Bayesian Pragmatics with Distributional Semantics: a Computational Approach to Metaphorical Predication

Reuben Cohn-Gordon and Leon Bergen
Stanford

Abstract

Previous Bayesian formalizations of metaphor interpretation have required hand-crafted semantics and choices of utterance. We propose an elaboration of (Kao, Bergen, and Goodman 2014) which rests on a word embedding semantics, thereby replacing hand-crafted knowledge with information encoded in pretrained word vectors. We show that this model can detect, interpret and paraphrase metaphor. This provides a particular instance of the more general project of scaling Bayesian models of pragmatics to real world data and distributional semantics.

TODO: fix log issue chris raises

TODO: fix detection generalize judgment case or get large scale case working: perhaps use ordering of large scale case for judgment redo animal cases on simple set do noun noun compounds

Metaphor presents a compelling theoretical challenge for semantics and pragmatics, with corresponding tasks for NLP, of metaphor identification, understanding and paraphrase. As such, it has proven to be a topic of interest among philosophers (Black 1955, Davidson 1978) and linguists (Lakoff and Johnson 2008, and recently has seen much work in computational modeling.

Pragmatic accounts of metaphor, stemming from (Davidson 1978) and (Grice 1975), attempt to derive metaphorical meaning as the inference of a language user reasoning about an expression heard in a given context. Grice suggests that a listener infers that a heard expression is metaphorical on account of the unlikelihood of its literal interpretation being true.

This pragmatic view of metaphor has been formalized within the paradigm of Bayesian pragmatics[1] by (Kao, Bergen, and Goodman 2014), by conducting joint inference over the attributes of a referent and a *question under discussion*. Excitingly, this Bayesian formalization offers not only a way to recognize that metaphorical expressions are not to be interpreted literally, but also a way to access the meaning the metaphor *does* convey.

[1]See Goodman and Stuhlmüller 2013 for an introduction to the Rational Speech Acts (RSA) model more generally

While this Bayesian model offers a natural formalization of the intuitions of earlier pragmaticists, and has allowed for rigorous experimental work (e.g. Frank and Goodman 2012), it has not yet been integrated with modern advances in computational linguistics, or adapted to deal with real-world data.

Given the success of pretrained word embeddings at a range of linguistic tasks (see Turney and Pantel 2010), it is natural to wonder whether such embedding spaces can provide a knowledge base for a model of metaphor which is scalable to real world data.

Building on the work of (Kao, Bergen, and Goodman 2014), we design and implement a distributional Bayesian model of metaphor meaning, which applies RSA to a semantics grounded in pretrained word embeddings. This allows us to generalize the RSA model of metaphor well beyond the hand-constructed examples which previous pragmatic models require. We will refer to this distributional form of RSA as DistRSA.

We apply our model to three task, which we view as crucial to the computational processing of metaphorical language: metaphor identification, generation and paraphrase.

We consider not only copular metaphors of the form "A is a B" (e.g. "John is a shark."), but also metaphorical modification, e.g "fiery temper" or "golden opportunity". We consider the possibility that the meaning of metaphorical phrases such as these can be interpreted as the posterior position of the head noun, after the model observes the modifying adjective.

We begin with a brief introduction to the RSA paradigm. We then sketch out how RSA, in particular QUD RSA can be applied to a distributional model of semantics, and describe the technical challenges this involves.

1 A Working Definition of Metaphor

Metaphor exists in many syntactic forms, and generally eludes easy definition. For present purposes, we focus on metaphorical *predication*, and in particular, copular predication of the form: "A is a B". These are cases

where a predicate only applies to an entity in certain regards. What these regards are is contextually determined. Thus, *being a brick wall* conveys different things for each of the following:

(1) • "The piece of granite is a rock." : literal.

(2) • "The bread is a rock": with respect to being hard.
• Paraphrase: "The bread is stale."

(3) • "The unconscious man is a rock." : with respect to being unchanging.
• Paraphrase:"The patient is unresponsive."

The first example above is one of literal predication: the granite really is a rock. For the other two cases, however, only certain aspects of rocks are shared between subject and predicate. The task of metaphor understanding can be summed up as the identification of which elements these are.

Parallel to metaphorical predication, one also finds cases of modification, where an adjective modifies a noun in certain regards and not others. Thus, a *fiery temper* shares with fire different properties to a *fiery dish*, and in turn, both differs from a more literal usage, as with a *fiery stove*.

A key property of metaphor is its productivity: it is not possible to enumerate every metaphorical sentence or phrase and its meaning. However, it should be noted that metaphors very commonly conventionalize, to the point that predicates used metaphorically often absorb the metaphorical meaning into their literal meaning. Thus, "John is a fool." would rarely, at present, be interpreted to mean that John performs dances and songs in a monarch's court. One the other hand, "John is a butcher." could either mean that he processes meat (literal), or that he is violent (metaphorical), depending on the context.

If metaphor was not productive, it would be possible for a computer to understand metaphor simply by rote memorization - it is the semantic productivity of metaphor which makes it an interesting computational problem, which, in our opinion, merits the use of a probabilistic generative model.

2 Distributional Semantics and Metaphor

Distributional models of word meaning have proven hugely useful in a variety of applications in computational linguistics and NLP. These assign each word in a language a point in a high-dimensional vector space. It is possible to learn mappings from words to points roughly according to co-occurrence between neighboring words observed in large text corpora, so that words which appear in similar contexts are close in the vector space under cosine distance. Google's Word2Vec (Mikolov et al. 2013) and Stanford's GloVe (Pennington, Socher, and Manning 2014) are two standard pretrained sets of word vectors.

A shared property of both GloVe and Word2Vec is an approximately linear structure which, for various quadruples of words (A,B,C,D), such that A is to B as C is to D, the corresponding pretrained vectors approximately satisfy the equation $\overrightarrow{A} - \overrightarrow{B} = \overrightarrow{C} - \overrightarrow{D}$, where $\overrightarrow{A}$ is the word vector corresponding to the word A. For instance, the nearest word vector to the point $(\overrightarrow{\text{king}} - \overrightarrow{\text{man}} + \overrightarrow{\text{woman}})$ in the Word2Vec embedding space is $\overrightarrow{\text{queen}}$.

Intrinsically, word embeddings are used in deep learning systems, as a form of transfer learning: for instance, language models for translation and other tasks are greatly helped by using pretrained word vectors.

Because of the rich semantic information encoded in word embeddings, many attempts have been made to utilize them for computational tasks requiring semantic understanding, such as sentiment analysis (Socher et al. 2013) and abstractness detection (Turney and Littman 2003).

These usages of word vectors often blur the word/object distinction, in the sense that the pretrained embeddings can be understood to provide *world* knowledge rather than solely knowledge about language. Thus, the vector associated with the word "cat" can be interpreted as its denotation, giving information about the animal *cat*.

Metaphor is an obvious candidate for approaches that use distributional semantics: a wide variety of attempts have been made to leverage the information inherent in pretrained word vectors for the detection, interpretation and paraphrase of metaphor (see Shutova 2016 for a comprehensive overview of proposed systems.).

An intuition which is voiced in many of these approaches is that metaphorical predicates should agree with only certain features of the subject. For instance:

> "Computing a meaning always involves activating context-appropriate features and inhibiting or deactivating inappropriate features." Kintsch 2000

A similar point is made about adjective-noun (AN) composition by (Grefenstette 2013):

> "In turn, through composition with its argument, I expect the function for such an adjective to strengthen the properties that characterise it in the representation of the object it takes as argument."

In the philosophical literature on metaphor, a similar point is made by (Black 1955):

> "We can say that the principal subject is "seen through" the metaphorical expression - or, if we prefer, that the principal subject is "projected upon" the field of the subsidiary subject." - (Black 1955)

The Bayesian model of metaphor proposed by (Kao, Bergen, and Goodman 2014) offers a way of capturing the dynamic to which these two quotes allude: in this model, a listener jointly infers a Question Under Discussion (QUD), which dictates which aspects of the modifier or predicate to care about, and a world state, based on the information about the subject. However, this sort of joint inference of world state and topic or QUD is absent from previous distributional approaches to metaphor.

Our proposal is to unite these two fruitful avenues of research, by constructing an explicitly Bayesian RSA model of metaphor which uses a word vector semantics. In this setting, world states will become points in the space and, in a fortuitous alignment with Black's choice of language, QUDs will be orthogonal projections which act on their subject.

We now review the general RSA framework, as well as the non-literal extension to RSA (Kao, Bergen, and Goodman 2014) on which our model will build.

3 An Introduction to Rational Speech Acts

Bayesian probability provides an elegant and practical way of formalizing pragmatics (Goodman and Stuhlmüller 2013). For a comprehensive and basic introduction, we recommend `http://gscontras.github.io/ESSLLI-2016/chapters/1-introduction.html`.

The general form of the model posits listeners and speakers, both of which are represented as conditional probability distributions. Speakers are of the form P(U|W) while listeners are P(W|U), where W is the type of the world and U is the type of utterances. Thus, speakers are distributions over possible utterances given worlds, and listeners are distributions over possible worlds given utterances.

RSA works by defining successive levels of speaker and listener recursively. L_0, the most basic listener, conditions on the truth of an utterance u in order to update their prior on worlds. For this, we need a semantics, in the form of an interpretation function I of type (U,W $\rightarrow \{0,1\}$)[2]. Formally, where P(w) is the prior on worlds, and I is an interpretation function:

$$(4) \quad L_0(w|u) \propto I(u,w) * P(w)$$

The speaker, S_1 attempts, in turn, to produce the utterance u which maximizes a utility function U. For the simplest case of RSA, we define U as[3]:

$$(5) \quad U(u,w) = log(L_0(w|u))$$

Using the utility function in (5) and assuming a uniform prior on utterances, we obtain a distribution for S_1 proportional to the utility:

$$(6) \quad S_1(u|w) \propto U(u,w)$$

L_1, in turn, conditions on a pragmatic speaker, i.e. S_1, being in a world which would have produced the heard utterance:

$$(7) \quad L_1(w|u) \propto S_1(u|w) * P(w)$$

RSA requires hand-constructed inputs for two aspects of the system: the semantics used in L_0 and the choice of possible utterances in S_1. These are both controlling factors on the scalability and domain-genericity of RSA, since each case of RSA requires the provision of both. As we shall see, RSA with Questions under Discussion introduces a third such controlling factor: the need for a predetermined set of QUDs to be provided to the model.

4 RSA with Questions under Discussion

Extending the RSA paradigm, (Kao, Bergen, and Goodman 2014) introduce an additional feature in order to model non-literal meaning such as metaphor and hyperbole. This is the notion of a Question under Discussion, inspired by the closely related notion of the same name proposed by (Roberts 1996).

In the context of RSA, a QUD can be understood as a function of type (W$\rightarrow$powerset(W)), which discards certain information about the world. For instance, suppose we have a model in which worlds consist of triples of properties pertaining to a subject, e.g. the lawyer in the sentence "The lawyer is a shark.".

One such world might look like: ⟨clever=True,vicious=True,breathes underwater=False⟩.

As for the L_0 distribution, let us suppose it is a uniform distribution over those world triples in which *breathes underwater* is set to *False*. This represents our prior knowledge that lawyers cannot breathe underwater. The L_1 will have the same prior on worlds.

[2] This semantics can be generalized to 'soft RSA", where the result type of I is the real interval from 0 to 1.

[3] Basic enrichments to this setup include the subtraction of a cost term from U(u|w), and a so-called "rationality parameter" which multiplies U(u|w)

The speaker S_1 will have a uniform prior on utterances. In (**??**), this is a finite set of utterances like *shark*, *man*, etc. This is a modeling assumption, since linguistic agents do not really have a finite set of utterances (see Chomsky 2002).

Then one example of a QUD would be a function which mapped any triple to its first element[4]. This would be the *cleverness* QUD, i.e. the QUD that only cares about whether the lawyer is clever.

(Kao, Bergen, and Goodman 2014) has world states of exactly this sort. The key innovation of the model is that the speaker, of type P(w|u,q), for an utterance u and QUD q, chooses the utterance which causes the listener's world state after hearing u to be the same as the speaker's world state *under the QUD*. For (Kao, Bergen, and Goodman 2014), utterances are nouns describing the subject, such as "shark", "human", etc. Formally, the L_0 remains the same as in basic RSA, and the S_1 is defined as follows:

$$\text{(8)} \quad \text{U(u,w,q)} = \sum_{w'} \delta_{q(w)=q(w')} * L_0(w'|u)$$

$$\text{(9)} \quad S_1(u|w,q) \propto U(u,w,q)$$

The L_1 then *jointly infers* both a world state and a QUD[5]:

$$\text{(10)} \quad L_1(q,w|u) \propto S_1(u|q,w) * P(w) * P(q)$$

We will, unimaginatively, refer to RSA with the joint QUD inference described here as the QUD RSA model. QUD RSA behaves in ways which appear empirically plausible (see Kao, Bergen, and Goodman 2014 for experimental verification of this claim). Qualitatively, the model is able to infer QUDs for sentences like "John is a shark.", given a simple semantics, and sets of possible utterances and QUDs.

Note that the model is asymmetric, in the sense that "A is a B" results in a different meaning to "B is an A". This accords with linguistic intuitions; as observed by (Way 1991), "The politician is a butcher." does not mean the same as "The butcher is a politician.".

5 RSA in a Distributional Setting: DistRSA

World states and word denotations in QUD RSA, as described in (Kao, Bergen, and Goodman 2014), can already be understood as vectors over the set of two elements, {0,1}, or equivalently as relations between the subject and predicate. (For instance, suppose the meaning of *shark* is <vicious=True,breathes_underwater=False>. This can be rewritten as the vector <1,0>.) The intuition behind our model is to generalize QUD RSA to a vector space over the reals.

Transferring RSA, specifically QUD RSA, to a distributional semantics requires the provision of analogues for the key elements of the QUD RSA model, namely:

- Word Denotation
- World State
- QUD

In each case, there is a natural way to enrich QUD RSA into a distribution setting. Word denotation, as we would expect, is given by the word vector corresponding to the word in question, as supplied by a pre-trained word embedding. We use the 50 dimensional version of GloVe[6]. The meaning of a word, in this paradigm, is a point in this 50 dimensional space.

We treat the world state as a point in this space too. Note that in non-distributional QUD RSA, the world state and the word denotation are also of the same type as each other.

In non-distributional RSA, the L_0 prior on worlds is a uniform distribution over the finite set of possible worlds. For DistRSA, we have an infinite set of possible worlds, corresponding to the points in the word embedding space. Since we want to use to information encoded in the subject of a metaphor (e.g. "The lawyer" in "The lawyer is a shark."), we do not want a uniform distribution over these points.

Thus, for the metaphor "Time is a river.", we define the listener's prior as a multidimensional Gaussian distribution centered around the word vector for *time*. The choice of a Gaussian is made for two reasons: firstly, they are well-behaved mathematically, and allow us to calculate the L_0 distribution analytically, which is necessary for our computational model. Secondly, as noted above, the GloVe word encodes semantic similarity to some degree as cosine distance in the embedding space.

The multidimensional Gaussian distribution weights most heavily those points nearest to a given word, e.g. "Time" in "Time is a river.". The points in the distribution represent meanings of *time* that might not be encoded in its vector. For instance, some points in the distribution might be closer to $\overrightarrow{\text{irreversible}}$ or $\overrightarrow{\text{continuous}}$. Updating one's prior on what time is like, namely the Gaussian centered on $\overrightarrow{\text{time}}$ so that these points have more

[4] Or equivalently, to the set of worlds which agree on the first element.

[5] QUD RSA is one of a set of *lifted variable* extensions to RSA, where the L_m jointly infers both world state and other variables used in S_n or L_n, for m >n. For examples of lifted variable models, see Kao, Bergen, and Goodman 2014 and Bergen and Goodman 2015

[6] In general, we are agnostic as to the appropriate choice of word embedding space.

weight would represent an update in one's knowledge about time.

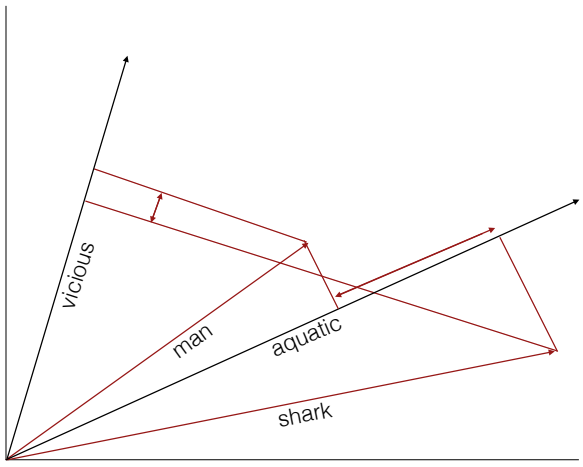
How does this update take place? In non-distributional RSA, the L_0 conditions on the truth of the utterance they hear. However, the notion of truth, which for ordinary RSA amounts to a function from worlds to {0,1} (or [0,1]) is no longer straightforward to define.

We will therefore need a new mechanism for updating the L_0 prior. For DistRSA, the prior on worlds is a Gaussian. Suppose it is centered on "time" and the observed utterance is "river". The prior is then updated so that more weight is placed on the point denoted by "river" (but for the formal definition, see section 6).

The final ingredient necessary to adapt QUD RSA to a distributional setting is the notion of QUD. Recall that the QUD in the non-distribution model maps from worlds to sets of worlds which agree on a particular part of the world state.

Since, in a distributional setting, the axes of a world state vector do not neatly correspond to its attributes[7], our distributional QUD cannot simply be a projection along an axis.

For this reason, the natural implementation of a QUD in a vector space is as an orthogonal projection, parametrized by some hyperplane, which maps from the original 300 dimensional space to a lower dimensional subspace. The simplest case is of a 1 dimensional projection, i.e. a line: here, every point in the original space is dropped in a perpendicular fashion onto the line. In the more general multidimensional case, given a hyperplane in our space we can obtain a function mapping points in the space to new positions, derived by dropping them perpendicularly onto the hyperplane. This projection is a linear transform to a subspace of the original vector space, pictured below in two dimension, for two word vectors, and two potential projection vectors (which here also correspond to words):



In this example, the vectors for *vicious* and *aquatic* each parametrize a QUD mapping all the points in the space (such as *man* and *shark*) to new points. These new points can be thought of as the positions of *man*, *shark* and so on in a new space which cares only about the position of the points with respect to the *vicious* vector.

Again, it is worth noting the similarly of the simple and distributional notions of QUD. In both cases, the QUD discards information, via a many-to-one mapping from worlds to sets of worlds[8].

The intuition behind the projection is that it picks out only certain features of the original vector. For instance, we could imagine a 1 dimensional subspace which measures how *predatory* an object is. This projection, if it exists, would map objects which are similarly predatory to nearby points in the space, and vice versa for points which differ in predatoriness. A higher dimensional hyperplane would care about n features, where n is the dimensionality of the hyperplane. The idea here is that a metaphor might convey multiple things at once. In saying that John is a shark, we are conveying both his speed and predatory nature, but not his scaliness.

So far we have defined our QUDs in terms of projections parametrized by hyperplanes in the word vector space, but have said nothing about how these hyperplanes are to be obtained. We find that words themselves can supply the projection hyperplanes, so that we can, for instance, map the Glove vectors into the subspace parametrized by the vector (a 1 dimensional hyperplane) corresponding to the word *predatory*. Thus, the L_1's set of QUDs can be chosen as the projections corresponding to a set of words, such as the set of animal features in (Kao, Bergen, and Goodman 2014). This means that we can think of our QUDs as words, such as adjectives, which measure aspects of the subject being predicated.

[7] If our space was formed by a basis of co-occurrence vectors, each word would be a dimension, but this is not the case once the dimensionality has been reduced.

[8] This is true in the distributional case because the projection maps more than one point to a single point. Thus, we can recast the projection as a function P from a point v to the inverse image of P(v).

The assumption that appropriate projections exist is based on the much observed linear substructure of word embeddings, which amounts to the claim that word vectors can be roughly decomposed as a sum of a set of other word vectors. While this linearity is very noisy, we find that it suffices for our purposes (largely, we suspect, as a result of the joint inference performed at L_1).

To illustrate how this works informally (but for a formal account, see section 6), we walk through a 2D example of our model, applied to the metaphor "The lawyer is a shark.". As with RSA generally, our model requires a set of possible utterances. In the present case, let us supply the possible utterances as *shark* and *fish*. Informally, we can describe the L_1 calculation of a single metaphorical expression, say "The lawyer is a shark." in distributional RSA as follows:

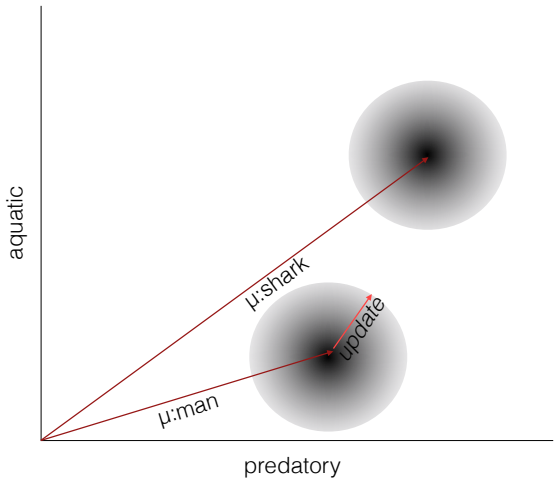
The L_1 world prior over worlds w is a Gaussian with mean at $\overrightarrow{\text{lawyer}}$. The prior over QUDs q, is (in the most simple model, but see section (6) for variations) a uniform distribution over a set of vectors, corresponding to a set of words. Here, the QUDs might be $\overrightarrow{\text{predatory}}$ and $\overrightarrow{\text{scaly}}$.

L_1 then performs joint inference over worlds and QUDs, to obtain a posterior on both, following the observation of the S_1 utterance given w,q.

The S_1, in turn, takes in a world state w (a vector in the space) and a QUD (a hyperplane in the space), and has a uniform prior[9] over a set of possible utterances. Here, let us assume that the possible utterances are *shark*, *fish* and *human*. The S_1 updates a uniform categorical distribution over the possible utterances to favor those which convey the S_1 world state to the listener.

The L_0 prior, in turn, is is identical to the L_1 world prior. To obtain the posterior, the L_0 makes the observation that $\overrightarrow{\text{shark}}$, which is the utterance they hear, is drawn from their prior distribution, and updates their prior according to this observation. The following diagram illustrates this for a 2D case, where the axes correspond to words - in practice, we use a word embedding space of 300 dimensions in which the axes have no special meaning.

[9]We later consider weighting this prior: see section (6.1).



Another way of thinking of DistRSA is as a spatial reference game (e.g. Golland, Liang, and Klein 2010), played in a high dimensional space. In this setting, the S_1 has knowledge of what lawyers (or a particular lawyer) is like, represented by a position of the lawyer in the vector space. They also have a QUD that they care about. They choose the noun which best conveys the known position of *lawyer* with respect to the QUD in question.

6 Technical Overview

While the general format of our model is analogous to previous models in the RSA paradigm, the vectorial setting produces a number of challenges which require novel solutions.

Moreover, previous implementations of RSA have employed prior distributions for both the speaker and listener with finite support. As a result, it is possible in standard RSA to perform exact inference for the L_0, S_1 and L_1. By contrast, our prior is a Gaussian and as such cannot be computed via exact inference. Furthermore, nesting causes approximate inference in the form of Markov Chain Monte Carlo to be unstable.

We therefore compute the L_0 and S_1 posteriors analytically, and only use a single Monte Carlo inference, applied at the L_1 level. We now describe how each of L_0, S_1 and L_1 is computed.

6.0.1 L_0 Definition

To calculate L_0 analytically, we make use of Gaussian conjugacy. Our L_0 posterior density function is defined as follows, where *subject* is the word being predicated (e.g. "lawyer" in "the lawyer is a shark"):

$$L_0(w|u) \propto P_{\mathcal{N}}(w|\mu = subject, \sigma = \sigma_1)P_{\mathcal{N}}(w|\mu = u, \sigma = \sigma_2) \quad (11)$$

The left hand term represents the prior on the world state, and the right hand term represents the observation. By the self conjugacy of Gaussians, (11) can be reduced analytically to a single Gaussian.

Note that truth is not a fundamental concept in distributional RSA. Instead, the semantics is dynamic, so that the literal meaning of an utterance is its effect on the L_0 prior. More precisely, the meaning of "A is a B" is the function from the L_0 prior on A to the L_0 posterior on A after observing B.

The variance of the prior and observation Gaussians, σ_1 and σ_2 respectively, are hyperparameters of our model. See section (6.1) for a discussion of their optimal values.

6.0.2 S_1 Definition

The prior for S_1, which is over possible utterances, is finite, and therefore is straightforward to compute. We refer to the hyperplane which parametrizes the QUD projection as $\overrightarrow{q}$, and abuse notation by having q(v) denote the vector resulting from a projection of v along q. Note that q(v) could either be represented in the dimensionality of the original space or the projection subspace. We will assume the former, so that if $v \in \mathbb{R}^n$, $q(v) \in \mathbb{R}^n$. Then our S_1 is almost identical to the plain QUD RSA model:

(12) $U(u,w,q) = \int_{w'} \delta_{q(w)=q(w')} * L_0(w'|u)$

(13) $S_1(u|w,q) \propto U(u,w,q)$

6.0.3 L_1 Definition

L_1 is defined as follows: TODO

(14) $L_1(w|u) \propto S_1(u|w) P_{\mathcal{N}}(w|\mu = subject, \sigma = \sigma_1)$

We use Hamiltonian Monte Carlo (Neal et al. 2011) to compute the L_1 posterior, a brand of inference algorithm which makes use of gradient information to choose its samples.

Hamiltonian Monte Carlo (HMC) greatly increases efficiency over other Monte Carlo methods, at the cost of requiring gradient information. Fortunately, it is possible for us to calculate gradients, using automatic differentiation.

L_1 must perform joint inference over QUDs and worlds. While the nature of the prior over worlds has already been discussed, two variants of L_1 are possible, as regards to prior on QUDs. The first is to have a categorical distribution over a set of projection hyperplanes (i.e. QUDs), for example those corresponding to n-tuples of words, according to the pretrained embedding.

The second is to have the QUD be a continuous variable. As a result of the way the projection is defined, only the angle and not the magnitude of the projection hyperplane matters. As such, we can have a uniform prior over unit hyperplanes. We will refer to these as two models as the Categorical and Non-Categorical L_1, respectively.

The advantage of the Categorical L_1 is that we can choose a particular set of QUDs, for instance, the vectors corresponding to a given set of words, and weight this prior according to the frequency of these words. This allows us to supply our model with a set of possible QUDs, and have it return a categorical distribution over them, as the L_1 QUD posterior.

The Non-Categorical model, on the other hand, has the contrasting benefit that no set of possible QUDs need be provided to the model. Furthermore, the use of Hamiltonian dynamics for performing the joint inference results in much faster performance.

6.1 Implementation

In section (3) we raised the issue of scalability of RSA, relating to the need for the provision of a hand-crafted semantics, and sets of possible utterances and possible QUDs. In distributional QUD RSA, we avoid the need for a hand-crafted semantics. However, we attempt to go further, by also removing the need for hand chosen utterances and QUDs. We do this by simply choosing sufficiently large sets of possible utterances and QUDs that they can remain constant no matter the metaphor in question. We create categorical distributions over possible utterances and QUDs by weighting the elements in these sets according to their frequency[10]. Our model is computationally efficient enough that this large inference is possible.

We implemented our model in both WebPPL and Python[11]. For the later, we make use of Tensorflow, and Edward in particular (see Tran et al. 2016), in order to perform HMC. Though the model in principle runs in both languages, the Tensorflow implementation is significantly faster, allowing us to run large inferences, over thousands of possible utterances and QUDs. Furthermore, Tensorflow and Edward are GPU compatible, allowing for significant speed up of the HMC.

For our word vectors, we experimented with a number of different variants of the Glove vectors, all available at `https://nlp.stanford.edu/projects/glove/`.

[10]For a measure of frequency, we use the Google N-grams unigrams.

[11]The code of our Python implementation is available at . Readers are warned that for reasons of efficiency, the code is written is a vectorized form, which makes the L_0 and S_1 somewhat opaque.

We obtained our best results with Glove840B, σ_1 of 1, and σ_2 of 0.01. We set the step size of our HMC to 0.01.

7 Evaluation

Our DistRSA model is evaluable in a variety of ways corresponding to various tasks. In order to test it rigorously, we run evaluations according to human judgments, both from corpora and our own experimental data. We compare our results to simpler baseline models which make use of distributional semantics but not RSA.

7.1 Task 1: QUD Identification

The first task we attempt is metaphor understanding: the task of finding a set of words which describe suitable QUDs for a given metaphor. As an example, we show results of the categorical L_1 on the animal-based set of possible utterances and QUDs supplied by (Kao, Bergen, and Goodman 2014, page 272). We obtain the following distributions over QUD words (truncated to the 5 most probable elements of the support):

man (is a)	shark	sheep	lion
1.	predatory	unfree	majestic
2.	unfree	dry	ferocious
3.	unattractive	wild	tame
4.	slimy	artless	fierce
5.	sighted	dependent	loyal

Qualitatively, these results are promising: for the most part, the QUDs identified seem appropriate. Note that the QUD should not be taken as a paraphrase of the metaphorically used noun. For instance, *tame* is a suitable QUD for "The man is a lion.", because tameness is a topic that the use of the metaphor resolves, even though lions are the opposite of tame. For quantitative evaluation, we obtain human judgments on the quality of metaphors in both the animal metaphor domain, and the domain of verbal metaphor.

We compare our results to a baseline model, in which the ranking on the QUDs is generated by cosine distance to the mean of the vectors for the subject and predicate. This baseline model (which serves as a null model for the usefulness of RSA in modelling metaphor distributionally) often works well, but fails in cases where the correct QUD is not similar to either the subject or predicate. As an example, results for the null model are as follows:

The advantage that DistRSA offers over the baseline is simply the dynamics of RSA generally: the model takes into account not only the observed utterance, but possible alternatives, giving rising to an "explaining away" effect. For example, the L_1 would be unlikely to infer that "The man is a cat" conveyed stubbornness, since "The man is an ox." is an alternative which performs this task better.

Table 1: Some data

man (is a)	shark	sheep	lion
1.	wild	wild	wild
2.	dangerous	small	large
3.	large	large	small
4.	predatory	dangerous	blind
5.	small	mean	big

For our quantitative testing of the QUD identification system, we consider three domains: the animal metaphors described above, a set of verbal metaphors, and a set of adjective-noun phrases collected by (Tsvetkov et al. 2014).

In order to show that domain-specific knowledge need not be input to the model, we perform our evaluations on the *non-categorical* L_1, with a fixed set of the top 1000 most common nouns as possible utterances.

7.2 Task 2: Metaphor Detection

A second task to which our model can be applied is metaphor detection. We model the process of metaphor detection as an inference as to the existence of a QUD projection at all. For a given expression, if the QUD inferred by the L_1 to have the most weight is the trivial projection (i.e. the identity function), then we conclude that the expression is literal. Otherwise, we conclude that the expression is metaphorical. For instance, this system should identify "The lawyer is a shark." as metaphorical and "The lawyer is a judge." as literal.

For this task, we use the labeled corpus of data provided by (Tsvetkov et al. 2014). This consists of 884 metaphorical, and 884 non-metaphorical, adjective-noun phrases. For instance, an example of a literal AN phrase is "hollow tree", while an example of a metaphorical one is "hollow victory".

We use 500 of each of the two lists of AN phrases as a test set for our system, and fit hyperparameters of our categorical model on the rest. In particular, we have half the probability mass of the prior assigned to the trivial QUD, and half distributed uniformly between the projections parametrized by the top 500 most common adjectives.

The results of this experiment are detailed in figure (7.2) - the system achieves an overall accuracy of

[NB: this is currently still being tuned: results look to be about at 70% accuracy at the moment. Aim is to tune

to 80%]

Table 2: Some data

	Gold Label: T	Gold Label: F
Prediction: T	%	%
Prediction: F	%	%

8 Discussion

While our model was designed to the end of creating computational solutions for metaphor, it offers some important theoretical conclusions.

In this paper, we have considered two sorts of metaphor: copular predication and adjectival modification. These can be seen as two instances of composition: copular predication (λx. x is a B) takes a noun and forms a sentence - it is of type (e→t). Adjectival modification (λx. y x) takes a noun and returns a noun phrase - it is of type (e→e).With these two instances of composition in mind, we can consider the theoretical question of the distinction between literal and metaphorical meaning.

8.1 Distinguishing Literal and Metaphorical Meaning

Prima facie, it seems that some predications and modifications are metaphorical, while others are literal. As far as our model is concerned, metaphorical meaning is distinguished from literal meaning by requiring a QUD to be interpreted[12].

A natural question which arises, therefore, is as to the extent of metaphorical language. For instance, many AN phrases normally treated as "literal" could be understood to require a QUD for interpretation. This line of argument is pushed by philosophers such as Quine:

> "Quine pointed out that a red apple is red on the outside while a pink grapefruit is pink on the inside, and Partee took that example to be similar to the case of "flat" which applies differently in "flat tire","flat beer" and "flat note"...". - (Lahav 1989)

The point here is that even the most seemingly literal modifier, the color term *red*, in fact has different meanings depending on the context and the noun to which it is applied. In our terms, the QUD for "flat tire" is a projection which cares about *deflation*, while for "flat beer", it is a *fizziness* projection.

[12]It is important to note that the literal-metaphorical distinction is separate from the semantics-pragmatics distinction. In our model, the former concerns the use of QUDs, while the latter concerns whether the listener is L_0 or L_n for n>0.

A similar argument can be made regarding predicative metaphor of the form "A is a B". For instance, "John is a musician." might either mean that John is musically talented, or that he plays music for a career. In other words, the aspect in which John is a musician is underspecified, and needs to be inferred.

Following this line of thinking to its most radical conclusion, we could entertain the following possibility:

(15) All modification and predication requires the inference of QUDs (and by our definition, is metaphorical)

We see this as roughly comparable to the theoretical position of (Roberts 1996), who envisions that all utterances in natural language are interpreted with respect to a question under discussion.

The results from the metaphor detection experiment shed some light on this issue. AN phrases, as seen in the metaphor detection experiment, get assigned differing weights for the trivial QUD projection in the L_1 inference. One way of interpreting the weight assigned to this QUD is as the degree of literalness of the phrase or sentence in question. On this approach, metaphor and literal meaning are part of a continuum; the more metaphorical an utterance is, the more the QUDs matter for its interpretation.

The issue is further complicated by a diachronic dimension; predication which once would have required pragmatic inference of a QUD no longer does. For instance, to understand "John is a fool.", it is highly unlikely that a listener will be unaware of the conventionalized meaning of "fool" as someone stupid and have to infer it. Thus, it does not seem to be the case that metaphorical pragmatic inference is required to understand what is conveyed by "John is a fool", "Jane is a cougar." or "Time flies.".

8.2 Compositional Distributional Semantics

Logical semantics for natural language capture compositionality well, but falls short on certain aspects of word meaning, particularly in representing the similarity or difference between words. The reverse is true for distributional models of NL semantics, which offer useful similarity metrics but no straightforward means of composition. As such, an active topic of research (Socher et al. 2013, Coecke, Sadrzadeh, and Clark 2010) is *compositional distributional semantics*. The key challenge of this research is to calculate meanings (i.e. vectors) for phrases and sentences, in terms of the meanings for words.

Our model sheds some light on this problem when L_1 inference is recast as a method of noun-adjective composition. The idea is this: a noun phrase like "fiery temper" should exist in the same space as "temper", since they are of the same type[13]. We can therefore make the following claim:

(16) If the meaning of [NOUN] is the prior of L_1, then the meaning of [ADJECTIVE NOUN] is the posterior of L_1 after hearing [ADJECTIVE].

In other words, the move from L_1 prior to posterior (i.e. the process of inference) can be understood as the function by which an adjective modifies a noun.

If this hypothesis is true, it should be possible to calculate vectors for units such as AN phrases using our model, provided that the adjective in question is used metaphorically (but see section (8) for a discussion of the extent to which most or all adjectival modification is metaphorical).

9 Conclusions and Further Work

On the one hand, we have developed a model of metaphor which performs well on a range of tasks. On the other, we have extended RSA to a distributional semantics, showing that the Bayesian approach to pragmatics can scale successfully. This scaling not only absolves the need for a hand-built semantics, but also for a hand-specified set of possible utterances and QUDs, since we can supply these both in a procedural way.

Of course, there are many ways in which the language which DistRSA models is not natural. For instance, we only treat very simple cases of metaphor, and model them in a way which ignores context, as well as the effect of words other than the subject and predicate. Furthermore, the speaker in our model has a finite set of possible utterances as a prior. This runs against basic theoretical insights regarding the productivity of language.

To address these shortcomings, a natural extension to our model would be the use of a neural speaker and listener. Not only would this allow DistRSA to intake unprocessed language from an NL corpus, it would also result in a speaker who could produce potentially infinite utterances.

[13]The analogy between types and vector spaces is explored formally by Coecke, Sadrzadeh, and Clark 2010, which proposes a mode of composition for distributional semantics which shares properties of standard semantic composition.

References

Bergen, Leon and Noah D Goodman (2015). "The strategic use of noise in pragmatic reasoning". In: *Topics in cognitive science* 7.2, pp. 336–350.

Black, Max (1955). "XII-METAPHOR". In: *Proceedings of the Aristotelian Society*. Vol. 55. 1. The Oxford University Press, pp. 273–294.

Chomsky, Noam (2002). *Syntactic structures*. Walter de Gruyter.

Coecke, Bob, Mehrnoosh Sadrzadeh, and Stephen Clark (2010). "Mathematical foundations for a compositional distributional model of meaning". In: *arXiv preprint arXiv:1003.4394*.

Davidson, Donald (1978). "What metaphors mean". In: *Critical inquiry* 5.1, pp. 31–47.

Frank, Michael C and Noah D Goodman (2012). "Predicting pragmatic reasoning in language games". In: *Science* 336.6084, pp. 998–998.

Golland, Dave, Percy Liang, and Dan Klein (2010). "A game-theoretic approach to generating spatial descriptions". In: *Proceedings of the 2010 conference on empirical methods in natural language processing*. Association for Computational Linguistics, pp. 410–419.

Goodman, Noah D and Andreas Stuhlmüller (2013). "Knowledge and implicature: Modeling language understanding as social cognition". In: *Topics in cognitive science* 5.1, pp. 173–184.

Grefenstette, Edward (2013). "Category-theoretic quantitative compositional distributional models of natural language semantics". In: *arXiv preprint arXiv:1311.1539*.

Grice, H Paul (1975). "Logic and conversation". In: *1975*, pp. 41–58.

Kao, Justine T, Leon Bergen, and Noah Goodman (2014). "Formalizing the Pragmatics of Metaphor Understanding." In: *CogSci*.

Kintsch, Walter (2000). "Metaphor comprehension: A computational theory". In: *Psychonomic Bulletin & Review* 7.2, pp. 257–266.

Lahav, Ran (1989). "Against compositionality: the case of adjectives". In: *Philosophical studies* 57.3, pp. 261–279.

Lakoff, George and Mark Johnson (2008). *Metaphors we live by*. University of Chicago press.

Mikolov, Tomas et al. (2013). "Distributed representations of words and phrases and their compositionality". In: *Advances in neural information processing systems*, pp. 3111–3119.

Neal, Radford M et al. (2011). "MCMC using Hamiltonian dynamics". In: *Handbook of Markov Chain Monte Carlo* 2, pp. 113–162.

Pennington, Jeffrey, Richard Socher, and Christopher D Manning (2014). "Glove: Global Vectors for Word Representation." In: *EMNLP*. Vol. 14, pp. 1532–1543.

Roberts, Craige (1996). "Information structure in discourse: Towards an integrated formal theory of pragmatics". In: *Working Papers in Linguistics-Ohio State University Department of Linguistics*, pp. 91–136.

Shutova, Ekaterina (2016). "Design and evaluation of metaphor processing systems". In: *Computational Linguistics*.

Socher, Richard et al. (2013). "Recursive deep models for semantic compositionality over a sentiment treebank". In: *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*. Vol. 1631, p. 1642.

Tran, Dustin et al. (2016). "Edward: A library for probabilistic modeling, inference, and criticism". In: *arXiv preprint arXiv:1610.09787*.

Tsvetkov, Yulia et al. (2014). "Metaphor detection with cross-lingual model transfer". In:

Turney, Peter D and Michael L Littman (2003). "Measuring praise and criticism: Inference of semantic orientation from association". In: *ACM Transactions on Information Systems (TOIS)* 21.4, pp. 315–346.

Turney, Peter D and Patrick Pantel (2010). "From frequency to meaning: Vector space models of semantics". In: *Journal of artificial intelligence research* 37, pp. 141–188.

Way, E Cornell (1991). *Knowledge representation and metaphor*. Vol. 7. Springer Science & Business Media.