5

Bayesian Inference Semantics for Natural Language

Jean-Philippe Bernardy, Rasmus Blanck, Stergios Chatzikyriakidis, Shalom Lappin and Aleksandre Maskharashvili

In most classical theories of formal semantics (Montague, 1974, Barwise and Cooper, 1981, Keenan and Falz, 1985, Heim and Kratzer, 1998, Peters and Westerståhl, 2006), the meaning of a sentence is identified with its truth conditions. These are specified as functions from a domain of indices (possible worlds, contexts, situations, etc.) to the range of Boolean values {0,1}. Inference is treated as a species of logical implication, such that every valuation (or model) for which the premises of an argument are true is one for which the conclusion is true.

While truth-conditional theories of meaning have offered important insights into the semantics of natural languages, they have also missed at least two central elements of the way in which humans interpret the expressions of their language. First, these theories cannot accommodate vagueness, which is a pervasive feature of natural languages. Speakers take most sentences to be more or less true, monadic and relational predicates to apply to a greater or lesser extent to n-tuples of objects, adverbs to be approximately true of the events and situations denoted by verb phrases, etc. A common response to this issue is to treat semantic theory as a component of an idealised linguistic competence which is categorical in nature. Vagueness is then relegated to performance

Probabilistic Approaches to Linguistic Theory.
Jean-Philippe Bernardy, Rasmus Blanck,
Stergios Chatzikyriakidis, Shalom Lappin,
Aleksandre Maskharashvili (eds.).

and processing effects.[1] In order to have any substance, this response must provide a treatment of performance that permits one to predict the observed properties of vagueness when it is combined with the proposed theory of competence. To the best of our knowledge, no such account has yet been constructed. In fact, it is reasonable to suggest that vagueness and ambiguity promote information-theoretic efficiency in communication (Piantadosi et al., 2011). If this is the case, then semantic vagueness is integral to the design of natural language, and it cannot be relegated to a side effect of performance factors.

Second, updating meaning to take account of new information is an important aspect of interpretation. While dynamic semantic approaches (Heim, 1982, Groenendijk and Stokhof, 1991, Kamp and Reyle, 1993, Chierchia, 1995, Cann et al., 2009) capture certain aspects of this process, and some are even amenable to computational implementation (Itegulov et al., 2018, Bernardy and Chatzikyriakidis, 2019a,b), they are largely restricted to extending scope, binding, and co-reference across sentences in discourse. They also tend to rely on the introduction of special-purpose update mechanisms, which are added to the machinery of classical semantic theories, to express these dynamic phenomena.

In this chapter we present Bayesian Inference Semantics (BIS). This system assigns probability conditions to inferences, and it defines functions for the typed constituents of sentences that generate these conditions compositionally. This framework permits us to capture vagueness through probability distributions for predicates, and the sentential assertions that are constructed from them. Vagueness is, then, a core property of expressions in our account. This allows us to provide natural representations of scalar adjectives and vague classifier terms, while these are problematic for classical semantic theories. Using probability distributions over the definitions of predicates also permits us to handle the sorites paradox in a straightforward way. We sustain the fuzzy boundaries of classifiers through these distributions, without invoking sharp borders between objects to which classifier terms apply, and those to which they do not.

BIS is designed to handle probabilistic inferences over a wide range of syntactic constructions and semantic types. We illustrate these with an inference test suite. The role of new information in updating the interpretation of a sentence falls directly out of our Bayesian models for estimating the probability values of the sentences in the premises and the conclusions of an argument. We do not require special-purpose

[1] A similar problem arises in syntax, where the defence of classical binary accounts of grammaticality rely heavily on the competence–performance distinction. See Lau et al. (2017) for a critical discussion of this approach.

update mechanisms, and we capture the effect of new information on the interpretation of all major constituents of a sentence. The lexical content of the premises of an argument specifies the priors of the conditional probability in terms of which the posterior probability of the conclusion is estimated. As information is added through new premises, possibly modifying the lexical content of the premises, the probability value of the conclusion changes.

Many classical formal semantic theories take the indices of interpretive functions to be possible worlds of the sort used in Kripke frame semantics (Montague, 1974). If these are understood as maximal worlds in which every proposition of the language has a defined truth-value, then they raise serious problems of representational tractability (Lappin, 2015, 2018). Probability theorists also frequently talk about distributing probability over possible worlds (Halpern, 2017). In fact, they restrict these worlds to the set of situations corresponding to the outcomes for which probabilities are specified in a random variable. The other events and situations required for a complete world are marginalised out of the probability distribution. Therefore, we avoid the intractability of representing full worlds by using the non-maximal situations of probability theory.

Estimating the probability of a sentence involves reasoning under uncertainty. The obvious question is, then, what is the source of uncertainty that grounds our probabilistic semantics. Epistemicists like Williamson (1994) use probability to model vagueness as uncertainty about the boundary within which a predicate applies to the set of objects of which it is true. They posit the existence of such a boundary, but take it to be unknown in the general case. Goodman and Lassiter (2015), Lassiter and Goodman (2017) adopt a similar view in assuming a contextually determined predicate boundary which hearers estimate in interpreting the utterances of a particular speaker.

By contrast, Edgington (2001) and Lappin (2018) reject the existence of absolute boundaries for property terms, and take them to be inherently indeterminate. Lappin (2018) regards the uncertainty that generates vagueness to be the result of the language acquisition process in which speakers learn to apply the predicates and other expressions of their language to real-world situations under supervision of competent speakers. When language learners use these terms in new situations they generalise them by estimating the likelihood that a given predication holds, according to the competent speakers of the language. This underdetermination of meaning can persist into mature language use. It is an effect of semantic learning as a supervised process.

BIS is compatible with either of these accounts of vagueness. It spec-

ifies a probability distribution over the boundaries of a property for a set of objects in a property space. One could take this distribution to be accessible to further constraints to the point that the boundary becomes fully determinate. On this approach, enough information about the world may resolve the probability of a predication to 1 or 0. Alternatively, we could take the distribution to be resistant to such determination. On that assumption, the boundary over which probability is distributed becomes intrinsically unknowable, and hence it effectively disappears. As both perspectives seem viable, we do not rule out either of them here.

In a classical logic an inference is valid iff under every assignment for which the conjunction of the premisses is evaluated as 1, the conclusion also receives the value 1. By contrast, a probabilistic semantics takes the Boolean truth-values {0,1} to be limit points in a distribution of possible values [0,1]. Inferences receive conditional probabilities $P(Conclusion|Premisses) = p$, where p is the the likelihood that the conclusion holds, given that the premisses do.

In designing our sampling and evaluation procedures for BIS we have sought to sustain classical validity as determined by first-order logic, as well as valid inferences that depend on standard interpretations of non-logical predicates. Our implementation is still under development, and so we have not yet succeeded in capturing all of the classical inferences in our test set.

In the work presented here we have relied on our collective judgements to determine probabilities that our models assign to non-valid inferences. These models are intended as a proof of concept for our project. We anticipate that in future work we will use crowd sourcing and systematic corpus analysis to obtain more robust data for specifying the probability values of the inferences in our test set. We will also experiment with deep learning methods to extract interpretations of predicates and other expressions, in a way that provides reliable wide coverage constraints for the models that we use to generate probabilities for inferences.

A central feature of BIS is the uniform assignment of priors to particular lexical constituents. We do not introduce any lexeme-specific knowledge.[2] Clearly, to obtain priors which incorporate real-world knowledge, properly grounded in speakers' representations of meaning, it is necessary to update these priors with suitable corpus-based representations. This is possible in principle, but here we are presenting

[2]As always when using Bayesian modelling, the priors have an influence on the behaviour of the system (§5.4).

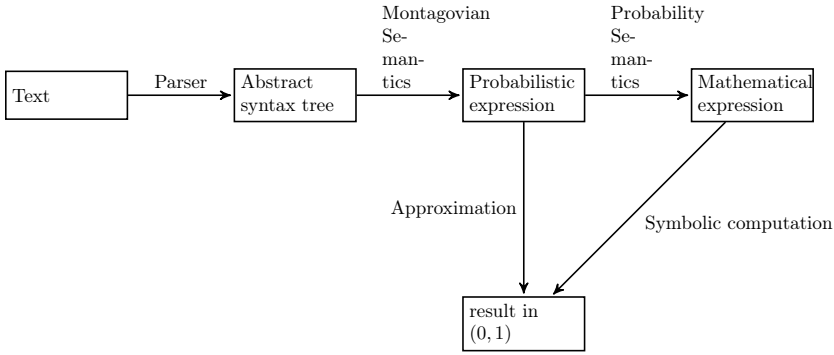

FIGURE 1: Phases in our system. Syntax is first interpreted as probabilistic expressions. Such expressions can be given a precise mathematical semantics. They can also be evaluated approximately, using Monte Carlo methods.

a "blank-state" system, with uniform priors, as a platform for experimenting with Bayesian semantics. Developing it into a wide-coverage framework that generates the interpretation of naturally occurring sentences by correctly predicting their probability conditions is a significant research challenge. The ultimate viability of our proposed program will depend upon meeting this challenge.

The core idea of BIS is that properties and entities are represented as spaces consisting of points (discrete or continuous). These points can be measured, and their density is computed by summation (the discrete case) or integration (the continuous case). In general, the probability of a predication is estimated by measuring the density of the relevant entities in the property space corresponding to the predicate.

Our system, BIS, consists of several subsystems, shown schematically in Fig. 1 (as labelled arrows) together with the intermediate representations that they use (in boxes). We describe and analyse each of these components in the body of the chapter. To get a sense of the intermediate representations, and how the parts of the system are articulated, we go through an example inference here, presented schematically.

Consider the following inference problem:

$$\frac{\text{Most birds fly}}{\text{A few birds fly}}$$

We wish to compute the probability

$$P(\text{A few birds fly} \mid \text{Most birds fly})$$

and test if its value is closer to 1 than 0.

The sentences are first parsed, yielding abstract syntax trees. In practice we use the Grammatical Framework (GF) tool (Ranta, 2004), but this is not essential to our account. Any parser which produces syntactic structures compatible with Montague-style categories would be suitable (see §5.1 for details). The parses that we obtain for the premise and the hypothesis, respectively, are the following.

$$P = CltoS\ Pos\ (S1\ (QNP\ most\ bird)(fly))$$
$$H = CltoS\ Pos\ (S1\ (QNP\ aFew\ bird)\ (fly))$$

The abstract syntax is then translated to a representation language which makes random variables explicit. It also makes their spaces explicit, and the measures thereof. Using these features, the probabilities of all propositions of interest can be expressed precisely. This language, and its notation is described in detail in §5.2[3]. To generate these intermediate representations, we first must express our (lack of) prior knowledge about the common nouns, verbs, etc. present in the problem. To do so we gather the lexical items and introduce them as random variables in the appropriate spaces. The premise(s) are added as extra conditions, obtained by a compositional semantics described in §5.3. These conditions effectively update the distributions of the representations of lexical items, yielding a global space of situations Ω.

$$\begin{aligned}\Omega = [&bird : Pred\\ &fly : Pred\\ &p : \mathsf{measure}([x : Ind; b : bird(x); f : fly(x)]) >\\ &\quad \theta_m \mathsf{measure}([x : Ind; b : bird(x)])]\end{aligned}$$

We are assuming here a proportion θ_m corresponding to the meaning of "most" (see §5.3.3 for details). To make the language more concise, we unify the language of spaces and the language of propositions. Effectively, we sample p over the space of *proofs* of the given proposition. We leave the space of predicates $Pred$ unspecified here. Possible choices are spelled out in §5.3, but the reader can glance at Fig. 7 for a preview.

The truth value of the conclusion is expressed as a probability measure of a proposition over the whole space Ω that we just defined, with a suitable proportion θ_f for "few".

$$\begin{aligned}X = P_{\omega:\Omega}([x : Ind; b : \omega.bird(x); f : \omega.fly(x)]) >\\ \theta_f \mathsf{measure}([x : Ind; b : \omega.bird(x)])\end{aligned}$$

[3]For the impatient, the notation $x : T$ indicates that x is of type T; The dot notation $x.f$ represents accessing field f in a record x. Record types are introduced with brackets a around of a dictionary mapping fields to their types.

The expression above can be turned into a mathematical term using the semantics for spaces and probabilities (Definition 1). In our example, the expression begins with integration over the spaces of predicates:

$$\sum_{bird:Pred} \sum_{fly:Pred} \frac{\mathbf{1}(P \wedge Q)}{\mathbf{1}(P)}$$

In fact, we are using here a generalisation of summation and integration, but adopting the symbol $\sum$ for this purpose (cf. §5.2). The conditions P and Q are given by

$$P = \mathsf{measure}([x : Ind; b : bird(x); f : fly(x)]) > \theta_m \cdot \mathsf{measure}([x : Ind; b : bird(x)])$$

$$Q = \mathsf{measure}([x : Ind; b : bird(x); f : fly(x)]) > \theta_f \cdot \mathsf{measure}([x : Ind; b : bird(x)])$$

The integrations and measures are further expanded once $Pred$ is made concrete. We can see that $P \wedge Q = P$ if $\theta_m > \theta_f$, and so in this simple case, the integral evaluates to 1. Therefore, the inference is (stochastically) certain.

However, a more precise model would let θ_m or θ_f be random variables, modelling the epistemic or intrinsic uncertainty of the meaning of the quantifiers. In such a model, the inference would evaluate to a value in the [0,1] interval, depending on the exact choice of priors.

Unfortunately, for the vast majority of cases the integrals would not be computable symbolically. In this kind of situation, one typically resorts to simulated sampling, using Monte Carlo methods (see §5.2.3). We follow this approach, and it is through sampling that we evaluate integrals.[4]

Sections 5.1 to 5.3 detail the phases of our system, outlined above. In §5.1, we define and illustrate the range of constructions that BIS handles, including generalised quantifiers, scalar modifiers, and vagueness cases including the sorites cases, specifying a suitable syntax. We also describe our method of parsing sentences in Ranta's (2004) typed Grammatical Framework (GF).

In §5.2 we construct a Logic with Measurable Spaces (LMS) to make precise the notions of random variables over potentially complex spaces. To ensure the compatibility with Montague semantics, it consists of a typed logic. We support Bayesian reasoning by associating types with probability densities. We define operators for measuring spaces, and

[4]We describe an important exception in §5.3.8.

estimating the probabilities of propositions in terms of the volumes of their relevant subareas.

In §5.2.3 we describe a Markov Chain Monte Carlo (MCMC) method for sampling properties, and estimating the probability of a sentence. Such methods have been used before in linguistic applications, for example by Goodman and Stuhlmüller (2014). Even though all models sample the density of property spaces to estimate the probability of a predication, with Monte Carlo techniques certain models will converge faster than others, depending on the geometric representations of properties and objects. In fact, under certain conditions explained in §5.2.3, estimating probabilities may be computationally intractable. For this reason we discuss two different classes of representation of individuals and predicates. In particular, in §5.3.8 we consider casting properties as boxes of uniform density. This allows us to compute their density symbolically in some instances, avoiding intractability in several important cases.

In Section 5.3, we present our Bayesian semantics for natural language. A compositional semantics is specified for each GF parse structure assigned to a sentence. We describe our inference test suite in §5.4, and we assess the extent to which BIS covers it.

Section 5.5 surveys related work in probabilistic semantics. Finally, in §8.5 we offer our conclusions, and we indicate directions for future work.

5.1 Scope: Phenomena and Grammar

The initial component in the sequence of semantic interpretation is the GF parsing of a natural language expression. The parse trees satisfy the homomorphism requirement of Montague (1970, 1974). They provide the domain for a compositional mapping to semantic types, which are probabilistic in our system. We identify here a subset of syntactic constructions suitable for our needs. We single out natural language phenomena that play an important role in probabilistic inference. We use GF to formulate a set of grammar rules that our parser applies in order to generate the parse trees on which our rules of semantic interpretation operate.

Our abstract grammar (Fig. 2) consists of context-free rules, while specific features of English syntax are specified in GF.

A context-free rule in the GF style is written from right to left, e.g., if X -> Y Z is context-free rule, in GF it would be encoded as a constant R of type Y -> Z -> X, we we usually write as R : Y -> Z -> X. One can see this as follows: R takes two arguments of type Y and Z and

```
    All,Most,Few,AFew,Every,GenericPl, Many : Quant;

    Tall, Short: Adj;

    ListenToOudMusic, TryHairTransplantTreatment, EnjoyTabouli : VP;

--polarity items--
    Pos,Neg: Pol;

--common nouns and relative common nouns--
    Linguist, BaldMan, ToupeeWearer,  BasketballPlayer : CN;
    Non : CN -> CN;
    Qual : Adj -> CN -> CN;
    THAT : RP;
    MakeRCL: RP -> VP -> RCl;
    MakepolarRS: Pol -> RCl -> RS;
    RelativiseCN : CN -> RS -> CN;

    PercentOf : Card -> CN -> NP;
    Exactly, AtLeast, MoreThan : Card -> Quant;
    QNP : Quant -> CN -> NP ;

--adverbs of frequency--
    Never, Always, Rarely, Probably, Often, Frequently,
    Occasionally, Generally, Regularly : VP -> AVP;

--Units of measure--
    Centimeter, Foot : Unit;
    Measure : Card -> Unit -> Adj -> VP;

--comparatives--
    Equal, More, Less : CompOperator;
    CanPlayChords : CompOperator -> Card -> VP;
    ComparVP : CompOperator -> Adj -> NP -> VP;
    MoreVP : CompOperator -> NP -> VP;

--sentences and clauses--
    S1  : NP -> VP -> Cl;
    CltoS : Pol -> Cl -> S;
    If, But, And, or : S -> S -> S;
    Not : S -> S;
```

FIGURE 2: Syntactic categories in our GF parses

returns the result of type X. That is, the rule R builds an object of type X from objects of type Y and Z. Below, we may call constants of GF, e.g. R : Y -> Z -> X, as rules and as constants interchangeably.

We modify standard syntactic treatments of English to achieve a more natural mapping between syntactic and semantic representations for the phenomena that we are concerned with. We are not providing a wide-coverage English grammar driven parser, but a proof of concept system for probabilistic inference. For instance, our rule IsA : CN -> VP allows us to parse "is a guitarist" as a VP, where "guitarist" is a CN.

Most rules are simple and straightforward. Some of them enrich the stylistic diversity of our constructions, such as relative clauses modifying noun phrases. Also for stylistic diversity, we allow for multi-word phrases to be taken as CNs or VPs. For instance "basketball player" and "toupee wearer" are common nouns, while "enjoy tabouli" is a VP.

As Fig. 2 shows, our grammar has four kinds of constants to build clauses and sentences. Two of them are for generating/parsing sentences from sentences (discussed in §5.1.1). The rest of the rules we use to build sentences from NPs, VPs, and polarity items. In particular, we build a clause using the rule S1 : NP -> VP -> Cl. A sentence may have a positive or negative polarity. We model that by taking a clause and a polarity item, and build a sentence out of them, which is encoded by the rule CltoS : Pol -> Cl -> S.

We list below the rules which are relevant for probabilistic reasoning, and we illustrate them with examples, which form part of our test suite (§5.4).

5.1.1 Logical connectives and polarities

Our test suite contains logically complex sentences built with logical connectives, including conditionals. They are interesting mainly because of their semantic properties. On standard, logical approaches to the interpretation of such sentences, the truth-conditional meanings of connectives play the major role in defining the semantics of a complex sentence. This corresponds to the rules If, But, etc. of type S -> S -> S.

5.1.2 Universal quantification

Using the rule QNP : Quant -> CN -> NP, we produce a quantified NP from a common noun and a quantifier. An important quantifier is All, corresponding to universal quantification, which figures in the usual Aristotelian syllogisms.

The test suite example T30 (p. 212) illustrates an inference that

relies on the interpretation of the universal quantifier.

(T30) P1. All intermediate logic students are Stones fans.
P2. John is an intermediate logic student.
H. John is a Stones fan.
Label:Quantifier, FOL validity

In classical logic, universal statements such as "All As are Bs" are interpreted as inclusion of subsets corresponding to the predicates A and B. Because a system developed for probabilistic reasoning may interpret the universal quantifier through a probabilistic approach to set inclusion, it is useful to test chained quantifiers, as in Example T76 (p. 220).

(T76) P1. All violinists are musicians.
P2. All musicians read music.
H. All violinists read music.
Label:Quantifiers, FOL validity

5.1.3 Generalised quantifiers and generics

We have a wide range of quantifiers besides the universal, including Few, and Most. We also support explicit percentages, using an *ad hoc* rule PercentOf.

Generic plurals can be expressed in several ways (see e.g. the work of Carlson (1982)), but, restricting ourselves to English, we use bare plurals (GenericPl), as in Example T11 (p. 209).

(T11) P1. Turkish coffee drinkers frequently enjoy a shot of Arak.
P2. Most people that enjoy a shot of Arak also listen to classical oud music.
H. Turkish coffee drinkers listen to classical oud music.
Label:Quantifier, Temporal Adverb

We also deal with instantiations of generalised quantifiers and generics, illustrated in Example T9 (p. 208).

(T9) P1. Stones fans often prefer The Doors to The Beatles.
P2. John is a Stones fan.
H. John prefers The Doors to The Beatles.
Label:Quantifier, FOL validity

5.1.4 Modal Adverbs

Modal adverbs play an important role in probabilistic inference. Adverbs of frequency such as Usually, Often, Never connect categorical judgments to probabilistic ones.

(T6) P1. All basketball players are probably tall.
H. Most basketball players are tall.
Label:Quantifier, Modal Adverb

As an illustration, in the premise of T6 (p. 208), we have "all basketball players are probably tall", which gives rise to the hypothesis "most basketball players are tall" because of *probably*. Otherwise, one would infer the stronger hypothesis that all basketball players are tall.

Some adverbs can switch the polarity of an hypothesis when they are applied to premises, as in Example T2 (p. 207).

(T2) P1. Prolog programmers are always intermediate logic students.
P2. Intermediate logic students rarely read music.
H. Prolog programmers don't read music.
Label:Quantifier, Modal Adverb

In general, it is necessary to study cases with frequency adverbs and quantifiers (including generics and generalised quantifiers) carefully, due to the complex interaction between the two. The semantic content of the adverb, as we model it, is the same as that of a quantifier (generalised or universal, depending on the modal adverb). Some quantifiers may not be encoded in single lexical items (like "most" and "few"), but require multi-word expressions that specify explicit numerical values. We intend our general approach to be independent of the details of the system that we build. Thus we take special care in designing our test cases to ensure that they illustrate general semantic patterns that any adequate probabilistic inference system must capture.

5.1.5 Gradation, Adjectives, and Comparatives

Many natural languages, including English, can derive comparative forms from their respective positive forms (Klein, 1980). For example, positive adjectives such as "tall" give rise to the comparative "taller". In our grammar, comparatives are supported by the rule ComparVP : CompOperator -> Adj -> NP -> VP, where a CompOperator can be Equal, More, or Less.

(T15) P1. Mary is tall.
P2. John is taller than Mary.
H. John is tall.
Label:Comparative Adjective, Transitivity

This is relevant to probabilistic reasoning, because the presence of a gradient of probability for the predicate ("John is tall" is more probable than "Mary is tall") corresponds to the applicability of the comparative ("John is taller than Mary").

5.1.6 Units of measure

We also experiment with units of measure. We study the ability of the system to learn the relationship between observations expressed in terms of measurable quantities and qualitative judgments. In order to encode quantities, we need to add numbers to our grammar. We can parse verb phrases such as "6 feet tall".

We are interested in whether a system can *learn the meaning of an adjective* from the premises which provide information about the property corresponding to the adjective, expressed as degrees, with the help of numerical information. Our test suite contains the following example T38 (p. 213).

(T38) P1. Mary is 190 centimeters tall. Mary is tall.
P2. Molly is 184 centimeters tall. Molly is tall.
P3. Ruth is 180 centimeters tall. Ruth is tall.
P4. Helen is 178 centimeters tall. Helen is tall.
P5. Athena is 166 centimeters tall. Athena isn't tall.
P6. Artemis is 157 centimeters tall. Artemis isn't tall.
P7. Joanna is 160 centimeters tall. Joanna isn't tall.
P8. Kate is 162 centimeters tall. Kate isn't tall.
P9. Christine is 163 centimeters tall.
H. Christine isn't tall.
Label:Quantifier, Modal Adverb

While T38 (p. 213) and T15 (p. 209) look similar, they illustrate two different aspects of our system. T15 (p. 209) tests if it captures the relation between "tall" and "taller". T38 (p. 213) requires it to correctly reflect the graded way in which the adjective "tall" applies to an object (when it is true that a person/object is tall, and when it is not).

5.2 Logic with Measurable Spaces

In this section we describe a Logic with Measurable Spaces (LMS). LMS is the representation language connecting parse structures to math-

ematical expressions of probabilities. We will use it to compose the meaning of inferences from the meaning of premises and hypotheses. As a first approximation, one can see LMS as a precise formalisation of informal notations used when manipulating logical expressions involving random variables. Readers familiar with these concepts can skip this section on first reading. But it will be helpful for understanding subsequent definitions.

LMS draws inspiration from several sources. Like descriptive logics, it aims at making truth computable. It features Sigma spaces (akin to Sigma types in Martin-Löf type-theory). It internalises the notion of the cardinality of spaces, expressed here as measures of spaces, as a tool for encoding the notion of event probability. Fox and Lappin (2005) use the cardinality of predicate spaces in their semantic system, for different purposes.

The syntax of LMS is comprised of two categories: spaces (ranged over by metasyntactic variables A, B, C, etc.), and expressions (ranged over by metasyntactic variables e or ϕ, ψ for Boolean expressions.)

For the purpose of this theory, we limit Boolean expressions to comparison between real-valued expressions ($e_1 \leq e_2$) and conjunctions thereof ($\phi \wedge \psi$); as well as boolean variables Real-valued expressions arise from variables, arithmetic operators (abstracted as *op*), and the measures of spaces— discussed below. Additionally we have support for pairs. We won't construct pairs, but we can extract their first or second components *via* the functions π_1 and π_2. The main objects of interest are *spaces*. Every space has two aspects: an underlying support *type* and a probability distribution over it. The types are formed by the unit type, Booleans, reals, functions, and products.

In LMS, types are used as in a programming language, to verify that nonsensical expressions are disallowed. We do not follow the tradition of intuitionistic logic, in that we ignore the inhabitants of types. Specifically, we do not consider types as propositions via the Curry-Howard isomorphism. LMS does not include quantification over all types, nor over all spaces. Instead, the *densities of spaces* are their logical content. Before turning to density we give a brief overview of LMS typing and its consequences. We use two judgments. First, the judgment $\Gamma \vdash e : \tau$, which is the standard typing judgment for terms in the simply typed lambda calculus. We call Boolean-valued expressions *propositions*, so if $\Gamma \vdash \phi : \mathit{Bool}$ holds, it means that ϕ is a proposition in context Γ. Secondly, the judgment $\Gamma \vdash A : \mathsf{Space}\,\sigma$ expresses that A is a space over the ground type σ, in a context Γ.

Because expressions are simply typed, they inherit the usual normalisation properties of typed lamdba terms (Barendregt, 1992). Any

Spaces:

$$
\begin{array}{rr}
A, B, \ldots ::= \mathsf{IsTrue}(\phi) & \text{filter space} \\
\mid \Sigma(x : A)B & \text{sigma space} \\
\mid \mathsf{Distr}(d) & \text{base distribution space} \\
\mid \{e \mid x : A\} & \text{image of } A \text{ under } \lambda x.e
\end{array}
$$

Expressions:

$$
\begin{array}{rr}
\phi, \psi, e ::= x & \text{variable} \\
\mid k & \text{constants} \\
\mid \phi \wedge \psi & \text{conjunction} \\
\mid e_1 \leq e_2 & \text{comparision} \\
\mid \pi_1(e) \mid \pi_2(e) & \text{projections} \\
\mid op(e_i) & \text{arithmetic operators} \\
\mid \diamond & \text{uninformative object} \\
\mid \mathsf{measure}(A) & \text{internalisation of measure}
\end{array}
$$

Types:

$$
\begin{array}{l}
\tau, \sigma ::= Unit \\
\mid Bool \\
\mid \mathbb{R} \\
\mid \tau \to \sigma \\
\mid \tau \times \sigma
\end{array}
$$

FIGURE 3: Syntax of LMS

$$\frac{\Gamma \vdash \phi : \mathit{Bool}}{\Gamma \vdash \mathsf{IsTrue}(\phi) : \mathsf{Space}\, \mathit{Unit}}$$

$$\frac{\Gamma \vdash A : \mathsf{Space}\, \tau \qquad \Gamma, x : \tau \vdash B : \mathsf{Space}\, \sigma}{\Gamma \vdash \Sigma(x : A)B : \mathsf{Space}\, (\Sigma(x : \tau)\sigma)}$$

$$\frac{\Gamma \vdash e_i : \mathbb{R}}{\Gamma \vdash \mathsf{Distr}(d_1[e_i]) : \mathsf{Space}\, \mathbb{R}} \qquad \frac{\Gamma, x : \tau \vdash e : \sigma \qquad \Gamma \vdash A : \mathsf{Space}\, \tau}{\Gamma \vdash \{e \mid x : A\} : \mathsf{Space}\, \sigma}$$

$$\frac{\Gamma \vdash e_1 : \tau \qquad \Gamma \vdash e_2 : \sigma[e_1/x]}{\Gamma \vdash (e_1, e_2) : \tau \times \sigma} \qquad \frac{\Gamma \vdash e : \tau \times \sigma}{\Gamma \vdash \pi_1(e) : \tau} \qquad \frac{\Gamma \vdash e : \tau \times \sigma}{\Gamma \vdash \pi_2(e) : \sigma}$$

$$\Gamma \vdash \diamond : \mathit{Unit} \qquad \frac{\Gamma \vdash \phi : \mathit{Bool} \qquad \Gamma \vdash \psi : \mathit{Bool}}{\Gamma \vdash \phi \wedge \psi : \mathit{Bool}} \qquad \frac{\Gamma, x : \tau \vdash e : \sigma}{\Gamma \vdash \lambda x.e : \tau \to \sigma}$$

$$\frac{\Gamma \vdash e_0 : \tau \to \sigma \qquad \Gamma \vdash e_1 : \tau}{\Gamma \vdash e_0(e_1) : \sigma} \qquad \Gamma \vdash \mathit{true} : \mathit{Bool}$$

$$\Gamma \vdash \mathit{false} : \mathit{Bool} \qquad \frac{\Gamma \vdash e : \mathit{Bool}}{\Gamma \vdash \mathbf{1}(e) : \mathbb{R}} \qquad \Gamma \vdash k : \mathbb{R} \qquad \frac{\Gamma \vdash e_i : \mathbb{R}}{\Gamma \vdash \mathit{op}(e_i) : \mathbb{R}}$$

FIGURE 4: Typing rules for LMS. In the above *op* stands for an arbitrary arithmetic operator of arbitrary arity, with e_i being its operands. Similarly, we list only one logical connective ($\wedge$); others follow the same pattern.

closed term of type $\mathbb{R}$ is a real number.

We now focus on spaces and distributions over them. We have four basic space constructions:

1. Given a basic with n parameters $d(x_1, ..., x_n)$, we have the space $\mathsf{Distr}(d(e_1, ..., e_n))$ (each of the parameters can be assigned any real-valued expression).
2. We can construct a space whose density is 1 when a proposition ϕ is true and 0 otherwise. It is written $\mathsf{IsTrue}(\phi)$.
3. We can construct sigma spaces. Given a space A and a space $B[x]$, we can write $\Sigma(x : A)B[x]$ for the the sigma space. The support type for this sigma space is the pair of types which support respectively A and $B[x]$. The associated distribution is a *joint* distribution. Indeed, the distribution associated with B can

depend on the value x sampled from A.

4. We can take the image of a space A under a function f. This space is written $\{f(x) \mid x : A\}$. (In fact, we generalise to, and allow, any expression dependent on x instead of just $f(x)$.)

These constructions are listed in Fig. 4.

Formally, we do not manipulate densities directly, thus avoiding theoretical difficulties, in particular for $\{f(x) \mid x : A\}$. Instead, we generalise the notion of integration so that it does not just apply to distributions, but to arbitrary spaces. For this purpose we use the symbol $\sum$, as it is a natural extension of the summation operator.[5]

Definition 1 If $\Gamma \vdash A : \mathsf{Space}\,\alpha$ and $\Gamma, x : \alpha \vdash e : \mathbb{R}$, we define $\sum_{x:A} e$ (which can be read as the integral of e for x ranging over A), by induction on A:

$$\sum_{x:\mathsf{Distr}(d)} e = \int_{\mathbb{R}} \mathrm{PDF}(d, x) \cdot [\![e]\!] dx$$

$$\sum_{x:\mathsf{IsTrue}(\phi)} e = \mathbf{1}([\![\phi]\!]) \cdot [\![e[\diamond/x]]\!]$$

$$\sum_{z:\Sigma(x:A)B} e = \sum_{x:A} \sum_{y:B} e[(x, y)/z]$$

$$\sum_{y:\{e \mid x:A\}} e_2 = \sum_{x:A} e_2[e/y]$$

Definition 2 (Evaluation of expressions) The value of an expression e is written $[\![e]\!]$ and defined by induction on the structure of expressions, as is standard in the lambda calculus. We know that evaluation terminates because of our type-system. The only case that merits attention is the evaluation of $\mathsf{measure}(A)$, which is specified by the following equation:

$$[\![\mathsf{measure}(A)]\!] = \sum_{x:A} 1$$

The expression $\sum_{x:A} 1$ integrates over the whole space the constant value 1, thus "counting" the elements of that space. Therefore it is the *measure* of the space A. Overloading the notation, we also write $\mathsf{measure}(A)$ for the measure of the space A as a meta-theoretical expression (not an LMS expression), with the same definition.

Definition 3 (Expected value) We define the *expected value* of e for a

[5] Technically, we define the integrator and the evaluation of expressions (the next two definitions) by mutual induction on the structure of space and expressions.

random variable x distributed in A as follows:

$$E_{x:A}(e) = \frac{\sum_{x:A} e}{\mathsf{measure}(A)}$$

Remark:

$$E_{z:(\Sigma(x:A)B)}(e) = E_{x:A}(E_{y:B}(e[(x,y)/z]))$$

Notation:

$$E_{x_0:A_0,\ldots,x_n:A_n}(e) = E_{x_0:A_0}(\ldots E_{x_n:A_n}(e))$$

Finally, we can define the *probability* of a proposition ϕ over a random variable x ranging in A as the proportion of (the measure of) the space A where ϕ holds.

Definition 4

$$P_{x:A}(\phi) = E_{x:A}(\mathbf{1}(\phi))$$

An equivalent definition is the following:

$$P_{x:A}(\phi) = \frac{\mathsf{measure}(\Sigma(x:A)\mathsf{IsTrue}(\phi))}{\mathsf{measure}(A)}$$

In general, for probabilistic inference, we define a space of possible situations Ω, and evaluate the expected truth value of some proposition ϕ over this space. The space Ω typically has a complex structure.

We now verify that $P_{x:A}(\phi)$ satisfies the expected properties of probabilities, starting with the following lemma:

Lemma 1 $\sum_{x:A}$ *is a linear operator:*

(i) $\sum_{x:A}(k \cdot t) = k \cdot \sum_{x:A} t$ *if k does not depend on x*

(ii) $\sum_{x:A}(t+u) = \sum_{x:A} t + \sum_{x:A} u$

Proof. By induction on the structure of A. □

When a space A has zero measure, the probabilities over it are undefined. Otherwise, the Kolmogorov laws of probability are respected. It is easy to verify that any probability is positive, and that the probability of *true* is 1. The last law (in its finite variant) needs a bit more work, and its proof follows.

Theorem 2 *If $\phi \wedge \psi = false$, then*

$$P_{x:A}(\phi \vee \psi) = P_{x:A}(\phi) + P_{x:A}(\psi)$$

Proof.

$$\begin{aligned} E_{x:A}(\phi \vee \psi) &= \sum_{x:A} \mathbf{1}(\phi \vee \psi) && \text{by def.} \\ &= \sum_{x:A} (\mathbf{1}(\phi) + \mathbf{1}(\psi)) && \text{because } \phi \wedge \psi = false \\ &= \sum_{x:A} \mathbf{1}(\phi) + \sum_{x:A} \mathbf{1}(\psi) && \text{by linearity of } \sum_{x:A} \\ &= E_{x:A}(\phi) + E_{x:A}(\psi) && \text{by def.} \end{aligned}$$

The result is obtained by dividing by $\mathsf{measure}(A)$. □

The property that probabilities are positive can be checked in a similar way. The assumption of unit measure ($P_{x:A}(true) = 1$) is a simple consequence of the definition.

5.2.1 Dealing with equality

In some situations it is useful to use equality of real-valued expressions (for example "John is as tall as Mary"). Perhaps the most obvious way to encode equality between x and y is by using $\mathsf{IsTrue}(x = y)$. Assuming that x and y are both taken in a space A of strictly positive measure, we can naively write the space B of equal x and y as follows.

$$B = \Sigma(x : A)\Sigma(y : A)\mathsf{IsTrue}(x = y)$$

Unfortunately, the above definition is problematic, because $x = y$ is stochastically impossible for real-valued x and y.[6] Consequently $\mathsf{measure}(B) = 0$. Therefore, when evaluating probabilities involving B, one gets division by zero, and so the probabilities are undefined using the definitions given above.

A theoretical approach We need to replace $\mathsf{IsTrue}(x = y)$ with another space $x \equiv y$, such that the density of $x \equiv y$ is zero when $x \neq y$, but whose total measure is 1 (instead of 0). This can be done conceptually by increasing the density at the points where $x = y$. To do this, we must first extend the language of spaces with the $\mathsf{Factor}(e)$ construction, which acts like $\mathsf{IsTrue}(\phi)$, where but e gives directly the factor to be used in the integration (which can thus be greater than 1). That is:

$$\sum_{x:\mathsf{Factor}(e_1)} e_2 = [\![e_1]\!] \cdot [\![e_2[\diamond/x]]\!]$$

[6] Readers who are not familiar with this stochastic property can convince themselves informally that it holds by considering that getting x and y to be equal requires an impossible alignment of infinite precision. We see this formally by carrying out the computation of integrals as defined above.

To be sure, its typing rule is as follows:

$$\frac{\Gamma \vdash e : \mathbb{R}}{\Gamma \vdash \mathsf{Factor}(e) : \mathsf{Space}\, Unit}$$

We can now come back to the problem of filtering a space by with the equality $x = y$. We know to do this with $\mathsf{Factor}()$, which multiplies the integrand by a certain factor. We need to pick a sufficiently large factor, so that integrating it over a 0-measure area produces the result 1. But we can only achieve this with an infinitely large factor.

One may think that no such space exists, but, fortunately, a space of this kind has already been extensively studied. It is known as the *Dirac δ function*. Classically, δ has a single parameter, and its density is 0 when this parameter is nonzero. Its defining property is:

$$\int_{-\infty}^{\infty} f(x)\delta(x)\, dx = f(0)$$

In terms of spaces, the same property becomes:

$$\sum_{x:\mathsf{Distr}(\delta)} t = t[0/x]$$

Hence we can define $x \equiv y$ to be for $\mathsf{Factor}(\delta(x - y))$.

We can now compute the measure of our motivating example B:

$$\begin{aligned}
&\mathsf{measure}(\Sigma(x : A)\Sigma(y : A)x \equiv y) \\
=&\sum_{x:A}\sum_{y:A}\sum_{p:\mathsf{Factor}(\delta(y-x))} 1 \\
=&\sum_{x:A}\sum_{z:\{y-x \mid x:A\}}\sum_{p:\mathsf{Factor}(\delta(z))} 1 && \text{by substitution} \\
=&\sum_{x:A}\sum_{z:\{y-x \mid x:A\}}\sum_{z:\mathsf{Distr}(\delta)} 1 \\
=&\sum_{x:A}\sum_{z:\{y-x \mid x:A\}} 1 && \text{by } \delta \text{ property} \\
=&\sum_{x:A}\sum_{y:A} 1 \\
=&\mathsf{measure}(A)^2
\end{aligned}$$

We see that involving $x \equiv y$ does not make the measure of spaces 0, and hence probabilites remain well-defined. Computing symbolic integration involving δ is not possible in every case, but we refer the reader to Shan and Ramsey (2017) for a generic method.

A numerical approach Perhaps more disturbing than δ not being always computable, is the fact that it is not well suited to Monte Carlo methods, which we describe in §5.2.3. We are faced with the same problem that we encountered originally. If we randomly sample any x and y, and their numerical representations have a high resolution, then it will be extremely rare that $x = y$, and the Monte-Carlo approximation will not converge.

A possible solution to this problem is to increase the density in a non-zero region around the points such that $x = y$, in a smooth fashion. One way to do this is to take the density of the space $x \equiv y$ to be a Gaussian curve of a suitably small standard deviation σ[7], and which has its maximum at $x = y$:

$$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-y}{\sigma})^2}$$

Like all probability density functions, the Gaussian has density 1, and we thereby avoid spaces of zero measure.

While this approach is satisfying, choosing a suitable value for σ is not always straightforward. If it is too small, then we fall into the original pitfall: most of the time the density of the space will be too small to contribute significantly to the integral. Conversely, if σ is too large, then we get an excessively imprecise result. Unless otherwise stated, we have run our models with $\sigma = 1$.

5.2.2 Record notation

When dealing with complex structures involving nested Σ spaces, the expressions for projections become quickly inscrutable. For this reason we use the record notation for such spaces and the corresponding projections.

Definition 5 (Record spaces and projections) Formally, record spaces are defined by translation to Σ spaces, as follows:

$$[x_1 : A_1; \ldots; x_n : A_n] = \Sigma(x_1 : A_1)\Sigma(x_2 : A_2) \ldots A_n$$

Additionally if $e : [x_1 : A_1; \ldots; x_n : A_n]$, then $e.x_i$ is a shorthand for $\pi_1(\pi_2(\pi_2(\ldots e)))$ (the number of repetitions of π_2 is the index of the field in the record).

For similar reasons, we use a shorthand notation for the expected value over several variables, defined as follows:

$$E_{x_0:A_0,\ldots,x_n:A_n}(e) = E_{x_0:A_0}(\ldots(E_{x_n:A_n}(e)))$$

[7] In fact, if f_σ is a Gaussian function with mean 0 and standard deviation σ, then $\delta(x) = \lim_{\sigma \to 0} f_\sigma(x)$

5.2.3 Approximation via sampling

Unfortunately, in the majority of cases (with the notable exceptions of those discussed in §5.3.8 and §5.3.4), the mathematical expressions produced by the semantics given in §5.2 contain integrals which cannot be evaluated symbolically. Hence, we are forced to resort to a numerical approximation algorithm to evaluate them. We use a variant of Gibbs sampling, which is itself an instance of a Markov Chain Monte Carlo (MCMC) method. The algorithm that we use closely follows the one described by Goodman and Stuhlmüller (2014).

All Monte Carlo methods are based on the same principle, which can be outlined as follows. To evaluate $P_{x:A}(\phi[x])$: 1. Sample a random x in A; 2. Check if $\phi[x]$ holds for a chosen value of x; 3. Repeat this process a large number of times. The ratio of the number successes to the number of tries converges to $P_{x:A}(\phi[x])$ as the number of tries tends to infinity.

In certain cases it is very hard to find any sample $x : A$. If (say) A contains an $\mathsf{IsTrue}(\psi)$ space where ψ is satisfied one time in a million, then it will be necessary to try a million samples until one try can be counted. In our application, these kind of situations will happen whenever 1. sets with many hypotheses are considered, 2. very strong hypotheses are tested. For example, "99.9 percent of men walk" requires such a precise arrangement of parameters that most samples will end up being discarded when this condition is checked.

To mitigate this problem, MCMC methods do not sample elements independently. Rather, each new sample x is based on a previous sample. Typically, only a single parameter is changed at every step. On average, the next sample is chosen to be as probable as the previous one, or more so. This way, the system is able to find many (probable) samples. But samples can form (probably) disconnected regions in the chain space, and thus certain configurations may end up being explored more thoroughly than other, equally (or more) probable ones.

Ultimately, it is up to the designer of the underlying problem to avoid the pitfalls of the approximation methods. Because the phrasing of the hypotheses are infinitely variable for any natural language, we cannot avoid these pitfalls entirely. However, certain semantic designs will be more prone to problems than others. We discuss the issues that these design questions raise in §5.4.4.

5.3 Probabilistic Compositional Semantics

Following Montague, our semantics assumes an assignment of syntactic categories to types. These assignments are standard, except that we use

real numbers for adjectival phrases, as we shall see in §5.3.2.

$$\begin{aligned} Pred &= Ind \to Prop \\ AP &= Ind \to \mathbb{R} \\ VP &= Ind \to Prop \\ NP &= VP \to Prop \\ Quant &= CN \to NP \end{aligned}$$

Additionally, CNs are interpreted as spaces over individuals ($CN = \mathsf{Space}\, Ind$). Adjectival phrases are treated as scalars, and so they are realised as real-valued functions.

While Montague leaves all types abstract, we instead use *spaces*. That is, we give certain types a density. This is done for all categories which can be quantified over, including individuals (Ind) and predicates ($Pred$).

5.3.1 Predicates and Individuals

It is common in current work in computational linguistics to represent words by points in a vector space. As an initial implementation of this approach we define the density of this space through a Gaussian distribution. The intuition behind this model is to take a simple prior which does not assume any bias.[8] Formally, the space of individuals is taken to be a multi-variate normal distribution of dimension k, with k sufficiently large, depending on the complexity of the problem at hand.

$$Ind = Normal(0,1)^k$$

Applying the idea that relationships between concepts can be captured by the dot product of their representations, we also represent predicates with a vector in the same distribution. An individual is said to satisfy a given predicate if the dot product of their representations is above a given bias— a bias which also is part of the representation of the predicate. An example predicate is shown in Fig. 5. Perhaps more vividly, one can represent predicates as a hyperplane— individuals falling on one side of it are deemed to satisfy it. This idea is expressed formally as follows:

$$Pred = \left\{ \lambda x.d \cdot x + b > 0 \,\middle|\, d : Normal(0,1)^k, b : Normal(0,1) \right\}$$

We sustain the property that predicates are Boolean-valued functions ($\vdash Pred : \mathsf{Space}\,(\mathbb{R}^k \to Bool)$ in LMS), as demanded by Montague-

[8] In what follows, we will see that each orthogonal dimension of the vector space can be used to represent an indepent property of the individual. Conversely, non-orthogonal dimensions can be used to represent correlated properties.

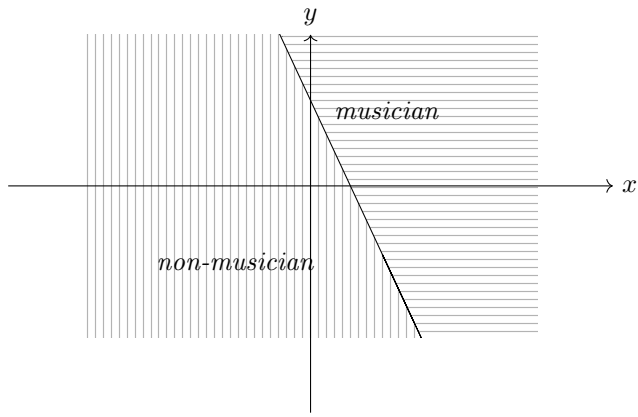

FIGURE 5: A representation of the predicate "musician" and its negation. The line patterns indicate the corresponding subspaces.

style semantics.

Given these basic concepts, we can specify a compositional semantics for simple phrases. In Montague semantics, the phrase "John is tall" is represented as $tall(john)$. Both the predicate *tall* and the individual *john* remain unspecified in that there is no information available about them beyond their types. But in our probabilistic semantics we can quantify the uncertainty attached to *john* and *tall* by construing them in their respective *spaces* instead of their bare *types*.

We can evaluate the expected truth value of $tall(john)$. In this case, the set of possible situations is:

$$\begin{aligned}\Omega = [&john : Ind\\ &tall : Pred]\end{aligned}$$

We evaluate the probability of John being tall as follows:

$$\begin{aligned}&E_{\omega:\Omega}(\omega.tall(\omega.john))\\ =&\sum_{john:Ind}\sum_{tall:Pred}\mathbf{1}(tall(john))\\ =&\sum_{john:Normal(0,1)^k}\sum_{b:Normal(0,1)}\sum_{d:Normal(0,1)^k}\mathbf{1}(d\cdot john+b>0)\\ =&0.5\end{aligned}$$

The computation of the integrals is done by a simple symmetry argument. In the absence of further information, our semantics estimates that John has a 50% chance of being tall. This result is a direct con-

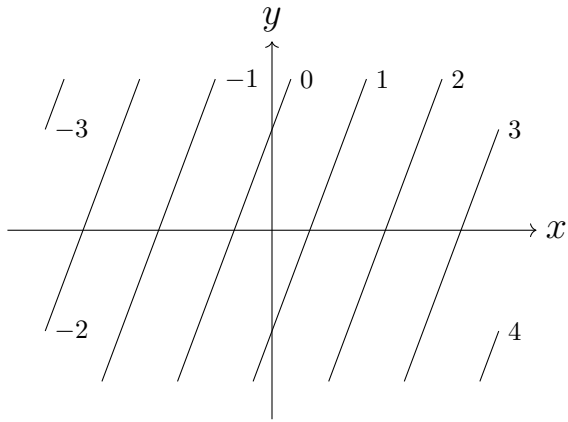

figure 6: A possible representation of "tall". Because tallness is computed with a dot product, lines of equal tallness are parallel, and evenly spaced as tallness increases. Individuals are deemed "tall" if their tallness is positive, shown above as the hatched area.

sequence of the choice of priors that we have proposed. Other choices are possible, depending on the domain of application.

Common Nouns We express common nouns not as predicates, but as spaces. This is similar to the idea of representing nouns as types rather than predicates (Ranta, 1994, Luo, 2012, Chatzikyriakidis and Luo, 2017)). As for predicates, the space of spaces is too large to be sampled, and so we take the space of common nouns to be a restriction on the space of individuals to a random predicate:

$$CN = \{\Sigma(x : Ind)(p(x)) \mid p : Pred\}$$

5.3.2 Graded adjectives

Our system supports scalar predicates and comparatives. We use real-valued functions for adjectival phrases:

$$\vdash A : \mathsf{Space}\,(Ind \to \mathbb{R})$$

A graded predicate can be easily converted to a regular one with the following function, in the simply typed lambda calculus:

$$\begin{aligned} &is : A \to Pred \\ &is = \lambda g.\lambda x.g(x) > 0 \end{aligned}$$

Comparatives can be defined as the comparison of such measures:

$$\begin{aligned} &more : A \to A \to Bool \\ &more = \lambda g.\lambda x.\lambda y.g(x) > g(y) \end{aligned}$$

So far, we have merely considered the sign of the real-valued expression $b + d \cdot x$ as the applicability of a predicate. But it can instead be interpreted as a degree to which the individual x satisfies the property characterised by (b, d). We can specify the space of adjectival phrases by

$$A = \left\{ \lambda x.d \cdot x + b \,\middle|\, d : Normal(0,1)^k, b : Normal(0,1) \right\}$$

The situation is depicted graphically in Fig. 6.

With these formal instruments we can describe simple probabilistic inference problems in detail. For example, what can we infer about the tallness of John from the observation "John is taller than Mary", expressed as the following space:

$$\begin{aligned} \Omega = [&john : Ind \\ &mary : Ind \\ &tall : A \\ &p1 : \mathsf{IsTrue}(more(tall, john, mary))] \end{aligned}$$

This space is a subspace comprised of two individuals and a gradable adjective, such that one of the individual (john) is higher on the scale than the other. We can then evaluate the probability of the sentence "john is tall" if "john is taller than mary", which is, by definition:

$$E_{\omega:\Omega}(is(\omega.tall, \omega.john))$$

Using LMS semantics, it is equal to:

$$\frac{\int_{\omega:\Omega} \mathbf{1}(()is(\omega.tall, \omega.john))}{\int_{\omega:\Omega} 1}$$

Even in this relatively simple case, computing the integrals symbolically is intractable. However, with MCMC sampling, we can get the approximation 0.662.

5.3.3 Generalised quantifiers

We turn to generalised quantifiers. We need them to interpret sentences such as "most birds fly" compositionally. On a standard reading, "most"

is a constraint on the ratio between the cardinality of sets:

$$most(cn, vp) = \frac{\#\{x : cn(x) \wedge vp(x)\}}{\#\{x : cn(x)\}} > \theta \tag{5.37}$$

for a suitable threshold θ. Alternatively, to avoid possible division by 0 and allow for vacuously true quantification, we have

$$most(cn, vp) = \#\{x : cn(x) \wedge vp(x)\} > \theta \ \#\{x : cn(x)\}. \tag{5.38}$$

In our framework, we replace set cardinalities with measures of spaces:

$$most(cn, vp) = \mathsf{measure}(\Sigma(x : cn)\mathsf{IsTrue}(vp(x))) > \theta\mathsf{measure}(cn) \tag{5.39}$$

The latter equality is a valid proposition in our probabilistic logic, because space measures are themselves well-formed expressions.

As a first approximation, we can let θ be a constant. For example, if "most" is meant in to designate a large portion of the population, this threshold will have a value close to one, let us say 0.9. (This is what we do when running our tests, for simplicity.) Other generalised quantifiers can be defined in the same way with a different value for θ. In our examples we define *many* with $\theta = 0.75$. However, it is possible, in fact desirable, to sample θ from a suitable distribution, so that its posterior would depend on linguistic and contextual observations.[9]

Now consider the following inference. "If many logicians are musicians, then it is likely that any given logician is a musician". We model the relevant possible situations as follows:

$$\begin{aligned}\Omega = [&musician : Pred\\ &logician : IndSubset\\ &p1 : \mathsf{IsTrue}(many\, logician\, musician)]\end{aligned}$$

Given the premises, the estimate for the conclusion is

$$E_{\omega:\Omega,x:Ind;p2:logician(x)}(\omega.musicianx) \tilde{=} 0.887$$

The model considers all possible parameter values (vectors/biases) for musicians and logicians. Then, it discards those such that

$$E_{y:Ind,p:\mathsf{IsTrue}(logician(y))}(\mathbf{1}(musician(y))) \leq \theta$$

[9] A particularly useful prior to use for θ is the $Beta(\alpha, \beta)$ distribution, which corresponds to having made $\alpha + \beta$ observations (α negative ones and β positive ones). This way we can control the initial value of as the ratio of positive over total obervations $\beta/(\alpha + \beta)$. We can also control how much θ is sensitive to new observations: the greater the sum $\alpha + \beta$ the less the threshold will be influenced by new observations.

Interestingly, because the models that we are building implement generalised quantifiers through correlation of predicates, we get 'inverse' correlation as well. Therefore, assuming that "many logicians are musicians", in the absence of further information, and given an individual x that is a musician, we predict a high probability for $logician(x)$.

$$E_{\omega:\Omega,x:Ind;p2:musician(x)}(\omega.logician x) \tilde{=} 0.566$$

The model's assumptions can be augmented with the hypothesis that most individuals are not logicians. This lowers the probability of a random musician being a logician appropriately.

$$\begin{aligned}\Omega = [&musician : Pred\\ &logician : Pred\\ &p0 : most\, anything\, (not \circ logician)\\ &p1 : \mathsf{IsTrue}(many\, logician\, musician)]\end{aligned}$$

In this case

$$E_{\omega:\Omega,x:Ind;p2:musician(x)}(\omega.logician(x)) \tilde{=} 0.12$$

Now consider a more complex inference involving three predicates and four propositions. Assume that

1. Most animals do not fly.
2. Most birds fly.
3. Every bird is an animal.

Can we conclude that "most animals are not birds"? We model the possible situations as follows:

$$\begin{aligned}\Omega = [&animal : Pred\\ &bird : Pred\\ &fly : Pred\\ &p1 : (most\, animal(not \circ fly))\\ &p2 : (every\, bird\, animal)\\ &p3 : most\, bird\, fly]\end{aligned}$$

The evaluation of the conclusion that we obtain is

$$E_{\omega:\Omega}(most\, \omega.animal(not \circ \omega.bird)) \tilde{=} 0.773$$

This result holds by virtue of the fact that only models similar to the one pictured in Fig. 7 conform to the premises. One way to satisfy "Every bird is an animal" is to assume that "animal" holds for every individual, because this is compatible with all hypotheses. Then "Most animals don't fly" implies that the "fly" predicate has a large (negative)

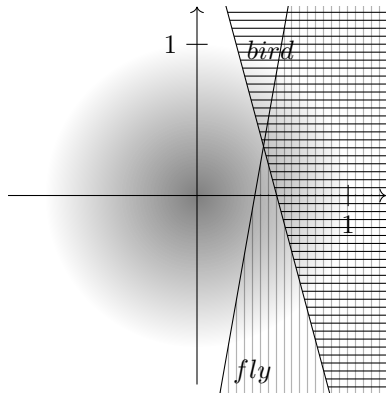

FIGURE 7: A probable configuration for the predicates in the bird example. We ignore the "animal" predicate, which can be assumed to hold for every individual. The grey area suggests the density of arbitrary individuals, a 2-dimensional Gaussian distribution in this case. Birds lie in the areas with a horizontal (and hatch) patterned areas. Flying individuals are in the vertical (and hatch) patterned areas. Note that the density of individuals in the "bird" area is small compared to that in the intersection (crosshatched area). In this model, the observations "Most individuals are not birds", "most individuals don't fly" and "Most birds fly" hold together.

bias. Finally, "Most birds fly" can be satisfied only if "fly" is highly correlated with "bird" (the predicate vectors have similar angles), *and if the bias of "bird" is even more negative than that of "fly"*. Consequently, "bird" also has a large negative bias, and the conclusion holds.

5.3.4 Comparatives and quantifiers

Using an MCMC method to compute expected values inside the computation of an expected value (itself using MCMC) is expensive. It beneficial to consider techniques which can alleviate these costs. One such technique involves the combination of generalised quantifiers and comparatives.

Consider the phrase: "Socrates is wiser than most men". On a straightforward application of our method, as described so far, MCMC sampling proceeds to:

1. sample the predicate "man";
2. sample "wise" as a gradable predicate;
3. sample "socrates";
4. sample n men, and verify that that Socrates is wiser than θn of them, with n large

On this procedure, a step in the Markov Chain of the sampler will *always* run the expensive last step of sampling n men. Consequently, the evaluations of conclusions exemplifying this construction typically do not converge in reasonable time.

One way to speed up the computation is to use quantiles to directly evaluate comparisons with generalised quantifiers, defined as follows.

Definition 6 If $\vdash A : \mathsf{Space}\,(Ind \to \mathbb{R})$, we define

$$x = Quantile\,\theta\,A \text{ iff. } \mathsf{measure}(\Sigma(y : A)(y < x)) = \theta\,\mathsf{measure}(A)$$

Theorem 3 *If the (generalised) quantifier q is associated with a threshold θ, we can show that:*

$$q\,cn\,(\lambda y.more\,g\,x\,y) = g(y) \geq Quantile\,\theta\,\{g(x) \mid x : cn\}$$

This revision of the expression containing generalised quantifiers significantly improves computational cost, when running a sampler. On the left-hand-side of the equal sign in Theorem 3, one needs to do an inner evaluation of the condition every time a new y is sampled ('socrates' in our example), while, on the right-hand-side, this is not necessary. Indeed, the quantile depends only on g and cn, but not on y. With this optimisation, MCMC converges on examples such as the one above.

5.3.5 Comparatives and subsectivity

It is common to classify adjectives according to several of their inferential properties. Subsective adjectives in this classification are adjectives like *skilful*, *big*, *small* (Kamp, 2013, Partee, 2007). Subsective adjectives denote properties which are contingent on the noun class that they modify. For instance, a skilful surgeon is someone who is skilful as a surgeon, but not necessarily skilful in general. Subsectivity has interesting interactions with gradability. Gradable subsective adjectives involve a grade parameter which is class-dependent. For something to be considered a small elephant in a given context, we take a different threshold than when assessing whether an animal is small as a mouse. The interpretations of comparative forms of these adjectives should encode these properties.

We express a gradable subsective adjectival phrase "small elephant" as "smaller than most elephants", thus reducing this phrase to the case studied in §5.3.4. We obtain

$$subsective\, g\, cn\, y = g(y) \geq Quantile\, \theta\ \{g(x) \mid x : cn\}$$

for a suitable threshold θ.

5.3.6 Semantic Learning

Bayesian models are updated to accommodate new observations. This gives rise to learning. We have seen that our framework takes account of data provided in the form of qualitative statements, including those containing generalised quantifiers. We can also accommodate information provided in sequence of observed situations. Consider the following series of statements:

- Mary is 190 centimeters tall.
- Mary is tall.
- Kate is 162 centimeters tall.
- Kate isn't tall.
- Christine is 178 centimeters tall.

The possible situations corresponding to them can be captured as follows:

$$\begin{aligned}\Omega = [\, & mary : Ind \\ & pM : tallness(mary) \equiv centimeters(190) \\ & qM : \mathsf{IsTrue}(tall(mary)) \\ & kate : Ind \\ & pK : tallness(kate) \equiv centimeters(162) \\ & qK : \mathsf{IsTrue}(not(tall(kate))) \\ & christine : Ind \\ & pC : tallness(christine) \equiv centimeters(178) \\ ]\end{aligned}$$

The definition of *centimeters* is given in §5.3.7. At this stage it is sufficient to assume that this definition correctly maps a measure in centimeters to a degree of tallness, as outlined in §5.3.2. Also, we use the equality (≡) space to deal with equality, as discussed in §5.2.1.

Given these assumptions, we can estimate the probability of Christine being tall:

$$P_{\omega:\Omega}(tall(christine))$$

Ω gives a density for the threshold of tallness which corresponds to the criterion of "tall". Hence the system learns dynamically, via a number of observations, the meaning of the adjective "tall"[10].

5.3.7 Units of measure

BIS supports reasoning with units of measure, as in "John is 6 feet tall". To interpret such sentences, we relate heights expressed as numbers to the notion of "tallness", (i.e. the grade associated with the "tall" adjective).

We treat "feet" similarly to other units of measure (as in"John is 180 cm tall"). The interpretation of measure words captures the scaling that they impose. We introduce an additional layer of meaning, which ensures that all units of measure are interpreted *a priori* in the same way. Each unit of measure u is represented by three parameters, $\alpha_u, \beta_u, \gamma_u$, each drawn from a normal distribution. The transformation from degree to numerical value is given by the expression $t_u(x) = \gamma_u x^{\alpha_u x + \beta_u}$. The numbers provided in the input ("6", "180") are then compared with the transformed measure predicates corresponding to the adjective. (In

[10]It can only learn the meaning within the bounds of this model. Here, this is not simply the threshold for tallness, as in the model of Lassiter and Goodman (2017). Rather, it find a direction in the space of individuals which corresponds to height, and a threshold along this dimension.

our example $t_{feet}(john \cdot v_{tall}) = 6$.) This allows BIS to simultaneously infer posterior distributions for individuals, graded predicates and units of measure.

5.3.8 Uniform boxes

As we have seen in the case of generalised quantifiers, one must often compute the measure of a subset of individuals satisfying some predicate. The form for such an expression is

$$\mathsf{measure}(\Sigma(x : Ind)(\mathsf{IsTrue}(p(x)))) = \sum_{x:Ind} \mathbf{1}(p(x))$$

Because individuals are elements in a high-dimensional space, if the density of individuals $p(x)$ is non-trivial, the above integral is often not computable symbolically. This is the case, for example, if individuals receive a Gaussian distribution. Instead the integral must be approximated numerically, typically through a Monte Carlo method.

Another option for the space of predicates is to use boxes, which are cuboids whose faces are orthogonal to the axes of the underlying Euclidean space. We take the density of individuals in this framework to be uniform.

$$\begin{aligned} Ind &= Uniform(-1,1)^k \\ Pred &= \left\{ \forall i. ||x_i - c_i|| < d_i \;\middle|\; c : Uniform(-1,1)^k, d : Beta(1,4) \right\} \end{aligned}$$

Here the computation can proceed symbolically. If we assume, for simplicity, that the box fits in the (-1,1) cuboid, we can compute the volume of a common noun defined by c_i and d_i as follows:

$$\begin{aligned} &\mathsf{measure}(\Sigma(x : Ind)(\mathsf{IsTrue}(\forall i. ||x_i - c_i|| < d_i))) \\ =&\mathsf{measure}(\Sigma(x : Ind)(\bigwedge_i \mathsf{IsTrue}(||x_i - c_i|| < d_i))) \\ =&\prod_i \mathsf{measure}(\Sigma(x_i : Uniform(-1,1))(\mathsf{IsTrue}(||x_i - c_i|| < d_i))) \\ =&\prod_i d_i \end{aligned}$$

Thanks to this symbolic estimation of predicative density the treatment of (generalised) quantifiers is computationally much cheaper with the box-based model of predicates and individuals than with the Gaussian-based technique. We show in Fig. 8 how the situation depicted in Fig. 7 is adapted to this new model.

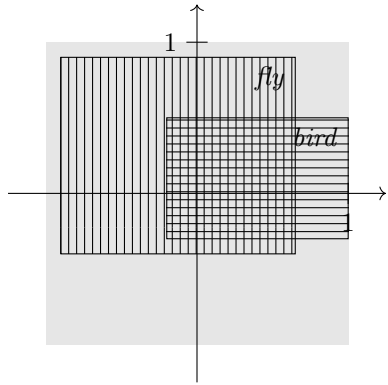

figure 8: A probable configuration for the predicates in the bird example, now with box-predicates and a uniform distribution of individuals.

Graded adjectives

Like the Gaussian-based model, the box-based model supports graded predicates by forming a grade g such that the corresponding non-graded predicate holds when $g(x) > 0$, *and* the area where this predicate holds ($\Sigma(x : Ind)gx > 0$) forms a box. We define the grade of a property defined by a center c and and width d for the individual x as $s(x, c, d)$. The space for graded adjectives A is

$$s(x, c, d) = 1 - \max\left\{ \frac{||x_i - c_i||}{d_i} \,\middle|\, i \in [1..n] \right\}$$

$$A = \left\{ \lambda x.s(x, c, d) \,\middle|\, c : Uniform(-1, 1)^k, d : Beta(1, 4) \right\}$$

This model has the additional convenient property that the space $\Sigma(x : Ind)gx > \alpha$ is a (possibly empty) box for every α. A corollary is that the predicate "more (λy. s(y,c,d)) x" is a box for every c, d and x. Consequently, generalised quantifiers over these expressions can also be computed efficiently.

We remark that the maximal degree of satisfaction with this model is 1. This property can be used in practice in models which make the absolute satisfaction grade a feature of natural language sentences.

Relative clauses

Boxes are closed under intersections. Thus if we use the expression $P \wedge Q$ to denote the intersection of the predicates P and Q, we have $(P \wedge Q)^l_i = \max(P^l_i, Q^l_i)$ and $(P \wedge Q)^h_i = \min(P^h_i, Q^h_i)$. The centre and the width of the box ($(P \wedge Q)^c$ and $(P \wedge Q)^d$ respectively) are recovered using the usual formula.

5.4 Test suite

We have constructed a test suite to illustrate BIS' coverage. We have done this, rather than adopting any of the existing test suites for inference, because none of the latter (e.g. the FraCaS test suite (Cooper et al., 1996), RTE (Dagan et al., 2006), or SNLI (Bowman et al., 2015)) are designed to assess probabilistic inference. These test suites rarely exhibit probabilistic (or quantitative) reasoning via generalised quantifiers (most, etc.), or adverbs (frequently, probably, etc.). They also do not deal with semantic learning.

Our test suite contains 84 examples, provided in §5.7. Each has one or more premises followed by a hypothesis (conclusion). Examples are annotated with respect to the semantic phenomena that occur in the inference. The phenomena were chosen because of their importance for semantic frameworks designed to handle probabilistic reasoning. While most examples in the test suite involve probabilistic reasoning, others are classically valid entailments.

Our models estimate the likelihood of an inference as the conditional probability of the conclusion, given the premises. The latter impose restrictions on the models generated to evaluate the conclusion.

The test suite is comprised of examples that are, in effect, generalised Aristotelian syllogisms. It additionally involves monotonic reasoning, probability, gradation, comparatives, and the relation of comparatives to adjectives. The latter phenomena are not taken up in traditional logical frameworks. To design examples with probabilistic inferences, we focus on semantic phenomena in natural language that introduce uncertainty into predicative judgments. For now, we limit the scope of our work to English. See §5.1 for the syntactic constructions that we associate with the examples in our test set.

Moreover, we annotate our examples with the following tags:

- FOL Validity
- Entailment
- Probable Inference
- Improbable Inference
- Contradiction
- FOL Contradiction

The first three tags fall into positive categories of entailment whereas the last three are their negative scale counterparts. Our choice of these tags is motivated by the fact that we believe that it is beneficial to test a system on classical, Aristotelian syllogisms. Since they involve quantifiers (exists and every), we want to be able to test how successful our system is in handling those, classical cases. But at the same

time, we want to also consider cases of entailment where the sentences cannot adequately be accounted within FOL. For example, one may have probabilistic elements in the proposition p_1 and thus goes beyond FOL. In our test suite we want cases where we have clear entailment involving p_1, which cannot be encoded using the FOL syntax (e.g. p_1 entails p_1 cannot be encoded by FOL but it is true entailment).

Furthermore, we have Probable Inference and its negative counterpart, Improbable Inference. Many of our examples fall into these categories (due to the nature of our test suite). In particular, generalized quantifiers give rise to such inferences, which are not true entailments in logical sense, but qualify for probable inferences (or improbable ones).

There are several reasons for choosing this approach rather than labelling the examples with expected values in the range $[0, 1]$. First, the probability estimates are only numerical approximations, and so an exact value cannot be relied upon. Secondly, the output values are sensitive to the details of the chosen priors (e.g. thresholds for generalised quantifiers ("many", "most", etc.). In particular, we want to assess two different sets of priors based on different geometric configurations.

5.4.1 Principles for evaluation

There are no commonly accepted principles for evaluating the performance of a probabilistic inferencing system. For the purposes of the present chapter, we evaluate the performance of our models using the following general principles.

1. FOL validity & Entailment: An example tagged with one of these labels is counted as correctly evaluated if the model calculates the probability of the conclusion given the premises to be greater than 0.99. Symbolically, this is expressed as the condition $p(\text{conclusion}|\text{premises}) > 0.99$. Hence, the difference between FOL validity and entailment is treated as qualitative rather than quantitative.
2. Probable inference: Such an example is counted as correctly evaluated if the model calculates the probability of the conclusion given the premises to be consistently greater than the probability of the conclusion as calculated without any premises. In other words, if the premises consistently raise the probability of the conclusion. If $p(\text{conclusion}) < 1$, this can be symbolically expressed as the condition
$$p(\text{conclusion}|\text{premises}) > p(\text{conclusion}).$$
3. Improbable inference: Such an example is correctly evaluated if the premises lower the probability of the conclusion, as com-

pared to the probability of the conclusion without any premises. If $p(\text{conclusion}) > 0$ this can be expressed by

$$p(\text{conclusion}|\text{premises}) < p(\text{conclusion}).$$

4. FOL contradiction & Contradiction: $p(\text{conclusion}|\text{premises}) < 0.01$

5.4.2 Unclear examples

Some examples in the testsuite are tagged by the special label "Unclear". These are examples where the priors are inadvertently not sufficiently specified in the premises, and the evaluation therefore only depends on the priors of the model. These priors may well differ from the real-world priors, as well as between the two models. The examples of this kind, even if they may be probable inferences in the real world, are not instances of generally acceptable argument forms since substituting one predicate for another across the premises and conclusion may yield an inference with a radically different probability. To illustrate this effect, we compare two of the unclear examples.

(T83) P1. many logicians are musicians.
P2. john is a musician.
H. john is a logician.
Label: UNCLEAR, QUANTIFIER

According to Bayes' theorem, the posterior probability $p(\text{logician}|\text{musician})$ of the conclusion can be calculated by

$$p(\text{logician}|\text{musician}) = \frac{p(\text{musician}|\text{logician})}{p(\text{logician})} p(\text{musician}).$$

The prior $p(\text{musician}|\text{logician})$ is specified by P1, but the remaining factors are entirely dependent on the model's priors. That is, the relative frequency of logicians and musicians across the domain impact the posterior probability, and also whether the posterior probability is higher or lower than the prior probability $p(\text{logician})$. To judge whether this inference should be counted as a probable inference, or as something else, we need access to the priors of the model, or to real-world knowledge about the proportion of logicians and musicians over the population.

Now consider the related example T84:

(T84) P1. many logicians are musicians.
P2. most people aren't logicians.
P3. john is a musician.

H. john is a logician.
Label: unclear, quantifiers

Here, we add an extra premise: the real-world knowledge that logicians are few. This fixes the prior $p(\text{logician})$, but $p(\text{musician})$ is still dependent on the model's priors. Indeed, we can now see the effect of the different priors of our two models. In the Gaussian model, $p(\text{musician}) \tilde{=} 0.5$, and the example evaluates to $\tilde{=} 0.1$, which is much lower than chance. By contrast, in the box model $p(\text{musician}) \tilde{=} 0.15$, and the example evaluates to $\tilde{=} 0.18$, a value somewhat greater than chance.

Since these examples depend on the priors of different predicates, it is not generally possible to assign them a tag assessing the strength of the inference. Since there then is nothing to compare the models' outputs to, the 13 unclear examples of the testsuite are excluded from the evaluation, leaving 71 examples where we have assessed the strength of the inference. The full list of unclear examples is: T4, T5, T14, T39, T40, T49–T52, T71, T79, T83, T84.

5.4.3 Evaluation of the system using the test suite

To assess the coverage and efficiency of both our Gaussian and our box models, we performed a sample run of each model across the whole test suite. The examples are run sequentially, with a 30 second timeout (triggered if no sample could be found in the space of situations described by the premises). The timeout was chosen so as to make most of the examples converge within the allotted time.[11] In overall performance the box model is superior to the Gaussian model in speed. This is illustrated by the fact that the box model yields an accurate output for example T11 (p. 209) about four times faster than the Gaussian.

Among the 71 tagged examples in the test suite, there are nine examples that current system does not cover. These are examples T7, T12, T17, T43–T46, T56, and T60. The phenomena that these examples exhibit include inherently binary predicates, and expressions like "kind of". Example T62 is also not handled, because it is designed to test three-valued probabilistic logic, which is out of scope here.

According to the principles presented above, the Gaussian model gives correct estimates for 50 of the 71 tagged examples, corresponding to an accuracy of about 70%. The box model fares somewhat better,

[11] Four examples (T4, T41, T48, T51) require substantially more time to converge in the box model. A factor 10 increase is rarely sufficient. Example T52 fails to converge within any reasonable timespan in the box model. The same holds for example T18 in the Gaussian model.

correctly estimating 54 (76%) of the examples. We stress that, because the number of examples is small, these accuracy rates give only a rough indication of the system's performance across a wide range of phenomena. We offer error analyses in the next section.[12]

For an illustration of how our models represent vagueness, and how to evaluate examples containing vague predicates, consider example T61.

(T61) P1. Mary is 190 centimeters tall.
P2. Mary is tall.
P3. Kate is 162 centimeters tall.
P4. Kate isn't tall.
P5. Christine is 185 centimeters tall.
H. Christine is tall.
Label:PROBABLE INFERENCE, POSITIVE ADJECTIVES, VAGUENESS

In the sample run, the Gaussian model returns 1. This is unexpected, because 'tall' is a vague predicate, and there is no a priori reason to regard someone with a height of 185cm as tall. In fact this result is an artefact of the sampling method. In any given run, a threshold for tallness is sampled, and in this run, that threshold is below 185cm.

In a probabilistic setting it is important to consider the average over a larger number of runs. With 100 runs (where the average starts to converge) we get results in the range 0.80–0.90. It is illuminating to compare this with a run for T70 (p. 219), where premise P5 above is replaced with "Christine is 179 centimeters tall". Averaged over 100 runs, the results are in the range 0.70–0.80. This suggests an incremental effect on the likelihood of a predication. The closer Christine's height is to that of a person who is clearly tall, the more likely is it that she is tall. A similar effect is observed with the box model, but not as sharply.

BIS generally works well on propositionally valid inferences. It also performs well with examples containing generalised quantifiers, modal adverbs, percentage determiners, and comparatives, when the number of predicates in the premises and conclusion is relatively small. The problematic examples are those with a combination of generalised quantifiers and 3 or more predicates. Also examples where the system

[12]Our code and a working version of BIS is available publicly at this URL:https://github.com/GU-CLASP/Bayesian-Inference-Semantics-for-Natural-Language . It can be run with either of the two models that we have described here. We invite interested reader to experiment with the system to get a sense of its capacity and its limitations.

has to relate the applicability of a gradable predicate to a particular unit of measurement can raise difficulties.

5.4.4 Error analysis

Example T57 (p. 217) shows how transitivity might fail in the Gaussian model. For that example, the Gaussian model returns a very low score, 0.11, while the box model succeeds with 0.85. The reason for this is that the generic plural in expressions like "if you ..., then you ..." is interpreted as involving a certain degree of inclusion of the predicates in question. But A and B overlapping to a certain degree, and B and C overlapping to the same degree implies little concerning the degree of overlap between A and C. A similar effect is observed in example T79 (p. 221), and this effect is also likely to cause the failure of T63 (p. 218) and T68 (p. 219). By contrast, the box model interprets inclusion of A in B by placing the box for A strictly inside the boundaries of B. This is easier to obtain, by sampling the dimensions for the box A within the box B.

Example T18 (p. 210) illustrates another kind of error. Here, the Gaussian model fails to produce an output, while the box model gives an expected high value. Indeed, the Gaussian model fails to evaluate even the second premise "Most linguists that know formal language theory dislike experimental work" on its own. This suggests that this premise is stochastically hard to satisfy in the geometry of the Gaussian model, and that sampling fails.

A third kind of error involves relating units of measurement to predicates and to other units of measurement. Both models struggle with some cases of this kind, such as T37–38, T41, and T72–T74.

(T74) P1. kate is 190 centimeters tall.
P2. kate is tall.
P3. helen is 180 centimeters tall.
P4. helen isn't tall.
P5. christine is 190 centimeters tall.
H. christine is tall.
Label: ENTAILMENT, VAGUENESS

For this example, both the Gaussian and the Box model return scores around 0.90 (averaged over 100 runs). While these are high scores, the example is tagged as an entailment, and for the evaluation to count as a success we would want a score even closer to 1. Note that the incremental effect pointed out in connection to example T61 (p. 218) remains: if we change Christine's height to 191 cm, the score increases

to 0.95, and at 195 cm the score is consistently 0.99 in both models. This suggests that our model fail to correctly correlate the units of measurement to the vague predicate associated with them.

Finally, T20 (p. 210) is an outlier, exhibiting odd behaviour in both models. In the Gaussian model we consistently get a value of around 0.95. The box model, gives probability of around 0.40. Both are consistently lower than the unconditioned probability of the conclusion, but the example is tagged as a probable inference. One probable cause for this failure is that an easy way of satifying the premises is to make the first predicate (guitarist) very small and the third predicate (prefer The Doors to the Beatles) very large. Then it becomes hard to find a guitarist, and even harder to find one that doesn't prefer The Doors to the Beatles. Another reason could be that the box model is bounded by the unit box, while the Gaussian one is unbounded. This is likely to have an impact on the properties of the space covered by the different predicates.

As a conclusion, a system like BIS has many components, and it is essential to exercise it on many examples to convince oneself of its correctness. Yet, due to the sensitivity to the priors, and the imprecision of Gibbs sampling for certain models, constructing and running tests is more an art than a science. Despite these difficulties, we feel that our test suite is a success: precisely, it highlights the strengths of the system (robustness to generalised quantifiers, relation between gradable adjectives) and its weaknesses, which we have listed above.

5.5 Related work

5.5.1 Distributing Probability Over Possible Worlds

Van Eijck and Lappin (2012) propose a theory in which probability is distributed over the set of possible worlds. The probability of a sentence is the sum of the probability values of the worlds in which it is true. Our work seeks to work out some of the the ideas presented by van Eijck and Lappin (2012). Specifically, we restrict possible worlds to concrete spaces, in the form of priors. This makes it possible to model them as possible situations in the sense of the bounded sets of outcomes corresponding to the distributions of random variables. In contrast van Eijck and Lappin (2012) do not take account of priors. They assign probability to maximal worlds, leaving it unclear how these distributions are computed.

Van Eijck and Lappin also suggest an account of semantic learning which seems to require the wholistic acquisition of all the classifier predicates in a language in a correlated way.

Our system avoids these problems. Our models sample only the individuals and properties (vector dimensions) required to estimate the probability of a given set of statements. Learning is achieved by representing external inputs (Bayesian evidence) as a filter over spaces.

5.5.2 Probabilistic Type Theory

Cooper et al. (2014, 2015) propose a compositional semantics within a probabilistic type theory (ProbTTR). On their approach, the probability of a sentence is a judgment on the likelihood that a given situation is of a particular type, specified in terms of ProbTTR. They also sketch a Bayesian treatment of semantic learning.

Cooper et al.'s semantics is not implemented, and so it is not entirely clear how probabilities for sentences are computed in their system. They do not offer an explicit treatment of vagueness or probabilistic inference.

In this work, we also extend types to support a probabilistic semantics, but we do not consider the probability that a *given* object inhabits a type. In fact, because we often use continuous spaces, this value will typically be 0. Instead we assign a density to the types, which allows us to estimate the expected values of a proposition over such spaces.

5.5.3 A Philosophical Account

Sutton (2017) uses a Bayesian view of probability to support a resolution of classical philosophical problems of vagueness in degree predication. His treatment of these problems is insightful, and it seems to be generally compatible with our implemented semantics. However, it operates at a philosophical level of abstraction, and so a clear comparison is not possible.

5.5.4 Rational Speech Act Theory

Goodman and Lassiter (2015), Lassiter and Goodman (2017) implement a probabilistic semantics and pragmatics, using the WebPPL probabilistic programming language. They regard the probability of a declarative sentence as the most highly valued interpretation that a hearer assigns to the utterance of a speaker in a specified context.

On this approach, speakers express unambiguous meanings in specified contexts through their utterances, and hearers estimate the likelihood of distinct interpretations as corresponding to those that the speaker intends to convey. Their account requires the existence of a univocal, non-vague speaker meaning that hearers seek to identify by distributing probability among alternative readings.

The Goodman–Lassiter account requires the specification of considerable amounts of real-world knowledge and lexical information in order

to support pragmatic inference. It appears to require the existence of a univocal, non-vague speaker's meaning that hearers estimate as the most likely one among competing readings.

On the other hand, and building on work by Sutton (2017), Cooper et al. (2015), we propose that the conditional probability of a predication expresses the likelihood that an idealised competent speaker of the language would apply the predicate to the argument, given the identified features of the object. By doing so, there is no need to assume the existence of a sharply delimited non-probabilistic reading for a predication that hearers attempt to converge on by assessing the probability of alternative readings. All predication consists in applying a classifier to new instances on the basis of supervised learning. We do not posit a contextually dependent cut-off boundary for graded predicates, but we suggest an integrated approach to graded and non-graded predication on which both types of property terms allow for vague borders.

Further advantages of our account include a probabilistic treatment of generalised quantifiers, which includes higher-order quantifiers like *most*, and a basic theory of semantic learning that is a straightforward extension of our sampling procedures for computing the marginal probability of a sentence in a model.

Goodman and Lassiter adopt a classical Montagovian treatment of generalised quantifiers, and their framework has limited coverage of syntactic and semantic structures. They also do not offer a theory of semantic learning.

Regardless, it is possible to integrate pragmatic elements in the semantics by building a rational speech act model *on top of* the models that we present here, following the method of Grove and Bernardy (2021), Grove et al. (2021).

5.5.5 Compositional Bayesian Semantics

The design of BIS is inspired by the Bayesian compositional semantic framework proposed by Bernardy et al. (2018). But BIS differs from this framework in a number of important respects. First, it has a comprehensive syntax–semantics interface through GF parsing. Secondly, it is intended to cover inference in a systematic way, including logically valid, as well as probabilistic arguments. Third, BIS has considerably wider coverage than the framework of Bernardy et al. (2018). It is constructed in such a way as to permit straightforward extension to new types of sentence structure and inference patterns.

5.5.6 Variational Inference

Emerson and Copestake (2017a,b) provide a probabilistic model in order to identify 'features' of objects in terms of the properties that apply to those objects. They develop their account as a graphical probabilistic model. They also interpret universal and existential quantifiers from a probabilistic perspective. "As are Bs" is represented as the conditional probability of B, given A, for all elements of the space, which is equal to the sum (or integral) over all elements. To compute this probability, they make use of variational inference for graphical probabilistic models.

5.5.7 Probabilistic Syllogisms

Pfeifer and colleagues (Pfeifer and Sanfilippo, 2018, Pfeifer, 2013, Gilio et al., 2015) study inference in a probabilistic setting by estimating the probability of the conclusion, given the probabilities of the premises. They employ the notion of p-validity defined by Adams (1998). To be p-valid the uncertainty of a conclusion in an inference should not increase the cumulative uncertainties of its premises.

Their approach differs from ours in several ways. The main one is that we build a model (using Bayesian updating of priors) where the premises hold, and then we estimate the probability of the hypothesis in this model. By contrast, they provide an analytic estimation of the conclusion, given its premises. They require that certain constraints on conditional probabilities hold. Conditional probabilities are primitives for modelling an implication ("if A then B"). This allows them to avoid problems in estimating $A \to B$ when A is false. On our account we take "if...then..." statements to be cases of material implication ($A \to B = \neg A \vee B$), instead of conditional probabilities.

Pfeifer and colleagues use conditional probability as a pivot. They apply a 3-valued logical system. For two events A and B, the conditional event $A|B$ is true if $A \wedge B$ is true, false if $A \wedge \neg B$ is true, and unspecified if B is false. In future work we will explore the interpretation of the "if...then..." construction as a conditional probability, and we may incorporate Pfeifer and colleagues' insights into our semantics.

As Suppes (1966) remarks, statistical syllogisms require a specific formulation in order to be well defined as probabilistic problems. Pfeifer and colleagues (Pfeifer and Sanfilippo, 2018, Pfeifer, 2013, Gilio et al., 2015) propose to add, for certain cases, several restrictions on classical, Aristotelian syllogisms, so that they become more informative from a probabilistic perspective. We handle these cases through priors, which Pfeifer and Sanfilippo (2018), Pfeifer (2013), Gilio et al. (2015) do not make use of. The fact that we do not require additional restrictions for

these examples motivates our prior driven approach. Note that in our account, this result also extends to non-classical syllogisms.

5.5.8 Probabilistic Programming Languages

Through LMS we describe types and an associated density of predicate spaces. This provides an alternative to probabilistic programming languages (Goodman et al., 2008, Borgström et al., 2013, Goodman and Stuhlmüller, 2014). These languages do not describe spaces as such. They specify functions that generate element of a certain type.

LMS offers several advantages. First, probabilistic programming languages generally do not (natively) allow the option of running an inference *within* another inference. LMS does this straightforwardly with the $\mathsf{measure}(e)$ expression. Secondly, the semantics of LMS is more straightforward than that of a probabilistic programming language. LMS does not allow sampling within expressions. Only spaces can refer to other spaces. Borgström et al. (2013) provides an instance of a probabilistic programming language equipped with a formal semantics. Third, constructing spaces is very similar to constructing types and logical formulas. We hope that LMS can be a useful tool for linguists who are familiar with the interpretation of natural language expressions through mapping of type theories (or similar logical systems).

In practice, LMS is implemented in a similar way to some probabilistic programming languages, by MCMC methods such as Gibbs sampling.

5.5.9 Modelling Predicates as Boxes

Boxes in Euclidean spaces are simple objects, and they have already been used for the geometric representation of predicates. Vilnis et al. (2018) apply boxes to encode WordNet lexical entries (unary predicates) in order to predict hypernyms. Like us, they take the distribution in the vector space to be uniform. The probability of a predicate is defined as the volume of the corresponding box. In our work, we use a Bayesian model. It is best suited to represent a small number of predicates, and to fully capture the uncertainty of the boundary for each box. Vilnis et al. (2018) opt for a neural network to learn a large number of box positions. This is appropriate, given that their data set is the complete WordNet hypernym hierarchy. Their model converges on a single mapping of predicates to precise box boundaries, rather than to a distribution over such mappings.

We have not yet tested the box representation of words proposed by Vilnis et al. (2018) for our task, but we plan to do so in future work. As our approach applies Bayesian sampling, we need to modify

the sizes of certain boxes to deal with a data set of this kind. Because their representations are learned for the purpose of detecting WordNet hypernymy, they do not need to contain additional lexical information not relevant to this task.

5.6 Discussion and Conclusion

5.6.1 Test suite improvements

There are several directions in which the current test suite can be developed. First, we can extend its breadth, with a larger data set of examples. This may allow the system to learn representations for lexical items (assuming that the representation of lexical items persist throughout the test suite.) Additionally, such an expanded test suite will provide the basis for deep-learning models. We intend to use crowdsource annotation for data collection.

We will also improve the test set qualitatively. We will identify new factors driving probabilistic inference, and we will extend the test suite with examples of these factors.

We have currently tagged each example with the main factors that they exhibit. We will revise these labels to produce a more rigorous and complete system for classifying inference types. This will facilitate crowd-sourced annotation.

It will also be useful to develop a partial order over examples according to the complexity of the phenomena that they exhibit. If we have a typology and a complexity hierarchy over examples, we will be in a better position to identify the sources of peculiar behaviour in our inference system.

5.6.2 Vagueness and the sorites paradox

Our system gracefully handles vague predicates. Vagueness is introduced through random variables which can occur within such propositions. When evaluating any proposition in a random context, we integrate over this random space, and can obtain any number within the interval [0,1] for an expected truth value. This is the core of probabilistic inference.

This approach formalises the widespread intuition that sentences are, in the general case, more or less true (or false). This indeterminacy arises whenever a distribution is sampled. Clear-cut paradigm cases of predicate (non-)satisfaction are, of course, still available. The proposition $tallness_j + k > tallness_j$, will be true even though $tallness_j$ is not fixed.

Our system is robust for the sorites paradox. In our system, a set of grains becomes a heap when their number is greater than some given

large (real) number. This number is given some indeterminacy by sampling it in a suitable distribution. Hence there is a "soft" transition between a small set of grains and a heap as this number increases.

5.6.3 Treatment of conditionals

Classically, conditional statements of the form "if A, then B" are interpreted as logical implication ($A \to B$) (also called material implication). Probability theory affords another possible interpretation: the probability of the event expressed by B depending on that expressed by A. If $A \to B$ is taken as an inference problem, then we adopt the natural probabilistic interpretation. In contrast, we choose the material conditional treatment of conditionals, when the conditional is embedded in one of the premises.

This choice is perhaps counterintuitive. To motivate it, consider the sentence: "if John is ill, he will not attend the meeting.".

Taking the sentence as a probabilistic inference, we compute the probability

$$P(\text{john attend the meeting}|\text{john ill})$$

The probability will depend on our prior knowledge about john and illness, specifically, the dependence between them. However, if the same sentence is found in a premise, it is understood by discarding all the situations in which *john* is simultaneously ill and attending the meeting. This is done by adding the expression john ill $\to$ john attend the meeting to the record of possible situations, effectively performing a Bayesian update on the distributional meaning of the lexical items.

We can contrast with categorical systems. There, instead of a Bayesian update, one records facts about the free variables mentioned in the conditional. When there are no such free variables, we find ourselves in a linguistic difficulty. One possibility is that the antecedent ('John is ill') is true, and it plays no role. The statement as a whole is thus wasteful, violating the Gricean maxims of quantity. The other possibility is that the antecedent is false, and then we have a counterfactual statement, which poses significant problems of its own.

In a probabilistic system the situation is more straightforward. As long as there are prior possible situations which are either compatible or incompatible with the hypothesis, the statement is informative, rather than a genuine counterfactual. In classical systems variables are either completely free or fixed, whereas probabilistic systems allow for intermediate situations.

Counterfactuals We can understand counterfactuals as locally-fictional stories. We suspend disbelief for the scope of the conditional. Classically, for example in a Montagovian system (Montague, 1970, 1974), we remove some hypothesis from the context. In probabilistic systems we are allowed the possibility of adding uncertainty to the variables mentioned in the conditional. If they were completely fixed, they could become random again. One can quantify the suspension of disbelief, rather than enforcing it categorically.

5.6.4 Mixed methods

There is a long tradition of using logic to interpret natural language. One way to see the present chapter is as departure from this tradition through the use of Bayesian models, which Goodman and Lassiter (2015) pioneered.

In contrast, we do not advocate a purely Bayesian semantics. Thanks to the symbolic evaluation of probabilities, it is possible to mix quantification over spaces, if prior/evidence is best modelled using distributions, with quantification over (non-probabilistic) types, when evidence is best modelled using traditional logical formulas (like in the case where no information about distributions is available, and we must assume that all possible cases are equally likely).

To implement these methods we need to reason about the relative measures of spaces symbolically, and this typically requires creative thought. We do not have access to a symbolic calculator that can automatically decide propositions involving non-trivial symbolic integrals.

5.6.5 Summary

In this chapter we have presented a Bayesian Inference Semantics system which captures probabilistic inferences through uniform priors for lexical items. It uses Bayesian modelling to capture informational updates. BIS is fully compositional, in the sense that the probability conditions of a sentence are calculated through functions corresponding to their syntactic constituents. We achieve probabilistic interpretations by assigning measurable spaces to both objects and properties. Estimating the probability of a predication reduces to measuring the density of the relevant objects in the space of the predicate's property. We have further experimented with two models of priors: semi-spaces of Gausian distributions, and boxes of uniform density in which density is made easier to compute by symbolic simplifications. The system supports a wide range of phenomena, which includes generalised quantifiers, modal adverbs, scalar modifiers, and vagueness. BIS captures intermediate or vague cases of predication, corresponding to the uncer-

tain intuitions speakers have in those cases. Our account also handles the sorites paradox. BIS is evaluated on a test suite constructed for probabilistic inference. We hope that the work that we present here will stimulate discussion and further work on the role of probability in the semantics of natural language.

Acknowledgements

The research reported in this paper was supported by grant 2014-39 from the Swedish Research Council, which funds the Centre for Linguistic Theory and Studies in Probability (CLASP) in the Department of Philosophy, Linguistics, and Theory of Science at the University of Gothenburg. We are grateful to our colleagues in CLASP for helpful discussion of some of the ideas presented here.

5.7 Appendix: Test Suite

(T1) P1. every violinist is a musician.
P2. musicians generally read music.
H. if john is a violinist, then john reads music.
Label: PROBABLE INFERENCE, QUANTIFIER, MODAL ADVERB

(T2) P1. prolog programmers are always intermediate logic students.
P2. intermediate logic students rarely read music.
H. prolog programmers don't read music.
Label: PROBABLE INFERENCE, QUANTIFIER, MODAL ADVERB

(T3) P1. many bald men are toupee wearers.
P2. toupee wearers always try hair transplant treatment.
P3. john is a bald man.
H. john tries hair transplant treatment.
Label: PROBABLE INFERENCE, QUANTIFIER, MODAL ADVERB

(T4) P1. john is a basketball player.
P2. basketball players are usually taller than non basketball players.
H. john is taller than most people.
Label: UNCLEAR (WORLD KNOWLEDGE), COMPARATIVE ADJECTIVE, MODAL ADVERB

(T5) P1. john is a short basketball player.
P2. basketball players are usually less short than non basketball players.
H. john is short.
Label: unclear (world knowledge), comparative adjective, modal adverb, subsectivity

(T6) P1. all basketball players are probably tall.
H. most basketball players are tall.
Label: probable inference, quantifier, modal adverb

(T7) P1. 40 percent of prolog programmers read music.
P2. violinists don't like prolog programmers that read music.
H. violinists don't like most prolog programmers.
Label: improbable inference, percentage determiner, modal adverb, negation, binary predicate

(T8) P1. prolog programmers definitely use tail recursion.
P2. many logicians are prolog programmers.
H. logicians use tail recursion.
Label: probable inference, quantifier, modal adverb

(T9) P1. stones fans often prefer the doors to the beatles.
P2. john is a stones fan.
H. john prefers the doors to the beatles.
Label: probable inference, quantifier, temporal adverb

(T10) P1. if you play for the leafs, then you are probably traded from the canadiens.
P2. if you are traded from the canadiens, then you often play in the montreal forum.
P3. john plays for the leafs.
H. john plays in the montreal forum.
Label: probable inference, quantifier, modal adverb, temporal adverb

(T11) P1. turkish coffee drinkers frequently enjoy a shot of arak.
P2. most people that enjoy a shot of arak also listen to classical oud music.
H. turkish coffee drinkers listen to classical oud music.
Label: probable inference, quantifier, temporal adverb

(T12) P1. if you regularly eat humus, then you also enjoy tabouli.
P2. most people that enjoy tabouli insist on having mint tea with food.
H. if you eat humus, then you insist on having mint tea with food.
Label: probable inference, quantifier, temporal adverb

(T13) P1. cricket players rarely hit a home run.
P2. mary hits a home run.
H. mary isn't a cricket player.
Label: probable inference, quantifier, temporal adverb, negation

(T14) P1. many jazz guitarists can play more than 100 chords.
P2. few violinists can play more than 10 chords.
P3. john can play more than 80 chords.
H. john is a jazz guitarist.
Label: unclear, quantifiers

(T15) P1. mary is tall.
P2. john is taller than mary.
H. john is tall.
Label: entailment, comparative adjective, transitivity

(T16) P1. mary isn't tall.
P2. mary is taller than john.
H. john isn't tall.
Label: entailment, comparative adjective, transitivity

(T17) P1. john is always as punctual as mary.
P2. sam is usually more punctual than john.
H. sam is more punctual than mary.
Label: probable inference, quantifier, modal/temporal adverb

(T18) P1. many linguists know formal language theory.
P2. most linguists that know formal language theory dislike experimental work.
H. many linguists dislike experimental work.
Label: probable inference, quantifier

(T19) P1. most conservatives don't usually support free university education.
P2. john supports free university education.
H. john isn't a conservative.
Label: probable inference, quantifier, modal/temporal adverb, negation

(T20) P1. if you are a guitarist, then you are probably a stones fan.
P2. if you are a stones fan, then you generally prefer the doors to the beatles.
H. it is not the case that every person that doesn't prefer the doors to the beatles is a person that isn't a guitarist.
Label: probable inference, modal adverbs, quantifies, relative clause, negation, conditional, comment: because the negation is in the relative clause, it acts with a narrow scope, see also 21 for wide scope

(T21) P1. if you are a guitarist, then you are probably a stones fan.
P2. if you are a stones fan, then you generally prefer the doors to the beatles.
H. every person that doesn't prefer the doors to the beatles isn't a guitarist.
Label: probable inference, modal adverbs, quantifies, relative clause, negation, conditional

(T22) P1. 80 percent of basketball players are tall.
H. basketball players are probably tall.
Label: probable inference, percentage determiner, modal adverb

(T23) P1. 80 percent of basketball players are tall.
H. few basketball players aren't tall.
Label: probable inference, percentage determiner, negation

(T24) P1. prolog programmers definitely use tail recursion.
P2. logicians are frequently prolog programmers.
H. logicians probably use tail recursion.
Label: probable inference, quantifier, modal adverb

(T25) P1. john is tall.
P2. all guitarists are logicians.
H. john is tall and all guitarists are logicians.
Label: fol validity, conjunction

(T26) P1. john is tall.
P2. all guitarists are logicians.
H. john is tall.
Label: fol validity, weakening

(T27) P1. john is taller than mary.
H. john is tall.
Label: probable inference, comparative adjective

(T28) P1. john is taller than mary.
P2. mary is taller than sam.
H. john is taller than molly.
Label: probable inference, comparative adjective, transitivity

(T29) P1. john is taller than mary.
P2. sam is taller than mary.
P3. kate is taller than christine.
H. kate is taller than john.
Label: probable inference, comparative adjective

(T30) P1. all intermediate logic students are stones fans.
P2. john is an intermediate logic student.
H. john is a stones fan.
Label: fol validity, quantifier

(T31) P1. john is a guitarist.
P2. most guitarists aren't logicians.
H. john isn't a logician.
Label: probable inference, negation, modus ponens

(T32) P1. john is a logician.
P2. most guitarists aren't logicians.
H. john isn't a guitarist.
Label: probable inference, negation, modus tollens

(T33) H. it is not the case that john is taller than most people.
Label: probable inference, law of great numbers (object), negation

(T34) H. a few people are shorter than john.
Label: probable inference, law of great numbers (subject)

(T35) H. it is not the case that few people are shorter than john.
Label: probable inference, law of great numbers (subject), negation

(T36) P1. john isn't a guitarist.
H. it is not the case that john is a guitarist.
Label: fol validity, wide-scope negation

(T37) P1. mary is 190 centimeters tall.
P2. mary is tall.
P3. molly is 184 centimeters tall.
P4. molly is tall.
P5. ruth is 180 centimeters tall.
P6. ruth is tall.
P7. helen is 178 centimeters tall.
P8. helen is tall.
P9. athena is 166 centimeters tall.
P10. athena isn't tall.
P11. artemis is 157 centimeters tall.
P12. artemis isn't tall.
P13. joanna is 160 centimeters tall.
P14. joanna isn't tall.
P15. kate is 162 centimeters tall.
P16. kate isn't tall.
P17. christine is 178 centimeters tall.
H. christine is tall.
Label: ENTAILMENT, POSITIVE ADJECTIVES, VAGUENESS

(T38) P1. mary is 189 centimeters tall.
P2. mary is tall.
P3. molly is 184 centimeters tall.
P4. molly is tall.
P5. ruth is 180 centimeters tall.
P6. ruth is tall.
P7. helen is 178 centimeters tall.
P8. helen is tall.
P9. athena is 166 centimeters tall.
P10. athena isn't tall.
P11. artemis is 157 centimeters tall.
P12. artemis isn't tall.
P13. joanna is 160 centimeters tall.
P14. joanna isn't tall.
P15. kate is 162 centimeters tall.
P16. kate isn't tall.
P17. christine is 163 centimeters tall.
H. christine isn't tall.
Label: ENTAILMENT, POSITIVE ADJECTIVES, VAGUENESS

(T39) P1. many jazz guitarists can play more than 2 chords.
P2. few violinists can play more than 1 chord.
P3. john can play more than 2 chords.
H. john is a jazz guitarist.
Label: unclear, quantifiers

(T40) P1. many jazz guitarists can play more than 5 chords.
P2. few violinists can play more than 2 chords.
P3. john can play more than 4 chords.
H. john is a jazz guitarist.
Label: unclear, quantifiers

(T41) P1. john is 180 centimeters tall.
P2. john is 6 foot tall.
P3. mary is 6 foot tall.
H. mary is 180 centimeters tall.
Label: entailment, comparative, measure

(T42) P1. mary is 190 centimeters tall.
P2. mary is tall.
P3. molly is 184 centimeters tall.
P4. molly is tall.
P5. ruth is 180 centimeters tall.
P6. ruth is tall.
P7. helen is 178 centimeters tall.
P8. helen is tall.
P9. athena is 166 centimeters tall.
P10. athena isn't tall.
P11. artemis is 157 centimeters tall.
P12. artemis isn't tall.
P13. joanna is 160 centimeters tall.
P14. joanna isn't tall.
P15. kate is 162 centimeters tall.
P16. kate isn't tall.
P17. christine is 171 centimeters tall.
H. (it is not the case that christine is tall) and (it is not the case that christine isn't tall).
Label: fol contradiction, positive adjectives, vagueness

(T43) P1. john is taller than mary.
P2. john is 185 centimeters tall.
P3. mary is 5 foot tall.
H. 185 centimeters is more than 5 foot.
Label: ENTAILMENT, COMPARATIVE, MEASURE

(T44) P1. john is taller than mary.
P2. mary is 5 foot tall.
H. john is more than 5 foot tall.
Label: ENTAILMENT, COMPARATIVE, MEASURE

(T45) P1. john is a guitarist.
P2. guitarists are kind of musicians.
H. john is a musician.
Label: PROBABLE INFERENCE, GENERALISED VAGUENESS, NOUN MODIFIER

(T46) P1. john is a guitarist.
P2. guitarists are pretty much musicians.
H. john is a musician.
Label: PROBABLE INFERENCE, GENERALISED VAGUENESS, NOUN MODIFIER

(T47) P1. john is a basketball player.
P2. basketball players are tall.
H. john is taller than 50 percent of people.
Label: PROBABLE INFERENCE, COMPARATIVE ADJECTIVE, MODAL ADVERB, PERCENTAGE DETERMINER

(T48) P1. many jazz guitarists can play more than 100 chords.
P2. few violinists can play more than 10 chords.
P3. john can play more than 80 chords.
H. john isn't a violinist.
Label: PROBABLE INFERENCE, QUANTIFIERS

(T49) P1. many jazz guitarists can play more than 100 chords.
P2. few violinists can play more than 10 chords.
P3. john can play less than 15 chords.
H. john is a violinist.
Label: unclear, quantifiers

(T50) P1. many jazz guitarists can play more than 100 chords.
P2. few violinists can play more than 10 chords.
P3. john can play more than 90 chords.
H. john is a jazz guitarist.
Label: unclear, quantifiers

(T51) P1. jazz guitarists are musicians.
P2. violinists are musicians.
P3. john is a musician.
P4. many jazz guitarists can play more than 100 chords.
P5. few violinists can play more than 10 chords.
P6. john can play more than 90 chords.
H. john is a jazz guitarist.
Label: unclear, quantifiers

(T52) P1. jazz guitarists are musicians.
P2. violinists are musicians.
P3. john is a musician.
P4. musicians can play more than 1 chord.
P5. many jazz guitarists can play more than 100 chords.
P6. few violinists can play more than 10 chords.
P7. john can play more than 90 chords.
H. john is a jazz guitarist.
Label: unclear, quantifiers

(T53) P1. john is a basketball player.
P2. basketball players are taller than most non basketball players.
H. john is taller than most non basketball players.
Label: probable inference, comparative adjective, modal adverb

(T54) P1. few people are basketball players.
P2. basketball players are taller than most non basketball players.
P3. john is a basketball player.
H. john is taller than most people.
Label: probable inference, comparative adjective, modal adverb

(T55) P1. 99 percent of people are non basketball players.
P2. basketball players are taller than most non basketball players.
P3. john is a basketball player.
H. john is taller than most people.
Label: probable inference, comparative adjective, modal adverb, percentage determiner

(T56) P1. if you regularly eat humus, then you also enjoy tabouli.
P2. people that enjoy tabouli insist on having mint tea with food.
H. if you regularly eat humus, then you insist on having mint tea with food.
Label: probable inference, quantifier, temporal adverb

(T57) P1. if you eat humus, then you also enjoy tabouli.
P2. people that enjoy tabouli insist on having mint tea with food.
H. if you eat humus, then you insist on having mint tea with food.
Label: probable inference, transitivity of implication, comment: doesn't work with planes, because the generic plural isn't transitive even combined with optimized universal

(T58) P1. if you eat humus, then you also enjoy tabouli.
P2. john eats humus.
H. john enjoys tabouli.
Label: probable inference, modus ponens

(T59) P1. if you eat humus, then you also enjoy tabouli.
P2. people that enjoy tabouli insist on having mint tea with food.
H. if john eats humus, then john insists on having mint tea with food.
Label: probable inference, transitivity of implication

(T60) P1. if you regularly eat humus, then you also enjoy tabouli.
P2. people that enjoy tabouli insist on having mint tea with food.
H. if john regularly eats humus, then john insists on having mint tea with food.
Label: probable inference, transitivity of implication

(T61) P1. mary is 190 centimeters tall.
P2. mary is tall.
P3. kate is 162 centimeters tall.
P4. kate isn't tall.
P5. christine is 185 centimeters tall.
H. christine is tall.
Label: probable inference, positive adjectives, vagueness

(T62) P1. few people are musicians.
P2. few people are non musicians.
H. (it is not the case that john is a musician) and (it is not the case that john is a non musician).
Label: fol contradiction, comparative adjective, grey area

(T63) P1. john is tall.
P2. few guitarists are logicians.
H. john is tall and few guitarists are logicians.
Label: fol validity, conjunction, quantifiers

(T64) P1. john is tall.
P2. john is a musician.
H. john is tall and john is a musician.
Label: fol validity, conjunction

(T65) P1. all guitarists are probably logicians.
H. most guitarists are logicians.
Label: probable inference, quantifiers, modal adverb

(T66) P1. all guitarists are logicians.
H. all guitarists are probably logicians.
Label: entailment, quantifier, modal adverb

(T67) P1. few guitarists are logicians.
H. all guitarists are logicians.
Label: contradiction, quantifiers

(T68) P1. all guitarists are probably logicians.
H. all guitarists are probably logicians.
Label: fol validity, quantifier, modal adverb

(T69) P1. all guitarists are logicians.
H. all guitarists are logicians.
Label: fol validity, quantifier

(T70) P1. mary is 190 centimeters tall.
P2. mary is tall.
P3. kate is 162 centimeters tall.
P4. kate isn't tall.
P5. christine is 179 centimeters tall.
H. christine is tall.
Label: probable inference, positive adjectives, vagueness

(T71) P1. mary is 190 centimeters tall.
P2. mary is tall.
P3. kate is 188 centimeters tall.
P4. kate isn't tall.
P5. christine is 189 centimeters tall.
H. it is not the case that christine isn't tall.
Label: unclear, negative adjectives, vagueness

(T72) P1. mary is 190 centimeters tall.
P2. mary is tall.
P3. kate is 189 centimeters tall.
P4. kate isn't tall.
P5. christine is 188 centimeters tall.
H. christine isn't tall.
Label: entailment, positive adjectives, vagueness

(T73) P1. mary is 190 centimeters tall.
P2. mary is tall.
P3. kate is 180 centimeters tall.
P4. kate is tall.
P5. helen is 170 centimeters tall.
P6. helen isn't tall.
P7. christine is 165 centimeters tall.
H. christine isn't tall.
Label: entailment, negative adjectives, vagueness

(T74) P1. kate is 190 centimeters tall.
P2. kate is tall.
P3. helen is 180 centimeters tall.
P4. helen isn't tall.
P5. christine is 190 centimeters tall.
H. christine is tall.
Label: entailment, vagueness

(T75) P1. few people are basketball players.
P2. basketball players are taller than most non basketball players.
P3. john is a basketball player.
H. john is taller than many people.
Label: probable inference, comparative adjective, modal adverb

(T76) P1. all violinists are musicians.
P2. all musicians read music.
H. all violinists read music.
Label: fol validity, quantifiers

(T77) P1. all violinists are musicians.
P2. all musicians read music.
H. 99 percent of violinists read music.
Label: ENTAILMENT, QUANTIFIERS, PERCENTAGE DETERMINER

(T78) P1. every guitarist is a logician.
P2. if every guitarist is a logician, then every musician reads music.
P3. john is a musician.
H. john reads music.
Label: FOL VALIDITY, IMPLICATION, QUANTIFIER

(T79) P1. john is a basketball player.
P2. basketball players are taller than non basketball players.
H. john is taller than 80 percent of people.
Label: UNCLEAR, COMPARATIVE ADJECTIVE, MODAL ADVERB, PERCENTAGE DETERMINER

(T80) P1. most animals don't fly.
P2. most birds fly.
P3. every bird is an animal.
H. most animals aren't birds.
Label: PROBABLE INFERENCE, QUANTIFIERS

(T81) P1. most birds fly.
H. a few birds fly.
Label: ENTAILMENT, QUANTIFIERS

(T82) P1. many logicians are musicians.
P2. john is a logician.
H. john is a musician.
Label: PROBABLE INFERENCE, QUANTIFIER

(T83) P1. many logicians are musicians.
P2. john is a musician.
H. john is a logician.
Label: UNCLEAR, QUANTIFIER

(T84) P1. many logicians are musicians.
P2. most people aren't logicians.
P3. john is a musician.
H. john is a logician.
Label: UNCLEAR, QUANTIFIERS

References

Adams, Ernest. 1998. *A Primer of Probability Logic*. Stanford: CSLI Publications.

Barendregt, Hendrik Pieter. 1992. Lambda calculi with types. *Handbook of logic in computer science* 2:117–309.

Barwise, J. and R. Cooper. 1981. Generalised quantifiers and natural language. *Linguistics and Philosophy* 4:159–219.

Bernardy, Jean-Philippe, Rasmus Blanck, Stergios Chatzikyriakidis, and Shalom Lappin. 2018. A compositional Bayesian semantics for natural language. In *Proceedings of the International Workshop on Language, Cognition and Computational Models, COLING 2018, Santa Fe, New Mexico*, pages 1–11.

Bernardy, Jean-Philippe and Stergios Chatzikyriakidis. 2019a. A computational treatment of anaphora and its algorithmic implementation. *Ms, University of Gothenburg* .

Bernardy, Jean-Philippe and Stergios Chatzikyriakidis. 2019b. A wide-coverage symbolic natural language inference system. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*. ACL.

Borgström, Johannes, Andrew D. Gordon, Michael Greenberg, James Margetson, and Jurgen Van Gael. 2013. Measure transformer semantics for Bayesian machine learning. *Logical Methods in Computer Science* 9:1–39.

Bowman, Samuel R, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of EMNLP*, pages 632–642.

Cann, Ronnie, Ruth Kempson, and Eleni Gregoromichelaki. 2009. *Semantics: An Introduction to Meaning in Language*. ISBN 9780521819626.

Carlson, Greg N. 1982. Generic terms and generic sentences. *Journal of Philosophical Logic* 11(2):145–181.

Chatzikyriakidis, S. and Z. Luo. 2017. On the interpretation of common nouns: Types versus predicates. In *Modern Perspectives in Type-Theoretical Semantics*, pages 43–70. Springer.

Chierchia, Gennaro. 1995. *Dynamics of Meaning: Anaphora, Presupposition, and the Theory of Grammar*. University of Chicago Press.

Cooper, R., D. Crouch, J. van Eijck, C. Fox, J. van Genabith, J. Jaspars, H. Kamp, D. Milward, M. Pinkal, M. Poesio, and S. Pulman. 1996. Using the framework. Technical report LRE 62-051r, The FraCaS consortium. ftp://ftp.cogsci.ed.ac.uk/pub/FRACAS/del16.ps.gz.

Cooper, R., S. Dobnik, S. Lappin, and S. 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, Sweden: Association of Computational Linguistics.

Cooper, R., S. Dobnik, S. Lappin, and S. Larsson. 2015. Probabilistic type theory and natural language semantics. *Linguistic Issues in Language Technology* 10:1–43.

Dagan, Ido, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *Machine learning challenges. Evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pages 177–190. Springer.

Edgington, Dorothy. 2001. The philosophical problem of vagueness. *Legal Theory* 7(4):371–378.

Emerson, Guy and Ann Copestake. 2017a. Semantic composition via probabilistic model theory. In *IWCS 2017 - 12th International Conference on Computational Semantics - Long papers*.

Emerson, Guy and Ann Copestake. 2017b. Variational inference for logical inference. *CoRR* abs/1709.00224.

Fox, Chris and Shalom Lappin. 2005. *Foundations of Intensional Semantics*. Blackwell.

Gilio, Angelo, Niki Pfeifer, and Giuseppe Sanfilippo. 2015. Transitive reasoning with imprecise probabilities. In S. Destercke and T. Denoeux, eds., *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 95–105. Cham: Springer International Publishing. ISBN 978-3-319-20807-7.

Goodman, Noah and Daniel Lassiter. 2015. Probabilistic semantics and pragmatics: Uncertainty in language and thought. In S. Lappin and C. Fox, eds., *The Handbook of Contemporary Semantic Theory, Second Edition*, pages 655–686. Malden, Oxford: Wiley-Blackwell.

Goodman, Noah, V. K. Mansinghka, D. Roy, K. Bonawitz, and J. Tenenbaum. 2008. Church: a language for generative models. In *Proceedings of the 24th Conference Uncertainty in Artificial Intelligence (UAI)*, pages 220–229.

Goodman, Noah and Andreas Stuhlmüller. 2014. The Design and Implementation of Probabilistic Programming Languages. http://dippl.org. Accessed: 2018-4-17.

Groenendijk, Jeroen and Martin Stokhof. 1991. Dynamic predicate logic. *Linguistics and Philosophy* 14(1):39–100.

Grove, Julian and Jean-Philippe Bernardy. 2021. Probabilistic compositional semantics, purely. In *Proceedings of LENLS18*.

Grove, Julian, Jean-Philippe Bernardy, and Stergios Chatzikyriakidis. 2021. From compositional semantics to bayesian pragmatics via logical inference. In *Proceedings of Natural Logic meets Machine Learning 2021*.

Halpern, Joseph Y. 2017. *Reasoning About Uncertainty*. Cambridge, MA, USA: MIT Press.

Heim, Irene. 1982. *The Semantics of Definite and Indefinite Noun Phrases*. Ph.D. thesis, UMass Amherst.

Heim, Irene and Angelika Kratzer. 1998. *Semantics in Generative Grammar*. Blackwell.

Itegulov, Daniyar, Ekaterina Lebedeva, and Bruno Woltzenlogel Paleo. 2018. Sensala: A dynamic semantics system for natural language processing. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 123–127.

Kamp, Hans. 2013. Two theories about adjectives. In *Meaning and the Dynamics of Interpretation*, pages 225–261. Brill.

Kamp, Hans and Uwe Reyle. 1993. *From Discourse to Logic*. Dordrecht: Kluwer Academic Publishers.

Keenan, Edward and L.M. Falz. 1985. *Boolean Semantics for Natural Language*. Berlin, New York: Springer.

Klein, Ewan. 1980. A semantics for positive and comparative adjectives. *Linguistics and Philosophy* 4(1):1–45.

Lappin, Shalom. 2015. Curry typing, polymorphism, and fine-grained intensionality. In S. Lappin and C. Fox, eds., *The Handbook of Contemporary Semantic Theory, Second Edition*, pages 408–428. Malden, MA and Oxford: Wiley-Blackwell.

Lappin, Shalom. 2018. Towards a computationally viable framework for semantic representation. In *Proceedings of the Symposium on Logic and Algorithms in Computational Linguistics 2018*, pages 47–63. Stockholm University, DiVA Portal for digital publications.

Lassiter, Daniel and Noah Goodman. 2017. Adjectival vagueness in a Bayesian model of interpretation. *Synthese* 194:3801–3836.

Lau, Jey Han, Alexander Clark, and Shalom Lappin. 2017. Grammaticality, acceptability, and probability: A probabilistic view of linguistic knowledge. *Cognitive Science* 41(5):1202–1241.

Luo, Z. 2012. Common nouns as types. In D. Bechet and A. Dikovsky, eds., *Logical Aspects of Computational Linguistics (LACL'2012). LNCS 7351*.

Montague, Richard. 1970. English as a formal language. In B. V. et al., ed., *Linguaggi nella Societa e nella Tecnica*.

Montague, Richard. 1974. The proper treatment of quantification in ordinary english. In R. Thomason, ed., *Formal Philosophy*. New Haven: Yale UP.

Partee, Barbara H. 2007. Compositionality and coercion in semantics: The dynamics of adjective meaning. *Cognitive foundations of interpretation* pages 145–161.

Peters, Stanley and Dag Westerståhl. 2006. *Quantifiers in Language and Logic*. Oxford University Press UK.

Pfeifer, Niki. 2013. The new psychology of reasoning: A mental probability logical perspective. *Thinking & Reasoning* 19(3-4):329–345.

Pfeifer, Niki and Giuseppe Sanfilippo. 2018. Probabilistic semantics for categorical syllogisms of figure II. In D. Ciucci, G. Pasi, and B. Vantaggi, eds., *Scalable Uncertainty Management*, pages 196–211. Springer International Publishing. ISBN 978-3-030-00461-3.

Piantadosi, Steven, Harry Tily, and Edward Gibson. 2011. Word lengths are optimized for efficient communication. *Proceedings of the National Academy of Sciences of the United States of America* 108:3526–9.

Ranta, Aarne. 1994. *Type-Theoretical Grammar*. Oxford University Press.

Ranta, Aarne. 2004. Grammatical framework. *Journal of Functional Programming* 14(2):145–189.

Shan, Chung-chieh and Norman Ramsey. 2017. Exact bayesian inference by symbolic disintegration. In *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages*, POPL, pages 130–144.

Suppes, Patrick. 1966. Probabilistic inference and the concept of total evidence. In J. Hintikka and P. Suppes, eds., *Aspects of Inductive Logic*, vol. 43 of *Studies in Logic and the Foundations of Mathematics*, pages 49–65. Elsevier.

Sutton, Peter R. 2017. Probabilistic approaches to vagueness and semantic competency. *Erkenntnis* .

van Eijck, Jan and Shalom Lappin. 2012. Probabilistic semantics for natural language. In Z. Christoff, P. Galeazzi, N. Gierasimszuk, A. Marcoci, and S. Smets, eds., *Logic and Interactive Rationality (LIRA), Volume 2*. University of Amsterdam: ILLC.

Vilnis, Luke, Xiang Li, Shikhar Murty, and Andrew McCallum. 2018. Probabilistic embedding of knowledge graphs with box lattice measures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 263–272. Association for Computational Linguistics.

Williamson, Timothy. 1994. *Vagueness*. Routledge.