Inferring Concept Hierarchies
from Text Corpora via Hyperbolic Embeddings

Matt Le[1] and **Stephen Roller**[1] and **Laetitia Papaxanthos**[2]
Douwe Kiela[1] and **Maximilian Nickel**[1]
[1]Facebook AI Research, New York, NY
[2]D-BSSE, ETH Zürich, Switzerland

Abstract

We consider the task of inferring `is-a` relationships from large text corpora. For this purpose, we propose a new method combining hyperbolic embeddings and Hearst patterns. This approach allows us to set appropriate constraints for inferring concept hierarchies from distributional contexts while also being able to predict missing `is-a`-relationships and to correct wrong extractions. Moreover – and in contrast with other methods – the hierarchical nature of hyperbolic space allows us to learn highly efficient representations and to improve the taxonomic consistency of the inferred hierarchies. Experimentally, we show that our approach achieves state-of-the-art performance on several commonly-used benchmarks.

1 Introduction

Concept hierarchies, i.e., systems of `is-a` relationships, are ubiquitous in knowledge representation and reasoning. For instance, understanding `is-a` relationships between concepts is of special interest in many scientific fields as it enables high-level abstractions for reasoning and provides structural insights into the concept space of a domain. A prime example is biology, in which taxonomies have a long history ranging from Linnaeus et al. (1758) up to recent efforts such as the Gene Ontology and the UniProt taxonomy (Ashburner et al., 2000; Gene Ontology Consortium, 2016; Apweiler et al., 2004). Similarly, in medicine, ontologies like MeSH and ICD-10 are used to organize medical concepts such as diseases, drugs, and treatments (Rogers, 1963; Simms, 1992). In artificial intelligence, concept hierarchies provide valuable information for a wide range of tasks such as automated reasoning, few-shot learning, transfer learning, textual entailment, and semantic similarity (Resnik, 1993; Lin, 1998; Berners-Lee et al., 2001; Dagan et al., 2010; Bowman et al., 2015; Zamir et al., 2018). In addition,
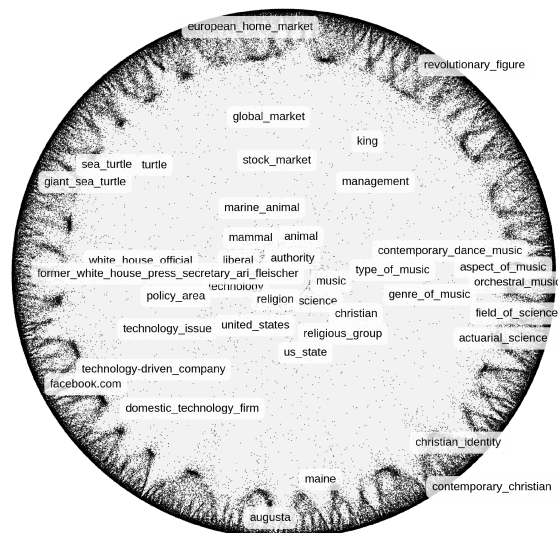

Figure 1: Example of a two-dimensional hyperbolic embedding of the extracted Hearst Graph.

`is-a` relationships are the basis of complex knowledge graphs such as DBPEDIA (Auer et al., 2007) and YAGO (Suchanek et al., 2007; Hoffart et al., 2013) which have found important applications in text understanding and question answering.

Creating and inferring concept hierarchies has, for these reasons, been a long standing task in fields such as natural language processing, the semantic web, and artificial intelligence. Early approaches such as WORDNET (Miller et al., 1990; Miller and Fellbaum, 1998) and CYC (Lenat, 1995) have focused on the manual construction of high-quality ontologies. To increase scalability and coverage, the focus in recent efforts such as PROBASE (Wu et al., 2012) and WEBISADB (Seitner et al., 2016) has shifted towards automated construction.

In this work, we consider the task of inferring concept hierarchies from large text corpora in an *unsupervised* way. For this purpose, we combine Hearst patterns with recently introduced hyperbolic

embeddings (Nickel and Kiela, 2017, 2018) as follows: Roller et al. (2018) showed that Hearst patterns can provide important constraints for hypernymy extraction from distributional contexts. However, it is also well-known that Hearst patterns suffer from missing and incorrect extractions, as words must co-occur in exactly the right pattern to be detected successfully. For this reason, we first extract potential `is-a` relationships from a corpus using Hearst patterns and build a directed weighted graph from these extractions. We then embed this Hearst Graph in hyperbolic space to infer missing hypernymy relations and remove wrong extractions. By using hyperbolic space for the embedding, we can exploit the following important advantages:

Consistency Hyperbolic entailment cones (Ganea et al., 2018) allow us to enforce transitivity of `is-a`-relations in the entire embedding space. This improves the taxonomic consistency of the model, as it enforces that $(x, \texttt{is-a}, z)$ if $(x, \texttt{is-a}, y)$ and $(y, \texttt{is-a}, z)$. To improve optimization properties, we also propose a new method to compute hyperbolic entailment cones in the Lorentz model of hyperbolic space.

Efficiency Hyperbolic space allows for very low dimensional embeddings of graphs with latent hierarchies and heavy-tailed degree distributions. For embedding large Hearst graphs – which exhibit both properties (e.g., see Figure 2) – this is an important advantage. In our experiments, we will show that hyperbolic embeddings allow us to decrease the embedding dimension by over an order of magnitude while outperforming SVD-based methods.

Interpretability In hyperbolic embeddings, similarity is captured via distance while the generality of terms is captured through their norms. This makes it easy to interpret the embeddings with regard to their hierarchical structure and allows us to get additional insights, e.g., about a term's degree of generality.

Figure 1 shows an example of a two-dimensional embedding of the Hearst graph that we use in our experiments. Although we will use higher dimensionalities for our final embedding, the visualization serves as a good illustration of the hierarchical structure that is obtained through the embedding.

2 Related Work

Hypernym detection Detecting `is-a`-relations from text is a long-standing task in natural language processing. A popular approach is to exploit high-precision *lexico-syntactic patterns* as first proposed by Hearst (1992). These patterns may be predefined or learned automatically (Snow et al., 2005; Shwartz et al., 2016; Nakashole et al., 2012). However, it is well known that such pattern-based methods suffer significantly from missing extractions as terms must occur in exactly the right configuration to be detected (Shwartz et al., 2016; Roller et al., 2018). Recent works improve coverage by leveraging search engines (Kozareva and Hovy, 2010) or by exploiting web-scale corpora (Seitner et al., 2016); but also come with precision trade-offs.

To overcome the sparse extractions of pattern-based methods, focus has recently shifted to distributional approaches which provide rich representations of lexical meaning. These methods alleviate the sparsity issue but also require specialized similarity measures to distinguish different lexical relationships. To date, most measures are inspired by the *Distributional Inclusion Hypothesis* (DIH; Geffet and Dagan 2005) which hypothesizes that for a subsumption relation (*cat*, `is-a`, *mammal*) the subordinate term (*cat*) should appear in a subset of the contexts in which the superior term (*mammal*) occurs. Unsupervised methods for hypernymy detection based on distributional approaches include *WeedsPrec* (Weeds et al., 2004), *invCL* (Lenci and Benotto, 2012), *SLQS* (Santus et al., 2014), and *DIVE* (Chang et al., 2018). Distributional representations that are based on positional or dependency-based contexts may also capture crude Hearst-pattern-like features (Levy et al., 2015; Roller and Erk, 2016). Shwartz et al. (2017) showed that such contexts plays an important role for the success of distributional methods. Camacho-Collados et al. (2018) proposed a new shared task for hypernym retrieval from text corpora.

Recently, Roller et al. (2018) performed a systematic study of unsupervised distributional and pattern-based approaches for hypernym detection. Their results showed that pattern-based methods are able to outperform DIH-based methods on several challenging hypernymy benchmarks. Key aspects to good performance were the extraction of patterns from large text corpora and using embedding methods to overcome the sparsity issue. Our work builds on these findings by replacing their
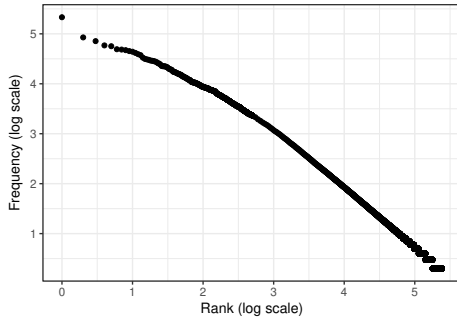
Figure 2: Frequency distribution of words appearing in the Hearst pattern corpus (on a log-log scale).

Pattern
X which is a (example | class | kind | . . .) of Y
X (and | or) (any | some) other Y
X which is called Y
X is `JJS` (most)? Y
X a special case of Y
X is an Y that
X is a !(member | part | given) Y
!(features | properties) Y such as X_1, X_2, . . .
(Unlike | like) (most | all | any | other) Y, X
Y including X_1, X_2, . . .

Table 1: Hearst patterns used in this study. Patterns are lemmatized, but listed as inflected for clarity.

embeddings with ones with a natural hierarchical structure.

Taxonomy induction   Although detecting hypernymy relationships is an important and difficult task, these systems alone do not produce rich taxonomic graph structures (Camacho-Collados, 2017), and complete taxonomy induction may be seen as a parallel and complementary task.

Many works in this area consider a taxonomic graph as the starting point, and consider a variety of methods for growing or discovering areas of the graph. For example, Snow et al. (2006) train a classifier to predict the likelihood of an edge in Word-Net, and suggest new undiscovered edges, while Kozareva and Hovy (2010) propose an algorithm which repeatedly crawls for new edges using a web search engine and an initial seed taxonomy. Cimiano et al. (2005) considered learning ontologies using Formal Concept Analysis. Similar works consider noisy graphs discovered from Hearst patterns, and provide algorithms for pruning edges until a strict hierarchy remains (Velardi et al., 2005; Kozareva and Hovy, 2010; Velardi et al., 2013). Maedche and Staab (2001) proposed a method to learn ontologies in a Semantic Web context.

Embeddings   Recently, works have proposed a variety of graph embedding techniques for representing and recovering hierarchical structure. 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 Poincaré ball and may be viewed as a continuous analogue to tree-like structures (Nickel and Kiela, 2017, 2018). Recently, Tifrea et al. (2018) also proposed an extension of GLOVE (Pennington et al.,

2014) to hyperbolic space. In addition, works have considered how distributional co-occurrences may be used to augment order-embeddings (Li et al., 2018) and hyperbolic embeddings (Dhingra et al., 2018). Further methods have focused on the often complex overlapping structure of word classes, and induced hierarchies using box-lattice structures (Vilnis et al., 2018) and Gaussian word embeddings (Athiwaratkun and Wilson, 2018). Compared to many of the purely graph-based works, these methods generally require supervision of hierarchical structure, and cannot learn taxonomies using only unstructured noisy data.

3   Methods

In the following, we discuss our method for unsupervised learning of concept hierarchies. We first discuss the extraction and construction of the Hearst graph, followed by a description of the Hyperbolic Embeddings.

3.1   Hearst Graph

The main idea introduced by Hearst (1992) is to exploit certain *lexico-syntactic patterns* to detect `is-a` relationships in natural language. For instance, patterns like "NP_y *such as* NP_x" or "NP_x *and other* NP_y" often indicate a hypernymy relationship $(u, \texttt{is-a}, v)$. By treating unique noun phrases as nodes in a large, directed graph, we may construct a *Hearst Graph* using only unstructured text and very limited prior knowledge in the form of patterns. Table 1 lists the only patterns that we use in this work. Formally, let $E = \{(u, v)\}_{i=1}^{N}$ denote the set of `is-a` relationships that have been extracted from a text corpus. Furthermore, let $w(u, v)$ denote how often we have extracted the relationship $(u, \texttt{is-a}, v)$. We then represent the extracted patterns as a weighted directed graph $G = (V, E, w)$

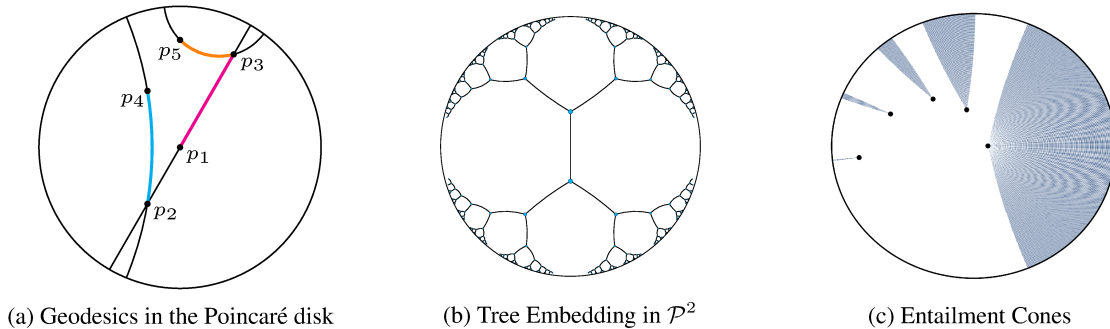(a) Geodesics in the Poincaré disk   (b) Tree Embedding in $\mathcal{P}^2$ (c) Entailment Cones

Figure 3: a) Geodesics in the Poincaré disk model of hyperbolic space. Geodesics between points are arcs that are perpendicular to the boundary of the disk. For curved arcs, midpoints are closer to the origin of the disk (p_1) than the associated points, e.g. (p_3, p_5). c) Entailment cones for different points in $\mathbb{P}^2$.

where V is the set of all extracted terms.

Hearst patterns afford a number of important advantages in terms of data acquisition: they are embarrassingly parallel across both sentences and distinct Hearst patterns, and counts are easily aggregated in any MapReduce setting (Dean and Ghemawat, 2004). Our own experiments, and those of Seitner et al. (2016), demonstrate that this approach can be scaled to large corpora such as COMMONCRAWL.[1] As Roller et al. (2018) showed, pattern matches also provide important contextual constraints which boost signal compared to methods based on the Distributional Inclusion Hypothesis.

However, naïvely using Hearst patterns can easily result in a graph that is extremely sparse: pattern matches naturally follow a long-tailed distribution that is skewed by the occurrence probabilities of constituent words (see Figure 2) and many true relationships are unlikely to ever appear in a corpus (e.g. "long-tailed macaque `is-a` entity"). This may be alleviated with generous, low precision patterns (Seitner et al., 2016), but the resulting graph will contain many false positives, inconsistencies, and cycles. For example, our own Hearst graph contains the cycle: (*area*, *spot*), (*spot*, *commercial*), (*commercial*, *promotion*), (*promotion*, *area*), which is caused by the polysemy of *spot* (location, advertisement) and *area* (location, topic).

3.2 Hyperbolic Embeddings

Roller et al. (2018) showed that low-rank embedding methods, such as Singular Value Decomposition (SVD), alleviate the aforementioned sparsity issues but still produce cyclic and inconsistent predictions. In the following, we will discuss how hyberbolic embeddings allow us to improve consistency via strong hierarchical priors in the geometry.

First, we will briefly review necessary concepts of hyperbolic embeddings: In contrast to Euclidean or Spherical space, there exist multiple equivalent models for hyperbolic space.[2] Since there exist transformations between these models that preserve all geometric properties (including isometry), we can choose whichever is best suited for a given task. In the following, we will first discuss hyperbolic embeddings based on the Poincaré-ball model, which is defined as follows: The Poincaré-ball model is the Riemannian manifold $\mathbb{P}^n = (\mathbb{B}^n, d_p)$, where $\mathbb{B}^n = \{\boldsymbol{u} \in \mathbb{R}^n : \|\boldsymbol{u}\| < 1\}$ is the *open n-dimensional unit ball* and where d_p is the distance function

$$d_p(\boldsymbol{u}, \boldsymbol{v}) = \cosh^{-1}\left(1 + 2\delta(\boldsymbol{u}, \boldsymbol{v})\right) \quad (1)$$

$$\delta(\boldsymbol{u}, \boldsymbol{v}) = \frac{\|\boldsymbol{u} - \boldsymbol{v}\|^2}{(1 - \|\boldsymbol{u}\|^2)(1 - \|\boldsymbol{v}\|^2)}$$

Hyperbolic space has a natural hierarchical structure and, intuitively, can be thought of as a continuous version of trees. This property becomes evident in the Poincaré ball model: it can be seen from Equation 1, that the distance within the Poincaré ball changes smoothly with respect to the norm of a point $\boldsymbol{u}$. Points that are close to the origin are relatively close to all other points in the ball, while points that are close to the boundary are relatively far apart[3]. This locality property of the distance is key for learning continuous embeddings of hierarchical structures and corresponds to the behavior of shortest-paths in trees.
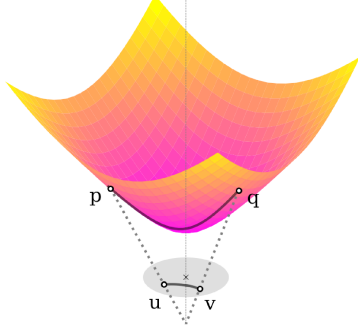

Figure 4: Mapping between $\mathbb{P}$ and $\mathbb{L}$. Points (p, q) lie on the surface of the upper sheet of a two-sheeted hyperboloid. Points (u, v) are the mapping of (p, q) onto the Poincaré disk using Equation 3.

Hyperbolic Entailment Cones Ganea et al. (2018) used the hierarchical properties of hyperbolic space to define entailment cones in an embedding. The main idea of Hyperbolic Entailment Cones (HECs) is to define for each possible point v in the space, an entailment region in the form of a hyperbolic cone $\mathcal{C}_v$. Points u that are located inside a cone $\mathcal{C}_v$ are assumed to be children of v. The width of each cone is determined by the norm $\|v\|$ of the associated base point. The closer v is to the origin, i.e., the more general the base-point is, the larger the width of $\mathcal{C}_v$ becomes and the more points are subsumed in the entailment cone. Figure 3c shows entailment cones for different points in $\mathbb{P}^2$. To model the possible entailment $(u, \texttt{is-a}, v)$, HECs then use the *energy* function

$$\mathcal{E}(u, v) = \max(0, \phi(\boldsymbol{u}, \boldsymbol{v}) - \alpha(\boldsymbol{v})) \quad (2)$$

In Equation (2), $\alpha(\boldsymbol{v})$ denotes the half-aperture of the cone associated with point $\boldsymbol{v}$ and $\phi(\boldsymbol{v}, \boldsymbol{u})$ denotes the angle between the half-lines $(\boldsymbol{v}\boldsymbol{u}$ and $(0\boldsymbol{v}$. If $u \in \mathcal{C}_v$, i.e., if the angle between $\boldsymbol{u}$ and $\boldsymbol{v}$ is smaller than the half aperture of $\mathcal{C}_v$, it holds that $\mathcal{E}(u, v) = 0$. If $u \notin \mathcal{C}_v$, the energy $\mathcal{E}(u, v)$ can then be interpreted as the smallest angle of a rotation bringing $\boldsymbol{u}$ into the cone associated with $\boldsymbol{v}$.

Lorentz Entailment Cones The energy function in Equation (4) is agnostic of the hyperbolic manifold on which the optimization is performed. Ganea et al. (2018) developed hyperbolic entailment cones in the Poincaré-ball model. However, as Nickel and Kiela (2018) pointed out, the Poincaré-ball model is not optimal from an optimization perspective as it is prone to numerical instabilities when points approach the boundary

of the ball. Instead, Nickel and Kiela (2018) proposed to perform optimization in the Lorentz model and use the Poincaré ball only for analysis and visualization. Here, we follow this approach and develop entailment cones in the Lorentz model of hyperbolic space which is defined as follows: let $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^{n+1}$ and let

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle_{\mathbb{L}} = -u_0 v_0 + \sum_{i=1}^{n} u_n v_n$$

denote the *Lorentzian scalar product*. The Lorentz model of n-dimensional hyperbolic space is then the Riemannian manifold $\mathbb{L}^n = (\mathbb{H}^n, d_\ell)$, where

$$\mathbb{H}^n = \{\boldsymbol{u} \in \mathbb{R}^{n+1} : \langle \boldsymbol{u}, \boldsymbol{u} \rangle_{\mathbb{L}} = -1, u_0 > 0\}$$

denotes the upper sheet of a two-sheeted n-dimensional hyperboloid and where the associated distance function on $\mathbb{L}$ is given as

$$d_\ell(\boldsymbol{u}, \boldsymbol{v}) = \cosh^{-1}(-\langle \boldsymbol{u}, \boldsymbol{v} \rangle_{\mathbb{L}})$$

Due to the equivalence of both models, we can define a mapping between both spaces that preserves all properties including isometry. Points in the Lorentz model can be mapped into the Poincaré ball via the diffeomorphism $p : \mathbb{H}^n \to \mathbb{P}^n$, where

$$p(u_0, u_1, \ldots, u_n) = \frac{(u_1, \ldots, u_n)}{u_0 + 1} \quad (3)$$

Furthermore, points in $\mathbb{P}^n$ can be mapped to $\mathbb{L}^n$ via

$$p^{-1}(u_1, \ldots, u_n) = \frac{(1 + \|\boldsymbol{u}\|^2, 2u_1, \ldots, 2u_n)}{1 - \|\boldsymbol{u}\|^2}$$

See also Figure 4 for an illustration of the Lorentz model and its connections to the Poincaré ball.

To define entailment cones in the Lorentz model, it is necessary to derive $\alpha(\boldsymbol{v})$ and $\phi(\boldsymbol{u}, \boldsymbol{v})$ in $\mathbb{L}^n$. Both quantities can be derived easily from the hyperbolic law of cosines and the mapping between $\mathbb{P}^n$ and $\mathbb{L}^n$. In particular, it holds that

$$\alpha(\boldsymbol{v}) = \sin^{-1}\left(2K/(-v_0 - 1)\right)$$

$$\phi(\boldsymbol{u}, \boldsymbol{v}) = \cos^{-1}\left(\frac{v_0 + u_0\langle \boldsymbol{u}, \boldsymbol{v} \rangle_{\mathbb{L}}}{\|\boldsymbol{v}'\|\sqrt{\langle \boldsymbol{u}, \boldsymbol{v} \rangle_{\mathbb{L}}^2 - 1}}\right)$$

Inference Given a Hearst graph $G = (V, E, w)$, we then compute embeddings of all terms in the following way: Let $\boldsymbol{v}_i$ denote the embedding of term i, let $M = |V|$ be the number of extracted terms, and

let $\Theta = \{v_i\}_{i=1}^M$ be the set of embeddings. To find an embedding that minimizes the overall energy, we then solve the optimization problem

$$\hat{\Theta} = \underset{\Theta \in \mathbb{H}^n}{\arg\min} \sum_{u,v \in V} \mathcal{L}(u,v) \qquad (4)$$

where

$$\mathcal{L}(u,v) = \begin{cases} \mathcal{E}(u,v) & \text{if } (u,v) \in E \\ \max(0, \gamma - \mathcal{E}(u,v)) & \text{otherwise} \end{cases}$$

is the max-margin loss as used in (Ganea et al., 2018; Vendrov et al., 2016). The goal of Equation (4) is to find a joint embedding of all terms that best explains the observed Hearst patterns.

To solve Equation (4), we follow Nickel and Kiela (2018) and perform stochastic optimization via Riemannian SGD (RSGD; Bonnabel 2013). In RSGD, updates to the parameters v are computed via

$$v_{t+1} = \exp_{v_t}(-\eta \operatorname{grad} f(v_t)) \qquad (5)$$

where $\operatorname{grad} f(v_t)$ denotes the *Riemannian gradient* and η denotes the learning rate. In Equation 5, the Riemannian gradient of f at v is computed via

$$\operatorname{grad} f(v_t) = \operatorname{proj}_{v_t}\left(g_\ell^{-1} \nabla f(v)\right)$$

where $\nabla f(v)$ denotes the *Euclidean* gradient of f and where

$$\operatorname{proj}_v(x) = v + \langle v, x \rangle_{\mathbb{L}} v$$
$$g_\ell^{-1}(v) = \operatorname{diag}([-1, 1, \ldots, 1])$$

denote the projection from the ambient space $\mathbb{R}^{n+1}$ onto the tangent space $\mathcal{T}_v \mathbb{L}^n$ and the inverse of the metric tensor, respectively. Finally, the exponential map for $\mathbb{L}^n$ is computed via

$$\exp_v(x) = \cosh(\|x\|_{\mathbb{L}})v + \sinh(\|x\|_{\mathbb{L}}) \frac{x}{\|x\|_{\mathbb{L}}}$$

where $\|v\|_{\mathbb{L}} = \sqrt{\langle v, v \rangle_{\mathbb{L}}}$ and $v \in \mathcal{T}_x \mathbb{L}^n$.

As suggested by Nickel and Kiela (2018), we initialize the embeddings close to the origin of $\mathbb{L}^n$ by sampling from the uniform distribution $\mathcal{U}(-0.001, 0.001)$ and by setting v_0 to $\sqrt{1 + ||v'||^2}$, what ensures that the sampled points are located on the surface of the hyperboloid.

4 Experiments

To evaluate the efficacy of our method, we evaluate on several commonly-used hypernymy benchmarks

(as described in (Roller et al., 2018)) as well as in a reconstruction setting (as described in (Nickel and Kiela, 2017)). Following Roller et al. (2018), we compare to the following methods for unsupervised hypernymy detection:

Pattern-Based Models Let $E = \{(x,y)\}_{i=1}^N$ be the set of Hearst patterns in our corpus, $w(x,y)$ be the count of how many times (x,y) occurs in E, and $W = \sum_{(x,y) \in E} w(x,y)$. We then consider the following pattern-based methods:

Count Model (p) This model simply outputs the count, or equivalently, the extraction probabilities of Hearst patterns, i.e.,

$$p(x,y) = \frac{w(x,y)}{W}$$

PPMI Model (ppmi) To correct for skewed occurrence probabilities, the PPMI model predicts hypernymy relations based on the Positive Pointwise Mutual Information over the Hearst pattern corpus. Let $p^-(x) = \Sigma_{(x,y) \in E} w(x,y)/W$ and $p^+(x) = \Sigma_{(y,x) \in E} w(y,x)/W$, then:

$$\text{ppmi}(x,y) = max\left(0, log\frac{p(x,y)}{p^-(x)p^+(y)}\right)$$

SVD Count (sp) To account for missing relations, we also compare against low-rank embeddings of the Hearst corpus using Singular Value Decomposition (SVD). Specifically, let $X \in R^{MxM}$, such that $X_{ij} = w(i,j)/W$ and $U\Sigma V^\top$ be the singular value decomposition of X, then:

$$\text{sp}(x,y) = u_x^\top \Sigma_r v_y$$

SVD PPMI (spmi) We also evaluate against the SVD of the PPMI matrix, which is identical to $\text{sp}(i,j)$, with the exception that $X_{ij} = \text{ppmi}(i,j)$, instead of $p(i,j)$. Roller et al. (2018) showed that this method provides state-of-the-art results for unsupervised hypernymy detection.

Hyperbolic Embeddings (HypeCones) We embed the Hearst graph into hyperbolic space as described in Section 3.2. At evaluation time, we predict the likelihood using the model energy $\mathcal{E}(u,v)$.

Distributional Models The distributional models in our evaluation are based on the DIH, i.e., the idea that contexts in which a narrow term x (ex: *cat*) may appear should be a subset of the contexts in which a broader term y (ex: *animal*) may appear.

	Detection (AP)					Direction (Acc.)			Graded (ρ)
	BLESS	EVAL	LEDS	SHWARTZ	WBLESS	BLESS	WBLESS	BIBLESS	HYPERLEX
Cosine	.12	.29	.71	.31	.53	.00	.54	.52	.14
WeedsPrec	.19	.39	.87	.43	.68	.63	.59	.45	.43
invCL	.18	.37	**.89**	.38	.66	.64	.60	.47	.43
SLQS	.15	.35	.60	.38	.69	.75	.67	.51	.16
p(x, y)	.49	.38	.71	.29	.74	.46	.69	.62	**.62**
ppmi(x, y)	.45	.36	.70	.28	.72	.46	.68	.61	.60
sp(x, y)	.66	.45	.81	.41	.91	**.96**	.84	.80	.51
spmi(x, y)	.76	.48	.84	.44	.96	**.96**	.87	.85	.53
HypeCones	**.81**	**.50**	**.89**	**.50**	**.98**	.94	**.90**	**.87**	.59

Table 2: Experimental results comparing distributional and pattern-based methods in all settings.

WeedsPrec   The first distributional model we consider is *WeedsPrec* (Weeds et al., 2004), which captures the features of x which are *included* in the set of more general term's features, y:

$$\text{WeedsPrec}(x, y) = \frac{\sum_{i=1}^{n} x_i \cdot \mathbb{1}_{y_i > 0}}{\sum_{i=1}^{n} x_i}$$

invCL   Lenci and Benotto (2012), introduce the idea of distributional exclusion by also measuring the degree to which the broader term contains contexts not used by the narrower term. The degree of *inclusion* is denoted as:

$$\text{CL}(x, y) = \frac{\sum_{i=1}^{n} \min(x_i, y_i)}{\sum_{i=1}^{n} x_i}$$

To measure the inclusion of x and y and the non-inclusion of y in x, invCL is then computed as

$$\text{invCL}(x, y) = \sqrt{\text{CL}(x, y) \cdot (1 - \text{CL}(y, x))}$$

SLQS   The SLQS model is based on the informativeness hypothesis (Santus et al., 2014; Shwartz et al., 2017), i.e., the idea that general words appear mostly in uninformative contexts, as measured by entropy. SLQS depends on the median entropy of a term's top k contexts:

$$E_x = \text{median}_{i=1}^{k}[H(c_i)]$$

where $H(c_i)$ is the Shannon entropy of context c_i across all terms. SLQS is then defined as:

$$\text{SLQS}(x, y) = 1 - E_x / E_y$$

Corpora and Preprocessing   We construct our Hearst graph using the same data, patterns, and procedure as described in (Roller et al., 2018): Hearst patterns are extracted from the concatenation of GigaWord and Wikipedia. The corpus is tokenized, lemmatized, and POS-tagged using CoreNLP 3.8.0 (Manning et al., 2014). The full set of Hearst patterns is provided in Table 1. These include prototypical Hearst patterns, like "animals [such as] big cats", as well as broader patterns like "New Year [is the most important] holiday." Noun phrases were allowed to match limited modifiers, and produced additional hits for the head of the noun phrase. The final corpus contains circa 4.5M matched pairs, 431K unique pairs, and 243K unique terms.

Hypernymy Tasks   We consider three distinct subtasks for evaluating the performance of these models for hypernymy prediction:

- *Detection*: Given a pair of words (u, v), determine if v is a hypernym of u.

- *Direction*: Given a pair (u, v), determine if u is more general than v or vise versa.

- *Graded Entailment*: Given a pair of words (u, v), determine the degree to which u is a v.

For detection, we evaluate all models on five commonly-used benchmark datasets: BLESS (Baroni and Lenci, 2011), LEDS (Baroni et al., 2012), EVAL (Santus et al., 2015), SHWARTZ (Shwartz et al., 2016), and WBLESS (Weeds et al., 2014), In addition to positive hypernymy relations, these datasets include negative samples in the form of random pairs, co-hyponymy, antonymy, meronymy, and adjectival relations. For directionality and graded entailment, we also use BIBLESS (Kiela et al., 2015) and HYPERLEX (Vulic et al., 2016). We refer to Roller et al. (2018) for an in-depth discussion of these datasets. For all models, we use the identical text corpus and tune hyperparameters on the validation sets.

	Animals			Plants			Vehicles		
	All	Missing	Transitive	All	Missing	Transitive	All	Missing	Transitive
p(x, y)	350.18	512.28	455.27	271.38	393.98	363.73	43.12	82.57	66.10
ppmi(x, y)	350.47	512.28	455.38	271.40	393.98	363.76	43.20	82.57	66.16
sp(x, y)	56.56	77.10	11.22	43.40	64.70	17.88	9.19	26.98	14.84
spmi(x, y)	58.40	102.56	12.37	40.61	71.81	14.80	9.62	17.96	3.03
HypeCones	**25.33**	**37.60**	**4.37**	**17.00**	**31.53**	**6.36**	**5.12**	**10.28**	**2.74**
Δ%	56.6	51.2	61.1	58.1	51.3	57.0	44.3	42.8	9.6

Table 3: Reconstruction of Animals, Plants, and Vehicles subtrees in WORDNET.

Table 2 shows the results for all tasks. It can be seen that our proposed approach provides substantial gains on the detection and directionality tasks and, overall, achieves state of the art results on seven of nine benchmarks. In addition, our method clearly outperforms other embedding-based approaches on HYPERLEX, although it can not fully match the count-based methods. As Roller et al. (2018) noted, this might be an artifact of the evaluation metric, as count-based methods benefit from their sparse-predictions in this setting.

Our method achieves also strong performance when compared to Poincaré GLOVE on the task of hypernymy prediction. While Tifrea et al. (2018) report Spearman's rho $\rho = 0.421$ on HY-PERLEX and accuracy $ACC = 0.790$ on WBLESS, our method achieves $\rho = 0.59$ (HYPERLEX) and $ACC = 0.909$ (WBLESS). This illustrates the importance of the distributional constraints that are provided by Hearst patterns.

An additional benefit is the efficiency of our embeddings. For all tasks, we have used a 20-dimensional embedding for HYPECONES, while the best results for SVD-based methods have been achieved with 300 dimensions. This reduction in parameters by over an order of magnitude clearly highlights the efficiency of hyperbolic embeddings for representing hierarchical structures.

Reconstruction In the following, we compare embedding and pattern-based methods on the task of reconstructing an entire subtree of WORDNET, i.e., the animals, plants, and vehicles taxonomies, as proposed by Kozareva and Hovy (2010). In addition to predicting the existence of single hypernymy relations, this allows us to evaluate the performance of these models for inferring full taxonomies and to perform an ablation for the prediction of missing and transitive relations. We follow previous work (Bordes et al., 2013; Nickel and Kiela, 2017) and report for each observed

relation (u, v) in WORDNET, its score ranked against the score of the ground-truth negative edges. In Table 3, *All* refers to the ranking of all edges in the subtree, *Missing* to edges that are not included in the Hearst graph G, *Transitive* to missing transitive edges in G (i.e. for all edges $\{(x, z) : (x, y), (y, z) \in E \land (x, z) \notin E\}$).

It can be seen that our method clearly outperforms the SVD and count-based models with a relative improvement of typically over 40% over the best non-hyperbolic model. Furthermore, our ablation shows that HYPECONES improves the consistency of the embedding due to its transitivity property. For instance, in our Hearst Graph the relation (*male horse*, is-a, *equine*) is missing. However, since we correctly model that (*male horse*, is-a, *horse*) and (*horse*, is-a, *equine*), by transitivity, we also infer (*male horse*, is-a, *equine*), which SVD fails to do.

5 Conclusion

In this work, we have proposed a new approach for inferring concept hierarchies from large text corpora. For this purpose, we combine Hearst patterns with hyperbolic embeddings which allows us to set appropriate constraints on the distributional contexts and to improve the consistency in the embedding space. By computing a joint embedding of all terms that best explains the extracted Hearst patterns, we can then exploit these properties for improved hypernymy prediction. The natural hierarchical structure of hyperbolic space allows us also to learn very efficient embeddings that reduce the required dimensionality substantially over SVD-based methods. To improve optimization, we have furthermore proposed a new method to compute entailment cones in the Lorentz model of hyperbolic space. Experimentally, we show that our embeddings achieve state-of-the-art performance on a variety of commonly-used hypernymy benchmarks.

References

Rolf Apweiler, Amos Bairoch, Cathy H Wu, Winona C Barker, Brigitte Boeckmann, Serenella Ferro, Elisabeth Gasteiger, Hongzhan Huang, Rodrigo Lopez, Michele Magrane, et al. 2004. Uniprot: the universal protein knowledgebase. *Nucleic acids research*, 32(suppl_1):D115–D119.

Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. 2000. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25.

Ben Athiwaratkun and Andrew Gordon Wilson. 2018. Hierarchical density order embeddings. In *Proceedings of the International Conference on Learning Representations*.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer.

Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chung-chieh Shan. 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. Association for Computational Linguistics.

Marco Baroni and Alessandro Lenci. 2011. How we BLESSed distributional semantic evaluation. In *Proceedings of the 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 1–10, Edinburgh, UK.

Tim Berners-Lee, James Hendler, and Ora Lassila. 2001. The semantic web. *Scientific american*, 284(5):34–43.

Silvere Bonnabel. 2013. Stochastic gradient descent on Riemannian manifolds. *IEEE Trans. Automat. Contr.*, 58(9):2217–2229.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Jose Camacho-Collados. 2017. Why we have switched from building full-fledged taxonomies to simply detecting hypernymy relations. *arXiv preprint arXiv:1703.04178*.

Jose Camacho-Collados, Claudio Delli Bovi, Luis Espinosa Anke, Sergio Oramas, Tommaso Pasini, Enrico Santus, Vered Shwartz, Roberto Navigli, and Horacio Saggion. 2018. Semeval-2018 task 9: hypernym discovery. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 712–724.

Haw-Shiuan Chang, Ziyun Wang, Luke Vilnis, and Andrew McCallum. 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. Association for Computational Linguistics.

Philipp Cimiano, Andreas Hotho, and Steffen Staab. 2005. Learning concept hierarchies from text corpora using formal concept analysis. *Journal of artificial intelligence research*, 24:305–339.

Ido Dagan, Bill Dolan, Bernardo Magnini, and Dan Roth. 2010. Recognizing textual entailment: Rational, evaluation and approaches–erratum. *Natural Language Engineering*, 16(1):105–105.

Jeffrey Dean and Sanjay Ghemawat. 2004. Mapreduce: Simplified data processing on large clusters. In *OSDI'04: Sixth Symposium on Operating System Design and Implementation*, pages 137–150, San Francisco, CA.

Bhuwan Dhingra, Christopher Shallue, Mohammad Norouzi, Andrew Dai, and George Dahl. 2018. Embedding text in hyperbolic spaces. In *Proceedings of the Twelfth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-12)*, pages 59–69, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. 2018. Hyperbolic entailment cones for learning hierarchical embeddings. *arXiv preprint arXiv:1804.01882*.

Maayan Geffet and Ido Dagan. 2005. The distributional inclusion hypotheses and lexical entailment. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 107–114. Association for Computational Linguistics.

Gene Ontology Consortium. 2016. Expansion of the gene ontology knowledgebase and resources. *Nucleic acids research*, 45(D1):D331–D338.

Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics.

Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, and Gerhard Weikum. 2013. YAGO2: A spatially and temporally enhanced knowledge base from wikipedia. *Artif. Intell.*, 194:28–61.

Douwe Kiela, Laura Rimell, Ivan Vulic, and Stephen Clark. 2015. Exploiting image generality for lexical entailment detection. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015)*, pages 119–124. ACL.

Zornitsa Kozareva and Eduard Hovy. 2010. A semi-supervised method to learn and construct taxonomies using the web. In *Proceedings of the 2010 conference on empirical methods in natural language processing*, pages 1110–1118. Association for Computational Linguistics.

Douglas B. Lenat. 1995. Cyc: a large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38.

Alessandro Lenci and Giulia Benotto. 2012. Identifying hypernyms in distributional semantic spaces. In *Proceedings of 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*, pages 75–79. Association for Computational Linguistics.

Omer Levy, Steffen Remus, Chris Biemann, and Ido Dagan. 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.

Xiang Li, Luke Vilnis, and Andrew McCallum. 2018. Improved representation learning for predicting commonsense ontologies. In *International Conference on Machine Learning Workshop on Deep Structured Prediction*.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the 14th International Conference on Machine Learning*, volume 98, pages 296–304.

Carolus Linnaeus et al. 1758. Systema naturae, vol. 1. *Systema naturae, Vol. 1*.

Alexander Maedche and Steffen Staab. 2001. Ontology learning for the semantic web. *IEEE Intelligent systems*, 16(2):72–79.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

George Miller and Christiane Fellbaum. 1998. Wordnet: An electronic lexical database.

George A Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J Miller. 1990. Introduction to wordnet: An on-line lexical database. *International journal of lexicography*, 3(4):235–244.

Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. Patty: a taxonomy of relational patterns with semantic types. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1135–1145. Association for Computational Linguistics.

Maximilian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems 30*, pages 6338–6347. Curran Associates, Inc.

Maximilian Nickel and Douwe Kiela. 2018. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *Proceedings of the Thirty-fifth International Conference on Machine Learning*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Philip Stuart Resnik. 1993. Selection and information: a class-based approach to lexical relationships. *IRCS Technical Reports Series*, page 200.

FB Rogers. 1963. Medical subject headings. *Bulletin of the Medical Library Association*, 51:114–116.

Stephen Roller and Katrin Erk. 2016. Relations such as hypernymy: Identifying and exploiting hearst patterns in distributional vectors for lexical entailment. *arXiv preprint arXiv:1605.05433*.

Stephen Roller, Douwe Kiela, and Maximilian Nickel. 2018. Hearst patterns revisited: Automatic hypernym detection from large text corpora. *arXiv preprint arXiv:1806.03191*.

Enrico Santus, Alessandro Lenci, Qin Lu, and S Schulte im Walde. 2014. Chasing hypernyms in vector spaces with entropy. In *14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 38–42. EACL (European chapter of the Association for Computational Linguistics).

Enrico Santus, Frances Yung, Alessandro Lenci, and Chu-Ren Huang. 2015. Evalution 1.0: an evolving semantic dataset for training and evaluation of distributional semantic models. In *Proceedings of the 4th Workshop on Linked Data in Linguistics: Resources and Applications*, pages 64–69.

Julian Seitner, Christian Bizer, Kai Eckert, Stefano Faralli, Robert Meusel, Heiko Paulheim, and Simone Paolo Ponzetto. 2016. A large database of hypernymy relations extracted from the web. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. *arXiv preprint arXiv:1603.06076*.

Vered Shwartz, Enrico Santus, and Dominik Schlechtweg. 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. Association for Computational Linguistics.

GO Simms. 1992. *The ICD-10 classification of mental and behavioural disorders: clinical descriptions and diagnostic guidelines*, volume 1. World Health Organization.

Rion Snow, Daniel Jurafsky, and Andrew Y Ng. 2005. Learning syntactic patterns for automatic hypernym discovery. In *Advances in neural information processing systems*, pages 1297–1304.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 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. Association for Computational Linguistics.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 697–706.

Alexandru Tifrea, Gary Bécigneul, and Octavian-Eugen Ganea. 2018. Poincaré glove: Hyperbolic word embeddings. *arXiv preprint arXiv:1810.06546*.

Paola Velardi, Stefano Faralli, and Roberto Navigli. 2013. Ontolearn reloaded: A graph-based algorithm for taxonomy induction. *Computational Linguistics*, 39(3):665–707.

Paola Velardi, Roberto Navigli, Alessandro Cuchiarelli, and R Neri. 2005. Evaluation of ontolearn, a methodology for automatic learning of domain ontologies. *Ontology Learning from Text: Methods, evaluation and applications*, 123(92).

Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2016. Order-embeddings of images and language. In *Proceedings of the International Conference on Learning Representations (ICLR)*, volume abs/1511.06361.

Luke Vilnis, Xiang Li, Shikhar Murty, and Andrew McCallum. 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. Association for Computational Linguistics.

Ivan Vulic, Daniela Gerz, Douwe Kiela, Felix Hill, and Anna Korhonen. 2016. Hyperlex: A large-scale evaluation of graded lexical entailment. *arXiv preprint arXiv:1608.02117*.

Julie Weeds, Daoud Clarke, Jeremy Reffin, David Weir, and Bill Keller. 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 City University and Association for Computational Linguistics.

Julie Weeds, David Weir, and Diana McCarthy. 2004. Characterising measures of lexical distributional similarity. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1015. Association for Computational Linguistics.

Wentao Wu, Hongsong Li, Haixun Wang, and Kenny Q Zhu. 2012. Probase: A probabilistic taxonomy for text understanding. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, pages 481–492. ACM.

Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. 2018. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3712–3722.