Bayesian Pragmatics with Distributional Semantics: a Computational Approach to Metaphor

Abstract

Previous Bayesian formalizations of the pragmatics of metaphor 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 interpret metaphors of the form "A is a B", by identifying the properties of B that are relevant to A. This work forms part of a more general project of scaling Bayesian models of pragmatics to real world data and distributional semantics.

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 (Shutova 2016).

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.

Within the paradigm of Bayesian pragmatics[1] (Kao, Bergen, and Goodman 2014) proposes a model which conducts joint inference over the attributes of a referent and a *question under discussion.* This Bayesian formalization offers not only a way to recognize that metaphorical expressions are not to be interpreted literally, like Grice suggests, but also a way to access the meaning the metaphor *does* convey.

This Bayesian model offers a natural formalization of the intuitions of earlier pragmaticists and has allowed for rigorous experimental investigation. A further use of RSA in is computational models of semantics and pragmatics; recent work (e.g. **andreas** Monroe and Potts 2015) has focused on integrating these Bayesian models with modern advances in computational linguistics.

Our contribution to this enterprise focuses on incorporating word vector semantics into RSA. To this end, we implement a model of RSA which has an infinite number of world states, which correspond to points in a word vector space. Furthermore, we implement a more complex variant of RSA (based on the work of Kao, Bergen, and Goodman 2014) that involves a notion of QUDs, so that the model reasons about which *aspects* of word meaning are relevant to a given metaphor.

This model, which we refer to as DistRSA, allows us to generalize the RSA model of metaphor well beyond the hand-constructed examples which previous pragmatic models require. We apply our model to two task, which we view as crucial to the computational processing of metaphorical language: metaphor understanding, and metaphor detection.

We focus on predicative metaphors of the form "A is a B" (e.g. "John is a shark."), as well as metaphorical modification, e.g "fiery temper" or "golden opportunity". We further 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 Defining 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 rock* conveys different things for each of the following:

(1) "That piece of granite is a rock."

[1] See Frank and Goodman 2012 for an introduction to the Rational Speech Acts (RSA) model more generally

(2) “That bread is a rock”

(3) “That unconscious man is a rock.”

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. For example, the bread shares the property of hardness with a rock, while the unconscious man shares the property of unresponsiveness. The task of metaphor understanding can be summed up as the identification of which properties of the predicate are relevant to a given metaphor.

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 into idiomatic phrases, 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.

While metaphors *can* conventionalize into idioms, it not would be possible for a computer to understand metaphor simply by rote memorization of phrasal meanings; the semantic productivity of metaphor prevents this. This aside, the problem remains of using context to establish when an utterance or phrase is literal and when it is metaphorical. It is this productivity and contextuality which, in our opinion, merits the use of a probabilistic generative model.

2 Distributional Semantics

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. Such word vectors can then encode both linguistic information (e.g. the similarity of verbs in the past tense) and world knowledge, e.g. the similarity of cats and dogs (see **finley2017analogies**).

Google’s Word2Vec (Mikolov et al. 2013) and Stanford’s GloVe (Pennington, Socher, and Manning 2014) are two standard models used to produce such vectors. Sets of word vectors produced by both models applied to very large amounts of text are publicly available.

Pretrained word embeddings appear to encode syntactic and semantic information (**mikolov2013linguistic**), and as such, are useful element in many NLP tasks such as sentiment analysis (Socher et al. 2013).

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).

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.”

Here, Grefenstette is suggesting that an adjective A, when modifying a noun N, targets certain properties of N and not others. He exemplifies this point as follows: “When I apply “angry” to “dog” the vector for the compound “angry dog” should contain some of the information found in the vector for “dog”. But this vector should also have higher values for the basis weights of “fighting”, “aggressive” and “mean”, and correspondingly lower values for the basis weights of “passive”, “peaceful”, “loves”.”

In the philosophical literature on metaphor, a more abstract version on this idea that only certain aspects matter is alluded to by (Black 1955), where for a metaphor

"A is a B", Black refers to "A" as the "principal subject" and "B" as the "subsidiary subject":

> "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)

Black offers the example of the metaphorical description of a war as a game of chess, commenting that "the chess vocabulary filters and transforms: it not only selects, it brings forward aspects of the battle that might not be seen at all through another medium".

The Bayesian model of metaphor proposed by (Kao, Bergen, and Goodman 2014) offers a way of capturing this dynamic of caring about certain aspects of the subject and not others: 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. 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 distributional semantics with Bayesian pragmatics by constructing an RSA model of metaphor which uses word vectors. In this setting, world states will become vectors in the space and QUDs will be orthogonal projections which map these vectors to vectors in subspaces of the space.

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 RSA

Bayesian probability provides an elegant and practical way of formalizing pragmatics (Frank and Goodman 2012). 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 $\rightarrow \{0,1\}$).Formally, where $P(w)$ is the prior on worlds, and I is an interpretation function:

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

For instance, consider a game in which a speaker is shown a picture of either a shark or a goldfish, and must try to communicate which picture they see to a listener who hasn't seen the picture. Further suppose that the speaker is only allowed to make one of the single word utterances *wet* or *vicious*.

Supposing that it is true that both sharks and goldfish are wet, but that only sharks are vicious, one might imagine that on hearing *wet*, a listener reasons as follows: if the speaker had wanted to convey that the picture was of a shark, they would have said *vicious*, since this would be the least ambiguous thing to say in this situation. Since they didn't say *vicious*, the picture is most likely of a goldfish.

We can formalize this reasoning process as follows in RSA. Here, the set of worlds W is {shark,goldfish}, and the set of utterances U is $\{wet, vicious\}^2$. We then define a semantics $\mathcal{I}$:

- $\mathcal{I}(shark, wet) = 1$
- $\mathcal{I}(shark, vicious) = 1$
- $\mathcal{I}(fish, wet) = 1$
- $\mathcal{I}(fish, vicious) = 0$

We can now model a literal listener, L_0, who on hearing an adjective uttered tries to deduce which animal, a shark or a goldfish, is being referred to. $P(w)$ is here taken to be a uniform distribution, which encodes the assumption that before hearing anything, L_0 thinks either picture is equally likely to be referred to.

(5) $L_0(w|u)) \propto \mathcal{I}(w,u) * P(w)$

S_1 chooses their utterance in order to best convey which world they are in (that is, which image they have been presented with). We define S_1's utility function in (6) and S_1 in (7):

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

(7) $S_1(u|w) \propto e^{U(u,w)}$

Note that S_1 is more likely to say *vicious* when presented with *shark* than *wet*. L_1, in turn, conditions on S_1 being in a world which would have produced the heard utterance:

[2] In this example (and in DistRSA), U is finite. This is a modeling assumption, since linguistic agents do not really have a finite set of utterances (see Chomsky 2002).

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

L_1, on hearing *wet*, will place more mass on *fish* than *shark*, which models the reasoning process described above.

RSA requires hand-constructed inputs for three aspects of the model: the semantics used in L_0, the set of possible utterances in S_1, and the set of world states in L_0 and L_1.

These three hand-constructed inputs are controlling factors on the scalability and domain-genericity of RSA, since each case of RSA requires the provision of all three. As we shall see, RSA with Questions under Discussion introduces a fourth such controlling factor: the need for a predetermined set of QUDs to be provided to the model.

4 RSA with QUDs

(Kao, Bergen, and Goodman 2014) proposes a model of metaphors of the form "A is a B" in which world states are a set of Boolean properties of A. Accordingly, a listener's prior belief is their knowledge about what A is like before hearing "B". The model must then account for how a listener is able to hear "A is a B", and only update their beliefs about A with regards to certain properties of B. For instance, on hearing that the lawyer is a shark, I may conclude that the lawyer is bloodthirsty, but not that she has gray skin.

To account for this, (Kao, Bergen, and Goodman 2014) introduces an additional feature to RSA. This is the notion of a Question under Discussion, inspired by the 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→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: ⟨c(lever)=True,v(icious)=True,a(quatic)=False⟩.

Denotations of utterances, like *shark* in the above example, are distributions over world states. As before, we define a function $\mathcal{I}$, which here takes a word x and a world y, and returns the probability density of y in the denotation of x. This allows us to define an L_0 model identical to that of standard RSA (see (5)):

(9) $L_0(w|u) \propto \mathcal{I}(u, w) * P(w)$

Here, $P(w)$ is the listener's prior over worlds, which encodes the information known about the subject of the metaphor (e.g. the lawyer) before hearing "shark". For instance, let us suppose that worlds in which *aquatic* is true of the laywer have little probability mass in this prior distribution. This represents our prior knowledge that lawyers cannot breathe underwater. This L_0 is much like the standard RSA L_0 defined in (5).

The first key difference to simple RSA comes at the S_1, which takes both a world and QUD. In this setting, a QUD is a function which maps any world triple to some element of that triple[3]. For example, the QUD which maps worlds to their first argument would be the *cleverness* QUD, the QUD that only cares about whether the lawyer is clever.

The S_1, 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*. Formally:

(10) $\mathrm{U(u,w,q)} = \log(\sum_{w'} \delta_{q(w)=q(w')} * L_0(w'|u))$

(11) $S_1(u|w, q) \propto e^{U(u,w,q)}$

The L_1 then *jointly infers* both a world state and a QUD[4]:

(12) $L_1(w, q|u) \propto S_1(u|q, w) * P(w) * P(q)$

We will, unimaginatively, refer to RSA with the joint QUD inference described here as the QUD RSA model. An example of QUD RSA in action is as follows. Let us suppose that the metaphor in question is "The lawyer is a shark.". We first supply a prior on possible worlds (for L_0 and L_1), on QUDs for L_1 and on utterances for S_1:

- World Prior[5]:
  - ⟨c=True←0.7,v=True←0.6,a=True←0.001⟩
- Utterance Prior:
  - *shark* : 0.5
  - *fish* : 0.5
- QUD Prior:
  - *clever* $(\lambda\langle x, y, z\rangle \to x)$: 1/3
  - *vicious* $(\lambda\langle x, y, z\rangle \to y)$: 1/3
  - *breathes-underwater* $(\lambda\langle x, y, z\rangle \to z)$: 1/3

[3] Or equivalently, to the set of worlds which agree on that element.

[4] 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

[5] The notation here of ⟨c(lever)=True← α,v(icious)=True← β,a(quatic)=True← γ⟩ denotes a distribution where the Boolean value for *clever* is sampled a Bernoulli distribution weighted at α to True, and likewise, *mutatis mutandis* for *vicious* and *aquatic*.

- The map from utterances to probability distributions used in the interpretation function:
  - *shark* : ⟨c=True←0.7,v=True←0.9,a=True←0.9⟩
  - *fish* : ⟨c=True←0.5,v=True←0.2,a=True←0.9⟩

We now show parts of the calculation for the L_0, L_1 and S_1:

Let $w_1 = \langle$ c=False,v=True,a=False $\rangle$

Let $w_2 = \langle$ c=True,v=False,a=False $\rangle$

$L_0(w_1|\mathit{shark}) = I(f, w_1) * P(f|\mathit{shark}) * P(w_1) = 0.143$

$S_1(\mathit{shark}|w_1, \mathit{vicious})$

$= \frac{\sum_{w'} q(w_1)=q(w')*L_0(w'|\mathit{shark})*P(u)}{\sum_u \sum_{w'} q(w_1)=q(w')*L_0(w'|\mathit{shark})*P(u)}$

$= 0.773$

$L_1(w, \mathit{vicious}|\mathit{shark})$

$= \frac{S_1(\mathit{shark}|w_1,\mathit{vicious})*P(w_1)*P(\mathit{vicious})}{\sum_{w'} S_1(\mathit{shark}|w',\mathit{vicious})*P(w')*P(\mathit{vicious})}$

$= 0.094$

By contrast, $L_0(w_2|\mathit{shark}) = 0.016$. In other words, the model predicts, given the utterance *shark*, that the lawyer is more likely vicious and not clever than the reverse. This conclusion is reached by the fact that *fish* would have been a more informative utterance if in w$_2$ is the speaker's world rather than w$_1$.

We can also marginalize out the world variable by summing over it, to obtain a posterior on QUDs. This tells us which QUD is most likely, given that the listener heard *shark*, when their prior reflected the properties of lawyers. The model correctly predicts that *vicious* is the best QUD, with 0.336 of the probability mass.

One of the properties of RSA generally that QUD RSA inherits is *explaining away.* Suppose, for instance, that we add a third possible utterance to our model, *fox.* Supposing that foxes are clever, the probability of the QUD being *clever* when the utterance heard by the L_1 is *shark* will decrease, in the presence of a more informative utterance for this QUD. Informally, this accords to the reasoning process: "my interlocutor could have said *fox*, but she didn't, so it is less likely they were conveying the lawyer's cleverness.".

A second property of QUD RSA is that the model is asymmetric, in the sense that "A is a B" results in a different meaning to "B is an A". This is because A and B in "A is a B" correspond to entirely different things in the model. *A* informs the listener prior, while *B* is the observation on the basis of which the prior is updated to the posterior.

Linguistically, this asymmetry seems to be a general feature of predicative statements, and *a fortiriori*, of predicative metaphors; "The politician is a butcher." does not mean the same as "The butcher is a politician.".

5 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,aquatic=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 both distributional and non-distributional QUD RSA, the world state and the word denotation are 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 a metaphor, such as "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 discussed in section (2), the GloVe word encodes semantic similarity roughly as cosine distance in the embedding space.

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

The multidimensional Gaussian distribution weights most heavily those points nearest to its mean, e.g. $\overrightarrow{\text{time}}$. 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 weight, represents in this model an update regarding one's knowledge about time.

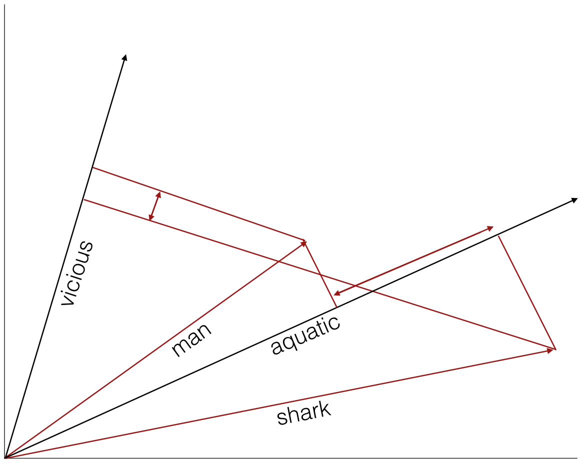
How does this update take place? In non-distributional RSA, the L_0 conditions on the truth of the utterance they hear. However, the notion of truth, which for ordinary RSA amounts to a function from worlds to {0,1} (or [0,1]) is no longer straightforward to define.

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 (**??**)).

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 full space to a lower dimensional subspace. The simplest case is of a 1 dimensional projection, 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 of using a projection to a hyperplane for the QUD rather than simply a projection to a line 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

[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 in typical word embedding spaces, the dimensions of which are much fewer than the size of the vocabulary.

[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).

we can think of our QUDs as words, such as adjectives, which measure aspects of the subject being predicated. We can generalize to projections onto n-dimensional hyperplanes by using n-tuples of words, instead of single words.

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).

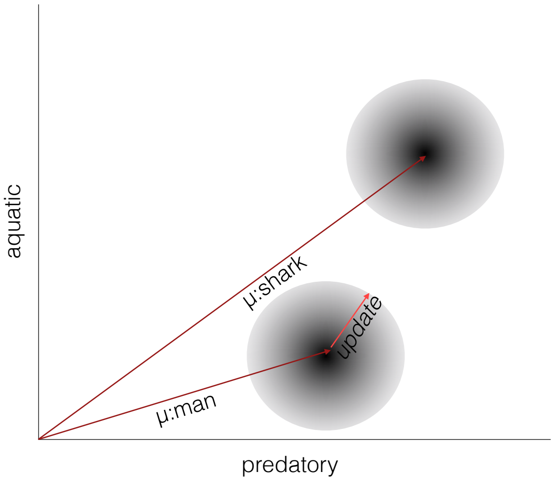
To illustrate how this works informally (but for a formal account, see section (**??**)), we walk through a 2D example of our model, applied to the metaphor “The lawyer is a shark.”. As with RSA generally, our model requires a set of possible utterances. In the present case, let us supply the possible 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 (**??**) 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 and 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.



Another way of thinking of DistRSA is as a spatial reference game (e.g. **golland2010game**), played in a word vector 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, due to the need to calculate a normalizing term which sums (or rather integrates) over an infinite set.

We therefore compute the L_0 and S_1 posteriors analytically, and use perform appromixate inference only at the L_1 level. We now describe how each of L_0, S_1 and L_1 is computed.

6.1 Model Definition

6.1.1 $\mathbf{L_0}$ 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”):

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

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

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, (**??**) can be reduced analytically to a single Gaussian.

Note that truth is not a fundamental concept in distributional RSA; we have no way of saying whether an utterance is true or false. 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. While this deviates from traditional truth-conditional semantics, it preserves the intuition that meanings should be things which allow linguistic agents to update their world knowledge.

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

6.1.2 $\mathbf{S_1}$ Definition

The prior for S_1, which is over possible utterances, is finite, and therefore is straightforward to compute[10]. 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:

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

(15) $\text{S}_1(u|w,q) \propto e^{U(u,w,q)}$

6.1.3 $\mathbf{L_1}$ Definition

L_1 is defined as follows, where P_{QUD} represents the prior on QUDs. As discussed below, we can either have this be a continuous distribution over hyperplanes corresponding to projection functions, or a discrete distribution of hyperplanes corresponding to a set of n-tuples of words. In either case, we assume it is a uniform distribution:

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

While we were able to use analytic methods to derive the L_0 and S_1 posteriors, we cannot do so for the L_1. Instead, we must use an approximate inference method. We try two different inference algorithms, Hamiltonian Monte Carlo (**neal2011mcmc**), a brand of Markov Chain Monte Carlo inference algorithm which makes use of gradient information to move from one sample to the next, and Variational Mean Field inference (see **blei2017variational**), an optimization based inference algorithm. In our first experiment we use the former, while we make use of the latter in our second, on account of its speed.

Variational Inference (VI) greatly increases efficiency over other inference methods, at the cost of requiring gradients for the S_1 and the functions the S_1 calls. Fortunately, it is possible for us to calculate these gradients, using automatic differentiation.

L_1 performs 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 (QUDs), for example those corresponding to n-tuples of words, according to the pretrained embedding. The output of our inference then gives us a categorical distribution over QUDs, corresponding to words (as well as a continuous distribution over worlds).

The second variant 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. However, the Non-Categorical model yields significantly less stable results, and as such, we use only the Categorical model in our experimentation.

[10] As discussed in section (4), having a finite set of utterances is theoretically objectionable, but for the time being, a necessary modeling assumption.

6.2 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, by using a semantics build on word vectors.

However, we attempt to go further, by also removing the need for hand chosen utterances and QUDs. This can be partly accomplished by the use of large sets of possible utterances and QUDs. We further use a language model[11] in order to choose possible utterances in a systematic fashion.

We do this by first taking a large set of nouns from WordNet (**miller1995wordnet**) and ranking them according to their probabilities of being completions of "[subject] is a" under the language model. We then take the first n nouns (with n varying depending on the experiment in question). So, for instance, to model "The lawyer is a shark.", we let the possible utterances be the best n completions of "The lawyer is a". The correct n varies depending on the precise task at hand - we discuss individual cases separately below.

As for the QUDs in the Categorical model, we take a set of adjectives from WordNet, and rank them according to their cosine distance from the mean of the subject (e.g. *lawyer*) and predicate (e.g. *shark*). This provides a rudimentary, but useful ranking of which words are the most relevant topics for the metaphor in question. We can then generate projections from these adjectives. If we want QUD projections into multidimensional hyperplanes, we can take the cross product of our set of adjectives with itself to generate n-tuples, and then obtain the QUD projections from these.

We create categorical distributions over possible utterances and QUDs by weighting the elements in these sets according to their frequency[12]. Our model is computationally efficient enough that quite large inferences are possible - we can supply several thousand possible utterances and QUDs, with a run time of around 30 minutes per metaphor.

We implemented our model in both WebPPL and Python[13]. For the later, we make use of Tensorflow, and Edward in particular (see **edward**), in order to perform HMC or VI. 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 inference.

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/`. We mean center the vectors, and use PCA to remove the top 4 orthonomal bases[14]. This second preprocessing step is inspired by (**mu2016geometry**), which suggests this for improving the quality of the Glove word embeddings in particular.

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 based on 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. Before considering quantitative evaluation, we show an example of the model, performing L_1 inference on a limited domain, the animal-based set of 34 possible utterances and 91 1D QUDs[15], generated by the adjectives supplied by (Kao, Bergen, and Goodman 2014, page 272).

Table (**??**) displays the L_1's top five QUDs (the five QUDs in the support of the L_1 distribution over QUDs with the most probability mass) for the subject "man" and the predicates "shark", "sheep" and "lion" respectively. Table (**??**) shows the same results, but from a vectorial 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[16]. 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.

We suggest that the advantage that DistRSA offers over the baseline is simply the dynamics of RSA generally:

[11] Introduced in (**jozefowicz2016exploring**) and available at `https://github.com/tensorflow/models/tree/master/lm_1b`

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

[13] The code of our Python implementation will be made publicly available.

[14] Rather than the bottom dimensions, as is usual for PCA. We keep the vectors in the full 50 dimensional space, rather than expressing them by their basis in the subspace.

[15] The hyperparameters are as follows: HMC step size: 0.25. Number of HMC samples: 1000. σ_1: 0.1. σ_2: 0.1. Speaker and listener priors: uniform.

[16] We experiment with other binary operators to combine the subject and predicate vectors (sum, weighted sum, product, difference) but find that mean works best overall.

Table 1: L1 model

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

Table 2: Baseline model

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

the model takes into account not only the observed utterance, but possible alternatives, giving rising to an "explaining away" effect discussed in section (4). 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.

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.

7.1.1 Mechanical Turk Experiment

Participants For quantitative evaluation, we obtained human judgments from 40 Mechanical Turk participants.

Materials We collected 15 metaphors from COCA (**davies2008corpus**), by searching for "[Singular Noun] is like a [Singular Noun]", and using hand-chosen pairs of nouns obtained from this search as our metaphor set[17].

Procedure We used Mechanical Turk to obtain crowd-sourced judgments regarding each metaphor. Each participant was shown all 15 of the metaphors (order randomized). For each metaphor, they were shown 6 adjectives, 3 from the L_1 and three from the baseline described above, which simply finds the words nearest to the mean of the subject and predicate (order also randomized). They were instructed to rate the relevance of each adjective to the metaphor, where relevance is defined as follows: "an adjective is relevant if it describes a property of the subject (e.g. "the man" in "The man is a shark") that the speaker of the sentence is trying to talk about". The format of the experiment is displayed in Figure (**??**). The baseline model was supplied with the same 50 QUDs to rank that constitute the L_1 QUD distribution support.

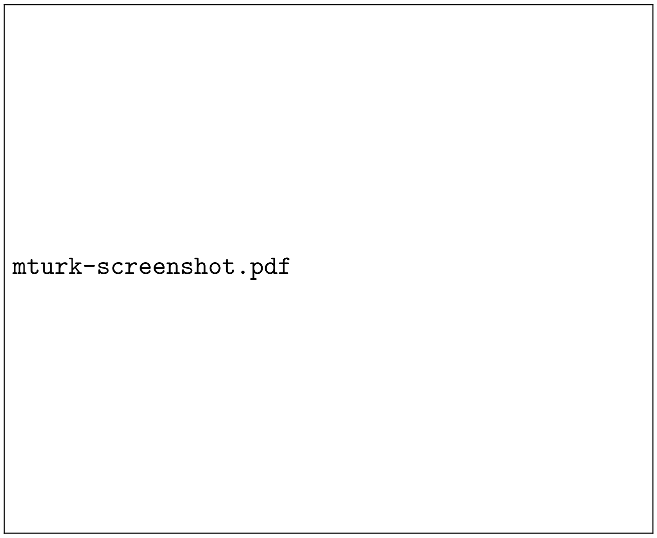


Figure 1: A slide from our Mechanical Turk experiment.

For the L_1, we took the first 50 possible utterances generated by our pretrained language model (**jozefowicz2016exploring**), and for QUDs, take the pairs of words[18] $\langle w_1, w_2 \rangle \in W \times W$, where W consists of the first fifty QUDs generated as described in section (**??**).

The hyperparameters of the model are as follows: VI step size = 0.0005. $\text{Sigma}_1 = 0.0005$. $\text{Sigma}_2 = 0.005$. Dimensions onto which QUD is projected: 2. Number of QUD tuples = 1225. Speaker prior: uniform. Listener prior: uniform. GloVe vectors: Twitter, 25 dimensional.

We obtain these hyperparameters by qualitative examination of the behavior of the model on a small validation set of metaphors. We find that the choice of GloVe vectors has little impact on the quality of results, but that the values of the sigma_1 and sigma_2 are important. Moreover, results seem to improve qualitatively when the QUDs project into a subspace of more than 1 dimension.

[17] Collecting data from COCA has precedent in other computational work on metaphor, e.g. (**neuman2013metaphor**) and (**turney2011literal**)

[18] In fact, we exclude all pairs where $w_1 = w_2$.

Figure 2: The mean user rating for our model vs. the baseline model, per metaphor. Error bars indicate bootstrapped 95% confidence intervals.

8 Results

8.1 Results for Task 1

We ran a linear mixed effects model, predicting slider rating from baseline vs. L_1 metaphor, with by-metaphor and by-participant random intercepts, to analyse the results of the experiment. We find a significant correlation (β=0.32,SE=0.01027, t=9.789, χ^2=94.57, p<0.0001) between an adjective being from our model and it being rated higher on the scale of relevance[19].

As can be seen in Figure (**??**), there is a lot of variation in the average quality of proposed adjectives across both models, between metaphors. For the baseline model, we note that there is a subset of metaphors for which the baseline will perform well, as discussed in section (**??**). These are metaphors where good QUDs q are such that the subject and predicate are close when projected along q. In these cases, averaging the subject and predicate will generally give rise to a good QUD.

However, further work is necessary to understand the conditions under which our model fails, and what could be done to improve it. An example of a failure case is the ranking of "adoptive" as the best QUD for the metaphor "The father is a shark." - in this case, it is not clear why the model chose this particular adjective. A further challenge, therefore, is to develop methods to interpret the choices made by the model, so that the model's posterior distributions can be understood in terms of the underlying geometry of the word embedding space.

8.2 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 (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 (**tsvetkov2014metaphor**). This consists of 884 metaphorical, and 884 non-metaphorical, adjective-noun phrases, collected by two annotators from text corpora. For instance, an example of a literal AN phrase is "hollow tree", while an example of a metaphorical one is "hollow victory".

We use 84 of each of the two lists of AN phrases as a training set for our system, for fitting hyperparameters of our categorical model. We test on the remaining 711 of each, having removed 89 metaphors in each to exclude words which did not appear in GloVe[20].

8.3 Results for Task 2

The results of this experiment are as follows. For each item in the test set, we see whether the trivial QUD has the most weight out of the set of QUDs. If it does, we say that the system has classified the AN phrase as being literal. The expectation is that the system should more commonly classify the phrases in the metaphorical AN set as metaphors and those in the literal AN set as literal.

This is indeed the case, with 356 out of 711 metaphorical AN phrases classified as literal, compared to 439 out of 711 literal AN phrases classified as literal. This yields p<0.0001 (with the null hypothesis that this result is drawn from a binomial distribution with equal likelihood of either outcome.).

This experiment suggests that the density of the trivial QUD is indeed an indicator of the literalness of a metaphor. A preliminary theoretical conclusion we might draw is as follows: while metaphorical and literal expressions lie on a continuum, the usefulness of QUDs is higher for metaphorical expressions.

While it is possible to fit a classifier to the data, by classifying those AN phrases as metaphors which yield a trivial QUD density below some fit threshold (e.g. 0.05), this results in accuracy of around 0.55, which is considerably less than comparable baselines (**tsvetkov2014metaphor**), which yields an accuracy of 0.86. As such, without further tuning, the system proposed here is not yet suitable for state of the art metaphor detection.

[19] Our p-value and χ^2 are obtained by a log-likelihood ratio test.

[20] The hyperparameters are as follows: HMC step size = 0.0005. Number of HMC samples = 400. Burn in: 285 removed. Sigma_1 = 0.0005. Sigma_2 = 0.01. Number of possible utterances = 30. Dimensions onto which QUD is projected (if not trivial QUD): 1. Number of QUD 1-tuples (including trivial QUD) = 31. Speaker prior: uniform. Listener prior: 0.9 on trivial QUD, 0.1 uniformly distributed between other 30 QUDs. GloVe vectors: twitter, 25 dimensional, with mean centering of vectors and removal of PCA top dimensions.

9 Discussion

9.1 The Extent of Metaphor

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.

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 proposed 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**)

The point here is that even the most seemingly literal modifier, the color term *red*, in fact has different meanings depending on 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. Similarly for predicative metaphor, "The grapefruit is pink.", like the AN phrase "the pink grapefruit" requires us to discern which aspect of the grapefruit is pink. Generalizing this line of thinking, we could entertain the following hypothesis:

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

The claim in (**??**) is that all modification and predication in natural language must be interpreted at the QUD RSA L_1, not at the L_0. It is important to make clear that this claim does not entail that there is no such thing as semantics, separate from pragmatics; in fact, our RSA model makes a very clear distinction between the two.

For DistRSA, the semantic meaning of each lexical item is a vector. The mapping from lexical items to vectors, for which we use GloVe, parametrizes the model's semantics. GloVe itself encodes pragmatic information; for instance, the co-occurrence of *shark* and *man* arising from instances of the metaphor "The man is a shark." in GloVe's training data, influences the position of the vectors for *shark* and *man*. However, this mapping from words to vectors need not, in principle, be influenced by pragmatics. For example, (Monroe and Potts 2015) is an RSA system which learns a semantics while making pragmatic observations, allowing the semantics to be separate. We hypothesize that a similar technique could in used in the context of DistRSA, allowing for a mapping from words to vectors which only captures information that the RSA would not account for itself.

Intersectivity and Metaphor Closely related to the issue of (**??**) is the status of *non-intersective* adjectives. Intersective adjectives are those which obey the following property: for an intersective adjective A, if A$_x$ is the set of things which are A, and for a noun N, if N$_x$ is the set of things in the extension of that noun, then the intersection of these two sets is the extension of [A N]. For example, if one believes that *red* is an intersective adjective, then the set of red apples would be the intersection of the set of red things and the set of apples.

As seen by Quine's position on "red" and "pink", there are grounds for skepticism that color terms are really intersective. Along the lines of (**??**), we might instead arrive at the position that all adjectives are non-intersective, being instances of metaphorical modification.

The results from the metaphor detection experiment shed some light on these issues. 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, metaphorical and literal meaning are part of a continuum; the more metaphorical an utterance is, the more the QUDs matter for its interpretation. From this perspective, (**??**) would amount to the claim that no AN phrase is fully literal.

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.".

9.2 Compositional Distributional Semantics

Logical semantics for natural language captures compositionality well, but falls short on certain aspects of word meaning, particularly in representing the similarity or difference between words. Distributional models of NL semantics address this by offering useful similarity metrics. As such, an active topic of research (Socher et al. 2013, **coecke2010mathematical**) is *compositional distributional semantics*. The key challenge of this research is to calculate meanings (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[21]. We can therefore make the following claim:

(18) 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 (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 compositional units such as AN phrases using our model, provided that the adjective in question is used metaphorically (but see section (**??**) for a discussion of the extent to which most or all adjectival modification is metaphorical). This is beyond the scope of the present work.

These proposals, (**??**) and (**??**), clearly need more empirical research before even initial conclusions can be reached. We simply raise the discussion as a reminder that distributional models such as the one proposed here may apply to predication and modification generally, rather than the rather narrow cases for which it first seems appropriate.

10 Conclusions

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 to multidimensional continuous distributions. 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 (a direction pursued for simpler RSA models by **andreas** and Monroe and Potts 2015). 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.

[21] The analogy between types and vector spaces is explored formally by **coecke2010mathematical** 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.

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.

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.

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.

Monroe, Will and Christopher Potts (2015). "Learning in the rational speech acts model". In: *arXiv preprint arXiv:1510.06807*.

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.

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.