Building Complementary Domain Taxonomies using Query Enrichment

Simoni S. Shah, Shraddha Bhattad, Sanket Lokegaonkar, Ganesh Ramakrishnan
Indian Institute of Technology, Bombay

Abstract

We propose a domain-agnostic framework for building and evolving a domain-specific taxonomy, given an initial set of well-organized data points. The idea is to automatically build and evolve the taxonomy with high precision and recall, but with minimal assistance from a domain expert. The approach used is to evolve two graphs simultaneously: one which is built using minimized involvement from the domain expert, and the other which is obtained by an automatic and controlled subsetting from a suitable Internet knowledge database (WordNet). While the former if high on precision, the latter provides better recall. Further, we define a mapping from the expert's graph to WordNet, hence providing a two-level domain taxonomy. We apply this framework on a dataset of text and videos that capture best practices of rural populations, and find encouraging results for the same.

1 Introduction

Large repositories of textual and multimedia based information need to be classified and organized based on their content, so that they may enable retrieval of the content and in turn support effective browsing, query-based retrieval, recommendation and other downstream applications. When the repository is specific to a particular domain, then, a certain amount of domain knowledge is required for moderation as well as curation, as the repository grows. However, availability of human domain experts may be limited and also expensive. We propose a domain-agnostic framework for building a domain taxonomy, by automating the process of building and evolving the taxonomy as far as possible, whilst achieving high precision and recall.

A domain-specific taxonomy built directly by extracting concepts from the domain-specific repository helps in achieving high specificity, which enables building more accurate and efficient machine models for classification and other purposes. In contrast, directly using any of the well-known large knowledge repositories such as the Wikipedia knowledge graph, WordNet [MF98], etc. often results in increased noise and topic drift. On the other hand, the vast expanse of concepts covered by these knowledge bases, as well as their multi-lingual support makes them very useful. Hence, it would be ideal to combine the best of both: the specificity and precision of a data-driven taxonomy, and the recall of an Internet-based knowledge base.

The novelty in design of this system comes from a two-level structure of the taxonomy which provides a systematic method of aggregating knowledge from the domain expert and knowledge from the Internet, and combining them. On one hand, knowledge from the domain helps in pruning out noise and defining a good sense of the domain. On the other hand, knowledge from the Internet enables addition of concepts with high recall.

The idea of our approach is to use human domain expertise to give an initial sense of the domain, by providing a manually curated repository. The expert is asked to organize a sufficiently large dataset, within a directory hierarchy, such that it is intuitively appealing to her. We use this curated dataset (we call it the "seed dataset") to bootstrap our algorithm. It provides us with a basic knowledge of the top-level entities in the domain, from which the taxonomy may be grown in a controlled fashion.

In order to implement the above, we define two graphs : a graph G_E that is built starting from the seed dataset and evolved using the help of a domain expert, and a graph G_W which is built automatically by defining mappings from nodes in G_E to nodes in WordNet. Since G_E is semi-automatically curated with the help of a domain expert, it is bound to be minimal in noise. On the other hand, the vast expanse of concepts in WordNet allow us to obtain a high level of recall in terms of related concepts. Hence, the two evolving graphs and the mappings between them allow us to define a combined taxonomy which covers the concepts in the growing dataset.

We choose WordNet in particular, due to its enhanced multilingual support, and a suitable means of disambiguation using synsets. However, the key challenge lies in defining mappings from nodes in the domain graph to a set of semantically nearby nodes in WordNet. This is because concepts defined in a domain-specific repository are often too specific for them to exist in a generalized repository such as WordNet. In our example data set, we find that close to 50% of the concepts in the domain graph do not exist in WordNet.

The problem of defining a mapping from a concept in the domain-specific graph G_E to related concepts in WordNet,

is formulated as a problem of querying. We use the idea of query enrichment by systematically searching for related concepts on the Internet and Querying WordNet with those related concepts. We use wikipedia for query enrichment, and find some encouraging results.

2 Related work

Ontology learning, as it is referred to in the literature, is the task of automatically, or semi-automatically inferring a taxonomy for a given domain, using textual data from corpora or the web [Bie05], [PKP+11]. Ontology learning has drastically reduced the overheads of manual ontology construction.

In the work by Kozareva and Hovy, [KH10], from an initial given set of root concepts and basic level terms, the authors first find lexico-syntactic patterns iteratively to harvest new terms from the Web. They also infer hypernym and holonym relations in the process. There has also been some work on extracting concepts from text and disambiguating them to entities in a knowledge base such as the pages of Wikipedia. Often referred to as Wikification [CR13]. In our method, types of relations are not as important, and the edges in the graph G_E are only evidences of semantic connection between nodes. Further, we rely on WordNet to give various types of relations. Our choice of WordNet follows from its versatality in various domains and multilingual support.

The KDD CUP 2010 winners and runners (respectively [SSYC06] and [TKB06]) use the notion of query enrichment for producing more robust classification of their queries to a target taxonomy. The idea is to map queries to a large intermediate taxonomy, in order to enrich the queries, and in turn map the intermediate taxonomy to the target taxonomy.

OntoLearn Reloaded [VFN13] produces an Ontology in the form of a Directed Acyclic Graph (DAG), by extracting terminology from the domain corpus, and also infer hypernym- based interconnections between them. They have also defined useful metrics for evaluating the effectiveness of such an Ontology building mechanism.

In this paper, we provide a semi-automatic method for ontology construction, starting from a well-organised domain corpus. In this sense, the seed domain corpus already provides a good outline of the domain. To the best of our knowledge, the idea of a two-level taxonomy, one expert-driven and the other Internet-driven, is a novel concept in the field of Ontology learning.

3 Problem formulation

Let $D = \{d_1 \cdots d_n\}$ be the seed dataset, such that the data points are organized by a domain expert in the form of a hierarchical set of directories. For example, a video titled "Palm_Sugar_Harvesting.wmv.mp4" is stored in a path "Agriculture/Traditional Art and Technologies/Jaggery_Sugar_Making/Making_Sugar_from_Date_Palm/". Let G_E be the graph that is created and evolved by inputs from the domain expert. Let G_W be the graph obtained by subsetting WordNet such that G_W is a subgraph of the WordNet graph (whose edges are defined by the hypernym-hyponym and meronym relations).

Our problem is to obtain an algorithm which (i)builds G_E from the initial data set D, (ii) Defines a mapping f from every node in G_E to a subset of semantically related nodes in WordNet, which shall constitute G_W and (iii)grows G_E and G_W whenever a new data instance suggests a new concept. This involves two steps: (a) Selectively consult the domain expert for addition of a new node into G_E (b) Define mappings for every new node in G_E, to a subset of nodes in WordNet, hence resulting in potential growth of G_W.

4 Algorithm

Our algorithm is broken into the following segments, each of which accomplishes a well defined task:

- Concept extraction from seed dataset
- Building and Evolving G_E
- Mapping every node in G_E to a subset of nodes in WordNet

Concept Extraction from Seed Dataset

The seed dataset is a well-organized repository, in which the data points have been organized into a hierarchical set of directories and provided by the domain expert. We begin with a simple concept extraction, by extracting keywords and n-grams from the names of the top l levels of the directory tree. The choice of l is left to the system designer, and should be a result of key considerations such as the breadth of the tree, levels up to which concepts are a majority and proper nouns are minimal, etc. In general, a greater l results in greater specificity of the domain taxonomy, which may be desirable to a certain extent. Further, the key words are passed through a lemmatizer to eliminate plural forms. Geographical locations may also be eliminated similarly. This constitutes the concept extraction and pre-processing phase, and the output of this is the set of nodes in G_E.

Building the graph G_E

The nodes of G_E are obtained from the above concept extraction module. Further, edges between these nodes are defined by the directory structure. Hence, if A was the parent directory of B, then there is an edge from the node corresponding to A, to the node corresponding to B, in the graph G_E.

Mapping from G_E to G_W

We define a mapping f from set of nodes in G_E to the power set of nodes in *WordNet*, such that for every node n in G_E, $f(n)$ gives a subset of nodes in WordNet, which are semantically close to n, given the disambiguated sense of n within the given domain. The nodes in G_E fall into one of the following categories: (i) There is an exact match; i.e. n is also a node in WordNet. We call such nodes "green nodes" (ii) The node n is not a node in WordNet. We call such a node a "red node".

For every green node, $f(n) = \hat{n}$, where $\hat{n}$ is the domain-relevant (disambiguated) synset of n in WordNet. For the case of red nodes, the task is more involved. The task of defining

a mapping into WordNet is perceived as the task of querying WordNet. This is preceeded by enhancing the node query with query-enrichment methods.

We use two parallel methods for query enrichment and subsequent mapping to WordNet. The result from both is a ranked list of WordNet nodes, which are semantically close to n.

- *Query enrichment using seed dataset directory names*
  For every occurrence of n in the directory tree, a query is made, which consists of keywords from the path from n to the root (ancestry of n). Every such query is submitted to the WordNet API as a query. WordNet outputs its result for a query, as a set of paths which start from the root "entity" to the most relevant leaf, such that there is a hyponym relation from a parent to a child, in that path. WordNet also provides a score against every path. Our algorithm picks the top 3 paths with maximum score, across all the paths output by WordNet for n. We then extract the bottom two nodes (for more specificity) of each of these paths, resulting in a maximum of 6 nodes in WordNet. Let this set of nodes be denoted as $f_s(n)$.
- *Query enrichment using Wikipedia*
  We query Wikipedia with the given node n. There are three cases: (i) n is a "direct hit", which means that there exists a wiki page for n, (ii) n is an "indirect hit", which means that there is no wiki page for n, but it occurs in some of the other wiki pages (iii) n is a "null hit", which means that n does not occur in any wiki page. For example, in the rural practices dataset, the red node "Permaculture" has a direct hit, "Biofertilizer" is an indirect hit and "Agniastra" is a null hit. For the "null hit" case, we get a null output for this sub task. For the other two cases, we perform the following set of steps:
  - Extract all the hyperlinked words in the top three wiki pages (in case of indirect hit) or the wiki page corresponding to n (in case of direct hit).
  - Filter the above hyperlinked words, by rejecting all those which are not nodes in WordNet. Let $w_1 \cdots w_m$ be the resulting accepted words, which are hyperlinked in wiki pages, and are also present in WordNet.
  - Rank $w_1 \cdots w_m$ in terms of proximity to the domain: We use word2vec to provide similarity scores between every w_i and every green node v_j. Let the number of green nodes be c. Then we define the seed similarity score $S(w_i)$ for w_i to be given by the average word2vec similarity between w_i and the set of green nodes. It is given by:

    $$S(w_i) \;=\; 1/c\, \Sigma_j [T(w_i, v_j)]$$

    where $T(w_i, v_j) \in [0,1]$ is the word2vec similarity score between w_i and v_j. In essence, $S(w_i)$ gives the semantic proximity of w_i with the domain, that is outlined by the seed dataset.

Evolving G_E

With every new data point that is added to the repository, a keyword extraction on the meta data helps us identify the key concepts that are related to that data point. Some of the concepts overlap with existing nodes in G_E. Others must be considered to add to G_E. The graph G_E is designed to evolve with minimal noise and high confidence. Hence the domain expert must be consulted before adding a node to G_E. However, the expert must not be over-burdened. The new nodes must be first screened to have sufficient proximity with the domain (using word2vec as described above). We could keep a log of such nodes such that they are recommended to the expert only when it gains sufficient popularity (appear in a sufficient number of data points).

Evolving G_W

For every new node n in G_E, G_W is grows using the mapping $f(n)$ as described above.

5 Evaluations

We evaluate our system on a large corpus of 5200 text/videos that pertain to best practices of rural people, which constituted our seed dataset. The dataset as well as our intermediate and final results are available on the following link: tinyurl.com/hv3j8j2 . The seed data set was manually curated by a domain expert, who organized it in a directory tree structure, such that the depth (distance from the root) of each leaf (data point) varies from 4 to 12. [1]

By a keyword extraction technique applied on the path of every data point in the seed data set, we obtained a total of 210 keywords which formed the nodes of the graph G_E. The edges of the graph were defined in accordance with the directory structure. Note that despite the tree- structure of the directory, the graph G_E need not be a tree. This is because the directory structure is constructed by the expert who is driven by the intuition of the domain. In our example, the node "Agriculture" appeared at different levels of the directory: as one of the top-level directories, and also within a parent directory named "Video Testimonies".

Out of the 210 nodes in G_E, 119 of them were present as nodes in WordNet as well. These are the green nodes. Synset disambiguation of these green nodes in WordNet was done manually by the domain expert. However, other automatic diambiguation techniques using semantic neighbourhood knowledge could also be used. The remaining 91 nodes do not exist in WordNet and are called the red nodes. Figure 1 shows a part of the generated graph G_E, with green and red coloured nodes.

Effectiveness of Query Enrichment:
Further, we observe several instances by which query enrichment has enhanced the relevance of the mapped nodes, to the domain. For example, for the red node "silvopasture", querying WordNet directly with this word gave non-relevant results. However, by enriching the query using keywords from the paths containing "silvopasture" in the seed data set, we

[1] We would like to acknowledge the generosity Sandeep Goradia, who is committed to working for the upliftment of people in the rural areas, and who made this well-curated data repository available to us.

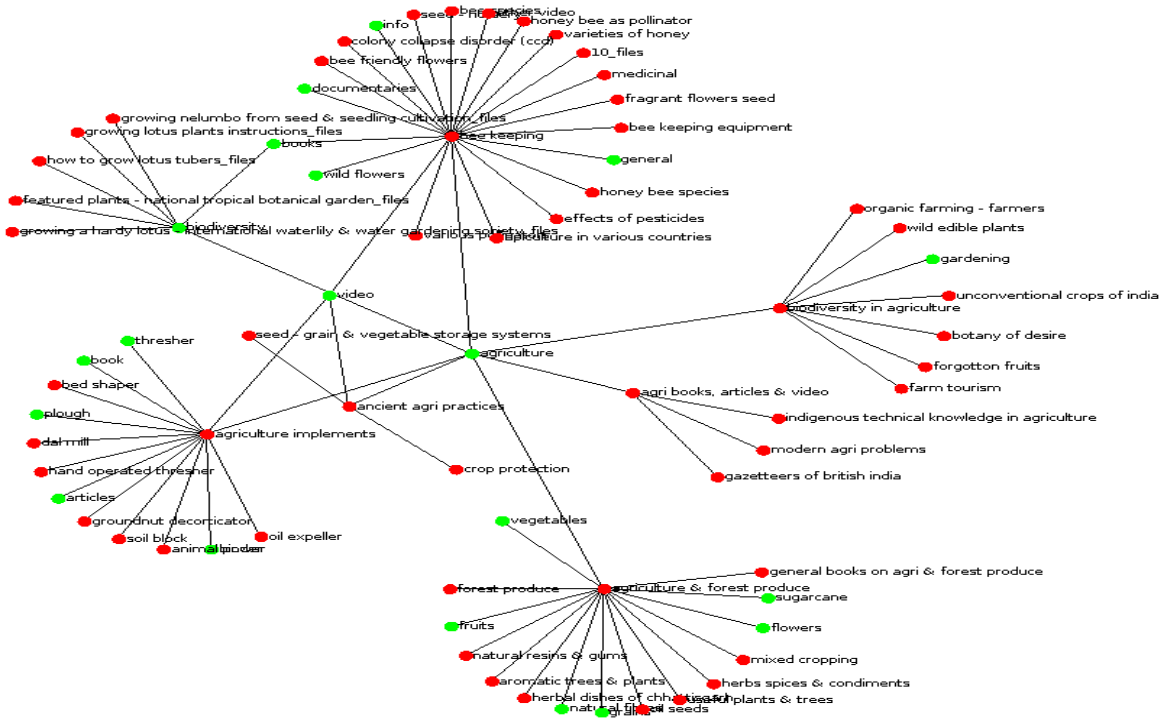

Figure 1: The seed graph G_E

were able to find relevant matches in WordNet, such as "cultivation", "production" and "irrigation". This shows that query enrichment provides a guided search within WordNet.

Analysis of Wikipedia Querying:

On the other hand, querying Wikipedia with red nodes, also proved fruitful. Only 15 out of the total red nodes, did not have any wiki page containing it. These are typically the names of specialized products or processes which may have great relevance within the domain, but do not figure in an Internet knowledge base. For such nodes, a mechanism for the expert to manually define the mappings, can be devised. For all other red nodes, querying Wikipedia resulted in a set of concepts (present in WordNet), with high recall, but it was high in noise as well. For example

- For the red node query "biodynamics", Wikipedia querying resulted in a mixture of words such as "blanc rudolf steiner rootstock riesling red wine quartz pruning powdery mildew port wine pinot", which were not relevant, and words such as "agriculture climate change horticulture bamboo processing" etc.
- For the red node query "bee keeping", some irrelevant words such as " ptolemy mythology livy latin iberian peninsula eratosthenes" were present along with relevant words such as "agriculture sericulture honey silk beeswax corn syrup mealworm cypress burlap pollination apiary"

Analysis of Word2vec Pruning:

Given the Wikipedia query output, word2vec pruning helps in filtering out noise to a very large extent. Hence, the noise is reduced to a minimum and most of the resulting WordNet nodes are relevant. However, in some cases, we observe that some relevant nodes are also eliminated. For instance, in cases of the above examples:

- For the query "biodynamics", the final results after word2vec pruning are "agriculture horticulture floriculture compost fertilizer flower garden garden grape gardening horsetail botanical garden greenhouse composting orchard botany". However, some relevant words such as "herbicide pomology" etc have been eliminated. This has resulted from the fact that our algorithm included a budget: a cap on the maximum number of WordNet nodes to be considered. By choosing a threshold-based algorithm, more relevant concepts can be accommodated.
- For the query "bee keeping" the set of WordNet node mappings after word2Vec pruning are "agriculture silkworm sericulture honey silk beeswax corn syrup mealworm cypress burlap pollination apiary sumac fungus ant".

The histogram in Figure 2 gives a comparative study of mapping results that are obtained from WordNet querying and Wikipedia querying. This analysis has resulted from manual evaluation of precision of mapping. It denotes the number of red nodes against the enrichment confidence. The enrichment confidence measures the precision of the mapping by manually evaluating the precision (percentage) of the mapped WordNet nodes. Hence, for every red node, the precision is the (percentage) fraction of suggested WordNet nodes for that

red node, which are relevant to the sense or meaning of the red node in the given domain. For instance, the histogram indicates that 34 out of 91 red nodes were mapped with precision greater than 80%, using wordnet enrichment. Further, while the WordNet querying shows more promise in terms of precision, the Wikipedia querying is higher on recall.

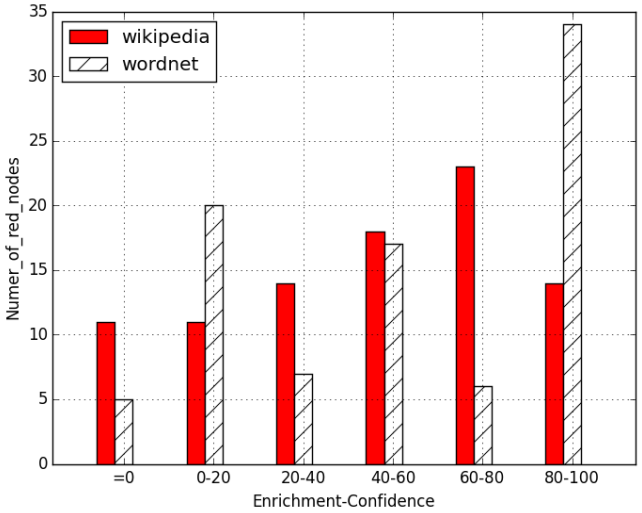

Figure 2: Comparison between Wikipedia and Wordnet Query Enrichment based on Precision

6 Conclusions and Future Work

The key conclusions that may be derived from our evaluations are as follows:

- A large percentage (close to 50%) of the concepts that are provided by the domain expert are not present in WordNet as is.
- Query enrichment using the paths in the data set hierarchy (that is provided by the domain expert) greatly enhances precision of WordNet querying.
- More than 80% of the concepts that are not present in WordNet, have relevant Wikipedia pages. Querying wiki helps in improving on recall.
- Pruning using word2vec helps us assess the relevance of wiki output, to the domain outlined by the expert.
- Both, the WordNet querying as well as Wikipedia querying intrinsically use domain expert knowledge, to provide results with better precision.
- The graphs G_E and G_W complement each other well, and the mapping between them implements a two-level taxonomy for the domain.
- The query enrichment and relying on the Internet knowledge base helps in greatly reducing the need for domain expertise for evolving the graphs.

In this paper, we have described a method to generate a dual-faceted domain taxonomy, with the help of a well-organized domain corpus. The idea is to evolve two graphs in parallel, which complement each other and have well-defined mappings between them. These mappings may also be exploited for down-stream applications such as search-retrieval, content moderation, recommender systems and other applications that could use this taxonomy.

On one hand, the seed graph helps in exploiting domain expertise to create and evolve a domain-specific graph with *high precision*, that may also aid further word-sense disambiguation and semantic inference of search queries and tags. On the other hand, the WordNet subsetting entails a graph with *high recall*, hence encompassing a larger set of words that belong to the domain. Using an Internet knowledge base such as WordNet comes with other advantages as well, such as multilingual support and word sense disambiguation techniques.

In this work, we show effectiveness of query enrichment, and the two-level complementary graph structure of the domain taxonomy. The tool may be enhanced by using more sophisticated models and techniques for keyword extraction, word sense disambiguation and semantic similarity. This may further reduce the dependence on domain experts for graph evolution.

References

[Bie05] Chris Biemann. Ontology learning from text: A survey of methods. In *LDV forum*, volume 20, pages 75–93, 2005.

[CR13] Xiao Cheng and Dan Roth. Relational inference for wikification. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1787–1796, 2013.

[KH10] Zornitsa Kozareva and Eduard Hovy. A semi-supervised method to learn and construct taxonomies using the web. In *Proceedings of the 2010 conference on empirical methods in natural language processing*, pages 1110–1118. Association for Computational Linguistics, 2010.

[MF98] George Miller and Christiane Fellbaum. Wordnet: An electronic lexical database, 1998.

[PKP+11] Georgios Petasis, Vangelis Karkaletsis, Georgios Paliouras, Anastasia Krithara, and Elias Zavitsanos. Ontology population and enrichment: State of the art. In *Knowledge-driven multimedia information extraction and ontology evolution*, pages 134–166. Springer-Verlag, 2011.

[SSYC06] Dou Shen, Jian-Tao Sun, Qiang Yang, and Zheng Chen. Building bridges for web query classification. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 131–138. ACM, 2006.

[TKB06] Domonkos Tikk, Zsolt T Kardkovacs, and Zoltan Bansaghi. Topic mapping: a tool for finding the

meaning of internet search queries. In *Intelligent Engineering Systems, 2006. INES'06. Proceedings. International Conference on*, pages 227–232. IEEE, 2006.

[VFN13] Paola Velardi, Stefano Faralli, and Roberto Navigli. Ontolearn reloaded: A graph-based algorithm for taxonomy induction. *Computational Linguistics*, 39(3):665–707, 2013.