A Generative Model of Words and Relationships from Multiple Sources

Stephanie L. Hyland[a,b] and **Theofanis Karaletsos**[a] and **Gunnar Rätsch**[a]
[a] Computational Biology Program, Memorial Sloan Kettering Cancer Center
1275 York Avenue, New York, NY 10065
[b] Tri-Institutional Training Program in Computational Biology and Medicine
Weill Cornell Medical College, 1305 York Ave, New York, NY 10021
{stephanie, theo, gunnar}@ratschlab.org

Abstract

Neural language models are a powerful tool to embed words into semantic vector spaces. However, learning such models generally relies on the availability of abundant and diverse training examples. In highly specialised domains this requirement may not be met due to difficulties in obtaining a large corpus, or the limited range of expression in average use. Such domains may encode prior knowledge about entities in a knowledge base or ontology. We propose a generative model which integrates evidence from diverse data sources, enabling the sharing of semantic information. We achieve this by generalising the concept of co-occurrence from distributional semantics to include other relationships between entities or words, which we model as affine transformations on the embedding space. We demonstrate the effectiveness of this approach by outperforming recent models on a link prediction task and demonstrating its ability to profit from partially or fully unobserved data training labels. We further demonstrate the usefulness of learning from different data sources with overlapping vocabularies.

Introduction[1]

A deep problem in natural language processing is to model the semantic relatedness of words, drawing on evidence from text and spoken language, as well as knowledge graphs such as ontologies. A successful modelling approach is to obtain an *embedding* of words into a metric space such that semantic relatedness is reflected by closeness in this space. One paradigm for obtaining this embedding is the *neural language model* (Bengio et al. 2003), which traditionally draws on local co-occurence statistics from sequences of words (sentences) to obtain an encoding of words as vectors in a space whose geometry respects linguistic and semantic features. The core concept behind this procedure is the *distributional hypothesis of language*; see Sahlgren (2008), that semantics can be inferred by examining the *context* of a word. This relies on the availability of a large corpus of diverse sentences, such that a word's typical context can be accurately estimated.

[1] A preliminary version of this work appeared at the International Workshop on Embeddings and Semantics at SEPLN 2015 (Hyland, Karaletsos, and Rätsch 2015).

In the age of web-scale data, there is abundant training data available for such models in the case of generic language. For *specialised* language domains this may not be true. For example, medical text data (Liu et al. 2015) often contains protected health information, necessitating access restrictions and potentially limiting corpus size to that obtainable from a single institution, resulting in a corpus with less than tens of millions of sentences, not billions as in (for example) Google n-grams. In addition to this, specialised domains expect certain prior knowledge from the reader. A doctor may never mention that anastrazole is a aromatase inhibitor (a type of cancer drug), for example, because they communicate sparsely, assuming the reader shares their training in this terminology. In such cases, it is likely that even larger quantities of data are required, but the sensitive nature of such data makes this difficult to attain.

Fortunately, such specialised disciplines often create expressive *ontologies*, in the form of annotated relationships between terms (denoted by underlines). These may be semantic, such as dog is a type of animal, or derived from domain-specific knowledge, such as anemia is an associated disease of leukemia. (This is a relationship found in the medical ontology system UMLS; see Bodenreider, 2004). We observe that these relationships can be thought of as additional *contexts* from which co-occurrence statistics can be drawn; the set of diseases associated with leukemia arguably share a common context, even if they may not co-occur in a sentence (see **Figure 1**).

We would like to use this structured information to improve the quality of learned embeddings, to use their information content to regularize the embedding space in cases of low data abundance while obtaining an explicit representation of these relationships in a vector space. We tackle this by assuming that each relationship is an *operator* which transforms words in a relationship-specific way. Intuitively, the action of these operators is to distort the shape of the space, effectively allowing words to have multiple representations without requiring a full set of parameters for each possible sense.

The intended effect on the underlying (untransformed) embedding is twofold: to encourage a solution which is more sensitive to the domain than would be achieved using only unstructured information and to use heterogeneous sources of information to compensate for sparsity of data. In addition

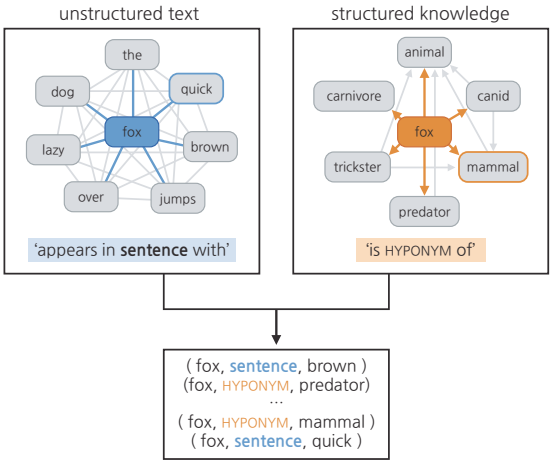

Figure 1: We unify structured and unstructured data sources by considering functional (e.g. hyponymic) relationships to be a form of *co-occurrence*, and considering sentence co-occurrence to be another type of functional relationship. Thus, our model is source-agnostic and uses true (S, R, T) triples as evidence to obtain an embedding of entities and relationships.

to this, since relationship operators can act on *any* part of the space, by learning these functions we can apply them to any word regardless of its source, allowing for link prediction on new entities in a knowledge base.

While we do not attempt to model higher-order language structure such as grammar and syntax, we consider a generative model in which the distance between terms in the embedded space describes the probability of their co-occurrence under a given relationship. Through this, we learn the joint distribution of all pairs in all relationships, and can ask questions such as *'What is the probability of anemia appearing in a sentence with imatinib[2]?'*, or *'What is the probability that anemia is a disease associated with leukemia?'* This introduces flexibility for subsequent analyses that require a generative approach.

This paper is laid out as follows: In Related Work, we describe relevant prior work concerning embedding words and relationships and place our contributions in context. In Modelling, we describe in detail the probabilistic model and our inference and learning strategy, including a link to code. In Experiments, we show an array of experimental results to quantitatively demonstrate different aspects of the model on datasets using WordNet and Wikipedia sources in supervised, semi-supervised and unsupervised settings, before summarising our findings in the Discussion section.

Related Work

The task of finding continuous representation for elements of language has been explored in great detail in recent and less-recent years. Bengio et al. (2003) described a neural architecture to predict the next word in a sequence, using distributed representations to overcome the curse of dimensionality. Since then, much work has been devoted to obtaining, understanding, and applying these distributed language representations. One such model is `word2vec` of Mikolov et al. (2013), which more explicitly relies on the distributional hypothesis of semantics by attempting to predict the surrounding context of a word, either as a set of neighbouring words (the skip-gram model) or as an average of its environment (continuous bag of words). We note later in the model section that the idealised version of skip-gram `word2vec` is a special case of our model with one relationship; appears in a sentence with. In practice, `word2vec` uses a distinct objective function, replacing the full softmax with an approximation intended to avoid computing a normalising factor. We retain a probabilistic interpretation by approximating gradients of the partition function, allowing us to follow the true model gradient while maintaining tractability. Furthermore, learning a joint distribution facilitates imputation and generation of data, dealing with missing data and making predictions using the model itself. We note that a generative approach to language was also explored by Andreas and Ghahramani (2013), but does not concern relationships.

Relational data can also be used to learn distributed representations of entities in knowledge graphs, entities which may correspond to or can be mapped to words. A general approach is to implicitly embed the graph structure through vertex embeddings and rules (or transformations) for traversing it. Bordes et al. (2011) scored the similarity of entities under a given relationship by their distance after transformation using pairs of relationship-specific matrices. Socher et al. (2013) describe a neural network architecture with a more complex scoring function, noting that the previous method does not allow for interactions between entities. The `TransE` model of Bordes et al. (2013) (and extensions such as Wang et al. (2014b), Fan et al. (2014), and Lin et al. (2015)) represents relationships as *translations*, motivated by the tree representation of hierarchical relationships, and observations that linear composition of entities appears to preserve semantic meaning (Mikolov et al. 2013). These approaches are uniquely concerned with relational data however, and do not consider distributional semantics from free text. Faruqui et al. (2015) and Johansson and Nieto Piña (2015) describe methods to modify pre-existing word embeddings to align them with evidence derived from a knowledge base, although their models do not learn representations *de novo*.

Similar in spirit to our work is Weston et al. (2013), where entities belonging to a structured database are identified in unstructured (free) text in order to obtain embeddings useful for relation prediction. However, they learn separate scoring functions for each data source. This approach is also employed by Fried and Duh (2014), Xu et al. (2014), Yu and Dredze (2014), and Wang et al. (2014a). In these cases, separate objectives are used to incorporate different data sources, combining (in the case of Xu et al. (2014)) the skip-gram objective from Mikolov et al. (2013) and the `TransE` objective of Bordes et al. (2013). Our method uses a single

[2]Imatinib is a tyrosine-kinase inhibitor used in the treatment of chronic myelogenous leukemia.

energy function over the joint space of word pairs with relationships, combining the 'distributional objective' with that of relational data by considering free-text co-occurrences as another type of relationship.

We have mentioned several approaches to integrating graphs into embedding procedures. While these graphs have been derived from knowledge bases or ontologies, other forms of graphs have also been exploited in related efforts, for example using constituency trees to obtain sentence-level embeddings (Tai, Socher, and Manning 2015).

The motivation for our work is similar in spirit to multitask and transfer learning (for instance, Caruana (1997), Evgeniou and Pontil (2004), or Widmer and Rätsch (2012)). In transfer learning one takes advantage of data related to a similar, typically supervised, learning task with the aim of improving the accuracy of a specific learning task. In our case, we have the unsupervised learning task of embedding words and relationships into a vector space and would like to use data from another task to improve the learned embeddings, here word co-occurrence relationships. This may be understood as a case of *unsupervised transfer learning*, which we tackle using a principled generative model.

Finally, we note that a recent extension of `word2vec` to full sentences (Jernite, Rush, and Sontag 2015) using a fast generative model exceeds the scope of our model in terms of sentence modeling, but does not explicitly model latent relationships or tackle transfer learning from heterogeneous data sources.

Probabilistic Modelling of Words and Relationships

We consider a probability distribution over triplets (S, R, T) where S is the *source word* of the (possibly directional) *relationship* R and T is the *target word*. Note that while we refer to 'words', S and T could represent any entity between which a relationship may hold without altering our mathematical formulation, and so could refer to multiple-word entities (such as `UMLS` Concept Unique Identifiers) or even non-lexical objects. Without loss of generality, we proceed to refer to them as words. Following Mikolov et al. (2013), we learn two representations for each word: $\mathbf{c}_s$ represents word s when it appears as a *source*, and $\mathbf{v}_t$ for word t appearing as a *target*.[3] Relationships act by altering $\mathbf{c}_s$ through their action on the vector space ($\mathbf{c}_s \mapsto G_R\mathbf{c}_s$). By allowing G_R to be an arbitrary affine transformation, we combine the bilinear form of Socher et al. (2013) with translation operators of Bordes et al. (2013).

The joint model is given by a Boltzmann probability density function,

$$\begin{aligned} P(S, R, T|\Theta) &= \tfrac{1}{Z(\Theta)} e^{-\mathcal{E}(S,R,T|\Theta)} \\ &= \tfrac{e^{-\mathcal{E}(S,R,T|\Theta)}}{\sum_{s,r,t} e^{-\mathcal{E}(s,r,t|\Theta)}} \end{aligned} \tag{1}$$

Here, the partition function is the normalisation factor over the joint posterior space captured by the model parameters $Z(\Theta) = \sum_{s,r,t} e^{-\mathcal{E}(s,r,t|\Theta)}$. The parameters Θ in this case are the representations of all words (both as sources and targets) and relationship matrices; $\Theta = \{\mathbf{c}_i, G_r, \mathbf{v}_j, \}_{i,j\in\text{vocabulary}}^{r\in\text{relationships}}$. If we choose an energy function

$$\mathcal{E}(S, R, T|\Theta) = -\mathbf{v}_T \cdot G_R\mathbf{c}_S \tag{2}$$

we observe that the $|R| = 1$, $G_R = \mathbb{I}$ case recovers the original softmax objective described in Mikolov et al. (2013), so the idealised `word2vec` model is a special case of our model.

This energy function is problematic however, as it can be trivially minimised by making the norms of all vectors tend to infinity. While the partition function provides a global regularizer, we find that it is not sufficient to avoid norm growth during training. We therefore use as our energy function the negative cosine similarity, which does not suffer this weakness;[4]

$$\mathcal{E}(S, R, T|\Theta) = -\frac{\mathbf{v}_T \cdot G_R\mathbf{c}_S}{\|\mathbf{v}_T\|\|G_R\mathbf{c}_S\|} \tag{3}$$

This is also a natural choice, as cosine similarity is the standard method for evaluating word vector similarities. Energy minimisation is therefore equal to finding an embedding in which the *angle* between related entities is minimised in an appropriately transformed relational space. It would be simple to define a more complex energy function (using perhaps splines) by choosing a different functional representation for R, but we focus in this work on the affine case.

Inference and Learning We estimate our parameters Θ from data using stochastic maximum likelihood on the joint probability distribution. The maximum likelihood estimator is:

$$\Theta^* = \text{argmax } P(\mathcal{D}|\Theta) = \text{argmax } \prod_n^N P((S, R, T)_n|\Theta) \tag{4}$$

Considering the log-likelihood at a single training example (S, R, T) and taking the derivative with respect to parameters, we obtain:

$$\begin{aligned} \frac{\partial \log P(S, R, T|\Theta)}{\partial \Theta_i} =& \frac{\partial}{\partial \Theta_i} \left[-\mathcal{E}(S, R, T|\Theta)\right] \\ & - \frac{\partial}{\partial \Theta_i} \left[\log \sum_{s,r,t} e^{-\mathcal{E}(S,R,T|\Theta)}\right] \end{aligned} \tag{5}$$

[3] Goldberg and Levy (2014) provide a motivation for using two representations for each word. We can extend this by observing that words with similar $\mathbf{v}$ representations share a *paradigmatic* relationship in that they may be exchangeable in sentences, but do not tend to co-occur. Conversely, words s and t with $\mathbf{c}_s \approx \mathbf{v}_t$ have a *syntagmatic* relationship and tend to co-occur (e.g. Sahlgren (2008)). Thus, we seek to enforce syntagmatic relationships and through transitivity obtain paradigmatic relationships of $\mathbf{v}$ vectors.

[4] We also considered an alternate, more symmetric energy function using the Frobenius norm of G;

$$\mathcal{E}(S, R, T|\Theta) = -\frac{\mathbf{v}_T \cdot G_R\mathbf{c}_S}{\|\mathbf{v}_T\|\|G_R\|_F\|\mathbf{c}_S\|}$$

However, we found no clear empirical advantage to this choice.

Given a smooth energy function the first term is easily obtained, but the second term is problematic. This term, derived from the partition function $Z(\Theta)$, is intractable to evaluate in practice owing to its double sum over the size of the vocabulary (potentially $\mathcal{O}(10^5)$). In order to circumvent this intractability we resort to techniques used to train Restricted Boltzmann Machines and use stochastic maximum likelihood, also known as persistent contrastive divergence (PCD); (Tieleman 2008). In contrastive divergence, the gradient of the partition function is estimated using samples drawn from the model distribution seeded at the current training example (Hinton 2002). However, many rounds of sampling may be required to obtain good samples. PCD retains a persistent Markov chain of model samples across gradient evaluations, assuming that the underlying distribution changes slowly enough to allow the Markov chain chain to mix. We use Gibbs sampling by iteratively using the conditional distributions of all variables (S, R, and T, see below) to obtain model samples.

In particular, we draw S, R and T from the conditional probability distributions:

$$\begin{aligned} P(S|r,t;\Theta) &= \frac{e^{-\mathcal{E}(S,r,t|\Theta)}}{\sum_{s'} e^{-\mathcal{E}(s',r,t|\Theta)}} \\ P(R|s,t;\Theta) &= \frac{e^{-\mathcal{E}(s,R,t|\Theta)}}{\sum_{r'} e^{-\mathcal{E}(s,r',t|\Theta)}} \\ P(T|s,r;\Theta) &= \frac{e^{-\mathcal{E}(s,r,T|\Theta)}}{\sum_{t'} e^{-\mathcal{E}(s,r,t'|\Theta)}} \end{aligned} \tag{6}$$

Thereby, we can estimate the gradient of $Z(\Theta)$ at the cost of these evaluations, which are linear in the size of the vocabulary.

Using this, following the objective from (5) further simplifies to a contrastive objective given a batch of B data samples and M model samples (each model sample obtained from an independent, persistent Markov chain):

$$\begin{aligned} \frac{\partial P(\mathcal{D}|\Theta)}{\partial \Theta_i} \simeq \frac{1}{M}\sum_{m=1}^{M}\left[\frac{\partial \mathcal{E}((S,R,T)_m|\Theta)}{\partial \Theta_i}\right] \\ - \frac{1}{B}\sum_{b=1}^{B}\left[\frac{\partial \mathcal{E}((S,R,T)_b|\Theta)}{\partial \Theta_i}\right] \end{aligned} \tag{7}$$

Interestingly, the model can gracefully deal with missing elements in observed triplets (for instance missing observed relationships). Learning is achieved by considering the partially observed triple as a superposition of all possible completions of that triple, each weighted by its conditional probability given the observed elements, using (6). This produces a gradient which is a weighted sum.

In the fully-observed case (which we sometimes call supervised in an abuse of terminology), the weighting is simply a spike on the observed state. Similarly, the model can predict missing values as a simple inference step. These properties make having a joint distribution very attractive in practical use, offsetting the conceptual difficulty of training. In our experiments, we exploit these properties to do principled semi-supervised and unsupervised learning with partially observed or unobserved relationships without needing an external noise distribution or further assumptions.

Implementation We provide the algorithm in Python (https://github.com/corcra/bf2). Since most of its runtime takes place in vector operations, we are developing a GPU-optimised version. We use `Adam` (Kingma and Ba 2015) to adapt learning rates and improve numerical stability. We used the recommended hyperparameters from this paper; $\lambda = 1 - 10^{-8}$, $\epsilon = 1 - 10^{-8}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$. Unless otherwise stated, hyperparameters specific to our model were: dimension $d = 100$, batch size of $B = 100$, learning rate for all parameter types of $\alpha = 0.001$, and three rounds of Gibbs sampling to obtain model samples.

Experiments

We will proceed to explore the model in five settings. First, an entity vector embedding problem on WordNet which consists of fully observed triplets of words and relationships. In the second case we demonstrate the power of the semi-supervised extension of the algorithm on the same task. We then show that a) adding relationship data can lead to better embeddings and b) that adding unstructured text can lead to better relationship predictions. Finally, we demonstrate that the algorithm can also identify latent relationships that lead to better word embeddings.

Data As structured data, we use the WordNet dataset described by Socher et al. (2013), available at http://stanford.io/1IENOYH. This contains 38,588 words and 11 types of relationships. Training data consists of true triples such as (feeling, has instance, pride).

We derived an additional version of this dataset by stripping sense IDs from the words, which reduced the vocabulary to 33,330 words. We note that this procedure likely makes prediction on this data more difficult, as every word receives only one representation. We did this in order to produce an aligned vocabulary with our *unstructured* data source, taken to be English Wikipedia (https://dumps.wikimedia.org/, August 2014). We extracted text using `WikiExtractor` (http://bit.ly/1Imz1WJ). We greedily identified WordNet 2-grams in the Wikipedia text. Two words were considered in a sentence context if they appeared within a five word window. Only pairs for which both words appeared in the WordNet vocabulary were included. We drew from a pool of 112,581 training triples in WordNet with 11 relationships, and 8,206,304 triples from Wikipedia (heavily sub-sampled, see experiments). To check that our choice to strip sense IDs was valid, we also created a version of the Wikipedia dataset where each word was tagged with its most common sense from the WordNet training corpus. We found that this did not significantly impact our results, so we chose to continue with the sense-stripped version, preferring to collapse some WordNet identities over assigning possibly-incorrect senses to words in Wikipedia.

WordNet Prediction Task We used our model to solve the basic prediction task described in Socher et al. (2013). In

this case, the model must differentiate true and false triples, where false triples are obtained by corrupting the T entry in the triple, e.g. $(S, R, T) \rightarrow (S, R, \tilde{T})$ (where $(S, R, \tilde{T})$ doesn't appear in the training data). The 'truth' of a triple is evaluated by its energy $\mathcal{E}(S, R, T)$, with a relationship-specific cut-off chosen by maximizing accuracy on a validation set (this is an equivalent procedure to the task as initially described). By learning explicit representations of each of the 38,588 entities in WordNet, our approach most closely follows the *'Entity Vector'* task in Socher *et al.* This is to be contrasted with the *'Word Vector'* task, where a representation is learned for each word, and entity representations are obtained by averaging their word vectors. We elected not to perform this task because we are not confident that composition into phrases through averaging is well-justified. Using the validation set to select an early stopping point at 66 epochs, we obtain a test set accuracy of 78.2% with an AUROC of 85.6%. The 'Neural Tensor Model' (NTN) described in Socher et al. (2013) achieves an accuracy of around 70% on this task, although we note that the simpler Bilinear model also described in Socher et al. (2013) achieves 74% and is closer to the energy function we employ. The improved performance exhibited by this simpler Bilinear model was also noted by Yang et al. (2015). Other baselines reported by Socher *et al.* were a single layer model without an interaction term, a Hadamard model (Bordes et al. 2012) and the model of Bordes et al. (2011) which learns separate left and right relationship operators for each element of the triple. These were outperformed by the Bilinear and NTN models, see Figure 4 in Socher et al. (2013) for further details. Hence, our model outperforms the two previous methods by more than 4%.

As a preliminary test of our model, we also considered the `FreeBase` task described by Socher et al. (2013). Initial testing yielded an accuracy of 85.7%, which is comparable to the result of their best-performing model (NTN) of about 87%. We chose not to further explore this dataset however, because its entities are mostly proper nouns and thus seemed unlikely to benefit from additional semantic data.

Semi-supervised Learning for WordNet We next tested the semi-supervised learning capabilities of our algorithm (see Inference and Learning). For this we consider the same task as before, but omit some label information in the training set and instead use posterior probabilities during the inference. For this we trained our algorithm with a subset of the training data (total 112,581 examples) and measured the accuracy of classifying into true and false relationships as before. The fully-observed case used only a subset of fully-observed data (varying amounts as indicated on the x-axis). For semi-supervised learning, we also used the remaining data, but masking the type of the relationship between pairs. In **Figure 2** we report the accuracy for different labelled/unlabelled fractions of otherwise the same dataset. We find that the semi-supervised method consistently performs better than the fully observed method for all analysed training set sizes. In this and the previous experiment, one Markov chain was used for PCD and a l_2 regulariser on G_R parameters with weight 0.01.

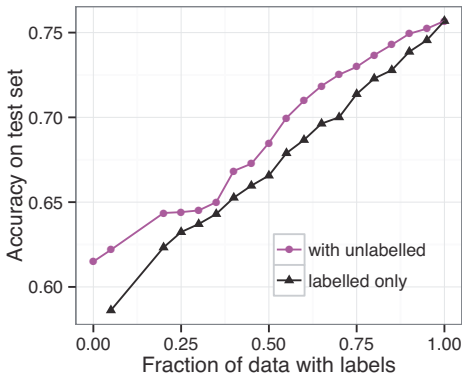

Figure 2: **Semi-supervised learning improves learned embeddings:** We tested the semi-supervised extension of our approach on the entity relationship learning task described in Socher et al. (2013) and previous subsection. Following Socher *et al.*, we predict if a triple (S, R, T) is true by using its energy as a score. For this we trained our algorithm with a subset of the training data (total 112,581 examples). The fully-supervised version only used the subset of fully labelled data (varying amounts as indicated on the x-axis). For semi-supervised learning, in addition we use the remaining data but where the type of the relationship between pairs is unknown. We find that the semi-supervised method consistently performs better than the fully supervised method (see main text for more details).

Adding Unstructured Data to a Relationship Prediction Task To test how unstructured text data may improve a prediction task when *structured* data is scarce, we augmented a subsampled set of triples from WordNet with 10,000 examples from Wikipedia and varied the weight κ associated with their gradients during learning. The task is then to predict whether or not a given triple (S, R, T) is a true example from WordNet, as described previously. **Figure 3** shows accuracy on this task as κ and the amount of structured data vary. To find the improvement associated with *unstructured data*, we compared accuracy at $\kappa = 0$ with $\kappa = \kappa^*$ (where κ^* gave the highest accuracy on the validation set; marked with $*$). We find that including free text data quite consistently improves the classification accuracy, particularly when structured data is scarce.

In this experiment and all following, we used five Markov chains for PCD and a l_2 regulariser on all parameters with weight 0.001.

Relationship Data for Improved Embeddings In this case, we assume *unstructured text data* is restricted, and vary the quantity of structured data. To evaluate the *untransformed* embeddings, we use them as the inputs to a supervised multi-class classifier. The task for a given (S, R, T) triple is to predict R given the vector formed by concatenating $\mathbf{c}_S$ and $\mathbf{v}_T$. We use a random forest classifier trained on

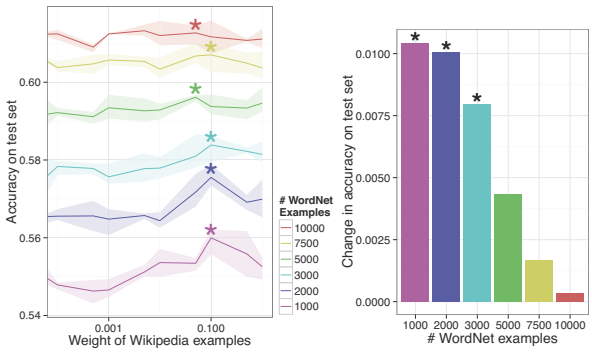

Figure 3: **Unstructured data helps relationship learning:** In addition to training on a set of known relationships, we use unstructured data from Wikipedia with varying weight (x-axis) during training. As before, with the goal to predict if a triple (S, R, T) is true by using its energy as a score. A validation set is used to determine the threshold below which a triple is considered 'true'. The solid line denotes the average of three independent experimental runs; shaded areas show the range of results. The bar plot on the right shows the difference in accuracy between $\kappa = 0$ and $\kappa = \kappa^*$, where κ^* gave the highest accuracy on a validation set. Significance at 5% (paired t-test) is marked by asterisk. We find then that unstructured Wikipedia can improve relationship learning in cases when labelled relationship data is scarce.

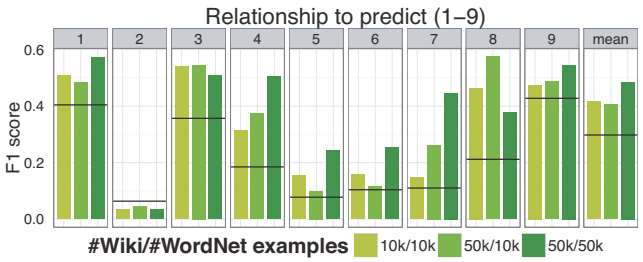

Figure 4: **Relationship data improves learned embeddings:** We apply our algorithm on a scarce set of Wikipedia co-occurences (10k and 50k instances) with varying amounts of additional, unrelated relationship data (10k and 50k relations from WordNet). We test the quality of the embedding by measuring the accuracy on a task related to nine relationships (has instance, domain region, subordinate instance of, member holonym, has part, has part, part of, member meronym, synset domain topic, type of; relationships similar to, domain topic were omitted for technical reasons). We used eight of the relationships together with the Wikipedia data to learn representations that are then used in a subsequent supervised learning task to predict the remaining ninth relationship based on the representations using random forests. Black lines denote results from `word2vec` trained on a Wikipedia-only dataset with 4,145,372 *sentences*.

the WordNet validation set using five-fold cross-validation.

To avoid testing on the training data (since the embeddings are obtained using the WordNet training set), we perform this procedure once for each relationship (11 times - excluding appears in sentence with), each time removing from the training data *all* triples containing that relationship. **Figure 4** shows the $F1$ score of the multi-class classifier on the left-out relationship for different combinations of data set sizes. We see that for most relationships, including more unstructured data improves the embeddings (measured by performance on this task). We also trained `word2vec` (Mikolov et al. 2013) on a much larger Wikipedia-only dataset (4,145,372 *sentences*) and trained a classifier on its vectors; results are shown as black lines. We see that our approach yields a consistently higher $F1$ score, suggesting that even data about *unrelated* relationships provides information to produce vectors that are semantically richer overall.

These results illustrate that embeddings learned from limited free text data can be improved by additional, unrelated relationship data.

Unsupervised Learning Of Relationships In our final experiment, we explore the ability of the model to learn embeddings from co-occurrence data alone, without specifying the relationships it should use. When using the model with just one relationship (trivially the identity), the model effectively reverts to `word2vec`. However, if we add a budget of relationships (in our experiments we use 1, 3, 5, 7, 11), the model has additional parameters available to learn affine transformations of the space which can differentiate how distances and meaning interact for the word embeddings without fixing this *a priori*. Our intuition is that we want to test whether textual context alone has substructure that we can capture with latent variables. We generate a training set of one million word co-occurrences from Wikipedia (using a window size of 5 and restricting to words appearing in the WordNet dataset, as described earlier), and train different models for each number of latent relationships. Inspired by earlier experiments testing the utility of supplanting WordNet training data with Wikipedia examples, we decide to test the ability of a model purely trained on Wikipedia to learn word and relationship representations which are predictive of WordNet triplets, *without* having seen any data from WordNet. As a baseline we start with $|R| = 1$ to test how well word embeddings from context alone can perform, indicated by the leftmost bar in **Figure 5**. We then proceed to train models with more latent relationships. We observe that, especially for some relationship prediction tasks, including this flexibility in the model produces a noticeable increase in $F1$ score on this task. Since we evaluate the embeddings alone, this effect must be due to a shift in the content of these vectors, and cannot be explained by the additional parameters introduced by the latent relationships. *We note that a consistent explanation for this phenomenon is that the model discovers contextual subclasses which are indicative of WordNet-type relationships. This observation opens doors to further explorations of the hypothesis regarding contextual subclasses and unsupervised relationships learning from different types of co-occurrence data.*

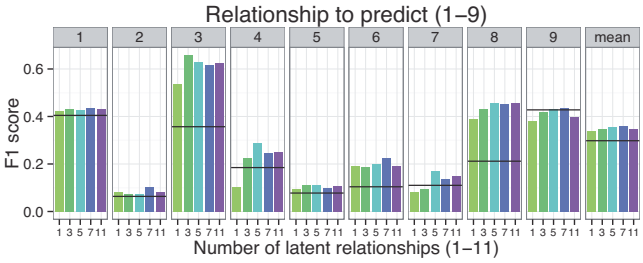

Figure 5: **Unsupervised learning of latent relationships improves embeddings**: We train a fully unsupervised algorithm with 1, 3, 5, 7 and 11 possible latent relationships on one million Wikipedia sentences. Initialisation is at random and without prior knowledge. To test the quality of the resulting *embeddings*, we use supervised learning of nine WordNet relationships with random forests. Depending on the relationship at hand, the use of multiple latent relationships during training leads to slightly, but consistently better accuracies using the computed embeddings for every of the nine relationships and also on average. Hence, the resulting embeddings using unsupervisedly learned latent relationships can be said to be of higher quality. Once again, black lines show results using `word2vec`.

We note that we did not perform an exhaustive search of the hyperparameter space; better settings may yet exist and will be sought in future work. Nonetheless, although the absolute improvement in $F1$ score yielded by this method is modest, we are encouraged by the model's ability to exploit latent variables in this way.

Discussion

We have presented a *probabilistic generative model of words and relationships* between them. By estimating the parameters of this model through stochastic gradient descent, we obtain vector and matrix representations of these words and relationships respectively. To make learning tractable, we use *persistent contrastive divergence* with Gibbs sampling between entity types (S, R, T) to approximate gradients of the partition function. Our model uses an energy function which contains the idealised `word2vec` model as a special case. By augmenting the embedding space and considering relationships as arbitrary *affine* transformations, we combine benefits of previous models. In addition, our formulation as a generative model is distinct and allows a more flexible use, especially in the missing data, semi- and unsupervised setting. Motivated by domain-settings in which structured *or* unstructured data may be scarce, we illustrated how a model that combines both data sources can improve the quality of embeddings, supporting other findings in this direction.

A promising field of exploration for future work is a more detailed treatment of relationships, perhaps generalising from affine transformations to include nonlinear maps. Our choice of cosine similarity in the energy function can also be developed, as we note that this function is insensitive to very small deviations in angle, and may therefore produce looser clusters of synonyms. Nonlinearity could also be introduced in the energy, using for example splines. Furthermore, we intend to encode the capacity for richer transfer of structured information from sources such as graphs as prior knowledge into the model. Our current model can take advantage of local properties of graphs to that purpose, but has no explicit encoding for nested and distributed relationships.

A limitation of our model is its conceptual inability to embed whole sentences (which has been tackled by averaging vectors in other work, but requires deeper investigation). Recurrent or more complex neural language models offer many avenues to pursue as extensions for our model to tackle this. A particularly interesting direction to achieve that would be a combination with work such as (Jernite, Rush, and Sontag 2015), which could in principle be integrated with our model to include relationships.

The intended future application of this model is exploratory semantic data analysis in domain-specific pools of knowledge. We can do so by combining prior knowledge with unstructured information to infer, for example, new edges in knowledge graphs. A promising such field is *medical language processing*, retrospective exploratory data analysis may boost our understanding of the complex relational mechanisms inherent in multimodal observations, and specific medical knowledge in the form of (for example) the UMLS can be used as a strong regulariser. Indeed, initial experiments combining clinical text notes with relational data between UMLS concepts from SemMedDB (Kilicoglu et al. 2012) have demonsrated the utility of this combined approach to predict the functional relationship between medical concepts, for example, `cisplatin treats diabetes`. We are in the process of expanding this investigation.

Acknowledgments

This work was funded by the Memorial Hospital and the Sloan Kettering Institute (MSKCC; to G.R.). Additional support for S.L.H. was provided by the Tri-Institutional Training Program in Computational Biology and Medicine.

References

Andreas, J., and Ghahramani, Z. 2013. A generative model of vector space semantics. *Association for Computational Linguistics (ACL)* 91.

Bengio, Y.; Ducharme, R.; Vincent, P.; and Janvin, C. 2003. A neural probabilistic language model. *J. Mach. Learn. Res.* 3:1137–1155.

Bodenreider, O. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic Acids Research* 32:D267–D270.

Bordes, A.; Weston, J.; Collobert, R.; and Bengio, Y. 2011. Learning structured embeddings of knowledge bases. In *Conference on Artificial Intelligence*.

Bordes, A.; Glorot, X.; Weston, J.; and Bengio, Y. 2012. Joint learning of words and meaning representations for open-text semantic parsing. In *International Conference on Artificial Intelligence and Statistics*, 127–135.

Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; and Yakhnenko, O. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems (NIPS)*, 2787–2795.

Caruana, R. 1997. Multitask Learning. *Machine Learning* 28(1):41 – 75.

Evgeniou, T., and Pontil, M. 2004. Regularized multi-task learning. In *International Conference on Knowledge Discovery and Data Mining*, 109–117.

Fan, M.; Zhou, Q.; Chang, E.; and Zheng, T. F. 2014. Transition-based knowledge graph embedding with relational mapping properties. In *Proceedings of the 28th Pacific Asia Conference on Language, Information, and Computation*, 328–337.

Faruqui, M.; Dodge, J.; Jauhar, S. K.; Dyer, C.; Hovy, E.; and Smith, N. A. 2015. Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*. Association for Computational Linguistics.

Fried, D., and Duh, K. 2014. Incorporating both distributional and relational semantics in word representations. In *Workshop Contribution at the International Conference on Learning Representations (ICLR), 2015*.

Goldberg, Y., and Levy, O. 2014. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*.

Hinton, G. E. 2002. Training products of experts by minimizing contrastive divergence. *Neural Computation* 14(8):1771–1800.

Hyland, S. L.; Karaletsos, T.; and Rätsch, G. 2015. A generative model of words and relationships from multiple sources. In *International Workshop on Embeddings and Semantics*, 16–20.

Jernite, Y.; Rush, A. M.; and Sontag, D. 2015. A fast variational approach for learning markov random field language models. *International Conference on Machine Learning (ICML)*.

Johansson, R., and Nieto Piña, L. 2015. Embedding a semantic network in a word space. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL): Human Language Technologies*, 1428–1433. Association for Computational Linguistics.

Kilicoglu, H.; Shin, D.; Fiszman, M.; Rosemblat, G.; and Rindflesch, T. C. 2012. Semmeddb: a pubmed-scale repository of biomedical semantic predications. *Bioinformatics* 28(23):3158–3160.

Kingma, D., and Ba, J. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR) 2015*.

Lin, Y.; Liu, Z.; Sun, M.; Liu, Y.; and Zhu, X. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of AAAI*.

Liu, Y.; Ge, T.; Mathews, K.; Ji, H.; and McGuinness, D. 2015. Exploiting task-oriented resources to learn word embeddings for clinical abbreviation expansion. In *Proceedings of BioNLP 15*, 92–97. Beijing, China: Association for Computational Linguistics.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*, 3111–3119.

Sahlgren, M. 2008. The distributional hypothesis. *Italian Journal of Linguistics* 20(1):33–53.

Socher, R.; Chen, D.; Manning, C. D.; and Ng, A. 2013. Reasoning with neural tensor networks for knowledge base completion. In *Advances in Neural Information Processing Systems (NIPS)*, 926–934.

Tai, K.; Socher, R.; and Manning, C. D. 2015. Improved semantic representations from tree-structured long short-term memory networks. *Association for Computational Linguistics (ACL)*.

Tieleman, T. 2008. Training restricted boltzmann machines using approximations to the likelihood gradient. In *International Conference on Machine Learning (ICML)*.

Wang, Z.; Zhang, J.; Feng, J.; and Chen, Z. 2014a. Knowledge graph and text jointly embedding. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics*, 1591–1601.

Wang, Z.; Zhang, J.; Feng, J.; and Chen, Z. 2014b. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 1112–1119.

Weston, J.; Bordes, A.; Yakhnenko, O.; and Usunier, N. 2013. Connecting language and knowledge bases with embedding models for relation extraction. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1366–1371.

Widmer, C., and Rätsch, G. 2012. Multitask Learning in Computational Biology. *JMLR W&CP. ICML 2011 Unsupervised and Transfer Learning Workshop.* 27:207–216.

Xu, C.; Bai, Y.; Bian, J.; Gao, B.; Wang, G.; Liu, X.; and Liu, T.-Y. 2014. Rc-net: A general framework for incorporating knowledge into word representations. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*, 1219–1228. ACM.

Yang, B.; Yih, W.; He, X.; Gao, J.; and Deng, L. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *Proceedings of the International Conference on Learning Representations (ICLR) 2015*.

Yu, M., and Dredze, M. 2014. Improving lexical embeddings with semantic knowledge. In *Association for Computational Linguistics (ACL)*, 545–550.