Natural Language Semantics Term Project

Michael Tao

1 Introduction

Representing natural language semantics through vector based techniques has become a common practice, but most of these methods come with heavy limitations. Few of these vector based methods contain enough structure to naturally allow operations such as the composition of phrases to be passed into the representation in any consistent way. Such issues can be seen in the example of a "red ball". Even if we have vectors representing "red", "ball", and "red ball" in general there are few representations that can naturally combine the vectors for"red" and "ball" to reproduce the vector for "red ball". Additionally these models cannot extrapolate from the representation of a "red ball" that we are dealing with a ball or that this ball is red. Another issue that frequently occurs is data scarsity, the longer a phrase is the less frequently we expect to see it. The application of such vector space representations is therefore limited to shorter terms, as data sparsity quickly comes into play when we try to represent the meaning of longer tokens such as phrases or sentences. Thus, it would be convenient to have a method that naturally allows for composing our vector representations of phrases. Such methods would allow for us to represent the composition of the phrases themselves, rather than to try to create a new vector for the composition of phrases directly from corpora and have to deal with sparsity.

One particularly promising method is context-theoretic semantics, which use the philosophy of *meaning as context*[1]. This philosophy states that the meaning of a phrase is precisely the way in which it is used. Within this framework there already exists approaches for data-driven taxonomy generation[3] , but these techniques do not take advantage of prexisting taxonomies. This paper discusses how one may approach taking advantage of pre-existing ontologies to create and apply context theories that represent taxonomies through the generation of a context theory for synsets, the elements found in WordNet. Through these synsets we are able to not only estimate vectors for terms that we don't see in the corpora used to define but also disambiguate the various senses of terms that, though spelled similarly, have differences in their definitions, like homonyms.

2 Context Theories

Context theories provide the structure necessary to develop theories of meaning from a mathemaical perspective. Their primary objective is to take structures discerned from natural language and represent them within the structure of abstract Hilbert spaces. By utilizing intuitive connections between the structure of Hilbert spaces such as addition, multiplication, and measures with certain structures found in natural language, they can be used to represent structure such as such the probability distributions of terms or semigroup structures in syntax[2]. From this connection between structures we are able to extract deeper meaning from our vector representations, such as the ability to derive meaning of previously unseen phrases with the composition of smaller phrases we already have vectors for.

Formally we define our context theory for a set of words A as a tuple $\langle A, S, \hat{}, \cdot \rangle$. The $\hat{} : A^* \mapsto L^1(S)$ operator gives us a means to connect between the structure that we're trying to represent from natural language and the mathematical structure that we're using to represent it. $L^1(S)$ is a lattice-ordered algebra in which we set the multiplication operator be $\cdot$. This means that $L^1(S)$ is the family of real valued functions from the set S such that they have finite L^1 norms:

$$L^1(S) = \{u : S \to \mathbb{R} | \|u\|_{L^1} < \infty\}$$

where we have that

$$\|u\|_{L^1} = \sum_{s \in S} |u(s)|.$$

In our case we view this as a probabilistic space by using an abstract Lebesgue space: a vector space with the L^1 norm and as well as operators $\vee$ and $\wedge$ that correspond to the union and intersection of objects respectively. This gives us that for $u, v, w \in L^1(S)^+$ and $\alpha \in \mathbb{R}$ the operators behave as follows:

$$\begin{aligned} (\alpha u)(s) &= \alpha u(s) \\ (u+v)(s) &= u(s)+v(s) \\ (u \vee v)(s) &= min(u(s), v(s)) \\ (u \wedge v)(s) &= max(u(s), v(s)) \end{aligned}$$

and

$$\begin{aligned} u \leq v &\Rightarrow \alpha u \leq \alpha v \\ u \leq v &\Rightarrow u + w \leq v + w \end{aligned}$$

The choice of multiplication operator is chosen such that for $x, y \in A^*$ we have $\hat{x} \cdot \hat{y} = \widehat{xy}$. This is so that we can maintain the structure of phrase composition through the mapping into our mathematical representation. To give this representation the ability to combat the data scarcity found in longer phrases we can simply take representations of shorter words and apply the $\cdot$ operator to represent the longer phrases.

2.1 Taxonomies

Ontologies like WordNet are governed by **is-a** relationships, which we will assume is a partial ordering. That is, A is a B implies that $A \leq B$. This assumption is not a very steep one as WordNet does not contain cycles and the **is-a** relationship naturally has an ordering: if A is a B and B is a A then they are intuitively the same thing. Our objective is create a vector representation of our terms through some ˆ operator such that the partial order prescribed by preexisting ontologies. However to begin the discussion there is some notation that must first be discussed to describe the underlying structure that must be held in a context theory for taxonomies.

An *ideal* I of a partially ordered set S is a subset $I \subset S$ such that $\forall x \in I, y \leq x \Rightarrow y \in I$. A *principal ideal* generated by x is the set $\downarrow(x) = \{y \in S : y \leq x\}$. Ideals give us a tool for looking at terms that are subsumed by the *is-a* relationship, for we see that $\downarrow(x)$ signifies all of the words that are instances of x. For example, if we are given any term (like "animal") we see that

$$\downarrow(\text{“animal”}) = \{\text{“dog”}, \text{“cat”}...\}$$

and we intuitively know that dogs and cats are instances of animals.

The primary issue is how to develop a context theory representing the probability of words occurring in different contexts. To do this we define the probability of our terms with a *Real Valued Taxonomy* - a set of concepts S with a partial order $\leq$ and a positive real function p. We set $\sum_{s \in S} p(s) = 1$ to make this probabilistic. If we let p signify the probability of a term x occurring in a particular context, the probability of the concept behind x, $\bar{x}$, will be the sum of the probability of any concept $\bar{y}$ such that $\bar{y} \leq \bar{x}$. Thus we define the operator $\hat{p}$ that represents the probability of a concept $\bar{x}$ given the probability of all of the terms that appear under it in the partial ordering:

$$\hat{p}(\bar{x}) = \sum_{\bar{y} \in \downarrow(\bar{x})} p(\bar{y}).$$

Now if we take our real valued taxonomy we then have a *Ideal Vector Completion* for our term x, we can define the elements of our vector space as $\psi(x)$, which takes the form

$$\psi(x) = \sum_{\bar{y} \in \downarrow(\bar{x})} p(\bar{y}) e_{\bar{y}}$$

where we let $\{e_y : y \in S\}$ be a set of independent basis vectors. This gives us a vector for the probability of a concept, the sum of the probabilities of the vectors for any term below it. It's immediately true that $\|\psi(x)\|_1 = \hat{p}(\bar{x})$, so we see that the measure of a concept is identical to the probability of it occurring.

2.2 Example Context Theory

For our purposes, we will assume that the context theory being used is similar to the context theory developed by Clarke [3]. Thus we provide a brief overview of Clarke's Quotient Algebra context theory to sketch how one might approach actually developing a context theory. For a more detailed description please refer [3].

The context theory in question represents each possible context as a dimension. This is the type of representation we will assume for the rest of this article. Essentially, we let S be the set of contexts in this context theory. In this particular case we first consider the vector space $\mathbb{R}^A$ and utilize tensor products as the multiplication operation for composition to obtain representation for higher dimensions. From this we get dimensions representing the form

$$w_1 \otimes ... \otimes w_{h-1} \otimes e_i \otimes w_{h+1} \otimes ... \otimes w_r.$$

The w_i are the words that exist at their respective slots in that context and e_i corresponds with an empty slot. For a given term the coefficient for the above dimension is the probability of that term appearing in a context at the empty slot. The dimensions from above are then stitched together using the direct sum making this something called a *tensor algebra.*

From the above construction we easily see that we have a representation that allows for dimensions representing the probability of seeing "x", "red x", "red x block" etc. However, the dimensionality of the representation increases in size exponentially and ends up covering redundant terms like "square flat x" and "flat square x" which have the same definition. Another flaw can be seen in"x" and "the x" where "the" adds almost no semantic information to the phrase. Therefore the dimensionality and redundancy are both reduced by declaring certain relationships between dimensions to be equivalent to each other through ideals. Mathematically the ideal mentioned here is the same as the one already mentioned but the way that it presents itself here is significantly different. In this ideal we consider the set of things that don't add to a representation and set their difference to be zero. After picking out some situations where we know we have equivalences we can write them as $\widehat{\text{“}x\text{”}} - \widehat{\text{“the }x\text{”}} = 0$ and compress the representation of both of these to a single dimension.

3 Ontology-driven Taxonomy Generation

3.1 Ontologies

Our primary task is to deal with how one might formulate a context theory in which the $\leq$ operator preserves the **is-a** partial ordering defined by an ontology. However, the ontologies we are looking at don't use words as their elements, but rather use the senses that words can take. These senses are implicit to individual contexts and not immediately discernible from instances of words by themselves. In the case of WordNet these are called synonym sets (synsets). Synsets discern various equivalence classes of words by words that are interchangeable for one another in contexts. They form a "many to many" correspondence between the space of words and meanings: synonyms all map to one synset but one word can have multiple distinct definitions. For our purposes we are interested in discerning the probabilities of the various contexts that individual synsets exist in. Therefore, ideally one would like to be able to discern the various senses of each word in our taxonomy to directly extract frequencies from a corpus. However that would require the ability to detect the sense of a term from its context, which is an open problem. In order to avoid this issue we will have to take a more indirect approach.

3.2 Synsets

From within an ontology such as WordNet we see that a word is precisely its synsets. The synsets define the various meanings that the word can take. Our philosophy therefore states that the synsets are the various contexts for which a word may exist in.

This tells us that for any particular word each of the synsets associated with it should exist in a set of contexts distinct from any other synset associated with that word. If two synsets were to have similar or even identical distributions of contexts, they would both represent the same concept and therefore be redundant. For any particular word we see that the contexts that the word occurs in corresponds to the totality of the contexts for which its synsets occur in. That is, the probability of a word occurring in a particular context is the sum of the probabilities of any of its synsets occurring.

Let us assume that we have a mapping $^{-}$ from the terms in a corpus and the synsets in the ontology. The first step in trying to define a context theory for our synsets is to try to associate the **is-a** partial ordering $\prec$ for synsets with the partial ordering of our existing context theory. However for any pair of words a, b they may contain multiple synsets, say $\bar{a}_1, \bar{a}_2$ for a and b_1 for b. If we happen to have partial ordering of synsest $\bar{a}_1 \leq \bar{b}_1$ we cannot guarantee that $a \leq b$ because the probability of a existing in any particular context is the sum of the probability of any of its senses occurring in that context. Also, if a is a b we cannot determine anything about the relationship of individual senses of words as the combination of two senses for b can subsume one sense of a. All we can really discuss is the connection between terms and the combination of their synsets. The word "combination" is a bit vague, but in our discussion we are dealing with probabilities of words and concepts occurring in different contexts. We can let the result of "combining" two objects be the probability of either object occuring - which is the summation of their probabilities. Therefore we know that

$$a \leq b \Leftrightarrow \sum_i \bar{a}_i \prec \sum_i \bar{b}_i.$$

We don't have any direct information on the contexts for which synsets exist in so we cannot straightforwardly compute an ideal vector completion for synsets. The ordering we are given gives us a partial ordering on the contexts themselves. Since we our goal is to discern the probabilities of synsets existing in certain contexts through the terms that represent them, what we'd look at the probabilities of various synsets is through a corpus. However, this would force us to look at terms and discern their senses which would require us to disambiguate word senses, which is an open and difficult problem. If we could discern some word senses we would immediately have the context vectors for the synsets that we could disambiguate in terms, but beyond that there is still a need to estimate the probability of our synsets in order to create a valid context theory for the synsets.

3.3 Deriving the Synset Context Theory

We'd like to define context vectors for the various synsets, but there is no direct way to derive them from the term probabilities as discussed previously. We do, however, have a large number of constraints on the probabilities of the contexts in which synsets occur in. These constraints are governed by by the relationships between words and the synsets that belong to them and constraints given to us by the ontology. These constraints present themselves as inequalities of linear equations and since all of the variables we are dealing with are probabilistic every variable must be non-negative. This gives us, with the exception of an objective function, an instance of linear programming. The choice for an objective function is not obvious, but our constraints make it so that any choice will produce a valid estimate given our knowledge of the synsets without attempting to actively disambiguate word senses.

3.3.1 Synset to Term Constraints

First let us consider the constraints that we must enforce between the synsets and terms. For each term let us recall that the probability of it occurring in any context is the sum of the probability of any of its senses occurring. Thus we explicitly have the constraint $\bar{t}_i = \sum_{j \in S_i} \bar{s}_j$ where t_i and s_j are context vectors for terms and synsets respectively and S_i is the set of synsets associated with term t_i. We have a constraint of this form for each of the terms in our original ontology so we must compose these vectors T, S and a matrix A from which we can derive linear constraints:

First off let us define $\dim(t_i) = N$ to be the dimensionality of our context vectors for terms. Each t_i is from the same space so $\dim(t_i) = N$ for any i. Since we are trying to develop the vectors s_j from the vectors t_i we see that we can at most let $\dim(s_j) = N$. Also, let us denote the total number of terms that we have by cT and the total number of synsets that we have by cS.

$$T = [t_i]$$

is vector representing the concatentation of all of the column vectors we have for representing terms. This gives us a vector of size $(N * cT) \times 1$.

Let us create a vector of the same nature as T except for the the synset vectors through the concatenation of synset column vectors. This gives us a vector of dimensionality $N * cS \times 1$

$$S = [s_j]$$

Finally, in order to represent the connection between the terms and synsets, in particular the additive property between the synsets of a term and the term itself we define a matrix A such that we can relate the terms and synsets through the equation

$$T = AS.$$

We can obtain the above equation by setting A to be defined by the block matrix

$$A = [\delta_{S_i,j}],$$

where $\delta_{S_i,j}$ is a function that takes the value I_N, the $N \times N$ identity matrix, if j is in S_i and 0_N, the $N \times N$ zero matrix, otherwise. This is used to represent that that s_j is a synset for t_i. The meaning behind this representation is easily realized when we view T, A, S all as block matrices where the entries in T and S are the original vectors. Each block-row i in A becomes an indicator for the synsets associated with the ith vector in T. This gives us a matrix of dimension $N * cS \times N * cS$.

Within the formalism of linear programming this constraint must be represented as inequalities, which we can easily do by rewriting this as, $T \geq AS$ and $-T \geq -AS$ to get the following matrices

$$\begin{bmatrix} T \\ -T \end{bmatrix} = \begin{bmatrix} A \\ -A \end{bmatrix} S.$$

3.3.2 Given Ontology Constraints

The ontology that the synsets come from has a set of constraints built into it, in particular the partial ordering that *is-a* implies. The *is-a* relationship implies that for any given synset the probability of it occurring any context must be greater than the probability of any of the elements below it existing in the same contexts. Thus we see that for any synset vector s and any $s' \in \downarrow(s)$ we have $s \geq s'$. Also, since we are dealing with a probabilistic context theory, the top level synset, if it exists, must have probability 1. In WordNet's case such a root exists and is called "entity". In such a case we see that$1_N = s_{entity}$ where 1 is a vector of size N that has 1 for every entry. If there is no root node, but rather a forest of them we need only say that their sum must be 1.

Now that we have all of the constraints implied by both the ontology and the context theory described we can deliver a final form of the constraint portion in the linear programming formulation of this problem:

$$\begin{bmatrix} T \\ -T \\ 1_N \\ -1_N \end{bmatrix} = \begin{bmatrix} A \\ -A \\ e^T_{entity} \\ -e^T_{entity} \end{bmatrix} S$$

where we let e_{entity} be the column vector which is 1 for the entity matrix if it occurs and 0 otherwise.

By solving this linear programming application we have a context theory that represents the meaning of synsets. This context theory is our best estimate of the meaning of synsets through the probability of valid contexts.

4 Applications

4.1 Word Sense Disambiguation

Word sense disambiguation is the determination of the appropriate sense of a word given a context.

So far we have been carefully trying to avoid dealing with this issue, as if we could disambiguate the senses of words with some ϕ we could have directly calculated the synset probabilities. To do that we would have calculated the probability distribution of a synset $\bar{x}_i$ existing in a context by looking at the frequency of the term x given the prior that $\bar{x}_i = \phi(x)$ within a corpus. However, now that we have a model for the distribution of the synsets we can decide on an appropriate sense for terms in a corpus.

In theory we should be able to look at the document and immediately see it that the context around a term in question determines one particular sense for that term. However, in a practical implementation this will not happen. We cannot store every configuration and usually end up having to deal with truncation issues. Not only must we truncate the size of the contexts that we look at, but it is additionally not realistic to expect to see every context that exists in any corpus at any scale to appear at all. In order to compensate for these issues we try to overcome the technical limitations in our context theories by broadening the size of the contexts we consider.

Suppose we have a token of a term x that we want to disambiguate in document d, within an appropriately sized contiguous window around x in d. Then it should be safe to assume that most if not all instances of the term are associated with the same tense. This window could be a paragraph, a couple of paragraphs, or even the whole document. If the term exists with a sufficiently high frequency within such a window we may then use the contexts derived within the window to sketch the distribution under which the term exists. From this experimental distribution we may compare it to the distributions of the various synsets of the term in question and choose a sense that fits the experimental distribution. The size of the window is important and difficult to choose. If we decide upon a window that is too large we may see more than one sense of a term frequently enough that the distribution we extract may be too noisy. Hence, we may fail to correctly choose the sense that the term takes on within the context in question. If the window is too small we may fail to sample the distribution of the term in question and be unable to determine a proper sense for the term.

First let us assume that we already have a good choice of window size chosen for us around a term x, which gives us a distribution of its contexts $\psi(x)$. Before we can directly compare x to the set of possible disambiguations, the synsets $\bar{x}_i$, we must account for the fact that $\psi(x)$ will not achieve a distribution similar to any of the synsets. Given a window of text we can't expect to see more than a incredibly small portion of the contexts for which the term can exist in to actually occur in the window. This ambiguity of which contexts the term occurs in means that among the dimensions we have in our context theory, only a fraction of the entries will be nonzero in $\psi(x)$. Also, $\psi(x)$ will respresent the relative appearance of terms with each other, so we will almost certainly have to renormalize $\psi(x)$ if we want to compare it to $\psi(\bar{x}_i)$. The choice of renormalization is also not nontrivial as there's no real way of knowing how much bias our window has toward certain contexts. Therefore, we must give an educated guess for a proper normalization of the context, the most obvious one being through a minimization of the distance between $\lambda\psi(x)$ and each of the $\psi(\bar{x}_i)$. Now when we compare our term distribution with the distribution of the synsets we are giving each synset the greatest opportunity to show that it is appropriate for a term. All that remains in our disambiguation is now is to compare the distances between x and the various senses it can take $\bar{x}_i$ to find the minimum distance, which gives us the following equation:

$$Sense = argmin_i \min_{\lambda} \|\psi(\bar{x}_i) - \lambda\psi(x)\|_{L_1}.$$

Choosing an appropriate window size is a tricky issue but there are a few simple ways to pick one. The most obvious is to choose a standard window size for all cases: the number of words away from the target term, one or many paragraphs, a chapter, etc. We can also choose a more dynamic approach such as picking a default window size and falling back to increased the window size by some prescribed amount until the window size is sufficiently large. A sufficient size can be determined by something like a threshold on how similar the distance of difference senses are to the target term are to one another.

$$\begin{aligned} Threshold \geq max_{i,j} \quad & (\min_\lambda \|\psi(\bar{x}_i) - \lambda\psi(x)\|_{L_1}) \\ & -(\min_\lambda \|\psi(\bar{x}_j) - \lambda\psi(x)\|_{L_1}). \end{aligned}$$

4.2 Connecting Terms and Synsets

As a direct result of the probability of a term being the sum of the probabilities of its synsets, we see that the vector the term is then the sum of the vectors of its synsets. Therefore if we have a context theory for the synsets we can produce a new context theory for terms that extends a pre-existing one to words that include terms for which we don't have vectors to represent for which its synsets exist can be discerned.

One of the main difficulties in maintaining an ontology is that the

5 Conclusion

We have developed a methodology for using a pre-existing context theory and ontology to generate a context theory for the elements of the ontology. Although ontologies such as WordNet take on a philosophy similar to *context-as-meaning* they only provide a relative framework for describing the probabilities of their objects in various contexts. Because of the similarity in philosophy our approach extends such ontologies in a natural way. We have proposed a method for estimating those probabilities to estimate a valid context theory, but without the ability to disambiguate senses or a properly driven objective function there is no way of knowing precisely determining vectors that correspond to the various senses of terms through linear programming. We have also discussed the utility of a context theory of synsets through two applications: to do word sense disambiguation and to discern the relationship between words and the concepts underlying them. In the generation of synset vectors we have not determined an appropriate optimization function for our problem and resolving an appropriate function would be a worthwhile direction to continue work in.

References

[1] Daoud Clarke. Context-theoretic Semantics for Natural Language an Algebraic Framework. *Framework*, (September), 2007.

[2] Daoud Clarke. *Context-theoretic semantics for natural language: an overview*, page 112119. Number March. Association for Computational Linguistics, 2009.

[3] Daoud Clarke and David Weir. Semantic Composition with Quotient Algebras. *Computational Linguistics*, (July):38–44, 2010.