Knowledge Completion for Generics using Guided Tensor Factorization

Hanie Sedghi*
Google Brain
Mountain View, CA, U.S.A.
hsedghi@google.com

Ashish Sabharwal
Allen Institute for Artificial Intelligence (AI2)
Seattle, WA, U.S.A.
AshishS@allenai.org

Abstract

Given a knowledge base or KB containing (noisy) facts about common nouns or generics, such as "all trees produce oxygen" or "some animals live in forests", we consider the problem of inferring additional such facts at a precision similar to that of the starting KB. Such KBs capture general knowledge about the world, and are crucial for various applications such as question answering. Different from commonly studied named entity KBs such as Freebase, generics KBs involve quantification, have more complex underlying regularities, tend to be more incomplete, and violate the commonly used locally closed world assumption (LCWA). We show that existing KB completion methods struggle with this new task, and present the first approach that is successful. Our results demonstrate that external information, such as relation schemas and entity taxonomies, if used appropriately, can be a surprisingly powerful tool in this setting. First, our simple yet effective knowledge guided tensor factorization approach achieves state-of-the-art results on two generics KBs (80% precise) for science, doubling their size at 74%-86% precision. Second, our novel taxonomy guided, submodular, active learning method for collecting annotations about rare entities (e.g., oriole, a bird) is 6x more effective at inferring further new facts about them than multiple active learning baselines.

1 Introduction

We consider the problem of completing a partial knowledge base (KB) containing facts about generics or common nouns, represented as a third-order tensor of *(source, relation, target)* triples, such as *(butterfly, pollinate, flower)* and *(thermometer, measure, temperature)*. Such facts capture common knowledge that humans have about the world. They are arguably essential for intelligent agents with human-like conversational abilities as well as for specific applications such as question answering. We demonstrate that state-of-the-art KB completion methods perform poorly when faced with generics, while our strategies for incorporating external knowledge as well as obtaining additional annotations for rare entities provide the first successful solution to this challenging new task.

Since generics represent classes of similar individuals, the truth value y_i of a generics triple $x_i = (s, r, t)$ depends on the quantification semantics one associates with s and t. Indeed, the semantics of generics statements can be ambiguous, even self-contradictory, due to cultural norms. As Leslie (2008) points out, 'ducks lay eggs' is generally considered true while 'ducks are female', which is true for a broader set of ducks than the former statement, is generally considered false.

To avoid deep philosophical issues, we fix a particular mathematical semantics that is especially relevant for noisy facts derived automatically from text: associate s with a categorical quantification from $\{all, some, none\}$ and associate t (implicitly) with $some$. For instance, "all butterflies pollinate (some) flower" and "some animals live in (some) forest". When presenting such triples to humans, they are phrased as: *is it true that all butterflies pollinate some flower?* As a notational shortcut, we treat the quantification of s as the categorical label y_i for the triple x_i. For example, *(butterfly, pollinate, flower)*

* This work was done while the author was affiliated with the Allen Institute for Artificial Intelligence.

Transactions of the Association for Computational Linguistics, vol. 6, pp. 197–210, 2018. Action Editor: Hinrich Schütze.
Submission batch: 6/2017; Revision batch: 9/2017; Published 4/2018.

is labeled all while *(animal, live in, forest)* is labeled $some$. Given a noisy KB of such labeled triples, the task is to infer more triples.

Tensor factorization and graph based methods have both been found to be very effective for expanding knowledge bases, but have focused on *named entity* KBs such as Freebase (Bollacker et al., 2008) involving relations with clear semantics such as *liveIn* and *isACityIn*, and disambiguated entities such as *Barack Obama* or *Hawaii*. Completing KBs that involve facts about generics, however, brings up new challenges, as evidenced by our empirical results when using existing methods.

It has been observed that Horn clauses often reliably connect predicates in the named-entity setting. For instance, for any person *x*, city *y*, and country *z*, *(x, liveIn, y) & (y, isACityIn, z) ⇒ (x, liveIn, z)*. With generics, however, clear patterns or reliable first-order logic rules are rare, in part due to each generic representing a collection of individuals that often have similarities with respect to some relations and differences with respect to others. For instance, *(x, liveIn, mountain)* is true for many *cats* and *caribou*, but there is little tangible similarity between the two animals and it is unclear what, if anything, can be carried over from one to the other. On the other hand, if we take two animals that share a 'parent' in some taxonomy (e.g., *reindeer* and *deer*), then the likelihood of knowledge transfer increases.

We propose to make use of additional rich background knowledge complementing the information present in the KB itself, such as a *taxonomic hierarchy* of entities (available from sources such as WordNet (Miller, 1995)) and the corresponding *entity types* and *relation schema*. Our key insight is that, if used appropriately, *taxonomic and schema information can be surprisingly effective in making tensor factorization methods vastly more effective for generics* for deriving high precision facts.

Intuitively, for generics, many properties of interest are themselves generic (e.g., living in forests, as opposed to living in a specific forest) and tend to be shared by siblings in a taxonomy (e.g., finch, oriole, and hummingbird). In contrast, siblings of named entities (e.g., various people) often differ substantially in the properties we typically care about and model (e.g., who they are married to, where they live, etc.). Methods that use type information are thus more promising for generics than for classical NLP tasks involving named entities. We propose three ways of using this information and empirically demonstrate the effectiveness of each on two variants of a KB of elementary level science facts (Dalvi et al., 2017).[1]

First, we observe that simply imposing **schema consistency** (Section 3.1) on derived facts can significantly boost state-of-the-art methods such as Holographic Embeddings (HolE) (Nickel et al., 2016b) from nearly no new facts at 80% precision to over 10,000 new facts, starting with a generics KB of a similar size. Other embedding methods, such as TransE (Bordes et al., 2013), RESCAL (Nickel et al., 2011), and SICTF (Nimishakavi et al., 2016) (which uses schema information as well), also produced no new facts at 80% precision. Graph-based completion methods did not scale to our densely connected tensors.[2]

Second, one can further boost performance by transferring knowledge up and down the **taxonomic hierarchy**, using the quantification semantics of generics (Section 3.2). We show that expanding the starting tensor this way before applying tensor factorization is complementary and results in a statistically significantly higher precision (86.4% as opposed to 82%) over new facts at the same yield.

Finally, we propose a novel **limited-budget taxonomy guided active learning** method to address the challenge of significant incompleteness in generics KBs, by quantifying uncertainty via siblings (Section 4). Dalvi et al. (2017) have observed that, when using information extraction methods, it is much harder to derive reliable facts about generics than about named entities. This makes generics KBs vastly incomplete, with no or very little information about certain entities such as caribou or oriole.

[1] We are unaware of other large generics KBs. Our method does not employ rules or choices specific to this dataset and is expected to generalize to other generics KBs, as and when they become available.

[2] On the smaller Animals tensor (to be described later), PRA (Lao et al., 2011) generated very few high-precision facts after 30 hours. SFE (Gardner and Mitchell, 2015) was unable to finish training a classifier for any relation after a day, in part due to the high connectivity of generics like *animal*. On the other hand, HolE is trained in a couple of minutes even on the larger Science tensor, and can be made even faster using the method of Hayashi and Shimbo (2017).

Our active learning approach addresses the following question: *Given a new entity*[3] $\tilde{e}$ *and a budget* B*, what is a good set* Q *of* B *queries about* $\tilde{e}$ *to annotate (via humans) such that expanding the original tensor with* Q *helps a KB completion method infer many more high precision facts about* $\tilde{e}$*?*

We propose to define a correlation based measure of the uncertainty of each unannotated triple (i.e., a potential query) involving $\tilde{e}$, based on how frequently the corresponding triple is true for $\tilde{e}$'s siblings in the taxonomic hierarchy (Section 4.1). We then develop a submodular objective function, and a corresponding greedy $(1 - 1/e)$-approximation, to search for a small subset of triples to annotate that optimally balances diversity with coverage (Section 4.2). We demonstrate that annotating this balanced subset makes tensor factorization derive substantially more new and interesting facts compared to several active learning baselines. For example, with a budget to annotate 100 queries about a new entity oriole, random queries lead to no new true facts at all (via annotation followed by tensor factorization), imposing schema consistency results in 83 new facts, and our proposed method ends up with 483 new facts. This demonstrates that well-designed intelligent queries can be substantially more effective in gathering facts about the new entity.

In summary, this work tackles, for the first time, the challenging task of knowledge completion for generics, by imposing consistency with external knowledge. Our efficient sibling-guided active learning approach addresses the paucity of facts about certain entities, successfully inferring a substantial number of new facts about them.

1.1 Related Work

KB completion approaches fall into two main classes: graph-based methods and those employing low-dimensional embeddings via matrix or tensor factorization. The former uses graph traversal techniques to complete the KB, by learning which types of paths or transitions are indicative of which relation between the start and end points (Lao et al., 2011; Gardner and Mitchell, 2015). This class of solutions, unfortunately, does not scale well to our setting (cf. Footnote 2). This appears due, at least in part, to different connectivity characteristics of generics tensors compared to named entity ones such as FB15k (Bordes et al., 2013). Advances in the latter set of methods have led to several embedding-based methods that are highly successful at KB completion for named entities (Nickel et al., 2011; Riedel et al., 2013; Dong et al., 2014; Trouillon et al., 2016; Nickel et al., 2016a). We compare against many of these, including variants of HolE, TransE, and RESCAL.

Recent work on incorporating entity type and relation schema in tensor factorization (Krompaß et al., 2014; Krompaß et al., 2015; Xie et al., 2016b) has focused on factual databases about named entities, which, as discussed earlier, have very different characteristics than generics tensors. Nimishakavi et al. (2016) use entity type information as a matrix in the context of non-negative RESCAL for schema induction on medical research documents. As a byproduct, they complete missing entries in the tensor in a schema-compatible manner. We show that our proposal performs better on generics tensors than their method, SICTF. SICTF, in turn, is meant to be an improvement over the TRESCAL system of Chang et al. (2014), which also incorporates types in RESCAL in a similar manner. Recently, Schütze et al. (2017) proposed a neural model for fine-grained entity typing and for robustly using type information to improve relation extraction, but this is targeted for Freebase style named entities.

For schema-aware discriminative training of embeddings, Xie et al. (2016b) use a flexible ratio of negative samples from both schema consistent and schema inconsistent triples. Their combined ideas, however, do not improve upon vanilla HolE (one of our baselines) on the standard FB15k (Bordes et al., 2013) dataset. They also consider imposing hierarchical types for Freebase, as entities may have different meanings when they have different types—an issue that typically does not apply to generics KBs. Komninos and Manandhar (2017) use type information along with additional textual evidence for knowledge base completion on the FB15k237 dataset. They learn embeddings for types, along with entities and relations, and show that this way of incorporating type information has a (small) contribution towards improving performance. Incorpo-

[3]Unless otherwise stated, we will henceforth use *entity* to refer to a singular common noun that represents a class or group of individuals, such as *animal, hummingbird, forest,* etc.

rating given first order logic rules has been explored for the simpler case of matrix factorization (Rocktaschel et al., 2015; Demeester et al., 2016). Existing first order logic rule extraction methods, however, struggle to find meaningful rules for generics, making this approach not yet viable in our setting.

Xie et al. (2016a) consider inferring facts about a new entity $\tilde{e}$ given a 'description' of that entity. They use Convolutional Neural Networks (CNNs) to encode the description, deriving an embedding for $\tilde{e}$. Such a description in our context would correspond to knowing some factual triples about $\tilde{e}$, which is a restricted version of our active learning setting.

Krishnamurthy and Singh (2013) consider active learning for a particular kind of tensor decomposition, namely CP or Candecomp/Parafac decomposition into a low dimensional space. They start with an *empty* tensor and look for the most informative slices and columns to fill *completely* to achieve optimal sample complexity. Their framework builds upon the *incoherence* assumption on the column space, which does not apply to generics KB.

Hegde and Talukdar (2015) use an entity-centric information extraction (IE) approach for obtaining new facts about entities of interest. Narasimhan et al. (2016) use a reinforcement learning approach to issue search queries to acquire additional evidence for a candidate fact. Both of these works, and others along similar lines, are advanced IE techniques that operate via a search for new documents and extraction of facts from them. This is different from the KB completion task, where the only source of information is the starting KB and possibly some details about the involved entities and relations.

2 Tensors of Generics

We consider knowledge expressed in terms of *(source, relation, target)* triples, abbreviated as (s, r, t). Such a triple may refer to *(subject, predicate, object)* style facts commonly used in information extraction. Each source and target is an *entity* that is a generic noun, e.g., animals, habitats, or food items. Examples of relations include *foundIn*, *eat*, etc. As mentioned earlier, with each generics triple (s, r, t), we associate a categorical truth value $q \in \{all, some, none\}$, defining the quantification semantics "q s r (some) t". For instance, "some animals live in (some) forest" and "all dogs eat (some) bone". *Given a set K of such triples with annotated truth values, the task is to predict additional triples K' that are also likely to be true.*

In addition to a list of triples, we assume access to **background information** in the form of entity types and the corresponding *relation schema*, as well as a *taxonomic hierarchy*.[4] Let E_T denote the set of possible entity types. For each relation r, the relation schema imposes a type constraint on the entities that may appear as its source or target. Specifically, using $[\ell]$ to denote the set $\{1, 2, \ldots, \ell\}$, the *schema* for r is a collection $\mathcal{S}_r = \{(\mathcal{D}_r^{(i)}, \mathcal{R}_r^{(i)}) \subseteq E_T \times E_T \mid i \in [\ell]\}$ of domain-range pairs with the following property: the truth value of (s, r, t) is $none$ whenever for every $i \in [\ell]$ it is the case that $s \notin \mathcal{D}_r^{(i)}$ or $t \notin \mathcal{R}_r^{(i)}$. For example, the relation *foundIn* may be associated with the schema $\mathcal{S}_{foundIn} = \{$*(animal, location), (insect, animal), (plant, habitat),* $\ldots\}$. Similarly, the taxonomic hierarchy defines a partial order $\mathcal{H}$ over all entities that captures the "isa" relation, with direct links such as *isa(dog, mammal)* or *isa(gerbil, rodent)*. We use this information to extract "siblings" of a given entity, i.e., entities that share a common parent (this may be easily generalized to any common ancestor).

3 Guided Knowledge Completion

We begin with an overview of tensor factorization for KB completion for generics. Let (s, r, t) be a generics triple associated with a categorical quantification label $q \in \{all, some, none\}$. For example, *((cat, havePart, whiskers), all), ((cat, liveIn, homes), some),* and *((cat, eat, bear), none)*. Predicting such labels is thus a multi-class classification problem. Given a set K of labeled triples, the goal of tensor factorization is to learn a low-dimensional embedding h for each entity and relation such that some function f of h best captures the given labels. Given a new triple, we can then use f and the learned h to predict the probability of each label for it. K often contains only "positive" triples, i.e., those with label *all* or *some*. A common step in discriminative training for h is thus *negative sampling*, i.e., generating additional triples that (are expected to) have

[4] We do not assume that the schema or taxonomy is perfect, and instead rely on these only for heuristic guidance.

label *none*.

With $[m]$ denoting the set $\{1, 2, \ldots, m\}$ as before, let $K = \{(x_i, y_i), i \in [m]\}$ be a set of triples $x_i = (s_i, r_i, t_i)$ and corresponding labels $y_i \in \{1, 2, 3\}$ equivalent to categorical quantification label $q_i \in \{all, some, none\}$. We learn entity and relation embeddings Θ that minimize the multinomial logistic loss defined as:

$$\begin{aligned}&\min_{\Theta} \sum_{i=1}^{m} \sum_{k=1}^{3} -\mathbb{1}\{y_i = k\} \log \Pr(y_i = k \mid x_i, \Theta) \\ &= \min_{\Theta} \sum_{i=1}^{m} \sum_{k=1}^{3} -\mathbb{1}\{y_i = k\} \log \sigma(y_i \, f(h_r, h_s, h_t))\end{aligned} \tag{1}$$

where $h_r, h_s, h_t \in \mathbb{R}^d$ denote the learned embeddings (latent vectors) for s, r, t, respectively, and $\sigma(\cdot)$ is the sigmoid function defined as $\sigma(z) = \frac{1}{1+\exp(-z)}$.

If the *all* categorical label for generics is unavailable,[5] we can simplify the label space to $\{some, none\}$, modeled as $y_i \in \{\pm 1\}$, and reduce the model to binary classification:

$$\min_{\Theta} \sum_{i=1}^{m} \log\left[1 + \exp\left[-y_i \, f(h_r, h_s, h_t)\right]\right]. \tag{2}$$

We remark that while this generics task with only two labels appears superficially similar to the standard KB completion task for named entities, the underlying challenges and solutions are different. For instance, the approach of using taxonomic information (as opposed to just entity types) as a guide is uniquely suited to generics KBs; the reason being that a generic entity refers to a *set* of individuals, with a natural subset/superset relation forming a taxonomy, whereas in standard KBs an entity refers to one specific individual. This prevents taxonomy based rules from providing useful information for standard KBs, while our results demonstrate their high value when reasoning with generics. Differences like this lead to differences in what is successful in each setting and what is not.

[5]This happens to be the case for current generics KBs, but is expected to change with increasing interest in the research community. A step in this direction is a recent version of the Aristo Tuple KB, http://allenai.org/data/aristo-tuple-kb, which includes *most* as a quantification label, in addition to *some*.

While all our proposed schemes are embedding oblivious, for concreteness, we describe and evaluate them for the **Holographic Embedding** or HolE (Nickel et al., 2016b) which models the label probability as:

$$f(h_r, h_s, h_t) = h_r^{\top}(h_s \circ h_t) \tag{3}$$

where $\circ : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ denotes circular correlation defined as:

$$[a \circ b]_k = \sum_{i=0}^{d-1} a_i b_{(i+k) \mod d}. \tag{4}$$

Intuitively, the k-th dimension of circular correlation captures how related a is to b when the dimensions of the latter are shifted (circularly, via the mod operation) by k. In particular $[a \circ b]_0$ is simply the dot product of a and b. As can be deduced from Eqns. (3)-(4), this model resembles circular convolution, but can capture, to some extent, relations that are asymmetric among the source and target entities. This is because $[a \circ b]$ is not the same as $[b \circ a]$ but is rather "flipped" ($[a \circ b]_k = [b \circ a]_{d-k}$). If we consider the $d \times d$ matrix M_{ab} of element-wise relationships between a and b, the HolE embedding of a relation r between a and b defines a weighted sum of circular anti-diagonals of M_{ab}.

Circular correlation can be computed using the fast Fourier transform (FFT), making HolE quite efficient in practice. Hayashi and Shimbo (2017) recently showed that HolE and complex embeddings (Trouillon et al., 2016), which is another state-of-the-art method for KB completion, are equivalent and differ only in terms of constraints on initial values. Further, they proposed a linear time computation for HolE by staying fully within the frequency domain of FFT.

3.1 Incorporating Types and Relation Schema (ITRS)

As described earlier, relation schema $\mathcal{S}_r$ imposes a restriction on sources and targets that may occur with a relation r. We can incorporate this knowledge both at training and at test times. Doing this at test time simply translates to relabeling schema-inconsistent predicted triples as *none*. Incorporating this knowledge at training time can be done as a constraint on the random negative samples that

the method generates to complement the given, typically positive, triples for training.

In general, the ratio of random negative samples from the entire tensor $\mathcal{T}$ and random negative samples from the schema consistent portion $\mathcal{T}'$ of $\mathcal{T}$ is a parameter that should be tuned such that the resulting negative samples mimic the true distribution of labels. It is worth noting that whether the locally closed world assumption (LCWA) holds or not plays an important role in determining this ratio. However, the idea of mixing the two kinds of negative samples has been used in the literature without considering the nature of the dataset, resulting in some seemingly contradicting empirical results on the optimal ratio (Li et al., 2016; Xie et al., 2016b; Shi and Weninger, 2017; Xie et al., 2017). As discussed later, we found sampling from $\mathcal{T}$ to work best on our datasets.

3.2 Incorporating Entity Taxonomy (IET)

It is challenging to come up with complex Horn or first order logic rules for generics, as each entity represents a class of individuals that may not all behave identically. However, we can derive simple yet highly effective rules based on categorical quantification labels, leveraging the fact that entities come from different levels in a taxonomy hierarchy.

Let p be the parent entity for entity set $\{c_i\}$. Note that c_i itself is a generic, that is, a class of individuals rather than a single individual. This allows one to make meaningful existential statements such as: if a property holds for all or most members of even one class c_i, then it holds for some (reasonable number of) members of its parent class p. We use the following rules:[6]

$$\begin{aligned}((p, r_j, t_j), all) &\Rightarrow \forall i\ \ ((c_i, r_j, t_j), all)\\ \forall i\ \ ((c_i, r_j, t_j), all) &\Rightarrow ((p, e_j, t_j), all)\\ \exists i\ \ ((c_i, r_j, t_j), all) &\Rightarrow ((p, e_j, t_j), some)\\ \exists i\ \ ((c_i, r_j, t_j), some) &\Rightarrow ((p, e_j, t_j), some)\end{aligned}$$

We apply these rules to address sparsity of generics tensors, making tensor factorization more robust. Specifically, given initial triples K, we use applicable rules to derive additional triples K', perform tensor factorization on $K \cup K'$, and then revisit the triples in K' using their predicted label probabilities. Note that this approach allows us to be robust to taxonomic errors: instead of assuming each triple in K' is true, we use this only as a prior and let tensor factorization determine the final prediction based on global patterns it finds.

[6]The last rule may not be appropriate for KBs where *some* may refer to the extreme case of a single individual. This is not the case for the KBs we use for our evaluation.

4 Active Learning for New or Rare Entities

To address the incomplete nature of generics KBs, we consider *rare* entities for which we have very few facts, or *new* entities which are present in the taxonomy but for which we have no facts in the KB. The goal is to use tensor factorization to generate high quality facts about such entities.

For instance, consider the task of inferring facts about *oriole*, where *all we know is that it is a bird*. We assume a restricted budget on the number of facts we can query (for human annotation) about *oriole*, using which we would like to predict many more high-quality facts about it.

Given a fixed query budget B, what is the optimal set of queries we should generate for human annotation about a new or rare entity $\tilde{e}$ for this task? We view this as an *active learning* problem and propose a two-step algorithm. First, we use taxonomy guided uncertainty sampling to propose a list L to potentially query. Next, we describe a submodular objective function and a corresponding linear time algorithm to choose an optimal subset $\widehat{L} \subseteq L$ satisfying $|\widehat{L}| = B$. We then use $\widehat{L}$ for human annotation, append the result to the original KB, and perform tensor factorization to predict additional new facts about $\tilde{e}$. For notational simplicity and without loss of generality, throughout this section, we consider the case where $\tilde{e}$ appears as the source entity in the triple; the ideas apply equally when $\tilde{e}$ appears as the target entity in the triple.

4.1 Knowledge Guided Uncertainty Quantification

We now discuss the active learning and specifically uncertainty sampling method we use to propose a list of triples to query. Uncertainty sampling considers the uncertainty for each possible triple $(\tilde{e}, r_i, e_i)$, defined as how far away from 0.5 the conditional probability is of this fact, given the facts we already

know from the KB (Settles, 2012). The question is how to model this conditional probability. A simple baseline is to consider **Random** queries, i.e., r, e are selected randomly from the list of relations and entities in the tensor, respectively.

To infer information about $\tilde{e}$, we propose the following approximation for the conditional probability of a new fact about $\tilde{e}$ given the KB. Let $\tilde{E}_{\tilde{e}} = \{e \mid corr(\tilde{e}, e) > 0\}$ be the set of entities that are correlated with $\tilde{e}$, $\Omega = \{((e_i, r_i, e'_i), y_i) \mid e_i \in \tilde{E}_{\tilde{e}}\}$ be the set of known facts about such entities, and y_i be the label for the triple (e_i, r_i, e'_i). We have:

$$\Pr(f(h_{r_i}, h_{\tilde{e}}, h_{e'_i})) \simeq \frac{1}{|\Omega|} \sum_{e_i \in \tilde{E}_{\tilde{e}}} corr(\tilde{e}, e_i)\ y_i. \quad (5)$$

However, in practice, we cannot measure $corr(\tilde{e}, e_i)$ for every entry in the KB as we do not have complete information about $\tilde{e}$. One simple idea is to consider that every entity is correlated with $\tilde{e}$: $corr(\tilde{e}, e_i) = 1\ \forall e_i \in E$. We will refer to this as **Schema Consistent** query proposal as this relates to summing over all possible (hence schema consistent) facts.

Since we have access to taxonomy information, we can do a more precise, **Sibling Guided**, approximation.[7] We propose the following approximation for $corr(\tilde{e}, e_i)$ for $e_i \in E$:

$$corr(\tilde{e}, e_i) = \begin{cases} 1 & \text{if } e_i \in \text{sibling}(\tilde{e}) \\ 0 & \text{otherwise} \end{cases} . \quad (6)$$

Eqns. (5) and (6) can be used to infer uncertain triples: if every sibling of $\tilde{e}$ has relationship r with an entity e', we can infer for "free" that this is the case for $\tilde{e}$ as well. On the other hand, when siblings disagree in this respect, there is more uncertainty about $(\tilde{e}, r, e')$ (according to (5) and (6)), making this triple a good candidate to query. In our example of *oriole*, the siblings are the *birds* that exist in the tensor, e.g., *hummingbird, finch, woodpecker, etc*. All of them *(eat, insect)* and hence we infer this for oriole. But there is no agreement on *(appearIn, farm)* and hence this is added to the query list.

[7] One may also define *corr* based on entity similarity in a distributional space. One challenge here is that such similarity generally doesn't preserve types. For example, *dog* may co-occur more often with and thus be "closer" to *bone* or *barking* in a distributional space, than to siblings such as *cat* or other pet animals, which are more helpful in our setting.

Algorithm 1: Active Learning for Query Proposal

input new entity $\tilde{e}$, KB, taxonomy, lower bound κ_M on agreement, lower bound τ_L on uncertainty, upper bound τ_U on uncertainty

1: extract list $S_{\tilde{e}}$ of sibling($\tilde{e}$) using taxonomy
2: for each $e_i \in S_{\tilde{e}}$, add all facts about e_i to Ω
3: **for** $(\tilde{e}, r_i, e'_i) \in \Omega$ **do**
4: use (5)-(6) to estimate $\Pr(f(h_{r_i}, h_{\tilde{e}}, h_{e'_i}))$
5: **if** $p \geq \kappa_M$ **then** add $(\tilde{e}, r_i, e'_i)$ to M
6: **if** $\tau_L \leq p \leq \tau_U$ **then** add $(\tilde{e}, r_i, e'_i)$ to L

output L, M

Algorithm 1 formalizes this process. Setting some upper (τ_U) and lower (τ_L) bounds on the conditional probability (Eqn. (5)) which quantifies the uncertainty, we reach a set $L = \{(\tilde{e}, r_i, e_i), i \in I\}$ of triples to query. Using another high threshold $\kappa_M > \tau_U$, we also infer the set $M = \{(\tilde{e}, r_j, e_j), j \in J\}$ of triples that a large majority of siblings agree upon, and hence $\tilde{e}$ is expected to agree with as well. Triples whose conditional probability estimate is between κ_M and τ_U are considered neither certain enough to include in M nor uncertain enough to justify adding to L for human annotation in hopes of learning from it. Similarly, triples with a conditional probability estimate lower than τ_L are discarded. The output of Algorithm 1 is the list L to query and the list M to add directly to the knowledge base.

4.2 Efficient Subset Selection

Given the list L as above (Algorithm 1), which we can write in short as $L = \{(r_i, e_i), i \in I\}$, the problem is to find the "best" subset $\widehat{L}$. A baseline for such a selection is to choose the top k queries. We will refer to this as **TK** subset selection.

Viewing subset selection as a combinatorial problem, we devise an objective $\mathcal{F}$ that models several natural properties of this subset. We then prove that $\mathcal{F}$ is **submodular**, that is, the marginal gain in $\mathcal{F}(L)$ obtained by adding one more item to L decreases as L grows.[8] Importantly, this implies that there is a simple known greedy algorithm that can efficiently compute a worst-case $(1 - 1/e)$-approximation of

[8] Formally, for $L'' \subseteq L' \subseteq L$ and for $l = (r_l, e_l) \in L \setminus L'$, we have $\mathcal{F}(L'' \cup l) - \mathcal{F}(L'') \geq \mathcal{F}(L' \cup l) - \mathcal{F}(L')$.

the global optimum of $\mathcal{F}$ (Nemhauser et al., 1978). We refer to this as **SM** subset selection.

Since queried samples will eventually be fed into tensor factorization, we would like $\widehat{L}$ to *cover* entities (for the other argument of the triple) and relations as much as possible. In addition, we would like $\widehat{L}$ to be *diverse*, i.e., prioritize relations and entities that are more varied.[9] At the same time, we would also want to minimize redundancy, i.e., avoid choosing relations (entities) that are too similar. Let $\mathcal{F}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}})$ denote our objective, where $R_{\widehat{L}}$, $E_{\widehat{L}}$ is the set of relations and entities in $\widehat{L}$, respectively. We decompose it as:

$$\mathcal{F}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}}) = w_{\mathcal{C}}\mathcal{C}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}}) + w_{\mathcal{D}}\mathcal{D}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}}) - w_{\mathcal{R}}\mathcal{R}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}}) \quad (7)$$

where the terms in RHS correspond to *coverage*, *diversity*, and *redundancy*, respectively, and $w_{\mathcal{C}}, w_{\mathcal{D}}, w_{\mathcal{R}}$ are the corresponding non-negative weights. Next, we propose functional forms for these terms. Note that any function that captures the described properties can be used instead, as long as the objective remains submodular.

Let R and E denote the set of relations and entities in the KB, respectively. The coverage simply captures the fraction of entity and relations that we have included in $\widehat{L}$:

$$\mathcal{C}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}}) = \frac{|R_{\widehat{L}}|}{|R|} + \frac{|E_{\widehat{L}}|}{|E|}.$$

The diversity for $\widehat{L}$ is the sum of the diversity measure of the entities and relations included in the set:

$$\mathcal{D}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}}) = \sum_{(r,e)\in\widehat{L}} [V_r + V_e],$$

$$V_r = \frac{|E_{S_r}| + |E_{T_r}|}{|E|}, \qquad V_e = \frac{|R_e| + |E_{S_e}|}{|R| + |E|}.$$

Here V_r and V_e represent the diversity measure of relation r and entity e, respectively. We use E_{S_r}, E_{T_r} to denote the set of sources and targets that appear for relation r in the KB, R_e as the set of relations in the KB that have e as their target, and E_{S_e} as the set of entities that appear as the first entity when e is the second entity of the triple in the KB. The diversity measure for each relation r is defined as the ratio of the number of entities that appear in the KB as its source or target, over the total number of entities. Similarly, for an entity e, its diversity is defined as the ratio of the number of relations involving e plus the number of source entities that co-occur with e in a relation, over the total number of relations and entities. Note that the diversity measure is an intrinsic characteristic of each entity and relationship, dictated by the KB and independent of the set L, and can thus be computed in advance.

As described above, redundancy is a measure of similarity between relations(entities) in $\widehat{L}$. Tensor factorization yields an embedding for each relation(entity) given the facts they participated in. Therefore, the learned embeddings are one of the best options for capturing similarities. Let h_e (and h_r) denote the learned embedding for entity e (and relation r, resp.). We define

$$\mathcal{R}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}}) = \sum_{r_1,r_2\in\widehat{L}} \|h_{r_1} - h_{r_2}\| + \sum_{e_1,e_2\in\widehat{L}} \|h_{e_1} - h_{e_2}\|.$$

This completes the definition of all pieces of our objective function, $\mathcal{F}$, from Eqn. (7). In Algorithm 2, we present our efficient greedy method to select a subset of L that approximately optimizes $\mathcal{F}$.

Despite being a greedy approach that simply adds the currently most valuable single query to $\widehat{L}$ and

Algorithm 2: Query Subset Selection

input KB, budget B, query list L from Alg. 1.

1: $\forall(r, e) \in L$, compute the diversity measure V_r, V_e
2: $\widehat{L} \leftarrow \emptyset$
3: **for** $j = 1$ to B **do**
4: $\quad\forall l \in L \setminus \widehat{L} : \mathcal{G}(l) = \mathcal{F}(\widehat{L} \cup l) - \mathcal{F}(\widehat{L})$, for $\mathcal{F}$ in (7)
5: $\quad$Select $l^* = \arg\max_{L\setminus\widehat{L}} \mathcal{G}(l)$
6: $\quad$Add l^* to $\widehat{L}$

output $\widehat{L}$

[9] This agrees with the sampling method of Chen et al. (2014) for factorizing coherent *matrices* with missing values, which chooses samples with probability proportional to their local coherence.

repeats, the submodular nature of $\mathcal{F}$, which we will prove shortly, guarantees that Algorithm 2 provides an approximation that, even in the worse case, is no worse than a factor of $1 - 1/e$ from the (unknown) true optimum of $\mathcal{F}$. This is formalized in the following theorem. Since addition preserves submodularity and the weights $w_{\mathcal{C}}, w_{\mathcal{D}}, w_{\mathcal{R}}$ are non-negative, we will show that each of the three terms in $\mathcal{F}$ is submodular.

Theorem 1. *Given a tensor KB, a budget B, and a candidate query list L, the quality $\mathcal{F}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}})$ of the output $\widehat{L}$ of Algorithm 2 is a $(1 - 1/e)$-approximation of the global optimum of $\mathcal{F}$.*

Proof. In order to prove the result, it suffices to show that $\mathcal{F}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}})$ in Equation (7) is submodular (Nemhauser et al., 1978). To this end, we show that for $L'' \subseteq L' \subseteq L$ and for $l = (r_l, e_l) \in L \setminus L'$,

$$\mathcal{F}(L'' \cup l) - \mathcal{F}(L'') \geq \mathcal{F}(L' \cup l) - \mathcal{F}(L').$$

Since addition preserves submodularity and the weights $w_{\mathcal{C}}, w_{\mathcal{D}}, w_{\mathcal{R}}$ are non-negative, it suffices to show that each term in $\mathcal{F}$ is submodular.

First, consider the *coverage* term, $\mathcal{C}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}})$. In order to prove that it is submodular, we verify:

$$\frac{(|R_{L'' \cup l}| - |R_{L''}|)}{|R|} \geq \frac{(|R_{L' \cup l}| - |R_{L'}|)}{|R|},$$
$$\frac{(|E_{L'' \cup l}| - |E_{L''}|)}{|E|} \geq \frac{(|E_{L' \cup l}| - |E_{L'}|)}{|E|}.$$

Note that for the numerators of each of the above lines, the difference can be either $+1$ or 0. Since $L'' \subset L'$, LHS is, by definition, never less than RHS and the inequalities holds.

Next, consider the diversity term, $\mathcal{D}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}})$. The above argument directly applies here as well.

Finally, consider the *redundancy* term. In order to show that $-\mathcal{R}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}})$ is submodular, note that when taking the difference between $\mathcal{R}(L'' \cup l)$ and $\mathcal{R}(L'')$ the terms that correspond to both entities (or both relations) being in L'' cancel out. The same holds for $\mathcal{R}(L' \cup l) - \mathcal{R}(L')$. We thus have:

$$\mathcal{R}(L'' \cup l) - \mathcal{R}(L'') = \sum_{r_l \in l, r_2 \in L''} \|h_{r_1} - h_{r_2}\| + \sum_{e_l \in l, e_2 \in L''} \|h_{e_1} - h_{e_2}\|$$
$$\mathcal{R}(L' \cup l) - \mathcal{R}(L') = \sum_{r_l \in l, r_2 \in L'} \|h_{r_1} - h_{r_2}\| + \sum_{e_l \in l, e_2 \in L'} \|h_{e_1} - h_{e_2}\|.$$

Since $L'' \subseteq L'$ and norms are non-negative,

$$\mathcal{R}(L'' \cup l) - \mathcal{R}(L'') \leq \mathcal{R}(L' \cup l) - \mathcal{R}(L').$$

The reverse inequality holds for the negation of both sides, proving that $-\mathcal{R}(\widehat{L}, R_{\widehat{L}}, E_{\widehat{L}})$ is submodular.

Combining the three items concludes the proof. ■

We will complement this theoretical guarantee in the experiments section (cf. Table 3) by empirically comparing the performance of our query proposal and subset selection methods with baselines.

5 Experiments

We begin with a description of the datasets and the general setup, then evaluate the effectiveness of our guided KB completion approach, and end with an evaluation of our active learning method.[10]

5.1 Dataset and Setup

To assess the quality of our guided KB completion method, we consider the only large existing knowledge bases about generics that we are aware of:

1. A **Science** tensor containing facts about various scientific activities, entities (e.g., animals, instruments, body parts), units, locations, occupations, etc. (Dalvi et al., 2017).[11] This starting tensor has a precision of about 80% and acts as a valuable resource for challenging tasks such as question answering. Our goal is to start with this tensor and infer more scientific facts at a similar or higher level of precision.

2. An **Animals** sub-tensor of the Science tensor, which focuses on facts about animals and also has a similar starting precision. Again, the goal is to infer more facts about animals.

The mainstream approach for KB completion is to focus on entities that are mentioned sufficiently often. For instance, the commonly used FB15K dataset guarantees that every entity appears at least 100 times. As a milder version of this, we focus on the subset of the starting tensors where every entity appears at least 20 times. The resulting statistics of the tensors we use here are shown in Table 1.

[10] Data and code available from the authors.

[11] Aristo Tuple KB v0, http://allenai.org/data/aristo-tuple-kb.

Dataset	# Entities	# Relations	# Triples
Animals	224	129	10,604
Science	1,255	1,513	66,643

Table 1: Datasets, with a 3/1/1 train/validation/test split.

This data, which is the only one we are aware of with generics, does not include $((s, r, t), all)$ style triples. We therefore use the objective function in Eqn. (2) rather than the multi-class one in Eqn. (1). Despite this limitation of the dataset and its superficial similarity to the binary classification task underlying standard (non-generics) KB completion, our results reveal that extending a generics KB is surprisingly difficult for existing methods.

Dalvi et al. (2017) use a pipeline consisting of Open IE (Banko et al., 2007) extractions, aggregation, and clean up via crowd-sourcing to generate the Science tensor. These facts come with a relevant WordNet (Miller, 1995) based taxonomy, entity types (derived from WordNet 'synsets'), and relation schema. Our method capitalizes on this additional information[12] to perform high quality knowledge completion.

Our **evaluation metric** is the accuracy of the top k triples generated by various KB completion methods. We also visualize entire precision-recall curves, where possible. While this metric requires human annotation and is thus more cumbersome than fully-automatic metrics, it is arguably more suitable for evaluating *generative tasks* with a massive output space, such as KB completion. In this setting, evaluation against a relatively small held out test set can be misleading—a method may be highly accurate at generating thousands of valid and useful triples even if it does not necessarily classify specific held out instances accurately. While measures such MAP and MRR have been used in the past to alleviate this, they provide only a partial solution to the inherent difficulty of evaluating generative systems. Annotation-efficient evaluation methods have recently been proposed to address this challenge (Sabharwal and Sedghi, 2017).

[12]In order to limit potential error propagation, we collapse the taxonomy to the top two levels in our experiments.

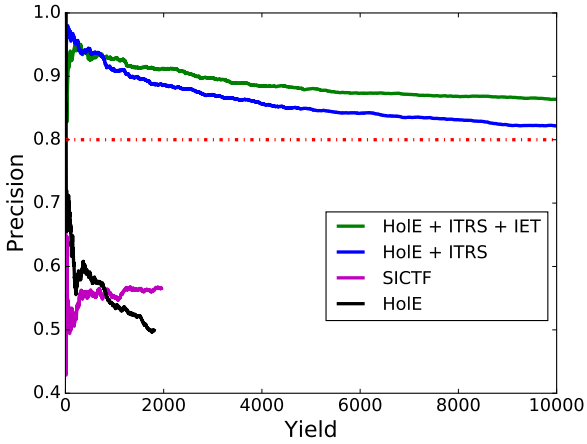

Figure 1: Precision-yield curves for various embedding-based methods on the Animals tensor. State-of-the-art named-entity inspired approaches (black, pink) have low precision even at a low yield. TransE is omitted due to its very low precision here, around 10%. Our method (HolE+ITRS+IET, green) doubles the size of the starting tensor at a precision of 86.4%.

5.2 Guided KB Completion

We first compare our method (Section 3) with existing KB completion techniques on the Animals tensor, and then demonstrate that its effectiveness carries over scalably to the larger Science tensor as well. In what follows, $\mathcal{T}$ denotes the tensor under consideration.

We examine two alternatives for generating negative samples: given a triple $(s, r, t) \in \mathcal{T}$, replace s with (1) any entity s' or (2) an entity s' of the same type as s. The resulting perturbed triple (s', r, t) is then treated as a negative sample if it is not present in $\mathcal{T}$. We also considered a weighted combination of (1) and (2), and found random sampling to be the most reliable on our datasets. This complies with the commonly used LCWA assumption not being applicable to these tensors.

As **baselines**, we consider extensions of three state-of-the-art embedding-based KB completion methods: HolE, TransE, and RESCAL. As mentioned earlier, two leading graph-based methods, SFE and PRA, did not scale well. Both vanilla TransE and RESCAL resulted in poor performance; we thus report numbers only for their extensions. Specifically, we consider 3 baselines: (1) HolE, (2) TransE+Schema, and (3) SICTF which extends

HolE	SICTF	HolE+ITRS+IET
farm,join,farm	penguin,has part,tooth	salmon,thrive in,water
family,join,family	mosquito,spread,parasite	animal,give birth to,animal
tree,resemble,tree	spider,has part,skin	duck,feed in,water
water,is known as,water	elephant,eat,fish	fish,migrate to,water
virus,attract,virus	shark,has part,skin	fish,thrive in,water
animal,resemble,animal	crab,eat,insect	turtle,swim in,water
tree,is known as,tree	snake,eat,fish	salmon,swim in,water
habitat,is known as,habitat	otter,has part,tooth	turtle,live in,water
envment.,is known as,envment.	meat,attract,hummingbird	animal,chew,food
man,join,man	spider,has part,claw	insect,destroy,tree
bird,give birth to,bird	turtle,has part,tooth	farm,possess,horse
region,is known as,region	human,eat,plant	fish,swim in,ocean
virus,derive from,virus	monkey,has part,wing	turtle,feed in,water
food,resemble,food	dolphin,has part,tooth	turtle,float in,water
bird,is known as,bird	carnivore,live in,water	dinosaur,walk on,leg
field,resemble,field	lizard,eat,fish	turtle,migrate to,water
fish,is known as,fish	pelican,has part,tooth	turtle,return to,water
bird,resemble,bird	caterpillar,turn into,bird	man,ride,cattle
grass,graze in,man	bee,pollinate,garden	turtle,swim in,ocean
animal,is known as,animal	virus,infect,bird	fish,float in,ocean

Table 2: Top 20 predictions by various methods, with invalid triples underlined and uninteresting ones , such as (X, is known as, X) or (Y, resembles, Y), shown in *italics*. While some of this assessment can be subjective, it is evident that our method, HolE+ITRS+IET, generates many more triples that are valid and interesting than competing approaches.

RESCAL and incorporates schema.

Figure 1 shows the resulting precision-yield curves for the predictions made by each method on the **Animals** dataset containing 10.6K facts. Specifically, for each method, we rank the predictions based on the method's assigned score and compute the precision of the top k predictions for varying k. As expected, we observe a generally decreasing trend as k increases. TransE+ITRS gave a precision of only around 10% and is omitted from the plot. We make two observations:

First, deriving new facts for these generics tensors at a high precision is challenging! Specifically, none of the baseline methods (black and pink curves), which represent state of the art for named-entity tensors, achieve a yield of more than 10% of $\mathcal{T}$ (i.e., 1K predictions) even at a precision of just 60%.

Second, external information, if used appropriately, can be surprisingly powerful in this setting. Specifically, simply incorporating relation schema (ITRS, blue curve) allows HolE-based completion to double the size of the starting tensor $\mathcal{T}$ by producing over 10K new triples at a precision of 82%. Further, incorporating entity taxonomy (IET, green curve) to address tensor sparsity results in the same yield at a statistically significantly higher precision of 86.4%.

It turns out that not only does our method result in substantially improved PR curves, it also generates **qualitatively more interesting** and useful generic facts about the world than previous methods. We illustrate this in Table 2, which lists the top 20 predictions made by various approaches. The triples shown in red are false predictions (e.g., *(penguin, has part, tooth)*, *(grass, graze in, man)*, *(caterpillar, turn into, bird)*) or uninteresting ones (e.g., *(water, is known as, water)*). As we see, a vast majority of the top 20 predictions made by both vanilla HolE and SICTF fall into these categories. On the other hand, our method, HolE+ITRS+IET, predicts 19 true tripes out of the top 20, including interesting scientific facts that were evidently missing from the starting tensor, such as *(salmon, thrive in, water)*, *(fish, swim in, ocean)* and *(insect, destroy, tree)*.

Finally, we evaluate our proposal on the entire **Science** dataset with 66.6K facts. Since graph-based methods did not scale well to the much smaller Animals dataset and other methods performed substan-

Query Proposal	Subset Selection	# New True Triples Inferred			
		From Anntation	Sibling Argument	Tensor Factorization	Total
Random	-	0	-	0	0
Schema Consistent	TK	73	-	10	83
Schema Consistent	SM	57	-	27	84
Sibling Guided	TK	96	17	211	324
Sibling Guided	**SM**	100	17	366	**483**

Table 3: Active Learning for new entities: Number of new facts inferred (from annotation, sibling agreement, tensor factorization, and in total) for a representative new entity $\tilde{e}$, when querying 100 facts about $\tilde{e}$ for human annotation.

tially worse there, we focus here on the scalability and prediction quality of our method. We found that HolE+ITRS+IET scales well to this high dimension, *doubling the number of facts* by adding 66K new facts at 74% precision. Although the Science tensor is 1,000 times larger than the Animals tensor, the method took only 10x longer to run (3 minutes on Animals tensor vs. 56 minutes on Science tensor, using a 2.8GHz, 16GB Macbook Pro). With additional improvements such as parallelization, it is easily possible to further scale the method up to substantially larger tensors.

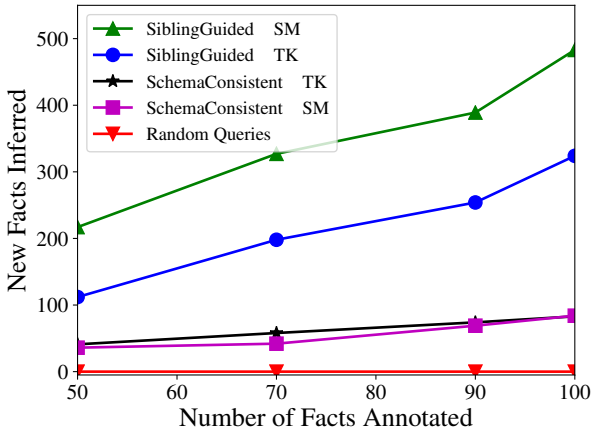

Figure 2: Active Learning for new entities: Total number of new inferred facts (y-axis) for various human annotation query sizes (x-axis). The use of subset selection (green triangles, top) and sibling information (blue circles, 2nd from top) vastly outperforms various baselines.

5.3 Active Learning for New Entities

To assess the quality of our active learning mechanism (Section 4), we consider predicting facts about a new entity $\tilde{e}$ that is not in the Animals tensor. For illustration, we choose $\tilde{e}$ from the Science tensor vocabulary while ensuring that it is present in the WordNet taxonomy.

The setup is as follows. We first use a *query generation mechanism* (Random, Schema Consistent, or Sibling Guided; cf. Section 4.1) to propose an ordered list L of facts about $\tilde{e}$ to annotate. Next, we perform *subset selection* (Top k or TK, Submodular or SM; cf. Section 4.2) on L to identify a subset $\widehat{L}$ of up to 100 most promising queries. These are then annotated and the true ones fed into tensor factorization as additional input to infer further new facts about $\tilde{e}$.

In Table 3, we assess the quality of $\widehat{L}$ in two ways, when $|\widehat{L}| = 100$: how many true facts does $\widehat{L}$ have and how many overall new facts does this annotation produce about $\tilde{e}$. Figure 2 provides a complementary view, focusing on the overall number of new facts inferred as $|\widehat{L}|$ increases. While these illustrative numbers are for a representative new entity, *reindeer*, the overall trend and order of numbers remained the same for other new entities we experimented with.

We mention some highlights from Table 3. First, not surprisingly, randomly choosing triples about $\tilde{e}$ to annotate is ineffective. Second, choosing schema consistent triples results in 73 true triples (out of 100) but these facts help tensor factorization very little, resulting in only 10 additional new triples about $\tilde{e}$. Our proposed sibling guided querying mechanism results not only in nearly all 100 facts being true along with 17 true facts inferred from sibling agreement (set M in Alg. 1), but also, combined with submodular subset selection for balancing diversity with coverage (Alg. 2), ultimately results in 483 new

facts about $\tilde{e}$. These facts cover interesting new information such as *(reindeer, eat, fruit)*, *(wolf, chase, reindeer)*, and *(reindeer, provide, fur)*.

Finally, the plot in Figure 2 demonstrates that the qualitative trends remain the same, irrespective of the number $|\widehat{L}|$ of queries annotated. Overall, our sibling guided queries with submodular subset selection (green triangles, top-most curve) ultimately results in 5.8 times more new facts about $\tilde{e}$ than a non-trivial, uncertainly based, schema consistent baseline (black stars, 3rd curve from the top). This attests to the efficacy of the method on this challenging problem and dataset.

6 Conclusion

This work explores KB completion for a new class of problems, namely completing generics KBs, which is an essential step for including general world knowledge in intelligent machines. The differences between generics and much studied named entity KBs make existing techniques either not scale well or produce facts at an undesirably low precision out of the box. We demonstrate that incorporating entity taxonomy and relation schema appropriately can be highly effective for generics KBs. Further, to address scarcity of facts about certain entities in such KBs, we present a novel active learning approach using sibling guided uncertainty estimation along with submodular subset selection. The proposed techniques substantially outperform various baselines, setting a new state of the art for this challenging class of completion problems.

Our method is applicable to KBs that have an associated entity taxonomy and relation schema. It is expected to be successful when information from siblings can be used to guide what is likely to be true and what is a good candidate to query for a given entity. We focus on KBs of generics where such information is available and—as we show—is highly valuable for effective KB completion.

Why does our use of types work substantially better in our setting than the use of types in various baselines? One hypothesis is the following. The use of complicated models requires substantial data and information. In our KB, the information appears so sparse and incomplete that using types in complicated ways is not productive. Our proposal instead attempts to use type information only to gently enhance the signal and reduce noise, before performing tensor decomposition. We hope this work will trigger further exploration of knowledge bases with generics, a key aspect of machine intelligence.

Acknowledgments

The authors would like to thank Peter Clark for fruitful discussions, valuable feedback, and crowdsourcing annotations; Matt Gardner for constructive comments and assessing graph-based completion methods on our datasets; and Udai Saini and Partha Talukdar for evaluating their CNTF approach on our datasets.

References

Michele Banko, Michael J. Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *IJCAI*, pages 2670–2676.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *ICMD*, pages 1247–1250. ACM.

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

Kai-Wei Chang, Wen-tau Yih, Bishan Yang, and Christopher Meek. 2014. Typed tensor decomposition of knowledge bases for relation extraction. In *EMNLP*, pages 1568–1579.

Yudong Chen, Srinadh Bhojanapalli, Sujay Sanghavi, and Rachel Ward. 2014. Coherent matrix completion. In *ICML*.

Bhavana Dalvi, Niket Tandon, and Peter Clark. 2017. Domain-targeted, high precision knowledge extraction. *TACL*, 5:233–246.

Thomas Demeester, Tim Rocktäschel, and Sebastian Riedel. 2016. Lifted rule injection for relation embeddings. In *EMNLP*, pages 1389–1399.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge Vault: A web-scale approach to probabilistic knowledge fusion. In *KDD*, pages 601–610. ACM.

Matthew Gardner and Tom M. Mitchell. 2015. Efficient and expressive knowledge base completion using subgraph feature extraction. In *EMNLP*, pages 1488–1498.

Katsuhiko Hayashi and Masashi Shimbo. 2017. On the equivalence of holographic and complex embeddings for link prediction. In *ACL*, pages 554–559.

Manjunath Hegde and Partha P. Talukdar. 2015. An entity-centric approach for overcoming knowledge graph sparsity. In *EMNLP*, pages 530–535.

Alexandros Komninos and Suresh Manandhar. 2017. Feature-rich networks for knowledge base completion. In *ACL*, pages 324–329.

Akshay Krishnamurthy and Aarti Singh. 2013. Low-rank matrix and tensor completion via adaptive sampling. In *NIPS*, pages 836–844.

Denis Krompaß, Maximilian Nickel, and Volker Tresp. 2014. Large-scale factorization of type-constrained multi-relational data. In *2014 International Conference on Data Science and Advanced Analytics (DSAA)*, pages 18–24. IEEE.

Denis Krompaß, Stephan Baier, and Volker Tresp. 2015. Type-constrained representation learning in knowledge graphs. In *International Semantic Web Conference*, pages 640–655. Springer.

Ni Lao, Tom Mitchell, and William W Cohen. 2011. Random walk inference and learning in a large scale knowledge base. In *EMNLP*, pages 529–539.

Sarah-Jane Leslie. 2008. Generics: Cognition and acquisition. *Pholosophical Review*, 117(1):1–47.

Xiang Li, Aynaz Taheri, Lifu Tu, and Kevin Gimpel. 2016. Commonsense knowledge base completion. In *ACL*, pages 1445–1455.

George A. Miller. 1995. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41.

Karthik Narasimhan, Adam Yala, and Regina Barzilay. 2016. Improving information extraction by acquiring external evidence with reinforcement learning. In *EMNLP*, pages 2355–2365.

George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. 1978. An analysis of approximations for maximizing submodular set functions - I. *Mathematical Programming*, 14(1):265–294.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *ICML*, pages 809–816.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2016a. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33.

Maximilian Nickel, Lorenzo Rosasco, and Tomaso A. Poggio. 2016b. Holographic embeddings of knowledge graphs. In *AAAI*, pages 1955–1961.

Madhav Nimishakavi, Uday Singh Saini, and Partha Talukdar. 2016. Relation schema induction using tensor factorization with side information. In *EMNLP*, pages 414–423.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *HLT-NAACL*, pages 74–84.

Tim Rocktaschel, Sameer Singh, and Sebastian Riedel. 2015. Injecting logical background knowledge into embeddings for relation extraction. In *NAACL*, pages 1119–1129.

Ashish Sabharwal and Hanie Sedghi. 2017. How good are my predictions? Efficiently approximating precision-recall curves for massive datasets. In *UAI*.

Hinrich Schütze, Yadollah Yaghoobzadeh, and Heike Adel. 2017. Noise mitigation for neural entity typing and relation extraction. In *EACL*, pages 1183–1194.

Burr Settles. 2012. *Active Learning*. Morgan & Claypool.

Baoxu Shi and Tim Weninger. 2017. ProjE: Embedding projection for knowledge graph completion. In *AAAI*, pages 1236–1242.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *ICML*, pages 2071–2080.

Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016a. Representation learning of knowledge graphs with entity descriptions. In *AAAI*, pages 2659–2665.

Ruobing Xie, Zhiyuan Liu, and Maosong Sun. 2016b. Representation learning of knowledge graphs with hierarchical types. In *IJCAI*, pages 2965–2971.

Qizhe Xie, Xuezhe Ma, Zihang Dai, and Eduard H. Hovy. 2017. An interpretable knowledge transfer model for knowledge base completion. In *ACL*, pages 950–962.