A Knowledge-graph based Taxonomy Construction Method

András London[1,2], János Zsibrita[2], and Rio Fear[3]

[1] University of Szeged, Institute of Informatics, Arpád tér 2, H-6720, Szeged Hungary
london@inf.u-szeged.hu
[2] Black Swan Data, Zászló utca 4, H-6722 Szeged, Hungary
[3] Black Swan Data, 15th Floor, 10 York Rd, London SE17ND, United Kingdom

1 Introduction

A taxonomy is a hierarchically organized categorization of concepts or entities, for example a Wikipedia category, an ACM Classification System, or an Amazon Product Category. For a great many companies around the world domain-specific taxonomies form a crucial component of the provision data-driven solutions: they can help in search optimization, browsing, organization and storage of information, and much more besides. However, the creation of taxonomies is invariably a highly manual process which is time-consuming, expensive and generally unsustainable at scale, especially in fast changing domains (e.g. news and certain products), therefore an effective method of automated taxonomy generation could be highly valuable. Automated taxonomy building has been well researched in the recent years. Most approaches apply NLP tools to a text corpus e.g. [2], some of them utilize knowledge-graphs, e.g. [5], like Wikipedia or WordNet, while others combine the previous approaches e.g. [3].

In this work we provide a simple, Wikipedia knowledge graph-based methodology to build topic focused taxonomies. We utilize the Wikipedia graph and regard the taxonomy construction as a series of basic graph algorithms performed using topic-specific seed input nodes. Our case-studies demonstrate that the method performs well in general with respect to standard statistics derived from comparison with expert-curated manual taxonomies.

2 Methods and results

We construct the Wikipedia-based knowledge graph proposed and deployed by Aspert et al. [1] available at https://lts2.epfl.ch/Datasets/Wikipedia/. This graph is a directed multigraph with multiple nodes and edge types. Specifically, there are two classes of node which represent either Wikipedia articles or Wikipedia category articles (i.e. category pages). These in turn may be connected by two classes of directed edge which represent 'links_to' and/or 'belongs_to' relationships. A 'links_to' type edge connects two nodes if a hyperlink exists between the corresponding articles (the direction of the edge is straightforward), while a 'belongs_to' type edge represents a hyperlink between an article node or (sub)category node and a category node. For illustration, see Fig. 1. The graph is stored in a Neo4J graph database; for detailed description of the graph structure and other technicalities, see [1].

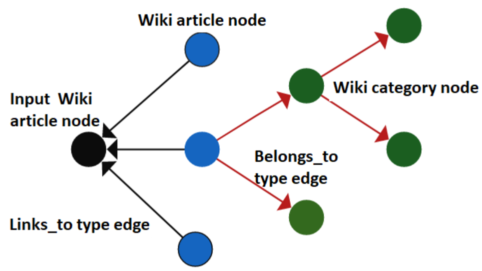

Fig. 1. Wikipedia graph structure. Blue (black) nodes: articles (as input). Green nodes: category pages. Black edges: hyperlinks connecting articles. Red edges: hyperlinks connecting articles or subcategories and parent categories.

2.1 Entity selection

The taxonomy generator is initialized with a collection of Wikipedia article type nodes $P = \{P^1, P^2, \dots\}$ and Wikipedia category type nodes $C = \{C^1, C^2, \dots\}$ which we process according to the following steps.

1. Construct a set $\mathscr{P} = \{P_1^1, P_2^1, \dots; P_1^2, P_2^2, \dots; \dots\}$ of all nodes which have a 'link_to' edge to one or more of the input pages, P.[4]
2. Start a depth-first traversal over each node $P_i^j \in \mathscr{P}$ for all 'belongs_to' type outgoing edges from P_i^j. At the first level this will result in the set $C_i^j = \{C_{i,1}^j, C_{i,2}^j, \dots\}$ of categories which the page P_i^j "belongs_to", at the second level the set of higher 'super'-categories of categories in C_i^j will be reached, and so on.)
   (a) If for a category node $C_{i,k}^j$, found during the traversal process $C_{i,k}^j \in C$ is satisfied, then add P_i^j to a "filtered" entity list $\mathscr{L}$;
   (b) Else, go to step 2, until all elements of $\mathscr{P}$ have been iterated over.

Note that in step 2. a stop criteria is required to restrict the maximum depth of the traversal process due to performance issues. In our experiments the criteria was set to a maximum depth level of four starting from the root node, provided that a category page in C had not already been reached.

2.2 Taxonomy creation

The next step is to classify each entity $e \in \mathscr{L}$ with a category and to provide a hierarchical category organization. For each e let C^e be the set of categories which e belongs to, that is, the neighborhood of e based on its outgoing 'belongs_to' type edges. Note that C^e is determined in step 2 of the entity extraction process. Let $\mathscr{C}$ be the set of all distinct

[4] After this step a fast filtering procedure can be applied by simply deleting any nodes from set $\mathscr{P}$ for which the node's corresponding Wikipedia page name either begins with a number (i.e. "2019_in_tennis") or contains the terms "by_year", "of_the_year", "List_of", or "_in_" (i.e. "Tennis_in_Hungary").

Taxonomy/category	TP/ Gold Players	TP/Gold Teams	TP/Gold All cat.	All Auto
American football	**90.65** (97/107)	**100** (32/32)	**74.59** (138/185)	4,068
Basketball	**89.15** (403/452)	**100** (30/30)	**89.67** (443/494)	5,526
Motorsport	**88.38** (784/88)	–	**86.12** (807/937)	5,862
Soccer	**79.8** (399/500)	**48.83** (294/602)	**62.96** (731/1161)	3,096
Tennis	**75.45** (206/273)	–	**67.17** (262/390)	2,077

Table 1. Coverage (ratio of true positives of automatically extracted entities and manually defined gold standard entities) results for several sports related taxonomies.

categories in $\bigcup_{e \in \mathscr{L}} C^e$. We define a bipartite graph over the disjoint node sets $\mathscr{L}$ and $\mathscr{C}$, where $e \in \mathscr{L}$ and $c \in \mathscr{C}$ are connected if e belongs to category $c \in \mathscr{C}$. Then, starting from c with the highest degree we greedily assign entities to categories step-by-step by removing the assigned entities and corresponding category in each step. Finally, to organize categories into a proper hierarchy one may use a pruning heuristics used e.g. in [4]

2.3 A case-study and evaluation

Domain-specific taxonomies are usually evaluated either by comparing them to manually-built (Gold Standard) taxonomies or by requesting feedback from experts in the field. One of our case-studies is targeted to build a taxonomy covering various sports.[5] Table 1 shows our experimental results regarding coverage (recall) values comparing the Gold Standard and automated taxonomy methods. It is noteworthy that the automated method finds many more relevant entities than the Gold Standard, however, for the purposes of this investigation this is a secondary concern to the primary aim of achieving a high recall compared to the Gold Standard. The high-precision reduction of irrelevant entities from the auto taxonomy (false positives) remains for future work.

References

1. Aspert, N., Miz, V., Ricaud, B., Vandergheynst, P.: A graph-structured dataset for Wikipedia research. In Proceedings of The 2019 World Wide Web Conference, pp. 1188-1193 (May 2019)
2. Hearst, M. A.: Automatic acquisition of hyponyms from large text corpora. In Proceedings of the 14th conference on Computational linguistics, Vol. 2, pp. 539-545 (August 1992)
3. Ponzetto, S. P., Strube, M.:Taxonomy induction based on a collaboratively built knowledge repository. Artificial Intelligence, 175(9-10), 1737-1756 (2011)
4. Medelyan, O., Manion, S., Broekstra, J., Divoli, A., Huang, A. L., Witten, I. H.:. Constructing a focused taxonomy from a document collection. In Extended Semantic Web Conference, pp. 367-381 (May 2013)
5. Ponzetto, S. P., Strube, M.: Deriving a large scale taxonomy from Wikipedia. In AAAI, Vol. 7, pp. 1440-1445 (July 2007)

[5]For example, for tennis the input Wikipedia page nodes were *Tennis, Association_of_Tennis_Professionals* and *Women's_Tennis_Association*, while the input category page node was *Tennis*.