}{lll} \text{lbl}(C_1) & = & \mathfrak{p}_{\mathfrak{J}}(C_1) \\ \ldots & & \\ \text{lbl}(C_w) & = & \mathfrak{p}_{\mathfrak{J}}(C_w) \end{array}\right] \end{array}\right]$$

Figure 6: Record containing conditional probabilities and priors for a classifier κ, where for $1 \leq u \leq v, E_u \in \mathfrak{R}(\mathbb{E}_1^\kappa) \cup \ldots \cup \mathfrak{R}(\mathbb{E}_n^\kappa)$, and where for $1 \leq u \leq w, C_u \in \mathfrak{R}(\mathbb{C}^\kappa)$

$$\pi(\text{FruitC}) = \left[\begin{array}{lll} \text{condps} & = & \left[\begin{array}{lll} \text{lbl}(\textit{Apple}) & = & \left[\begin{array}{lll} \text{lbl}(\textit{Red}) & = & 0.63 \\ \text{lbl}(\textit{Green}) & = & 0.37 \\ \text{lbl}(\textit{AShape}) & = & 0.97 \\ \text{lbl}(\textit{PShape}) & = & 0.03 \end{array}\right] \\ \text{lbl}(\textit{Pear}) & = & \ldots \end{array}\right] \\ \text{priors} & = & \left[\begin{array}{lll} \text{lbl}(\textit{Apple}) & = & 0.64 \\ \text{lbl}(\textit{Pear}) & = & 0.26 \end{array}\right] \end{array}\right]$$

Figure 7: Parts of record containing conditional probabilities and priors for the fruit classifier

4 Learning perceptual meanings from definitions

The work reviewed above showed how probabilistic classifiers can be trained from examples presented in interaction. However, this cannot be the whole story. Indeed, in terms of the dialogue acts for semantic coordination presented in Larsson and Myrendal (2017), we have only covered exemplification. What about partial definitions (explicifications)? What effect do they have on agent's takes on meanings, and how is learning from definitions related to learning from examples?

From the perspective of agents learning how to classify situations probabilistically, one might ask how agents learn the structure of the Bayes net (or as a special case, Naive Bayes classifier) used to classify situations. We propose to connect these two questions, by exploring the idea that the the dependency structure of Bayesian classifiers can be derived (learned) from definitions, and that one effect of a definition can be to update the structure of a Bayesian classifier. (We are not claiming that this is the only way agents can learn such structures.)

In the fruit recognition game, B learns how to take shape (a-shape or p-shape) and colour (red or green) into account when classifying apples and pears, by adjusting conditional probabilities and priors. Before going into learning new meanings from definitions, it might be helpful to show how learning new meanings from examples (demonstrations) could be accounted for.

4.1 Learning a new meaning from example

In Larsson and Cooper (2009), it is shown how *ontological* meaning (e.g. that kumqat is a type of fruit) can be learned from interaction, and how such learning can be modelled in TTR. We can imagine a version of the fruit recognition game where new fruits (i.e., new value types for the fruit variable type) are introduced by demonstration:

A: What fruit is this?
B: A pear.
A: Wrong, it's a Wax Jambu.
B: Okay.

In this example, B can learn both that Wax Jambus are fruits, and what they look like based on being provided with an example Wax Jambu that they can observe. From the context, B can figure out that Wax Jambus are fruits. In the general case such an inference can be based on a variety of factors, including the ongoing activity and linguistic evidence. In terms of a probabilistic classifier, learning this amounts to adding a new value (type) to the Fruit variable (type). This update can be formalised thus:

(18) $\text{FruitR}'\text{=FruitR}[\left[\text{cvals} = F.\text{cvals} \cup \{C\}\right]]$

Furthermore, B could add the new example to $\mathfrak{J}$ and update the conditional probabilities, as detailed above.

In the following, we will see how B can instead learn from partial definitions, which do not provide perceptually available evidence but do seem to offer help in guiding B's learning of the structure of the classifier, as well as associated probabilities.

4.2 Learning a new meaning from definition

We can imagine another language game where A asks B to fetch different fruits in a fruit storage, where several types are fruits are available, some of them unknown to B:

A: Get me an apple please
B: (fetches apple) there you go
A: Thanks. Now get me a Wax Jambu!
B: A Wax Jambu?
A: They are pear-shaped and red.

We can call this the *fruit fetching game*. Let's assume that our learning agent B from A's second utterance learns that Wax Jambus are fruits. However, B has not been presented with an example fruit to use for training. In this sense B does not yet know what Wax Jambus look like. It seems plausible that B in this case might be able to use A's definition of Wax Jambu to distinguish Wax Jambus from other fruits (even if this ability will not be as developed as it might later be after seeing several Wax Jambus).

How, then, could we model the effects of A's definition, which (with pronoun resolved) can be paraphrased as "Wax Jambus are pear-shaped and red"? Firstly, by adding a value type *WaxJambu* to the fruit classifier:

(19) $\text{FruitR}'\text{=FruitR}[\left[\text{cvals = FruitR.cvals} \cup \{\textit{WaxJambu}\}\right]]$

Secondly, by recomputing probabilities, assigning high values to $p(\textit{Pshape}||\textit{WaxJambu})$ and $p(\textit{Red}||\textit{WaxJambu})$, and lowering other probabilities accordingly. For simplicity, we assume here that the high values are 1 and that conditional probabilities for other values of the same variables are lowered to 0.

(20) $\text{FruitR}''\text{=FruitR}'[\left[\text{condps=}\left[\text{lbl}(\textit{WaxJambu})\text{=}\begin{bmatrix}\text{lbl}(\textit{PShape}) = 1.0\\ \text{lbl}(\textit{AShape}) = 0.0\\ \text{lbl}(\textit{Red}) = 1.0\\ \text{lbl}(\textit{Green}) = 0.0\end{bmatrix}\right]\right]]$

Hence, the ProbTTR implementation of the $\delta^+(T, D)$ function should be such that δ^+(⟦Wax Jambu⟧,⟦pear-shaped and red⟧) results in these updates.

Equipped with the updated mental fruit classifier, B now goes off to fetch a Wax Jambu in a storage room, despite never having seen one. One way of finding the right type of fruit in the storage is to simply going through the fruits in storage one by one and classify them, until one is classified as the sought type (here, Wax Jambu)[5].

4.3 Learning new evidence values

Above, A's definition only included evidence values that were already used in the fruit classifier. However, A may also introduce a unknown value previously unknown to B:

A: Get me a Mango please!
B: A Mango?
A: They have an oblong shape.

In this case, B needs to both add the new class value *Mango* and a new evidence value *Oblong* (for the variable *Shp*):

(21) $\text{FruitR}'\text{=FruitR}[\left[\text{values = FruitR.cvals} \cup \{\textit{Mango}\}\right]]$

(22) $\text{FruitR}''\text{=FruitR}'[\left[\text{evals = }\left[\begin{matrix}\text{lbl}(\textit{Shp}) \text{ =FruitR.evals.lbl}(\textit{Shp})\\ \cup \{\textit{Oblong}\}\end{matrix}\right]\right]]$

Finally, as before, the conditional probabilities are shifted to favour the evidence variable given in the definition:

(23) $\text{FruitR}'''\text{=FruitR}''[\left[\text{condps=}\left[\text{lbl}(\textit{Mango}) = \begin{bmatrix}\text{lbl}(\textit{PShape}) = 0.0\\ \text{lbl}(\textit{AShape}) = 0.0\\ \text{lbl}(\textit{Oblong}) = 1.0\end{bmatrix}\right]\right]]$

We assume here that B was familiar with the shape value type *Oblong*, but had not previously considered it relevant to fruit classification[6]A more complicated situation arises when a previously unknown value for a known variable is introduced, e.g. a new shape. In such cases, perceptually available

[5]One can imagine a continuation of the game, where B shows the retrieved fruit to A and receives feedback on whether it was right kind of fruit or not, and trains on this example in the normal way.

[6]An agent may know a very high number of shapes but not all of them will be relevant to classifying fruits. For such reasons, one might consider separating a general shape classifier (if such a thing is ever needed) from a classifier-specific one (in this case, specific to the fruit classifier). In general, even many many evidence types are of a general character (e.g. shape, colour and size), generic classifiers may be of less use than evidence classifiers that are adapted to specific tasks.

examples may be necessary to train the updated classifier on.

4.4 Learning new evidence variables

Finally, a definition may introduce a new evidence variable, along with a value:

A: Get me a Kumqat!
B: A Kumqat?
A: They are small

We assume that B is already has a *Size* classifier and knows that *Small* is a *Size* (along with, say, *MidSize* and *Large*). Given this, the resulting updates to B's fruit classifier could be described thus:

(24) $\text{FruitR}'\text{=FruitR}[\left[\text{values = FruitR.cvals} \cup \{\textit{Kumqat}\}\right]]$

(25) $\text{FruitR}''\text{=FruitR}''[\left[\text{evars = FruitR.evars} \cup \{\textit{Size}\}\right]]$

(26) $\text{FruitR}'''\text{=FruitR}''[\left[\text{evals} = \left[\text{lbl}(\textit{Size}) = \{\textit{Large}, \textit{MidSize}, \textit{Small}\}\right]\right]]$

(27) $\text{FruitR}''''\text{=FruitR}'''[\left[\text{condps=}\left[\text{lbl}(\textit{Kumqat}) = \begin{bmatrix}\text{lbl}(\textit{Small}) = 1.0\\ \text{lbl}(\textit{Large}) = 0.0\\ \text{lbl}(\textit{Midsize}) = 0.0\end{bmatrix}\right]\right]]$

This example also raises the question about partial definitions that only mention a value of one of the evidence variables. What should the conditional probabilities for a situation being of the value types for the other evidence variable types (not mentioned in the definition) given that the situation is of the new class value type? For now, we note that several options are available - assuming uniform distributions, or asking for more information ("What colour is a Kumqat? What shape?") and use the response to infer new conditional probabilities.

5 Definitions vs. examples

If we want to model how meanings are affected by both definitions and examples, we will need to say something about the trade-off between definitions and examples. For example, while a definition may be useful until examples have been observed, at some point the observed examples may override a definition. In the proposed account, definitions affect conditional probabilities only in the short run. Assuming conditional probabilities are recomputed when receiving new relevant observations, the probabilities resulting from proposed definitions (e.g. in the fruit fetching game) will be overwritten as soon as an observation of an instance of the defined concept has been made (an actual fruit of the defined type has been observed). This is perhaps not obviously wrong – it is at least theoretically possible that definitions are categorically superseded by observations – but a more flexible trade-off between definitions and examples (observations) would probably be desirable. There are ways of achieving this in the frequentist approach, e.g. by letting a definition lead to adding some relatively high number N of "fake" observations in line with the definition to $\mathfrak{J}$. By manipulation of N, the relative importance of definitions relative to observations can be regulated. If such approaches are deemed unsatisfying for theoretical or empirical reasons, it may be necessary to move to a different learning method. Future work thus includes working out alternative learning approaches that can better account for the trade-off between definitions and examples.

6 Conclusion

We have shown how (partial) definitions offered in word meaning negotiations can help learners structure probabilistic classifiers that are used to compute probabilistic semantic judgements. Technically, this was achieved by encoding a Naive Bayes classifier as a TTR record structure which can updated by definitions. Beyond what has been mentioned above, future work includes parsing natural language into an appropriate representation for updating classifiers, formulating a general update rule for carrying out such updates (of which several examples were given above), and generalising the account to Bayes nets (and other types of probabilistic classifiers). We also want to study actual definitions from human-human dialogues, rather than invented ones.

Acknowledgements

This work was supported by grant 2014-39 from the Swedish Research Council (VR) for the establishment of the Centre for Linguistic Theory and Studies in Probability (CLASP) at the University of Gothenburg.

References

Jon Barwise and John Perry. 1983. *Situations and Attitudes*. Bradford Books. MIT Press, Cambridge, Mass.

Robin Cooper. 2005. Records and record types in semantic theory. *Journal of Logic and Computation*, 15(2):99–112.

Robin Cooper. 2012. Type theory and semantics in flux. In Ruth Kempson, Nicholas Asher, and Tim Fernando, editors, *Handbook of the Philosophy of Science*, volume 14: Philosophy of Linguistics. Elsevier BV. General editors: Dov M. Gabbay, Paul Thagard and John Woods.

Robin Cooper. in prep. From perception to communication: An analysis of meaning and action using a theory of types with records (TTR). Draft available from `https://sites.google.com/site/typetheorywithrecords/drafts`.

Robin Cooper, Simon Dobnik, Shalom Lappin, and Staffan Larsson. 2014. A probabilistic rich type theory for semantic interpretation. In *Proceedings of the EACL 2014 Workshop on Type Theory and Natural Language Semantics (TTNLS)*, pages 72–79. Gothenburg, Association of Computational Linguistics.

Robin Cooper, Simon Dobnik, Shalom Lappin, and Staffan Larsson. 2015. Probabilistic type theory and natural language semantics. *Linguistic Issues in Language Technology 10*, pages 1–43.

Raquel Fernández and Staffan Larsson. 2014. Vagueness and learning: A type-theoretic approach. In *Proceedings of the 3rd Joint Conference on Lexical and Computational Semantics (∗SEM 2014)*.

J. Halpern. 2003. *Reasoning About Uncertainty*. MIT Press, Cambridge MA.

Staffan Larsson. 2013. Formal semantics for perceptual classification. *Journal of Logic and Computation*.

Staffan Larsson. 2015. Formal semantics for perceptual classification. *Journal of Logic and Computation*, 25(2):335–369. Published online 2013-12-18.

Staffan Larsson. 2020. Discrete and probabilistic classifier-based semantics. In *Proceedings of the Probability and Meaning Conference (PaM 2020)*, pages 62–68, Gothenburg. Association for Computational Linguistics.

Staffan Larsson, Jean-Philippe Bernardy, and Robin Cooper. 2021. Semantic learning in a probabilistic type theory with records. In *Proceedings of Workshop on Computing Semantics with Types, Frames and Related Structures 2021*.

Staffan Larsson and Robin Cooper. 2009. Towards a formal view of corrective feedback. In *Proceedings of the Workshop on Cognitive Aspects of Computational Language Acquisition*, pages 1–9. EACL.

Staffan Larsson and Robin Cooper. 2021. Bayesian classification and inference in a probabilistic type theory with records. In *Proceedings of NALOMA 2021*.

Staffan Larsson and Jenny Myrendal. 2017. Dialogue acts and updates for semantic coordination. *SEMDIAL 2017 SaarDial*, page 59.

Jenny Myrendal. 2015. *Word Meaning Negotiation in Online Discussion Forum Communication*. Ph.D. thesis, University of Gothenburg.

Jenny Myrendal. 2019. Negotiating meanings online: Disagreements about word meaning in discussion forum communication. *Discourse Studies*, 21(3):317–339.

Bill Noble, Kate Viloria, Staffan Larsson, and Asad Sayeed. 2021. What do you mean by negotiation? annotating social media discussions about word meaning. In *Proceedings of the Workshop on the Semantics and Pragmatics of Dialogue (SemDial 2021)*.

J. Pearl. 1990. Bayesian decision methods. In G. Shafer and J. Pearl, editors, *Readings in Uncertain Reasoning*, pages 345–352. Morgan Kaufmann.