ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

Catalog Integration Using Taxonomy

[I]V. Rasagnya, [II]K. Vinay Kumar
[I,II]Kakatiya Institute of Technology and Science, Warangal, Telangana India

Abstract
Every online Shopping website arranges products into its own categories and it becomes sometimes really difficult to put a product into a category which is existing as the product vendor puts the categories according to his or her own criteria. Hence to avoid the problem of categorization of the products the paper is proposed. The approach is based on a taxonomy-aware processing step that adjusts the results of a text-based classifier to ensure that products that are close together in the provider taxonomy remain close in the master taxonomy.

Keywords
Catalog Integration, Classification, Data Mining, Taxonomies.

I. Introduction

There has been a increased number of users who use the web for shopping and most of the search engines used in the shopping sites maintain a taxonomy of the category which they use to map the products and also use the same for updation when a new product arrives it would not be practically feasible if the product vendor is asked to give the category to the product as per the taxonomy maintained by the site hence we would be needing a product catalog to be maintained and updated as new products arrive at the shopping site .

The provider taxonomy may be different from the master taxonomy, but in most cases, there is still a powerful signal coming from the provider classification. Intuitively, products that are in nearby categories in the provider taxonomy, should be classified into nearby categories in the master taxonomy.

To illustrate this point, consider the example in the provider taxonomy is an excerpt from the taxonomy used by Shoppers Stop, and the master taxonomy is an excerpt from the taxonomy used by TACI searching site Shopping. Now, given a product tagged with a category from Shoppers Stop's (provider) taxonomy, we want to categorize it in the TACI searching site Shopping (master) taxonomy. Suppose we are given the product "Tablet computer" from the category Computers Category in the Shop-Aroundtaxonomy. If we use a text-based classifier to categorize this product into the TACI searching site taxonomy, it is unclear whether this product should be classified into computers/Desktop or computer/laptop or computer/mini PC. If we know that most of the products in the computer category.

Shop-Around category are categorized to the Computer category in TACI searching site, then we can conclude that most likely the new product should also be classified in Computer category.

However, for most products in the Computers Category from the Shop-Aroundtaxonomy, the classifier is actually unable to decide if they should be classified in computers. Therefore, we cannot use the categorization of the products in the same category as the new product to guide as for the correct decision. In this case, the taxonomy information can help to determine the correct categorization. As we go up the taxonomy tree of the Shop-Around taxonomy, we observe that many more products in the Desktop & laptops in Shop-Aroundcategory are classified to the computer category, as opposed to the Electronics category. Using this information we can conclude that most likely the products from the Computers Category should be mapped to laptop rather than Electronics.

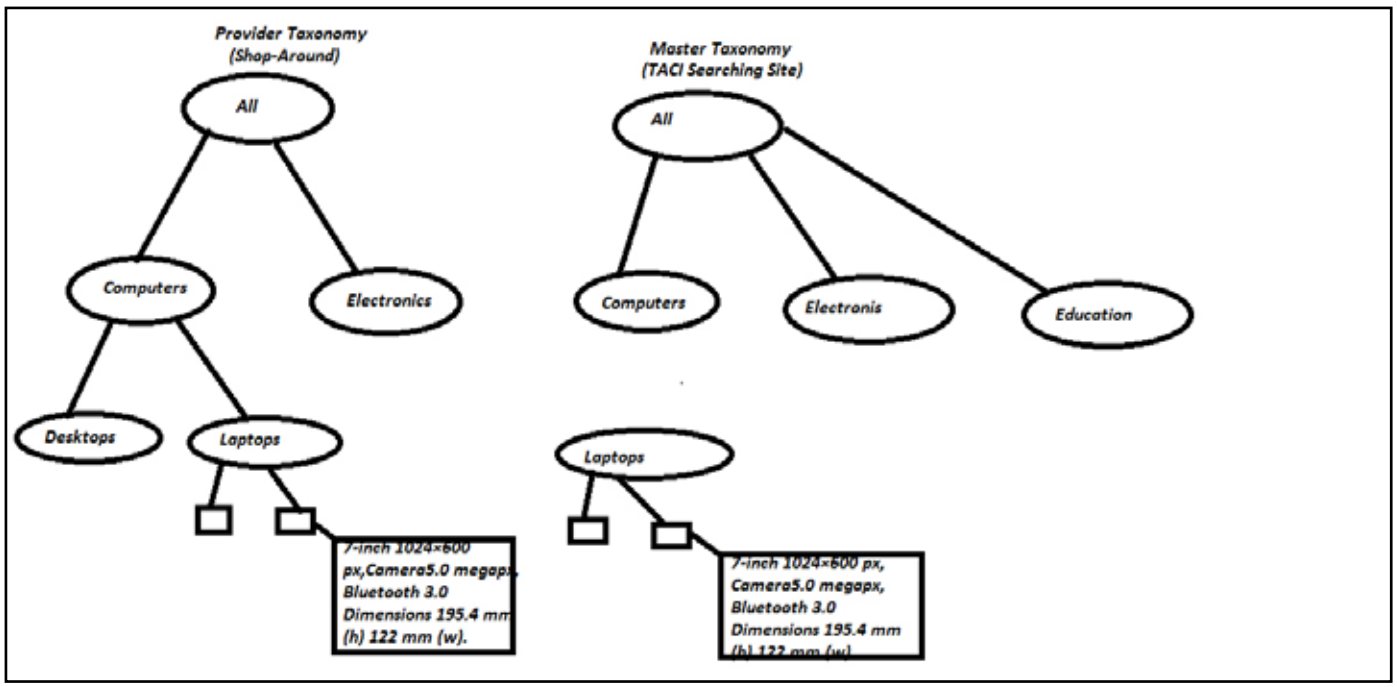

Fig. 1: A simple catalog integration example

II. Related Work

Agrawal and Srikant [1]. method scales to large data sets (like ours),. Sarawagi et al. introduce cross training , which is an approach to semi-supervised learning in the presence of multiple label sets.

Zhang and Lee have also developed approaches to catalog integration by using boosting and transductive learning .

Nandi and Bernstein [20] proposed an approach for matching taxonomies based on query term distributions. It performs the mapping at the taxonomy level, mapping categories from the source to the target,.

Metric labeling and structured prediction. In the metric labeling problem, the goal is to find the optimal labeling of some objects so that they minimize an assignment and a separation cost..

Structured prediction—the study of machine learning algorithms whose goal is to predict complex objects with internal statistical dependencies—is an active area of machine learning research [2]. It has direct applications to natural language processing, in which most prediction problems are structured in nature: sequences of syntactic or semantic labels for words in a sentence (part of speech tagging or entity recognition), syntactic trees (parsing), foreign language sentences (machine translation), graph matchings (word- or sentence-alignment) or logical forms (semantic parsing).

Ontology alignment and schema matching. There is a large body of work in ontology alignment. Representative examples include Glue [8], a system that uses machine learning to learn how to map between ontologies; and Iliads , a system which makes use of machine learning and logical inference techniques to output alignments. In general, the focus in ontology alignment is to map nodes of a source taxonomy to nodes of a target taxonomy.

III. Problem Definition

We now introduce some basic terminology, and formulate the taxonomy-aware catalog integration problem.

A product x is an item that can be bought at a commercial portal. Each product has a textual representation that consists of a name (a short sentence descriTACI searching site the product), and possibly a set of attribute-value pairs. For example, Fig. 1 shows a product whose name is "Tablet computer", and has also a description attribute with value "5.0inches, 1080x1920 pixels, IPS LCD, Quad core, 2300 MHz, 2048 MB RAM" Note that the name and the attributes of a product may vary across providers. For example, another provider may use the name "Tablet computer" for the same Computer Category, and have no description, or other attributes associated with the product.

A product taxonomy G = (C9, E9) is a directed acyclic graph (DAG) whose nodes C9 represent the set of possible categories into which products are organized. Each graph edge (c1, c2) Є Eg represents a subsumption (i.e., an "is-a") relationship between two categories c1 and c2.

Definition 1 (Taxonomy-Aware Catalog Integration).

Given a source catalog Ks and a target catalog Kt, use a taxonomy-aware process fT to learn a cross-catalog labeling function l=fT(Ks,KT).

The learning process fT that produces labeling vector makes use of the full taxonomy structure of the taxonomies S and T in order to define relationships between products in the source and target catalog, and guide the classification process.

IV. The Algorithm

Our problem can be formulated as an Integer Linear Program (ILP) or a Quadratic Integer Program (QIP), however the number of variables is proportional to the number of products in the source catalog, which is prohibitively large.

A. Taxonomy-Aware Algorithm

Algorithm 1 describes the Taxonomy Aware Catalog Integra-tion algorithm (henceforth referred to as the TACI algorithm). The algorithm assumes the existence of a base classifier trained on data from the target catalog. The input to the algorithm consists of a source catalog and a target taxonomy, as well as the parameters _, k, and _. The output of the algorithm is a labeling ' for the products in the source catalog.

In the loop of Lines 2-9, the algorithm applies the base classifier to each product. Based on the base classifier output probabilities, the algorithm either classifies the product to the top category given by the base classifier (Lines 4-6), or it leaves its classification open and stores the top k categories, sorted by probability (Lines 7-9). Given the set of open products and their top-k candidate target categories the algorithm computes the set of candidate source-category pairs (Line 10). In the loop of Lines 12-13, the algorithm computes the separation costs for all of the candidate pairs, and stores them in a hash table. Note that for each source-target pair we compute the value of h only once, and we never compute a separation cost that we will not use later on. In the loop of lines 14-15, the algorithm classifies the open products in. A product is assigned to the category among the top-k categories that minimizes the objective function.

Algorithm 1. TACI Algorithm
Input: source catalog $\mathcal{K}_s$, target taxonomy T, base classifier b, and parameters θ,k, and γ.
Output: a labeling vector ℓ

1: $F_s \leftarrow \emptyset$
2: **for all** $x \in P_s$ **do**
3: $\quad \tau^* \leftarrow \arg\max_{\tau \in C_t} Pr_b[\tau|x]$
4: $\quad$ **if** $Pr_b[\tau^*|x] \geq \theta$ **then**
5: $\quad\quad \ell_x \leftarrow \tau^*$
6: $\quad\quad F_\theta \leftarrow F_\theta \cup \{x\}$
7: $\quad$ **else**
8: $\quad\quad O_\theta \leftarrow O_\theta \cup \{x\}$
9: $\quad\quad$ Compute $\text{TOP}_k(x)$
10: Compute candidate pairs $H_{\theta,k}$
11: Initialize hash table Ψ to empty
12: **for all** $(\sigma,\tau) \in H_{\theta,k}$ **do**
13: $\quad \Psi[(\sigma,\tau)] = h(\sigma,\tau)$
14: **for all** $x \in O_\theta$ **do**
15: $\quad \ell_x \leftarrow \underset{\tau \in \text{TOP}_k(x)}{\text{argmin}} \{(1-\gamma)\text{A COST}x, \tau + \gamma\Psi[(s_x,\tau)]\}$

Fig. 2: TACI Algorithm

To avoid overfitting, when tuning the parameters _ and _ we do not update them unless we have a significant improvement in the accuracy (0.1 percent of the previous value in our experiments). Tuning parameter k. We set the parameter k, such that the accuracy of the base classifier over the top-k results (i.e., the fraction of times that the true category is contained in the top-k results) is above a certain threshold (99 percent in our experiments), or k reaches a predefined maximum (20 in our experiments). In this way, we guarantee that the TACI algorithm can achieve accuracy up to 99 percent.

V. Experimental Evaluation

In this section, we present the experimental evaluation of our

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

approach. The main goals of this evaluation are the following: 1) to show the benefits of our taxonomy-aware calibration step and compare the taxonomy-aware algorithm against other catalog integration approaches; 2) to evaluate the different cost and similarity functions we consider; and 3) to study the running time of our algorithm, and the sensitivity to parameter values.

1. Experimental Setup

Data sets. We use as master catalog the catalog of TACI searching site Shopping, which aggregates data feeds from retailers, distributors, resellers, and other commercial portals. We consider three providers: Shoppers Stop, E-Shop. The Shop-Aroundsource catalog contains 11 products; the E-Shop catalog has 6 products.

Fig 3: TACI Tree Structure

In all the experiments, we consider a target taxonomy that consists of all the categories in TACI searching site Shopping taxonomy that are related to consumer electronics (all of the data set products come from such categories).

In conclusion, the TACI algorithm is obviously affected from the setting of the parameters. However, when varying the parameters the accuracy plots are mostly smooth, and there is a range of values for all parameters for which the algorithm achieves performance comparable to that of the calibrated parameter values. Thus, TACI is not overly sensitive to the exact setting of the parameters, and it can perform comparably well for a wide range of parameter values

VI. Conclusion

In this paper, we presented an efficient and scalable approach to catalog integration that is based on the use of source category and taxonomy structure information. We also showed that this approach leads to substantial gains in accuracy with respect to existing classifiers.

While we focused on shopping scenarios, our techniques are relevant to many other important domains. In parti-cular, they are applicable to classification in any domain where there is a concept of a master taxonomy and there are information providers which use their own taxonomy to label the items that they provide. This includes important verticals such as Local, Travel, Entertainment, etc. One example in Entertainment is the integration of media for streaming purposes. For instance, the Xbox Dashboard now provides the ability to access movies and TV shows from multiple providers, such as Netflix, Hulu, different TV networks, etc. These providers use their own taxonomy to label movies and shows, and thus the need to properly organize them under a master taxonomy. As another example, in the Local domain, different providers may label restaurants in a different way. For example, one provider may tag a restaurant as "Ethnic/Greek" while another may tag it as "Mediterranean."

While our techniques were used for classification, they can also be used for other problems. For example, their output could be used as a feature for item matching, when we want to match elements classified under the master taxonomy (e.g., the products in the master catalog) to incoming offers from the providers.

References

[1] *R. Agrawal and R. Srikant, "On Integrating Catalogs," Proc. 10th Int'l Conf. World Wide Web (WWW), pp. 603-612, 2001.*

[2] *Predicting Structured Data, G. Bakir, T. Hofmann, B. Scho¨lkopf, A. Smola, B. Taskar, and S. Vishwanathan, eds. MIT Press, 2007.*

[3] *G. Bakir, T. Hofmann, B. Schlkopf, A. Smola, B. Taskar, and S. Vishwanathan, Predicting Structured Data. MIT Press, 2007.*

[4] *Y. Boykov and V. Kolmogorov, "An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 26, no. 9, pp. 1124-1137, Sept. 2004.*

[5] *Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization via Graph Cuts," IEEE Trans. Pattern Analysis Machine Intelligence, vol. 23, no. 11, pp. 1222-1239, Nov. 2001.*

[6] *C. Chekuri, S. Khanna, J.S. Naor, and L. Zosin, "Approximation Algorithms for the Metric Labeling Problem via a New Linear Programming Formulation," Proc. 12th Ann. ACM-SIAM Symp. Discrete Algorithms (SODA), pp. 109-118, 2001.*

[7] *H. Daume´ III, J. Langford, and D. Marcu, "Search-Based Structured Prediction," Machine Learning J., vol. 75, pp. 297-325, 2009*

[8] *A.Doan,J.Madhavan,R.Dhamankar,P.Domingos,and A Halevy, "Learning to Match Ontologies on the Semantic Web," The VLDB J., vol. 12, no. 4, pp. 303-319, 2003.*

[9] *T.FinleyandT.Joakims, "TrainingStructuralSVMswhenExact Inference is Intractable," Proc. Int'l Conf. Machine Learning (ICML) 2008.*

[10] *A. Fraser and D. Marcu, "Getting the Structure Right for Word Alignment: Leaf," Proc. Joint Conf. Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), 2007.*

[11] *J. Graca, K. Ganchev, and B. Taskar, "Learning Tractable Word Alignment Models with Complex Constraints," The Computational Linguistics J., vol. 36, no. 3, pp. 481-504, 2010.*

[12] *J. Kleinberg and E. Tardos, "Approximation Algorithms for Classification Problems with Pairwise Relationships:*

Metric Labeling and Markov Random Fields," J. ACM, vol. 49, no. 5, pp. 616-639, 2002.

[13] V. Kolmogorov and R. Zabih, "What Energy Functions can be Minimizedvia Graph Cuts?" IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 26, no. 2, pp. 147-159, Feb. 2004.

[14] A. Kulesza and F. Pereira, "Structured Learning with Approx-imate Inference," Proc. Neural Information Processing Systems (NIPS), 2007.

[15] P. Liang, H. Daume´ III, and D. Klein, "Structure Compilation: Trading Structure for Features," Proc. Int'l Conf. Machine Learning (ICML), 2008.

[16] T.-Y. Liu, Y. Yang, H. Wan, H.-J. Zeng, Z. Chen, and W.-Y. Ma, "Support Vector Machines Classification with a Very Large-Scale Taxonomy," SIGKDD Explorations Newsletter, vol. 7, no. 1, pp. 36-43, 2005.

[17] S. Melnik, H. Garcia-Molina, and E. Rahm, "Similarity Flooding: A Versatile Graph Matching Algorithm and Its Application to Schema Matching," Proc. 18th Int'l Conf. Data Eng. (ICDE), pp. 117-128, 2002.

[18] T.M. Mitchell, Machine Learning. McGraw-Hill, 1997.

[19] R.C. Moore, W. tau Yih, and A. Bode, "Improved Discriminative Bilingual Word Alignment," Proc. Ann. Meeting Assoc. Computa-tional Linguistics (ACL), 2007.

[20] A. Nandi and P.A. Bernstein, "Hamster: Using Search Clicklogs for Schema and Taxonomy Matching," Proc. VLDB Endowment, vol. 2, no. 1, pp. 181-192, 2009.

[21] C. Pesquita, D. Faria, A.O. Falco, P. Lord, and F.M. Couto, "Semantic Similarity in Biomedical Ontologies," PLoS Computa-tional Biology, vol. 5, no. 7, p. e1000443, 2009.

[22] E. Rahm and P. Bernstein, "A Survey of Approaches to Automatic Schema Matching," The VLDB J., vol. 10, no. 4, pp. 334-350, 2001.

[23] P. Ravikumar and J. Lafferty, "Quadratic Programming Relaxa-tions for Metric Labeling and Markov Random Field Map Estimation," Proc. 23rd Int'l Conf. Machine Learning (ICML), pp. 737-744, 2006.

[24] P. Resnik, "Using Information Content to Evaluate Semantic Similarity in a Taxonomy," Proc. 14th Int'l Joint Conf. Artificial Intelligence (IJCAI), pp. 448-453, 1995.

[25] S. Ross, G.J. Gordon, and J.A. Bagnell, "A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learn-ing," Proc. 14th Int'l Conf. Artificial Intelligence and Statistics (AIStats), 2011.

[26] A.M. Rush and M. Collins, "Exact Decoding of Syntactic Translation Models through Lagrangian Relaxation," Proc. 49th Ann. Meeting Assoc. Computational Linguistics (ACL), 2011.

[27] A.M. Rush, D. Sontag, M. Collins, and T. Jaakkola, "On Dual Decomposition and Linear Programming Relaxations for Natural Language Processing," Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP), 2010.

[28] S. Sarawagi, S. Chakrabarti, and S. Godbole, "Cross-Training: Learning Probabilistic Mappings between Topics," Proc. Ninth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2003.

[29] V. Stoyanov, A. Ropson, and J. Eisner, "Empirical Risk Minimiza-tion of Graphical Model Parameters Given Approximate Infer-ence, Decoding, and Model Structure," Proc. Int'l Conf. Artificial Intelligence and Statistics (AISTATS), 2011.

[30] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, "Large Margin Methods for Structured and Interdependent Output Variables," J. Machine Learning Research, vol. 6, 1453-1484, Sept. 2005

Author's Profile

V. Rasagnya, Student in Department of Computer Science, Kakatiya Instiute of Technology and Science, Warangal, Telangana India.

K. Vinay Kumar, Assistant Professor in Department of Computer Science, Kakatiya Instiute of Technology and Science, Warangal, Telangana India.