Erschienen in: WWW '15 Companion: Proceedings of the 24th International Conference on World Wide Web / Gangemi, Aldo et al. (Hrsg.). - New York : ACM, 2015. - S. 51-52. - ISBN 978-1-4503-3473-0
https://dx.doi.org/10.1145/2740908.2742753

Extracting Taxonomies from Bipartite Graphs

Tobias Kötter
Carnegie Mellon University
koettert@cs.cmu.edu

Stephan Günnemann
Carnegie Mellon University
sguennem@cs.cmu.edu

Michael R. Berthold
University of Konstanz
berthold@ieee.org

Christos Faloutsos
Carnegie Mellon University
christos@cs.cmu.edu

ABSTRACT

Given a large bipartite graph that represents objects and their properties, how can we automatically extract semantic information that provides an overview of the data and – at the same time – enables us to drill down to specific parts for an in-depth analysis? In this work in-progress paper, we propose extracting a taxonomy that models the relation between the properties via an *is-a* hierarchy. The extracted taxonomy arranges the properties from general to specific providing different levels of abstraction.

1. INTRODUCTION

Bipartite graphs, representing objects and their related properties, are ubiquitous when analyzing information sources available on the Web. Examples include online medical databases, where objects represent drugs and properties their characteristics or electronic document collections, where documents are objects and keywords their properties. In this work in-progress paper, we propose to extract taxonomies from such bipartite graphs.

An example of an object-property graph and the taxonomy we aim to extract is depicted in Fig. 1. On the left, the adjacency matrix of the graph is shown, with columns representing objects and rows representing properties. On the right, the taxonomy that describes the *is-a* relation between the *properties* is illustrated (e.g. vertebrate *is-a* subclass of animal). The key aspect we exploit is the principle of inheritance: Each concept in the hierarchy inherits the properties of its parent concepts, e.g. mammal *is-a* vertebrate and an animal. Furthermore, concepts inherit the objects of their children. The animal concept, e.g., inherits the objects beetle and worm from the invertebrate concept. To further improve understanding, we aim to identify similar concepts such as acronyms or synonyms (e.g. animal and beast) and represent them as a single vertex in the tree.

2. PROPOSED PRINCIPLE

We are interested in extracting taxonomy trees. Formally, given a bipartite graph $G = (O, P, E)$ with objects O, properties P and directed edges $E \subseteq O \times P$, we define:

DEFINITION 1. *A taxonomy tree $T = (V, R)$ consists of a set of vertices V and an is-a relation $R \subseteq V \times V$. Each vertex $v = (P_v, O_v) \in V$ represents a concept consisting of a set of properties $P_v \subseteq P$ and the objects $O_v \subseteq O$ exhibiting these properties. Each property is assigned to at most one vertex in the taxonomy tree, i.e. $P_v \cap P_{v'} = \emptyset$ for $v \neq v'$.*
Given a set of properties P_v, the set of objects O_v is uniquely defined as the set of objects $o \in O$ which possess the majority of the given properties P_v, i.e.

$$O_v = \{o \in O \mid \sum_{p \in P_v} [\![o \in O[p]]\!] > \sum_{p \in P_v} [\![o \notin O[p]]\!]\}$$

with $O[p] = \{o \in O \mid (o, p) \in E\}$ representing the set of objects that posses the property $p \in P$ and $[\![.]\!]$ being the Iverson bracket. The is-a relation R states that if $(v_1, v_2) \in R$, the concept v_1 is-a subclass of the concept v_2. Intuitively, the most general concept is the root vertex of the tree.

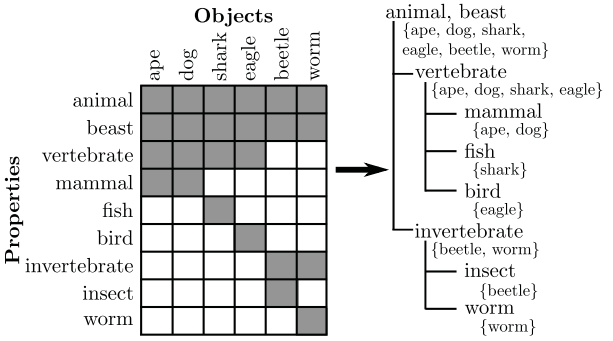

Figure 1: Input graph on the left and the corresponding taxonomy on the right. The *properties* are arranged in a hierarchy, representing an *is-a* relation.

To find an instantiation of such a tree that describes the data's patterns well, we refer to the principle of Minimum Description Length (MDL) [1]: a good tree provides a compact description of the data. Thus, according to MDL, our goal is to find a tree T that minimizes the overall codelength $C(T) + C(G|T)$ where $C(T)$ measures the codelength of the model itself and $C(G|T)$ the codelength of the data G when encoded with the model T.

Codelength of Model. The crucial idea we exploit is inheritance: Assuming a perfect hierarchy (see Fig. 1), any given concept vertex inherits the objects of its child vertices. Thus, instead of representing an object *multiple* times, it is sufficient to record it *once* in the child vertices and to propagate the information to the parent vertices. For example, if we know that the object ape is a mammal (see Fig. 1), and we know that mammals are vertebrates, we do not need to explicitly store the information that apes are vertebrates. *The information that apes are vertebrates can automatically be derived based on the structure of the taxonomy tree.* Based on this principle, we define the *model description cost* $C(T)$ as:

$$C(T) = L_{\mathbb{N}}(|V|) + L_{\mathbb{N}}(|O|) + L_{\mathbb{N}}(|P|) + \sum_{v \in V} C_M(v)$$

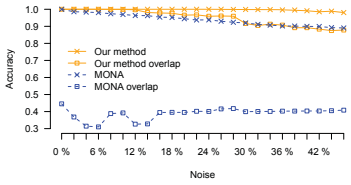

Figure 2: Accuracy of the extracted hierarchies for varying noise levels. Our method clearly outperforms the competing technique.

where we encode (i) the overall number of vertices, the number of objects, and the number of properties in the tree using the MDL optimal universal codelength $L_{\mathbb{N}}$ for integers [5], and (ii) each individual vertex of the tree requiring codelength of

$$C_M(v) = \log|V| + L_{\mathbb{N}}(|P_v|) + |P_v| \cdot \log|P| + L_{\mathbb{N}}(|O_v^{new}|) + |O_v^{new}| \cdot \log|O \setminus O_v^{inh}|.$$

Here, O_v^{inh} denotes all objects of vertex v that are inherited from its children and O_v^{new} those which are not. Since the objects O_v^{inh} are already represented by the child vertices, we only have to encode the objects O_v^{new}. Thus, for each vertex v of the tree we only encode (a) a pointer to its parent $p \in V$, requiring codelength of $\log|V|$. (b) The set (and number) of properties P_v used for this vertex. The codelength is $L_{\mathbb{N}}(|P_v|) + |P_v| \cdot \log|P|$ bits. (c) The set (and number) of objects O_v^{new}, using a similar coding as above.

Codelength of Data. Given the model, what will the codelength $C(G|T)$ of the data look like? The taxonomy tree in Fig. 1, for example, perfectly recovers the input graph, leading to zero cost. In general, we cannot assume that the data perfectly follows the given model. Since MDL, however, requires lossless compression, we additionally have to encode the errors introduced by the model (e.g. edges which are present in the graph but not represented by the taxonomy or vice versa). Since each error corresponds to one edge, the codelength of these description cost can be bounded by

$$C(G|T) = L_{\mathbb{N}}(|R|) + |R| \cdot (\log|P| + \log|O|)$$

where R denotes the set of errors.

Algorithmic aspects. The optimal taxonomy is obtained by minimizing the overall codelength, which is an NP-hard problem. To ensure efficiency, we developed a heuristic algorithm. The basic idea is to build the tree *top-down* by iteratively adding new concepts to the taxonomy. At the beginning, each property represents its own concept. We process these concepts in descending order of their specificity, i.e. we start with the most general concept, where the specificity of a property is defined by the number of objects it possesses. Thus, when, e.g., constructing the taxonomy of Fig. 1, the concept animal/beast and vertebrate are processed first. The second building block of our technique is a *bottom-up* search, where the potential parent of a new vertex is found in reverse order of their insertion (i.e. leafs first). The new vertex is attached to the parent which leads to the smallest value in the overall codelength.

Related works. Most taxonomy extraction approaches are linguistic based approaches developed in the field of natural language processing. Since they, e.g., base on lexical syntactic patterns to extract relations from text [4], they do not apply to graphs. Other methods base on the idea of set coverage [2]. They cannot handle data containing errors since concepts have to fully subsume each other. Monothetic hierarchical clustering [3] builds a hierarchy of clusters where members have some common property (e.g. all are vertebrates) – they follow a similar spirit as our method by describings concept via sets of properties. A limitation of these techniques is the generation of binary trees, and the restriction to disjoint clusters, i.e. objects appear only in a single branch of the hierarchy. In our work, objects might appear in multiple branches. In general, the crucial difference is that clustering is focused on grouping *objects*, while our goal is to hierarchically arrange the *properties*.

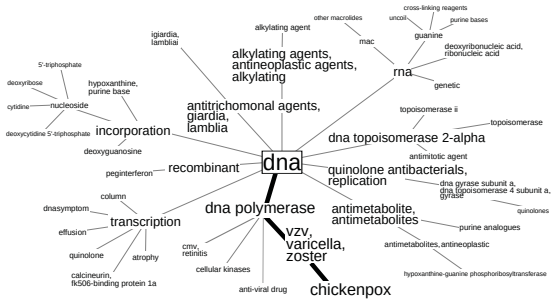

Figure 3: Subset of the deoxyribonucleic acid (DNA) branch automatically extracted from DrugBank.

3. EXPERIMENTAL ANALYSIS

Accuracy. We compared our method against the monothetic hierarchical clustering method MONA [3]. We synthetically created taxonomies (like the one in Fig. 1) and we added a varying degree of additive noise. We then compared the hierarchy found by our method with the ground truth hierarchy of the synthetic data. The same is done for MONA. The hierarchies are compared by using an adaption of the cophenetic correlation coefficient. To be fair to MONA, we created binary taxonomies. Each vertex has two children, each inheriting 50% of its parent's objects, starting with 5000 objects at the root. We distinguish two set-ups: (i) the object sets of the children are disjoint, (ii) they randomly overlap to 25%. Figure 2 shows that our method clearly outperforms MONA.

Case Study. We applied our method on the *DrugBank* data (http://www.drugbank.ca), describing 1,578 drugs and 5,000 experimental substances. Each drug is an *object*, and the *properties* are the drugs' categories, target information, and keywords. Figure 3 shows a part of the extracted taxonomy. *Each vertex represents a set of properties P_v. Edges denote the relation R.* The root in the center represents the most general concept. Fig. 3 illustrates the DNA branch of the extracted taxonomy. By following this branch we see that (a) VZV is the abbreviation for the varicella-zoster virus and (b) they are related to chickenpox. In fact chickenpox is commonly caused by the varicella-zoster virus which is a DNA virus. This knowledge allowed scientists in the past to develop a vaccine against chickenpox. Our detected taxonomy, thus, well matches the inherent properties of the data. Overall, these initial experiments show the high potential of our taxonomy extraction method.

Acknowledgements. This material is based upon work supported by the National Science Foundation (NSF) under Grants No. CNS-1314632 and IIS-1408924. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF or other funding parties. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

4. REFERENCES

[1] P. Grünwald. A tutorial introduction to the minimum description length principle. In *Advances in Minimum Description Length*. MIT Press, 2004.

[2] T. Jiang, A.-H. Tan, and K. Wang. Mining generalized associations of semantic relations from textual web content. *IEEE TKDE*, 19(2):164–179, 2007.

[3] L. Kaufman and P. J. Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009.

[4] Z. Kozareva, E. Riloff, and E. H. Hovy. Semantic class learning from the web with hyponym pattern linkage graphs. In *ACL*, pages 1048–1056, 2008.

[5] J. Rissanen. A universal prior for integers and estimation by minimum description length. *The Annals of statistics*, pages 416–431, 1983.