Predicting Institution Hierarchies with Set-based Models

Anonymous authors

Abstract

The hierarchical structure of research organizations plays a pivotal role in science-of-science research as well as in tools that track the research achievements and output. However, this structure is not consistently documented for all institutions in the world, motivating the need for automated construction methods. In this paper, we present a new task and model for predicting ancestor/descendant relationships of institutions based on their string names. We present a model that predicts ancestor relationships between the institutions by modeling set operations on the tokens in the institution name strings. The model overall outperforms all non set-based models and baselines on all, but one metric. We also create a dataset for training and evaluating models for this task based on the publicly available relationships in the Global Research Identifier Database.

1. Introduction

Academic, government and various industrial organizations are often hierarchically structured [Fan et al., 2012]. For example, the Canadian Institute for Theoretical Astrophysics is a sub-institution of University of Toronto and Harvard University is the super-institution of both Harvard Medical School and Harvard Divinity School.

The research area of “science-of-science” studies the interaction between various scientific agents (researchers, academic institutions, etc) and aims to develop tools and policies for accelerating science [Fortunato et al., 2018]. Science-of-science research is often carried out at lower levels of institutional hierarchies than at the top-levels. These hierarchies are used in analysis of the funding of institutions and when accompanied with temporal information, organizational hierarchies also provides insights in the evolution / growth of institutions as well as a fine-grained information the mobility of researchers, capturing department and career focus changes [Azoulay et al., 2011, Stephan, 2012]. Thus, correctly modeling organizational hierarchies is critical for the development of some widely-used tools in science-of-science as well as other institutional research.

The Global Research Identifier Database (GRID) is a resource meant to support such research efforts. GRID is manually curated and contains detailed information about research institutions, including their sub-institution/super-institution relationships. While the database is large and growing (containing almost 100k institutions at the time of writing), there are many institutions and rela-
tionships missing. For instance, the Vector Institute, which is a subsidiary of University of Toronto, has no super-institution institution in GRID and the Montreal Institute for Learning Algorithms is missing from the database. Similarly, the database does not contain fine grained information as particular research laboratories or smaller departments within universities’ colleges.

Relation extraction methods are typically used to create knowledge-bases containing facts such as which institutions are parents/children of one another. OpenIE [Etzioni et al., 2011, Fader et al., 2011, inter-alia] and Universal Schema [Riedel et al., 2013, Verga et al., 2016, Das et al., 2017, inter-alia], are able to extract complex relationship between entities from text. However they rely on rich contextual information to make accurate predictions. Unfortunately, such context is often unavailable. For example, funding agencies listed in research papers, assignees mentioned in patents and author/inventor affiliations included in articles and bibtex entries regularly appear as strings with little to no context relevant to making predictions about organization hierarchies. Additionally, one might have an existing knowledge-base of institutions (containing names, and potentially other features), but be missing hierarchical relationships between those institutions.

In this paper, we explore methods for predicting hierarchical relationships between institutions based solely on the spelling of the institution names and knowledge-base features. While some sub-institution/super-institution relationships can be predicted using simple token overlap between institution strings (e.g., Harvard Medical School sub-institution-of Harvard University), others require methods which can perform richer semantic understanding beyond simple token matching (e.g., Centre d’Investigation Clinique Pierre Drouin, Vandœuvre-lès-Nancy, France sub-institution-of Centre Hospitalier Universitaire de Nancy, Nancy, France). This setting of predicting sub-institution/super-institution relationships without additional context is important when trying to complete knowledge-bases such as GRID without intensive data gathering for contextual evidence.

We present a model that can be used to predict institutional sub-relationships using string names, institution locations and types. We develop a model that can satisfy the following desiderata — (a) We notice that word order is not strictly maintained for institution names (e.g. Yale University and Medical School of Yale University) and therefore our model should be able to represent names as “sets” of tokens. Moreover it should be able to represent “set-intersection” and “set-difference” operations between names of institution. (b) For better generalizability, our model should be able to learn representations of sets instead of operating in discrete token space.

This paper makes the following three contributions: (1) presents a new set based model that outperforms non set based models on predicting sub-institution/super-institution relationships (2) examines the difficulty of the task and shows when set based models work well, (3) creates and releases a new dataset from GRID for predicting sub-institution/super-institution relationships.

2. Background

2.1 Dataset & Preprocessing

Global Research Identifier Database (GRID), contains information about 91640 institutions from 217 countries. The institutions include universities, government agencies, companies, health care organizations and other places of research. GRID stores the institution name, location, type (university, hospital, government agency, etc), as well as parent institutions. GRID includes 11393 instances of
Figure 1: **GRID Institution Hierarchy**. A connected component in the GRID institution graph. Tokens highlighted in blue are members of the intersection of the sub-institution and super-institution strings. Tokens highlighted in red are not members of this intersection, however they are semantically conveyed in both the sub-institution and super-institution strings. Notice the GRID institution graph exhibits some moderately complex structure.

the sub-institution relationship between institutions. As our goal is to develop models that can predict the hierarchical structure of institutions, we would like to partition the relationship edges in GRID into training/development/test data. We do so by finding connected components in the institution hierarchy and randomly assign each connected components to a partition (using a 0.6/0.2/0.2 split). We show an example of this data in Figure 1.

To construct the GRID dataset, we extract the parent-child relationships from a set of institutions. All the parent-child related institutions are linked together to form a graph, which consists of a set of connected components where each component contains all the institutions related to each other. From this graph, we extract sub-institution relationships to get positive candidates and construct negative candidates in three ways: non sub-institution institutions in the same connected component, institutions in the same city in a different connected component, and randomly sample negatives.

### 2.2 Institution Hierarchy Prediction

We say that institution $D$ is a sub-institution (sub-institution) of institution $A$ if $A$ is an ancestor of $D$ according to the child-parent facts stored in GRID. sub-institution can be computed from the transitive closure of these child-parent facts. In our task, we are given the knowledge-base entry in GRID for each institution, which contains the name, location and type of the entity, and asked to predict if the sub-institution relationships exists between the two.

### 3. Set-based Models for Predicting Institution Hierarchies

In this section, we present a model for predicting whether one institution is a sub-institution of another. We notate this relation by $A \leftarrow B$, i.e., $A$ is a sub-institution of $B$. Our model makes use of the set-based model, Set-Transformers [Lee et al., 2018], to model the token overlap between the institution names.
Figure 2: **Set-based predictor.** Given two institution names represented as sets of tokens, the predictor embeds the sets, their intersection, and set difference. It uses these embeddings to compute a score representing the likelihood that the first institution is a sub-institution of the second.

### 3.1 Sets of Institution Tokens

We model the strings as sets of tokens since the intersection between the tokens does not depend on the order. For example, *National Natural Science Foundation of China* sub-institution *China Center of Advanced Science and Technology*, and the intersection *{China, Science}* can only be found if order doesn’t matter. In addition, it is more efficient to find the similarities and differences between the strings by viewing them as sets.

We observe that intersection between the tokens that make up two institution names is a strong indicator of whether one institution is a sub-institution of another. Consider the pair of institutions *University of Pennsylvania* and *The Wharton School of the University of Pennsylvania*. If we treat the institutions as sets of tokens, then their intersection—*{University, of, Pennsylvania}*—is telling of the pair’s relationship. In particular, when comparing the intersection to the original institution names, it is possible to arrive at the following heuristic: if for two string *s* and *s′*, the tokens of *s* all appear in *s′*, i.e., *s ∩ s′ = s*, then *s* is likely a super-institution of *s′*, i.e., *s′ ←− s*.

We present a model inspired by heuristics like this one that is designed to predict whether one institution is a sub-institution of another. The model takes an ordered pair of institution names as input, (*s*₁, *s*₂), and outputs a score: *h*(*s*₁, *s*₂) = *o*, with larger scores corresponding to greater model confidence in *s*₁ ←− *s*₂. The model is a function of four sets, each of which can be built from the two institution names: 1. the first institution represented as a set of tokens, 2. the second institution represented as a set of tokens, 3. the intersection between the first and second institution, and 4. the set difference between the second institution and the first, i.e., *s*₂ \ *s*₁.

More formally, let *t*ₙ be the tokens of a string *s*, and let *f* : *2*¹⁺ → *R*ᵈ be a function that returns the embedded representation of a token set, where *V* is the vocabulary defined by our model. Define the model, *h*, evaluated on two string, *s*₁ and *s*₂, as follows:

\[
h(s_1, s_2) = \langle f(t_{s_2}), f(t_{s_2} \cap t_{s_1}) \rangle - \langle f(t_{s_1}), f(t_{s_2} \cap t_{s_1}) \rangle - \langle f(t_{s_2}), f(t_{s_2} \setminus t_{s_1}) \rangle
\]  

(1)
where \(\langle \cdot, \cdot \rangle\) is the inner product of the two vectors.

As in the example in the beginning of this subsection, the first term in the model compares (via dot product) \(s_2\), the candidate super-institution, to intersection of the two institutions. As previously mentioned, if all tokens of \(s_2\) appear in \(t_{s_1} \cap t_{s_2}\), then it is likely that \(s_1 \leftarrow s_2\). Similarly, if the embedded representation of \(s_2\), i.e., \(f(t_2)\) is similar to the embedding of the intersection of the tokens of \(s_1\) and \(s_2\), i.e., \(f(t_{s_1} \cap t_{s_2})\), then \(s_2\) is likely a super-institution of \(s_1\). The second term in calculating the model score is closely related to the first. Since children institutions tend to exhibit a higher degree of specificity than their parents, if \(s_1 \leftarrow s_2\), then \(s_1\) should be different (i.e., low dot product) from its intersection with \(s_2\). This accounts for our choice to subtract the second term from the first. The first two terms used to compute the model score are based on similarities amongst the two institutions. If the two institutions are relatively different, term 1 might be small, but term 2 might be smaller, yielding a positive model score. To account for this, we add a third term of the model. The third term in the model compares the tokens in \(s_2\) (and potentially in the intersection) to the tokens which are unique to \(s_2\). If there is large overlap between these sets, it indicates that \(s_2\) is sufficiently different from \(s_1\), i.e., \(s_1\) is unlike to be a sub-institution of \(s_2\).

Figure 2 visualizes our model. Any set encoder function may be used for \(f\). The choice of this function is discussed further in Section 3.2.

### 3.1.1 Training Objective

We train our model in ranking objective. Given a triple of institutions \((D, A_{pos}, A_{neg})\) such that \(D\) sub-institution \(A_{pos}\) and \(D\) sub-institution \(A_{neg}\) is not true, we use the Bayesian Personalized Ranking objective as our loss function \(\sigma(h(D, A_{pos}) - h(D, A_{neg}))\) [Rendle et al., 2009].

### 3.2 Set Encoders

Our model relies on the function, \(f\), to provide a meaningful representations of sets of institution name tokens. The most performant set representations are those that obey permutation invariance [Zaheer et al., 2017]. Recently, Lee et al. [2018] present a state-of-the-art permutation invariant model based on transformers [Vaswani et al., 2017] without position embeddings. The transformer consists of layers of multi-head attention followed by feed forward networks. The equation for the feed forward network is \(\max(0, xW_1 + b_1)W_2 + b_2\) and the equation for mutli-head attention is

\[
\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_h)W_O
\]

where \(\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)\)

and \(\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{dk}})V\)

Given a set \(s\), which are either the tokens in one of the string or the result of the set intersection or set difference of the tokens in both strings, we pass the tokens in \(s\) into a transformer without position embeddings, since the tokens have no notion of order after applying set operations to them. The set representation is the average of the final representation after the last layer in the transformer for each token in the set.
3.3 Institution Features

In addition to the name, GRID contains the city, state, country, and type (i.e. Education, Government, Nonprofit) for each institution. To train our model, we apply the SetTransformer \( h \) to each feature and get a score for each feature. Then, the final score is learned as a weighted sum of the scores for each feature. Note that each feature has different parameters for the SetTransformer \( h \). The name, and city have a token vocabulary while the state, country and type have a character vocabulary.

Though we are training on additional metadata that the knowledge base contains, and our model applies to string where we assume a surface form of institutions, this is still a reasonable assumption since most of the metadata we train on like city, state, and country and usually included with the institution, especially in science to science research.

3.4 Learning the Weighted Coefficients

Note that the model \( h \) (Eq. 1) can be rewritten as

\[
h(s_1, s_2) = \langle f(t_{s_2}), f(t_{s_2} \cap t_{s_1}) \rangle - \langle f(t_{s_1}), f(t_{s_2} \cap t_{s_1}) \rangle - \langle f(t_{s_2}), f(t_{s_2} \setminus t_{s_1}) \rangle
\]

\[
= \langle [1 \ -1 \ -1], \ [\langle f(t_{s_2}), f(t_{s_2} \cap t_{s_1}) \rangle \ \langle f(t_{s_1}), f(t_{s_2} \cap t_{s_1}) \rangle \ \langle f(t_{s_2}), f(t_{s_2} \setminus t_{s_1}) \rangle \rangle
\]

To determine how important the specific combination of terms for our model \( h \) is, we replace the vector \([1 \ -1 \ -1]\) with a learnable weight vector, initialized randomly and initialized with our proposed model. This only represents a linear combination of the values

\[
[\langle f(t_{s_2}), f(t_{s_2} \cap t_{s_1}) \rangle \ \langle f(t_{s_1}), f(t_{s_2} \cap t_{s_1}) \rangle \ \langle f(t_{s_2}), f(t_{s_2} \setminus t_{s_1}) \rangle]
\]

4. Experiments

We evaluate the ability of SETTRANSFORMERS to predict sub-institution relationships between institutions in a retrieval-based task on the GRiD dataset. Given the name of an institution (query), we rank the candidate parent institution name strings. We evaluate the models performance in terms of mean average precision (MAP) and hits at K for K = 1, 5, 50, and 100. The descendant institutions are randomly assigned to be queries in a train/dev/test split. We compare our method to three kinds of baseline approaches - (1) simple token overlap model, (2) sequence order dependent representations of strings, and (3) set encoders other than transformers:

- **TokenSimilarity (TokSim)** - Rank candidate pairs by \( \frac{|t_{par} \cap t_{ch}|}{|t_{par}|} \)

- **OrderedLSTM (O-LSTM)** - Our order-based model with each string is encoded as the last hidden state representations encoded by an LSTM.

- **OrderedTransformer (O-Trans)** - Our order-based model with each string encoded by average of the tokens encoded by a transformer.

- **SetAverageEmbedding (S-AvgEmb)** - Our set-based model with a simple average of embeddings for the function \( f \).

- **SetTransformer (S-Trans)** - The proposed approach of this paper, our set-based model with a set transformer [Lee et al., 2018] for the function \( f \).
Table 1: **Comparison between SETTRANSFORMERS and ORDEREDLSTM.** This table presents examples for which SETTRANSFORMERS makes a correct prediction and ORDEREDLSTM makes an incorrect prediction. We observe that SETTRANSFORMERS correctly utilizes location information.

<table>
<thead>
<tr>
<th>Descendant Entity</th>
<th>Prediction by SETTRANSFORMERS</th>
<th>Prediction by ORDEREDLSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Northampton General Hospital, Northampton, England, United Kingdom</td>
<td>Northampton General Hospital NHS Trust, Northampton, United Kingdom</td>
<td>Northamptonshire Healthcare NHS Foundation Trust, Kettering, United Kingdom</td>
</tr>
<tr>
<td>NIHR Leeds Musculoskeletal Biomedical Research Unit, Leeds, United Kingdom</td>
<td>Leeds Teaching Hospitals NHS Trust, Leeds, United Kingdom</td>
<td>National Institute for Health Research Leeds, United Kingdom</td>
</tr>
</tbody>
</table>

Table 2: **Comparison between SETTRANSFORMERS and TOKENSIM.** SETTRANSFORMERS makes a correct predictions in these examples and TOKENSIM makes incorrect predictions. We observe that SETTRANSFORMERS seems to identify salient tokens to correctly make these predictions.

<table>
<thead>
<tr>
<th>Descendant Entity</th>
<th>Prediction by SETTRANSFORMERS</th>
<th>Prediction by TOKENSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>The National Institute for Strategic Studies, Kiev, Ukraine</td>
<td>National Academy of Sciences of Ukraine, Kiev, Ukraine</td>
<td>Ukrainian Institute of Public Health Policy, Kiev, Ukraine</td>
</tr>
<tr>
<td>NIHR Oxford Musculoskeletal Biomedical Research Unit, Oxford, United Kingdom</td>
<td>Oxford University Hospitals NHS Trust, Oxford, United Kingdom</td>
<td>Medical Diagnostics (United Kingdom), Yarnton, United Kingdom</td>
</tr>
</tbody>
</table>

Table 3: **Challenging Examples.** These examples highlight the difficulty of predicting sub-institutions. There are examples with very little textual overlap and examples with drastically different geographic locations.

<table>
<thead>
<tr>
<th>Descendant Entity</th>
<th>Ancestor Entity</th>
</tr>
</thead>
<tbody>
<tr>
<td>London Chest Hospital, London, United Kingdom Pharmacy and Poisons Board</td>
<td>Barts Health NHS Trust, London, United Kingdom Ministry of Health</td>
</tr>
<tr>
<td>BASF (Canada), Mississauga, Ontario, Canada</td>
<td>BASF (Germany), Ludwigshafen am Rhein, Germany</td>
</tr>
<tr>
<td>BASF (China), Shanghai, China</td>
<td>BASF (Germany), Ludwigshafen am Rhein, Germany</td>
</tr>
</tbody>
</table>
The order dependent models, O-LSTM and O-Trans use a scoring function of \( \langle f(t_s^2), f(t_s^1) \rangle \).

We also experiment with several different ways to combine the three terms of our set-based scoring function:

\[
\left[ \langle f(t_s^2), f(t_s^2 \cap t_s^1) \rangle, \langle f(t_s^1), f(t_s^2 \cap t_s^1) \rangle, \langle f(t_s^2), f(t_s^2 \setminus t_s^1) \rangle \right]
\]

This ablation allows us to compare the proposed linear combination of these terms in \( h \) (Eq. 1) to alternative scoring functions. First, we have **Original (Orig)**, which is has the scoring function \( h \) described above. Second, we have **Linear Combination (LC)**, a learned linear combination of the values. Third, we have **Linear Combination Initialized (LCI)**, a learned linear combination of the values initialized as the scoring function \( h \) described above. Finally, we compare against **Multilayer Perceptron (MLP)**, a one layer multi-layer perceptron.

Finally, we also experiment with using a token vocabulary (**Unig**) for every feature (including the state, country, and type), rather than just the institution name and city as in our model.

For each model, we report the mean and standard deviation of the score across 5 different random seeds. We bold the model with the highest mean score, and any model whose mean is within 1 standard deviation. Table 4 shows the result for this experiment. We also consider the performance using a subset of the features present in grid. First, we use only the string spelling of the institution name. The results are shown in table 6. Next, we consider using both the institution name and the location (city, state, and country) if available. The results are in table 5.

We observe that in general using all the features (the institution name, location, and type) increases the performance of all the models. The exception is for SetTransformers for Hits at 50 and Hits at 100, where using just the institution name and location is better than adding the type.

---

1. URL withheld to preserve anonymity
2. Hyperparameter search and configurations released in the code
and T
nearest
and outperforms the baseline. Note that the

\[ \cap \]

We measure the MAP performance on queries with varying level

Performance By Token Overlap

Table 4: Score using Institution and Location (city, state, country) and Type features

Table 5: Score using Institution and Location (city, state, country) features Only

4.1 Analysis

Performance By Token Overlap

We measure the MAP performance on queries with varying level of
token overlap with the candidates considered. Specifically, we filter examples based on the ratio of the number of tokens in the query \( \cap \) candidate to the number of tokens in the candidate. We observe that SETTRANSFORMERS achieves high accuracy when the input pair have many overlapping tokens and outperforms the baseline. Note that the \% of candidate tokens in query tokens is rounded to the

nearest .1, so 0% overlap could mean up to 5% overlap.

While the SETTRANSFORMERS outperforms ORDEREDLSTM and TOKENSIM for all levels of
token overlap, SETTRANSFORMERS offers its most significant improvements over ORDEREDLSTM
and TOKENSIM when there are fewer overlapping tokens between the input pair, except for when
there are no overlapping tokens. This shows that the set encoder models are capturing additional information than simple token overlap.

**Examples** Tables 1, 2, show example predictions for SETTRANSFORMERS as well as baseline approaches. Table 1 compares SETTRANSFORMERS’s predictions to ORDEREDLSTM. We notice that the set-based model captures the distinction of Northampton and Northamptonshire being indeed different tokens. SETTRANSFORMERS also seems to accurately capture the distinction between hospitals and healthcare agencies, but understands the importance of Leeds. Table 2 shows that SETTRANSFORMERS is able to better capture token importance as compared to the simple overlap model.

Table 3 show examples that we believe are quite difficult. These examples seem to indicate cases where external data or contextual features would be needed to determine the descendant / ancestor relationships. For example, one cannot know Pharmacy and Poisons Board is a sub-institution of Ministry of Health since there is no token overlap unless one knows the semantics of the word and some context about them. In addition, one cannot know BASF(Canada), Mississauga, Ontario,
Canada is sub-institution of BASF(Germany), Ludwigshafen am Rhein, Germany unless one knows that BASF is headquartered in Germany. We hypothesize that newswire, web data, or affiliation strings of authors could be a source for this evidence.

5. Related Work

The proposed task in this paper sits at the intersection of learned string similarity methods, relation prediction and extraction, and more generally set-based neural network models.

A fundamental component of the proposed approach is to use a parameterized method for comparing the spelling of two institution names. Parametric string similarity methods have a long history. Classic methods (Levenshtein, Longest Common Subsequence, Needleman-Wunsch [Needleman and Wunsch, 1970], and Smith-Waterman [Smith and Waterman, 1981]) measure similarity through parameters of the insert/edit/delete. Other work uses generative models [Dreyer et al., 2008, Andrews et al., 2012, 2014, Faruqui et al., 2016, Rastogi et al., 2016] or conditional random fields [McCallum et al., 2005] to model the string mutation sequences in an effort to learn a parameterized method for similarity within a domain. Recently, Gan et al. [2017] proposed a neural network model based on convolutional neural networks applied to character n-grams to model string similarity.

Relation extraction systems such as OpenIE, Universal Schema and others [Angeli et al., 2015, Verga et al., 2016, Das et al., 2017] are often interested in predicting relationships such as subsidiary or other similar relationships to that of this paper. However, these systems use natural language context such as sentences or paragraphs to make predictions rather than the string similarities themselves.

Set-based embedding models has also been well studied [Ravanbakhsh et al., 2016, Zaheer et al., 2017, Ilse et al., 2018, Hartford et al., 2018, Lee et al., 2018, Mena et al., 2018, Cotter et al., 2019, Bloem-Reddy and Teh, 2019]. Much of this work focuses on learning representations for sets that are order invariant. In other words, the representation formed for a set is the same for any permutation of the order of the input. Our proposed approach models tokens in the intersection / set difference of the institution names, making set-based models a natural choice as our encoder’s architecture. As in previous work, we use parametric models that encode the elements of each set to form a representation.

6. Conclusion

In this paper, we introduce a new dataset built from GRID for learning to predict the institution hierarchies using institution names and other metadata such as the location and type. We demonstrate that a learned model based on set transformers outperforms other set based and non set based methods as well as a simple token overlap baseline. We hope that this work draws interest to this challenging and important problem and that future work explores both richer string spelling based context-free models as well as models that make use of natural language text and a variety of contextual resources to predict institution hierarchies. In addition, rather than constructing set intersections by finding overlapping tokens, we will consider model-based methods for constructing set intersections (i.e., functions of geometric representations of entities, such as Box Embeddings (Vilnis et al, 2018) or Order Embeddings (Vendrov et al 2015)).
References


