Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 6069–6076
Marseille, 20-25 June 2022

HyperBox: A Supervised Approach for Hypernym Discovery using Box Embeddings

Maulik Parmar, Dr. Apurva Narayan
Independent Researcher, University of British Columbia
British Columbia, Canada, British Columbia, Canada
maulikres@gmail.com, apurva.narayan@ubc.ca

Abstract

Hypernymy plays a fundamental role in many AI tasks like taxonomy learning, ontology learning, etc. This has motivated the development of many automatic identification methods for extracting this relation, most of which rely on word distribution. We present a novel model *HyperBox* to learn box embeddings for hypernym discovery. Given an input term, *HyperBox* retrieves its suitable hypernym from a target corpus. For this task, we use the dataset published for SemEval 2018 Shared Task on Hypernym Discovery. We compare the performance of our model on two specific domains of knowledge: medical and music. Experimentally, we show that our model outperforms existing methods on the majority of the evaluation metrics. Moreover, our model generalize well over unseen hypernymy pairs using only a small set of training data.

Keywords: Box Embeddings, Hypernym Discovery, Hypernym relation

1. Introduction

In linguistics, hypernymy is a semantic relation between a hypernym denoting a superordinate and a hyponym denoting a subordinate. Hypernymy is a major semantic relation and a vital organization principle of semantic memory (Miller and Fellbaum, 1991). It is an asymmetric relation between a hypernym (supertype) and a hyponym (subtype), as in animal-dog and sport-tennis. Figure 1 shows some examples of a hypernym class along with some of their hyponyms. As we can see from Figure 1, hypernyms are a more general class of hyponym terms. It plays a crucial role in language understanding because it enables generalization, which lies at the core of human cognition. Therefore, it has been an active area of research in NLP for decades. Automatic hypernym discovery is useful in many tasks like taxonomy creation (Snow et al., 2006; Navigli et al., 2011), recognizing textual entailment (Dagan et al., 2013), and text generation (Biran and McKeown, 2013).

In this paper, we tackle the problem of hypernym discovery (Espinosa-Anke et al., 2016) instead of hypernym detection (Shwartz et al., 2017). Generally, evaluation benchmarks for modeling hypernymy are such that they are reduced to a binary classification task where one tries to predict if a hypernymy relation exists between candidate pairs. Hypernym detection uses this experimental setting and tends to suffer from lexical memorization phenomena (Levy et al., 2015) due to the inherent modeling of the datasets by supervised systems. Thus, to alleviate this problem (Espinosa-Anke et al., 2016) proposed to frame the problem as Hypernym Discovery i.e given a query term and search space of domain vocabulary, discover the best candidate hypernyms of input hyponym. This reformulation not only helps in alleviating the issues discussed above but also helps to use it with other downstream applications such as semantic search, query understanding etc. Motivated by this, the organizers of SemEval Task9 published a full-fledged benchmarking dataset (Camacho-Collados et al., 2018) for the novel task of hypernym discovery, which covered multiple languages and knowledge domains. In this paper, we test our hypothesis on a corpus of the English language in two domains: Music and

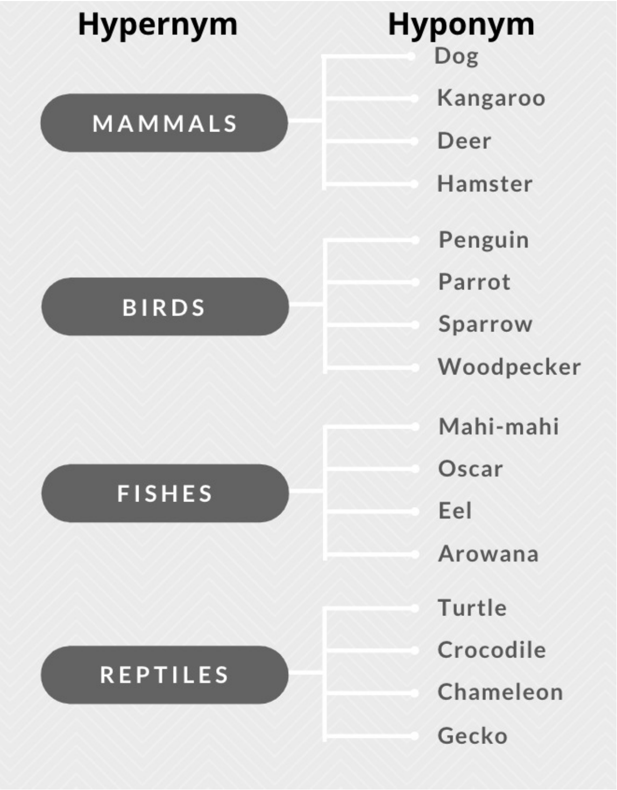

Figure 1: Examples of Hypernym-Hyponym pairs

Medical.

Two families of approaches to identify and discriminate hypernyms are prominent in Hypernym Discovery. Pattern-based approaches for relation extraction have been discussed for a while in the literature and are used to discover a variety of relations including general hypernymy relation. The pattern-based approach (Hearst, 1992; Navigli and Velardi, 2010; Pavlick and Pasca, 2017) to discover hypernymy was pioneered by Hearst (Hearst, 1992) where the author defined certain lexico-syntactic patterns (e.g X such as Y) to discover hypernymy relations between pairs from corpora. Hearst introduced many such patterns in the paper for hypernym discovery. But generally, these approaches suffer from low recall as the inherent assumption is that both hypernym hyponym pairs co-occur in a pattern. This is often not the case and leads to reduced recall.

The second line of approaches uses supervised techniques and distributional models (Sanchez and Riedel, 2017; Weeds et al., 2014; Santus et al., 2014) for the task of hypernym discovery. The general idea is to learn a function that takes as input the word embeddings of a query q and a candidate hypernym h and outputs the likelihood that there is a hypernymy relationship between q and h or outputs a distance in the embedding space between q and h. This decision function is learned in a supervised fashion using examples of pairs of words that are related by hypernymy and pairs that are not.

In this work, we consider the task of discovering hypernyms from large text corpora in a supervised way. We use the recently introduced Box Embeddings (Abboud et al., 2020) to discover hypernyms from a text corpus. (Abboud et al., 2020) proposed a spatio-translational embedding model, called BoxE that embeds entities as points, and relations as a set of hyper-rectangles (or boxes), which spatially characterize basic logical properties. Our approach is also based on Box embeddings. We show that our method *HyperBox* experimentally outperforms existing methods for hypernym discovery on most of the evaluation metrics. Our contributions are as follows:

- We introduce Box embeddings for hypernym discovery. To the best of our knowledge, this is the first model of its kind for hypernym discovery.
- Through extensive experiments on real-world datasets, we establish HyperBox's effectiveness in discovering Hypernyms.

2. Related Work

Hypernym Detection and Discovery: Traditionally, discovering hypernymic relations from text corpora has been addressed using both unsupervised and supervised approaches. The pattern-based approach is a popular unsupervised approach that uses lexico-syntactic patterns to discover hypernyms from text corpora. Hearst in her paper (Hearst, 1992) defined many such patterns for extracting hypernym relation. These high-precision patterns can also be learned automatically. However, it is well understood that the pattern-based approaches suffer significantly from missing hypernym extraction as terms must occur in exactly the right configuration to be detected.

Conversely, distributional approaches rely on a distributional representation for each observed word and are capable of discovering hypernymic relations between words even when they do not occur together explicitly in the text. Moreover, distributional approaches provide rich representations of lexical meaning. A variety of distributional methods for unsupervised hypernymy detection have been proposed (Weeds and Weir, 2003; Lenci and Benotto, 2012; Chang et al., 2018; Weeds et al., 2004) all rely on some variation of the distributional inclusion hypothesis: If x is a semantically narrower term than y, then a significant number of salient distributional features of x is expected to be included in the feature vector of y as well. Moreover, (Santus et al., 2014) proposed the distributional informativeness hypothesis i.e hypernyms tend to be less informative than hyponyms, and that they occur in more general contexts than their hyponyms.

Most of the recent work on the subject is however supervised and is based on using word embeddings as input for classification or prediction (Fu et al., 2014; Espinosa-Anke et al., 2016; Sanchez and Riedel, 2017; Baroni et al., 2012; Nguyen et al., 2017). (Shwartz et al., 2016) showed that pattern-based and distributional evidence can be effectively combined within a neural architecture.

Embeddings: Our approach is based on embeddings. (Yu et al., 2015) proposed a dynamic distance-margin model to learn term embeddings that capture properties of hypernymy. The model is trained on the pre-extracted taxonomic relation data and the resulting term embeddings are fed to an SVM classifier to predict hypernymy relation. However, one of the major drawbacks of this model is that they learn term pairs without considering their contexts, leading to a lack of generalization for term embeddings. Order-embeddings (Vendrov et al., 2016) represent text and images with embeddings where the ordering over individual dimensions forms a partially ordered set.

Hyperbolic embeddings represent words in hyperbolic manifolds such as the Poincare ball and may be viewed as a continuous analogue to tree-like structures (Nickel and Kiela, 2017; Nickel and Kiela, 2018). But these graph-based methods generally require supervision of hierarchical structure, and cannot learn taxonomies using only unstructured noisy data. (Luu et al., 2016) introduced a dynamic weighting neural network to learn term embeddings that encode information about hyper-

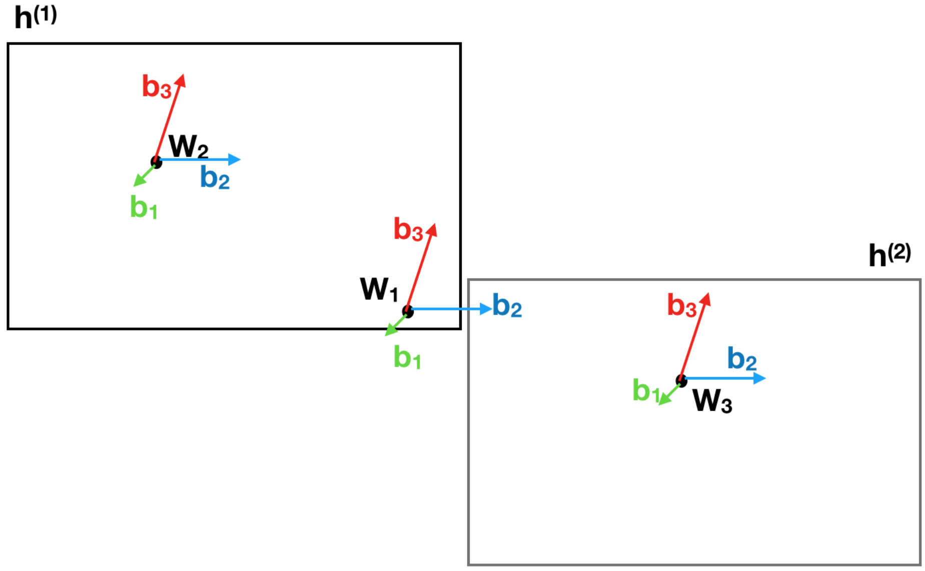

Figure 2: An example *HyperBox* model for three words w_1, w_2, w_3 in R^2 for hypernymy pairs $h(w_2, w_3)$, $h(w_2, w_1)$ and $h(w_1, w_3)$. The hypernymy relation is encoded by box embeddings $\mathbf{h^{(1)}}$ and $\mathbf{h^{(2)}}$. Every word w_i has an embedding $\mathbf{w_i}$, and $\mathbf{b_i}$ which defines a bump on other words, as shown with distinct colors.

nymy and their contexts, considering all terms between a hyponym and its hypernym in a sentence. The proposed model is trained on a set of hypernym relations extracted from WordNet (Miller, 1995). The embeddings are fed as features to an SVM classifier to detect hypernymy but the method still is not able to determine the directionality of a hypernym pair. (Vilnis et al., 2018; Li et al., 2019) proposed construction of a novel box lattice and accompanying probability measure to capture anticorrelation and disjoint concepts. (Abboud et al., 2020) introduced BoxE, a Box embeddings model that embeds entities as points, and relations as a set of hyper-rectangles (or boxes), which spatially characterize basic logical properties. Our approach is also based on Box embeddings.

3. HyperBox: Proposed Method

In this section, we present our model for hypernym discovery, *HyperBox*. The general idea is to learn a function that takes as input the word embeddings of a query q and a candidate hypernym h and outputs the score that there is a hypernymy relationship between q and h. To discover hypernyms for a given query q (rather than classify a given pair of words), we apply this decision function to all candidate hypernyms and select the most likely candidates. In this section, we start with the description of the *HyperBox* model. After this, we describe the distance function and the training objective used to train our *HyperBox* model for hypernym discovery. The code for HyperBox can be found at https://github.com/maulikres/HyperBox.

3.1. Hypernym Discovery using HyperBox

In this subsection, we describe *HyperBox*, an embedding model that encodes hypernym relation as axis-aligned hyper-rectangles (or boxes) and words as points in the d-dimensional Euclidean space.

Consider a vocabulary obtained from a corpus, which consists of a finite set **E** of words. Given a word w_i and word embedding dimension m, the model retrieves its embedding $e_i \in \mathbb{R}_m$ using a lookup table. These embeddings were learned beforehand on a large unlabeled text corpus. In *HyperBox*, every word $w_i \in \mathbf{E}$ is represented by two vectors $\mathbf{w_i}, \mathbf{b_i} \in \mathbb{R}^d$ in the d-dimensional Euclidean space, where $\mathbf{w_i}$ defines the base position of word, and $\mathbf{b_i}$ defines its translational bump, which translates all the words co-occuring in a hypernymy relation with w_i, from their base positions to their final embeddings by "bumping" them. We define base projection matrix $\phi_{base} \in \mathbb{R}^{d \times m}$ and bump projection matrix $\phi_{bump} \in \mathbb{R}^{d \times m}$ to obtain base position and translational bump for each word. The base position and translational bump of each word is obtained by projecting initial word embeddings using two matrix ϕ_{base} and ϕ_{bump} as follows:

$$\mathbf{w_i} = \phi_{base} \cdot e_i \tag{1}$$

$$\mathbf{b_i} = \phi_{bump} \cdot e_i \tag{2}$$

The final embedding of a word w_i relative to a hypernym pair $h(w_i, w_j)$ is hence given by:

$$\mathbf{w_i}^{\mathbf{h(w_i,w_j)}} = (\mathbf{w_i} + \mathbf{b_j}) \tag{3}$$

	Term	Hypernym(s)	Source
Medical	pulmonary embolism	pulmonary artery finding, trunk arterial embolus, embolism	SnomedCT
Music	Green Day	artist, rock band, band	MusicBrainz

Table 1: Some example hyponym terms and hypernyms extracted from different sources for both domain

$$\mathbf{w_j}^{\mathbf{h(w_i,w_j)}} = (\mathbf{w_j} + \mathbf{b_i}) \tag{4}$$

Essentially, the word representation is dynamic, as every word can have a potentially different final embedding relative to a different hypernym pair. The main idea is that every word translates the base positions of other word co-appearing in a pair, that is, for a hypernym pair $h(w_1, w_2)$, $\mathbf{b_1}$ and $\mathbf{b_2}$ translate $\mathbf{w_2}$ and $\mathbf{w_1}$ respectively, to compute their final embeddings.

In *HyperBox*, hypernym relation h is represented by 2 hyper-rectangles, i.e., boxes, $\mathbf{h^{(1)}}, \mathbf{h^{(2)}} \in \mathbb{R}^d$. Intuitively, this representation defines two regions in $\mathbb{R}^d$, one for hyponym and other for hypernym, such that a fact $h(w_i, w_j)$ holds when the final embeddings of w_i and w_j each appear in their corresponding position box. An example *HyperBox* model is shown in Figure 2 for d=2. Consider three words w_1, w_2, w_3 which are represented as a point, and a hypernymy relation is represented with two boxes $\mathbf{h^{(1)}}$ and $\mathbf{h^{(2)}}$. Every word is translated by the bump embeddings of all other words. For example, $h(w_2, w_3)$ is a true hypernym-hyponym pair in the model, since (i) $\mathbf{w_2}^{\mathbf{h(w_2,w_3)}} = (\mathbf{w_2} + \mathbf{b_3})$ is a point in $\mathbf{h^{(1)}}$ ($\mathbf{w_2}$ appears in the head box), and (ii) $\mathbf{w_3}^{\mathbf{h(w_2,w_3)}} = (\mathbf{w_3} + \mathbf{b_2})$ is a point in $\mathbf{h^{(2)}}$ ($\mathbf{w_3}$ appears in the tail box). Similarly, $h(w_1, w_3)$ is a true hypernym-hyponym pair in the model.

3.2. Scoring Function

We use the scoring function introduced by (Abboud et al., 2020). They define a distance function for evaluating entity positions relative to box positions such that the distance function grows slowly if a point lies inside a box (relative to the center of the box), but grows rapidly if the point is outside the box. This drive points more effectively into their target boxes and ensure they are minimally changed and can remain there once inside. Formally, let $\mathbf{u^{(i)}}, \mathbf{l^{(i)}} \in \mathbb{R}^d$ be the upper and lower boundaries of a box $\mathbf{h^{(i)}}$, respectively. Let $\mathbf{c^{(i)}} = (\mathbf{u^{(i)}} + \mathbf{l^{(i)}})/\mathbf{2}$ its center and $\omega^{\mathbf{(i)}} = (\mathbf{u^{(i)}} - \mathbf{l^{(i)}} + \mathbf{1})$ its width incremented by 1. A point $\mathbf{w_i}$ is inside a box $\mathbf{h^{(i)}}$ if $\mathbf{l^{(i)}} \leq \mathbf{w_i} \leq \mathbf{u^{(i)}}$. The distance function for the given word embeddings relative to a given target box is defined as follows:

$$\mathbf{dist}(\mathbf{w_i}^{\mathbf{h(w_1,w_2)}}, \mathbf{h^{(i)}}) = \\ \begin{cases} |{w_i}^{h(w_1,w_2)} - c^{(i)}| \oslash \omega^{(i)} & if w_i \in h^{(i)} \\ |{w_i}^{h(w_1,w_2)} - c^{(i)}| \circ \omega^{(i)} - \kappa & otherwise \end{cases} \tag{5}$$

where $\kappa = \mathbf{0.5} \circ (\omega^{\mathbf{(i)}} - \mathbf{1}) \circ (\omega^{\mathbf{(i)}} - {\omega^{\mathbf{(i)}}}^{\circ -\mathbf{1}})$, is a width-dependent factor, $\circ$ is element wise multiplication, $\circ - 1$ is element-wise inversion and $\oslash$ is element wise division.

The distance function, $\mathbf{dist}$ factors in the size of the target box in its computation for both the above cases. In the first case, when the point is in its target box, distance inversely correlates with box size, to maintain low distance inside large boxes and provide a gradient to keep points inside. In the second case, box size linearly correlates with distance, to penalize points outside larger boxes more severely. Moreover, κ is subtracted to preserve function continuity. More details about this distance function can be found out in (Abboud et al., 2020). Finally, the scoring function is defined as the sum of the L-2 norms of $\mathbf{dist}$ between both hyponym-hypernym pair and their respective boxes, i.e.:

$$score(h(w_1, w_2)) = \sum_{i=1}^{2} ||\mathbf{dist}(\mathbf{w}_i{}^{h(w_1,w_2)}, \mathbf{h}^{(i)})||_2 \tag{6}$$

3.3. Training Objective

Our next goal is to learn base and bump embeddings for each word as well as projection matrix and box embedding for both boxes. Given a training set of queries and their output, we optimize a negative sampling loss (Mikolov et al., 2013) to effectively optimize a distance-based model (Sun et al., 2019):

$$L = -log\sigma(\gamma - dist(v; q)) - \sum_{i=1}^{k} (1/k) log\sigma(dist(\acute{v}_i; q) - \gamma) \tag{7}$$

where γ represents a fixed scalar margin, $v \in [[q]]$ is a positive entity (i.e., answer to the query q), and $\acute{v}_i \notin [[q]]$ is the i-th negative entity (non-answer to the query q) and k is the number of negative samples.

4. Experiments

4.1. Datasets

We use the SemEval-2018 Task9-Hypernym Discovery dataset (Camacho-Collados et al., 2018) for our

Split	Music	Medical
Trial/Validation	15	15
Training	500	500
Test	500	500

Table 2: Number of terms (hyponyms) for each dataset in trial, training and test sets.

experiments. We use two datasets in the English language corresponding to two specific domains of medical and music. Some example input-output pairs (i.e. hyponym terms and corresponding hypernym lists) are shown in Table 1 for both datasets. Table 1 also reports the sources of hypernymy information beside each pair, which vary depending on the dataset.

Statistics of the two datasets used in the experiments of this paper are summarized in Table 2. The dataset was split equally in the training and testing set, while the trial data provided fewer examples and is to be used as a validation set. It should be noted that each term may be associated with one or (in most cases) more than one hypernym. Therefore, the effective number of hyponym-hypernym pairs for both datasets would be high. For example, the number of hyponym-hypernym pairs in the test gold standard is 4,116 for the medical dataset and 5,233 for the music dataset.

4.2. Evaluation Metrics

To compare the performance of our model with existing models on the SemEval-2018 Task9 dataset (Camacho-Collados et al., 2018), we also use the same set of metrics provided by the organizer of SemEval-2018 Task9. The hypernym discovery task is evaluated as a soft ranking problem. Models were evaluated over the top 15 (at most) hypernyms retrieved for each input term, and their performance was assessed through Information Retrieval metrics.

- **Mean Reciprocal Rank (MRR)** : For a single query, reciprocal rank is 1/rank where rank is the position of the first correct result in a ranked list of outcomes. For multiple queries Q, the MRR is defined as follows:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|} 1/rank_i \tag{8}$$

- **Mean Average Precision (MAP)**: MAP is a widely used metric to measure the performance of models in information retrieval. It is defined as:

$$MAP = \frac{1}{|Q|}\sum_{q\in Q} AP(q) \tag{9}$$

where Q is the number of queries or experimental runs, AP(·) refers to average precision, i.e. an average of the correctness of each individual obtained hypernym from the search space.

- **Precision@k (P@k)**: In addition to MRR and MAP, P@k is used which is defined as the number of correctly retrieved hypernyms at various thresholds (k=1,3,5,15, etc).

$$P@k = \frac{truepositives@k}{(truepositives@k) + (falsepositives@k)} \tag{10}$$

4.3. Experimental Setup

Training for the HyperBox model was conducted on an Intel Xeon CPU with 16 cores and 224 GB RAM. We run *HyperBox* on both the medical and music dataset. Initially, we train our word embeddings on a given raw corpus with an embedding dimension of 300. We use the same raw corpora that were provided by the organizers for the SemEval2018 Task9. For the medical dataset, a combination of abstracts and research papers provided by the MEDLINE (Medical Literature Analysis and Retrieval System) repository, which contains academic documents such as scientific publications and paper abstracts, is used. For the music domain, the raw corpus is a concatenation of several music-specific corpora, i.e., music biographies from Last.fm contained in ELMD 2.0 (Oramas et al., 2016), the music branch from Wikipedia, and a corpus of album customer reviews from Amazon (Oramas et al., 2017). The raw corpora along with training and test data can be downloaded from **https://competitions.codalab.org/competitions/17119**. *HyperBox* is trained using the Adam optimizer, to optimize negative sampling loss. Hyperparameter tuning was conducted over its learning rate, loss margin γ, dimensionality d, and the number of negative examples. We only report final hyperparameter values after hyperparameter tuning. We used an embedding size of 300 for both base and bump embeddings, and also for hypernym boxes (center and offset). Based on the performance on the validation set, we use a learning rate of 0.001, no of negative samples equal to 100, and γ (margin in negative sampling loss) equal to 2.

5. Results

A summary of the results is provided in Tables 3 and 4. We compare the performance of our model with the benchmark methods and models of the team participating in the SemEval 2018 Task9. It is worth noting that we don't use hyperbolic embeddings for comparison. Hyperbolic embeddings (Nickel and Kiela, 2017; Nickel and Kiela, 2018) have been shown to perform well in learning the hierarchical structure as observed in trees. But, to use hyperbolic embeddings we need a graph-structured dataset instead of the raw corpus. (Le et al., 2019) use Hearst Graphs to create such graph-like structure from the raw corpus. But for the given dataset, such a graph will be very sparse with many disconnected components. So in this work, we skip comparison with models using hyperbolic embeddings.

Model	MRR	MAP	P@5
Hypernyms under Siege (Shwartz et al., 2017)	5.01	1.95	2.15
Adapt (Maldonado and Klubička, 2018)	7.46	2.63	2.64
SJTU (Zhang et al., 2018)	9.15	4.71	4.91
vTE (Espinosa-Anke et al., 2016)	39.36	12.99	12.41
300-sparsans (Berend et al., 2018)	46.43	29.54	28.86
CRIM Supervised (Bernier-Colborne and Barrière, 2018)	57.34	39.95	43.00
HyperBox(Our)	**58.15**	**41.39**	**43.13**

Table 3: Results on the Music dataset

Model	MRR	MAP	P@5
Hypernyms under Siege (Shwartz et al., 2017)	2.10	0.91	1.08
Adapt (Maldonado and Klubička, 2018)	20.56	8.13	8.32
SJTU (Zhang et al., 2018)	25.95	11.69	11.69
vTE (Espinosa-Anke et al., 2016)	41.07	18.84	20.71
300-sparsans (Berend et al., 2018)	40.60	20.75	21.43
CRIM supervised (Bernier-Colborne and Barrière, 2018)	37.63	**28.51**	25.63
HyperBox(Our)	**43.71**	27.79	**30.22**

Table 4: Results on the Medical dataset

The Adapt team (Maldonado and Klubička, 2018) uses skip-gram word embeddings for hypernym discovery. They use the traditional word2vec similarity function to discover hypernym from a raw corpus. The SJTU team (Zhang et al., 2018) uses neural term embeddings for hypernym discovery. They use different neural networks like LSTM, CNN, GRU to learn term embeddings to discover hypernym from a raw corpus. The vTe team uses a supervised distributional framework for hypernym discovery which operates at the sense level, by exploiting semantic regularities between hyponyms and hypernyms in embeddings spaces and integrating a domain clustering algorithm (Espinosa-Anke et al., 2016). The 300-sparsans (Berend et al., 2018) team uses a system based on sparse coding and a formal concept hierarchy obtained from word embeddings. The CRIM team (Bernier-Colborne and Barrière, 2018) uses a hybrid approach by combining methods based on unsupervised Hearst patterns and supervised projection learning. We use the supervised model of CRIM (Bernier-Colborne and Barrière, 2018) for a fair comparison with all the existing models. Most of these models use symmetric similarity functions. As a result of this, even if we interchange hyponym-hypernym pairs their symmetric similarity function will output the same score which is undesirable. Unlike this, *HyperBox* doesn't face such a problem because it uses a Box structure and order.

We can see from Tables 3 and 4 that our method *HyperBox* outperforms all existing benchmark models on most of the metrics. The SOTA results are bolded in the table. This is because our *HyperBox* model is able to learn the anti-symmetric and hierarchical relation "hypernymy" very well. (Abboud et al., 2020) showed that the Box embedding model is fully expressive, and is capable of learning symmetry, anti-symmetry, inversion, composition, hierarchy, intersection, and mutual exclusion.

6. Conclusion

HyperBox provides an effective way for solving the problem of Hypernym discovery. Unlike the Hypernym detection task which reduces to a binary classification task, hypernym discovery focuses on retrieving hypernyms from a large text corpus. *HyperBox* encodes words as points and hypernym relation as axis-aligned hyper-rectangles (or boxes) in the d-dimensional euclidean space. Moreover, Box embeddings have been shown to learn antisymmetric and hierarchical relations very well due to their distinctive box structure.

As part of future work, we hope to combine *HyperBox* with existing unsupervised approaches like Hearst patterns to form a hybrid approach to solve the problem of hypernym discovery.

7. Bibliographical References

Abboud, R., Ceylan, İ. İ., Lukasiewicz, T., and Salvatori, T. (2020). Boxe: A box embedding model for knowledge base completion. In *Proceedings of the Thirty-Fourth Annual Conference on Advances in Neural Information Processing Systems (NeurIPS)*.

Baroni, M., Bernardi, R., Do, N.-Q., and Shan, C.-c. (2012). Entailment above the word level in distributional semantics. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 23–32, Avignon, France, April. Association for Computational Linguistics.

Berend, G., Makrai, M., and Földiák, P. (2018). 300-sparsans at SemEval-2018 task 9: Hypernymy as interaction of sparse attributes. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 928–934, New Orleans, Louisiana, June. Association for Computational Linguistics.

Bernier-Colborne, G. and Barrière, C. (2018). CRIM at SemEval-2018 task 9: A hybrid approach to hypernym discovery. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 725–731, New Orleans, Louisiana, June. Association for Computational Linguistics.

Biran, O. and McKeown, K. (2013). Classifying taxonomic relations between pairs of Wikipedia articles. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 788–794, Nagoya, Japan, October. Asian Federation of Natural Language Processing.

Camacho-Collados, J., Delli Bovi, C., Espinosa-Anke, L., Oramas, S., Pasini, T., Santus, E., Shwartz, V., Navigli, R., and Saggion, H. (2018). SemEval-2018 task 9: Hypernym discovery. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 712–724, New Orleans, Louisiana, June. Association for Computational Linguistics.

Chang, H.-S., Wang, Z., Vilnis, L., and McCallum, A. (2018). Distributional inclusion vector embedding for unsupervised hypernymy detection. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 485–495, New Orleans, Louisiana, June. Association for Computational Linguistics.

Dagan, I., Roth, D., Zanzotto, F., and Sammons, M. (2013). *Recognizing Textual Entailment: Models and Applications*. Morgan and Claypool.

Espinosa-Anke, L., Camacho-Collados, J., Delli Bovi, C., and Saggion, H. (2016). Supervised distributional hypernym discovery via domain adaptation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 424–435, Austin, Texas, November. Association for Computational Linguistics.

Fu, R., Guo, J., Qin, B., Che, W., Wang, H., and Liu, T. (2014). Learning semantic hierarchies via word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1199–1209, Baltimore, Maryland, June. Association for Computational Linguistics.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics*.

Le, M., Roller, S., Papaxanthos, L., Kiela, D., and Nickel, M. (2019). Inferring concept hierarchies from text corpora via hyperbolic embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3231–3241, Florence, Italy, July. Association for Computational Linguistics.

Lenci, A. and Benotto, G. (2012). Identifying hypernyms in distributional semantic spaces. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 75–79, Montréal, Canada, 7-8 June. Association for Computational Linguistics.

Levy, O., Remus, S., Biemann, C., and Dagan, I. (2015). Do supervised distributional methods really learn lexical inference relations? In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 970–976, Denver, Colorado, May–June. Association for Computational Linguistics.

Li, X., Vilnis, L., Zhang, D., Boratko, M., and McCallum, A. (2019). Smoothing the geometry of probabilistic box embeddings. In *International Conference on Learning Representations*.

Luu, A. T., Tay, Y., Hui, S. C., and Ng, S. K. (2016). Learning term embeddings for taxonomic relation identification using dynamic weighting neural network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 403–413, Austin, Texas, November. Association for Computational Linguistics.

Maldonado, A. and Klubička, F. (2018). ADAPT at SemEval-2018 task 9: Skip-gram word embeddings for unsupervised hypernym discovery in specialised corpora. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 924–927, New Orleans, Louisiana, June. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space.

Miller, G. A. and Fellbaum, C. (1991). Semantic networks of english. *Cognition*, 41(1):197–229.

Miller, G. A. (1995). Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November.

Navigli, R. and Velardi, P. (2010). Learning word-class lattices for definition and hypernym extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1318–1327, Uppsala, Sweden, July. Association for Computational Linguistics.

Navigli, R., Velardi, P., and Faralli, S. (2011). A graph-based algorithm for inducing lexical taxonomies from scratch. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Three*, IJCAI'11, page 1872–1877. AAAI Press.

Nguyen, K. A., Köper, M., Schulte im Walde, S., and Vu, N. T. (2017). Hierarchical embeddings for hypernymy detection and directionality. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 233–243, Copenhagen, Denmark, September. Association for Computational Linguistics.

Nickel, M. and Kiela, D. (2017). Poincaré embeddings for learning hierarchical representations. In I. Guyon, et al., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Nickel, M. and Kiela, D. (2018). Learning continuous hierarchies in the lorentz model of hyperbolic geometry.

Oramas, S., Anke, L. E., Sordo, M., Saggion, H., and Serra, X. (2016). ELMD: An automatically generated entity linking gold standard dataset in the music domain. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3312–3317, Portorož, Slovenia, May. European Language Resources Association (ELRA).

Oramas, S., Nieto, O., Barbieri, F., and Serra, X. (2017). Multi-label music genre classification from audio, text, and images using deep features.

Pavlick, E. and Pasca, M. (2017). Identifying 1950s american jazz musicians: Fine-grained isa extraction via modifier composition. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL-2017)*, pages 2099–2109, Vancouver, Canada.

Sanchez, I. and Riedel, S. (2017). How well can we predict hypernyms from word embeddings? a dataset-centric analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 401–407, Valencia, Spain, April. Association for Computational Linguistics.

Santus, E., Lenci, A., Lu, Q., and Schulte im Walde, S. (2014). Chasing hypernyms in vector spaces with entropy. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers*, pages 38–42, Gothenburg, Sweden, April. Association for Computational Linguistics.

Shwartz, V., Goldberg, Y., and Dagan, I. (2016). Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2389–2398, Berlin, Germany, August. Association for Computational Linguistics.

Shwartz, V., Santus, E., and Schlechtweg, D. (2017). Hypernyms under siege: Linguistically-motivated artillery for hypernymy detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 65–75, Valencia, Spain, April. Association for Computational Linguistics.

Snow, R., Jurafsky, D., and Ng, A. Y. (2006). Semantic taxonomy induction from heterogenous evidence. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 801–808, Sydney, Australia, July. Association for Computational Linguistics.

Sun, Z., Deng, Z.-H., Nie, J.-Y., and Tang, J. (2019). Rotate: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations*.

Vendrov, I., Kiros, R., Fidler, S., and Urtasun, R. (2016). Order-embeddings of images and language.

Vilnis, L., Li, X., Murty, S., and McCallum, A. (2018). Probabilistic embedding of knowledge graphs with box lattice measures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 263–272, Melbourne, Australia, July. Association for Computational Linguistics.

Weeds, J. and Weir, D. (2003). A general framework for distributional similarity. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 81–88.

Weeds, J., Weir, D., and McCarthy, D. (2004). Characterising measures of lexical distributional similarity. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 1015–1021, Geneva, Switzerland, aug 23–aug 27. COLING.

Weeds, J., Clarke, D., Reffin, J., Weir, D., and Keller, B. (2014). Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Yu, Z., Wang, H., Lin, X., and Wang, M. (2015). Learning term embeddings for hypernymy identification. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 1390–1397. AAAI Press.

Zhang, Z., Li, J., Zhao, H., and Tang, B. (2018). SJTU-NLP at SemEval-2018 task 9: Neural hypernym discovery with term embeddings. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 903–908, New Orleans, Louisiana, June. Association for Computational Linguistics.

8. Language Resource References

Camacho-Collados, J., Delli Bovi, C., Espinosa-Anke, L., Oramas, S., Pasini, T., Santus, E., Shwartz, V., Navigli, R., and Saggion, H. (2018). SemEval-2018 task 9: Hypernym discovery. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 712–724, New Orleans, Louisiana, June. Association for Computational Linguistics.