Automatic Domain Term Recognition Method for Search Engine Query Logs (Postprint)

Authors: Liu Tong, Weijian Ni, Liu Mei

Date: 2017-10-11T00:00:00+00:00

Abstract

Purpose: To address the limitations of traditional domain term recognition methods based on static domain corpora, this paper proposes a novel method for automatically identifying domain terms from search engine query logs.

Method: We use a four-partite graph to abstractly represent query logs and apply a manifold ranking algorithm on it to obtain a ranking of all candidate terms with respect to domainness, selecting the top-ranked terms as domain terms.

Results: Experiments on real search engine query logs confirm that the proposed method achieves better domain term recognition performance, with approximately 20% improvement over baseline methods on the Precision@n metric.

Limitations: The coverage of identified domain terms partially depends on the initial query words selected by domain experts, which imposes certain requirements on the expertise of domain experts.

Conclusion: This method can construct high-quality domain term collections without requiring large-scale domain corpora or extensive manual annotation in advance, demonstrating high practical value.

Full Text

Preamble

Automatic Domain Terminology Identification from Search Engine Query Logs

Liu Tong, Ni Weijian, Liu Mei
(College of Information Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China)

Abstract

[**Objective**] To address the limitations of traditional domain terminology identification methods based on static domain corpora, this paper proposes a novel approach for automatically identifying domain terminology from search engine query logs. [**Methods**] The method employs a four-partite graph to abstractly represent query logs and applies a manifold ranking algorithm on this structure to obtain a domain-relevance ranking of all candidate terms. Terms ranked at the top are selected as domain terminology. [**Results**] Experiments on real search engine query logs demonstrate that the proposed method achieves superior domain terminology identification performance, improving Precision@n metrics by approximately 20% compared to baseline methods. [**Limitations**] The coverage of identified domain terminology partially depends on the initial query terms selected by domain experts, which imposes certain requirements on their expertise. [**Conclusion**] This method can construct high-quality domain terminology collections without requiring large-scale domain corpora or extensive manual annotation beforehand, offering significant practical value.

Keywords: Domain terminology; Search engine; Query logs; Manifold ranking
Classification Number: TP391.1

1. Introduction

Domain terminology refers to phrases that frequently appear in corpora specific to particular fields, such as "Semanotus bifasciatus" and "leaf spot disease" in agriculture. Compared to general vocabulary, domain terms encode rich domain knowledge, making domain dictionaries fundamental resources for various information processing and analysis tasks. Existing domain dictionaries are primarily constructed through two approaches: manual compilation (e.g., AGROVOC, UMLS) and automatic extraction from domain-specific corpora such as news articles, scientific literature, Wikipedia, patent documents, and domain websites. While manual construction achieves high accuracy, it consumes substantial human resources and proves difficult to maintain when domain knowledge evolves. Automatic methods, though more efficient, heavily depend on corpus quality. However, obtaining high-quality domain corpora presents practical challenges: simultaneously acquiring corpora with broad domain coverage and strong domain relevance is difficult, and such corpora are typically static with low update frequency, making them ill-adapted to evolving domain knowledge. Consequently, traditional domain terminology identification methods face significant limitations in real-world applications.

In contrast to conventional domain corpora, search engine query logs represent a novel type of linguistic resource automatically collected by search engines. These logs record complete user-search engine interactions, including submitted queries, timestamps, search results, and user clicks. Query logs exhibit two key characteristics: (1) **Massiveness:** Widespread search engine usage generates vast query logs with extensive coverage across virtually all domains; (2)

Timeliness: Query logs are continuously updated in real-time, capturing the latest information needs across domains. These features make query logs rich repositories of domain terminology that can serve as important data sources for terminology identification. Since query logs are automatically collected, no prior preparation of large-scale domain corpora is required.

2. Methodology

2.1 Four-Partite Graph Representation

We model query logs using a four-partite graph structure. The manifold ranking algorithm is then applied to this graph to rank candidate terms by their domain relevance.

2.2 Manifold Ranking Algorithm Application

The manifold ranking algorithm aims to rank nodes in a graph by leveraging its intrinsic manifold structure. The algorithm constructs a weighted nearest-neighbor graph from a sample set, where some nodes are manually assigned initial interest scores. These scores propagate iteratively through the weighted graph until reaching a stable state, with higher-ranked nodes indicating greater interest. Extensive experiments demonstrate that manifold ranking converges well and produces effective interest-based rankings.

For domain query identification, domain experts first specify an initial set of domain queries. Applying manifold ranking on the four-partite graph yields a ranking of all queries in the log. Top-ranked queries exhibit strong relevance to the initial domain queries and can be considered domain queries themselves. Since the four-partite graph is heterogeneous, we must first convert it to a homogeneous graph of query nodes. We compute query similarity across user and URL dimensions based on edge weights in the original graph, designing the following similarity formula using cosine similarity:

For the user dimension:

$$sim_{User}(q_i, q_j) = \frac{\sum_{u \in U} w_{qu}(q_i, u) \cdot w_{qu}(q_j, u)}{\sqrt{\sum_{u \in U} w_{qu}^2(q_i, u)} \cdot \sqrt{\sum_{u \in U} w_{qu}^2(q_j, u)}}$$

For the URL dimension:

$$sim_{URL}(q_i, q_j) = \frac{\sum_{p \in P} w_{qp}(q_i, p) \cdot w_{qp}(q_j, p)}{\sqrt{\sum_{p \in P} w_{qp}^2(q_i, p)} \cdot \sqrt{\sum_{p \in P} w_{qp}^2(q_j, p)}}$$

We then linearly combine these similarities:

$$sim(q_i, q_j) = \alpha \cdot sim_{User}(q_i, q_j) + (1 - \alpha) \cdot sim_{URL}(q_i, q_j)$$

where α is a parameter controlling the contribution ratio of user and URL dimensions. Since webpage topics are typically more focused than user search interests, the URL dimension usually carries more weight; we set $\alpha = 0.2$ in experiments.

Using this similarity measure, we construct a query graph where each node represents a query and edge weights are computed by the formula above. To prevent self-reinforcement during ranking, we set $sim(q_i, q_i) = 0$ for all queries.

Let W be the adjacency matrix of the query graph. The manifold ranking algorithm proceeds as follows:

1. **Graph preprocessing:** To ensure convergence, we normalize W both row-wise and column-wise to obtain matrix $S = D^{-1/2}WD^{-1/2}$, where D is a diagonal matrix with $D_{ii} = \sum_j sim(q_i, q_j)$.

2. **Domain score initialization:** Experts specify an initial domain query set, defining vector $y = (y_0, y_1, \ldots, y_{|Q|})$ where $y_i = 1$ if query q_i belongs to the initial set and $y_i = 0$ otherwise. This vector represents prior domain scores.

3. **Score propagation:** Define vector $f = (f_0, f_1, \ldots, f_{|Q|})$ representing posterior domain scores. Iteratively compute until convergence:

$$f^{(t+1)} = \alpha \cdot S \cdot f^{(t)} + (1 - \alpha) \cdot y$$

   where $\alpha \in [0, 1)$ is a smoothing parameter controlling the contribution of prior scores versus neighbor propagation.

4. **Domain query output:** Specify the number n of domain queries to extract, then select the top n queries ranked by their final scores f_i.

2.3 Domain Terminology Identification

Candidate Term Generation: After identifying domain query set QD, we consider all URLs clicked through QD in the query logs as domain-relevant webpages:

$$PD = \{p \mid (\forall q \in QD) \wedge (p \in Click(q))\}$$

We crawl these webpages, filter HTML tags, and obtain a domain-specific web corpus. All phrases in this corpus become candidate terms. However, many domain terms cannot be correctly segmented by conventional Chinese word segmenters (e.g., "双条杉天牛" often gets split into individual characters). Therefore, after segmentation, we merge fragments using sliding windows of lengths 2, 3, and 4 to generate all possible n-grams as candidate terms.

To measure n-gram cohesion, we use an extended multivariate mutual information metric. For n-gram $C = c_1 c_2 \ldots c_n$, let $p(C)$ and $p(c_i)$ denote probabilities.

We design:

$$eMI(C) = \frac{\log \frac{p(C)}{\prod_{i=1}^{n} p(c_i)}}{\sqrt{|C|^{\gamma}}}$$

Compared to traditional metrics, the $\sqrt{|C|^{\gamma}}$ parameter penalizes low-frequency n-grams when γ is large, reducing noise. We set $\gamma = 2$ in experiments.

Since mutual information values are not comparable across different n-gram lengths (tending to increase with n), we normalize within each length group:

$$neMI(C) = \frac{eMI(C)}{\frac{1}{|S_{|C|}|} \sum_{C' \in S_{|C|}} eMI(C')}$$

where $S_{|C|}$ is the set of n-grams with the same length as C. N-grams exceeding a threshold are selected as candidate terms.

Edge Weight Calculation: We compute weights for edges between URLs and candidate terms in the four-partite graph' s subgraph GPT. The weight reflects their association strength, designed similarly to query similarity formulas:

$$w_{pt}(p,t) = \frac{count(p,t)}{len(p)} \cdot \log \frac{|P|}{\sum_{p' \in P} I(p',t)}$$

where $count(p,t)$ is the frequency of term t in webpage p, $len(p)$ is the total number of candidate terms in p, and $I(p,t)$ indicates term presence.

After obtaining edge weights, we convert GPT to a homogeneous term graph and apply manifold ranking to obtain domain-relevance scores for all candidate terms. The top m candidates form the final domain terminology set.

3. Experiments

3.1 Experimental Setup

We used publicly available query logs from a commercial search engine. After filtering invalid URLs and associated queries, the experimental dataset statistics are shown in .

We selected "crop pest and disease control"as the target domain. A PhD student in plant protection from an agricultural university was invited to annotate initial domain queries and evaluate the correctness of extracted terms.

Since complete domain terminology sets are typically unavailable, we use precision-based metrics. Given that manifold ranking produces a relevance-ordered list, we employ Precision@n (P@n) to evaluate ranking accuracy:

$$P@n = \frac{\text{Number of correct terms in top } n}{\text{Total terms in top } n}$$

We test with $n = 100, 200, 300, 400, 500$.

Few studies use query logs for terminology identification; we compare against the method in [11], which requires manually annotated domain URLs. For fair comparison, we replace their required URLs with those obtained from our first-stage domain queries.

3.2 Experimental Results

Parameter Sensitivity: We analyze the smoothing parameter α in $[0.5, 1.0]$ with step 0.1. Results in [Figure 2: see original paper] show that terminology identification improves as α increases, peaking at $\alpha = 0.9$. This indicates that while neighbor propagation is important, prior domain scores from manually specified queries remain essential. When $\alpha = 1.0$ (ignoring prior scores), performance drops significantly.

Convergence Analysis: With $\alpha = 0.9$, we analyze convergence using Sum of Squared Differences (SSD) between consecutive iterations. [Figure 3: see original paper] shows both stages converge quickly: domain query identification in ~150 iterations and terminology identification in ~100 iterations.

Comparison: [Figure 4: see original paper] compares our method against the baseline. Our approach outperforms the baseline across all P@n metrics, with average precision reaching 74% for top 500 terms—demonstrating substantial practical value. shows sample extracted terms, where starred entries indicate errors like "玉米螟的" (possessive form) and "蚜虫蚜虫" (repetition). These errors occur when high-frequency words co-occur with seed terms, suggesting the need for more careful seed selection and improved cohesion metrics.

4. Conclusion

Search engine query logs represent a massive, dynamic resource rich in domain terminology. This paper proposes a method that models query logs as a four-partite graph and applies manifold ranking to identify domain terms. The approach automatically extracts domain-specific terminology from non-domain corpora, avoiding the need for large-scale domain corpora required by traditional methods. By leveraging the structural characteristics of query logs, it achieves accurate terminology identification with minimal manual annotation. Experiments on real query logs demonstrate high convergence speed and accuracy. Future work includes incorporating semi-supervised and active learning to further reduce dependence on initial domain queries.

References

[1] Liu Chunyan, An Xiaomi, Hou Renhua. Vocabulary Standard Development Methodology and Its Application in Information and Documentation Fields[J]. Library and Information Service, 2014, 58(9): 91-95.

[2] Caracciolo C, Stellato A, Morshed A, et al. The AGROVOC Linked Dataset[J]. Semantic Web, 2013, 4(3): 341-348.

[3] Bodenreider O. The Unified Medical Language System (UMLS): Integrating Biomedical Terminology[J]. Nucleic Acids Research, 2004, 32(S1): D267-D270.

[4] Bonin F, Dell' Orletta F, Venturi G, et al. A Contrastive Approach to Multi-word Term Extraction from Domain Corpora[C]. In: Proceedings of the 7th International Conference on Language Resources and Evaluation. 2010: 19-27.

[5] Hua Bolin. Extracting Information Method Term from Chinese Academic Literature[J]. New Technology of Library and Information Service, 2013(6): 68-75.

[6] He Yuanbiao, Le Xiaoqiu, Zhang Fan. Research on Keyphrase Extraction from Scholarly Article Outline[J]. New Technology of Library and Information Service, 2014(3): 73-79.

[7] Zeng Wen, Xu Shuo, Zhang Yunliang, et al. The Research and Analysis on Automatic Extraction of Science and Technology Literature Terms[J]. New Technology of Library and Information Service, 2014(1): 51-55.

[8] Dorji T C, Atlam E-S, Tata S, et al. Extraction, Selection and Ranking of Field Association (FA) Terms from Domain-specific Corpora for Building a Comprehensive FA Terms Dictionary[J]. Knowledge and Information Systems, 2011, 27(1): 141-161.

[9] Qu Peng, Wang Huilin. Patent Term Extraction for Information Analysis[J]. Library and Information Service, 2013, 57(1): 130-135.

[10] Gu Jun, Wang Hao. Study on Term Extraction on the Basis of Chinese Domain Texts[J]. New Technology of Library and Information Service, 2011(4): 29-34.

[11] Yan Xinglong, Liu Yiqun, Fang Qi, et al. Domain-Specific Terms Extraction Based on Web Resource and User Behavior[J]. Journal of Software, 2013, 24(9): 2089-2100.

[12] Jiang D, Pei J, Li H. Mining Search and Browse Logs for Web Search: A Survey[J]. ACM Transactions on Intelligent Systems and Technology, 2013, 4(4): Article No. 57.

[13] Ji Peipei, Yan Xiaoyan, Cen Yonghua. A Survey of Term Recognition and Extraction for Domain-specific Chinese Text Information Processing[J]. Library and Information Service, 2010, 54(16): 124-129.

[14] Song Peiyan, Lu Qing, Liu Ningjing. A New Method for Knowledge Unit Automatic Extraction Using Definitions of Terms[J]. Journal of Intelligence, 2014, 33(4): 139-143.

[15] Xiong Liyan, Tan Long, Zhong Maosheng. An Automatic Term Extraction System of Improved C-value Based on Effective Word Frequency[J]. New

Technology of Library and Information Service, 2013(9): 54-59.

[16] Foo J, Merkel M. Using Machine Learning to Perform Automatic Term Recognition[C]. In: Proceedings of the LREC 2010 Workshop on Methods for Automatic Acquisition of Language Resources and Their Evaluation Methods, Malta. 2010: 49-54.

[17] Da Silva Conrado M, Pardo T, Rezende S O. A Machine Learning Approach to Automatic Term Extraction Using a Rich Feature Set[C]. In: Proceedings of NAACL HLT 2013 Student Research Workshop. 2013: 16-23.

[18] Loukachevitch N V. Automatic Term Recognition Needs Multiple Evidence[C]. In: Proceedings of the 8th International Conference on Language Resources and Evaluation. 2012: 2401-2407.

[19] Jiang D, Leung K W T, Yang L, et al. Query Suggestion with Diversification and Personalization[J]. Knowledge-Based Systems, 2015, 89: 553-568.

[20] Rose D E, Levinson D. Understanding User Goals in Web Search[C]. In: Proceedings of the 13th International Conference on World Wide Web. ACM, 2004: 13-19.

[21] Zhai Haijun, Guo Jiafeng, Wang Xiaolei, et al. Mining Named Entities from Query Logs[J]. Journal of Chinese Information Processing, 2010, 24(1): 71-76, 116.

[22] Xu G, Yang S H, Li H. Named Entity Mining from Click-through Data Using Weakly Supervised Latent Dirichlet Allocation[C]. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2009: 1365-1374.

[23] Jain A, Pennacchiotti M. Domain-independent Entity Extraction from Web Search Query Logs[C]. In: Proceedings of the 20th International Conference Companion on World Wide Web. ACM, 2011: 63-64.

[24] Dalvi B, Xiong C, Callan J. A Language Modeling Approach to Entity Recognition and Disambiguation for Search Queries[C]. In: Proceedings of the 1st International Workshop on Entity Recognition & Disambiguation. ACM, 2014: 45-54.

[25] Zhou D, Weston J, Gretton A, et al. Ranking on Data Manifolds[J]. Advances in Neural Information Processing Systems, 2004, 16: 169-176.

[26] Singhal A. Modern Information Retrieval: A Brief Overview[J]. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 2001, 24(4): 35-43.

[27] Van de Cruys T. Two Multivariate Generalizations of Pointwise Mutual Information[C]. In: Proceedings of the Workshop on Distributional Semantics and Compositionality. Association for Computational Linguistics, 2011: 16-20.

Author Contributions

Liu Tong: Implemented and improved the research methodology, wrote the manuscript.
Ni Weijian: Conceived the research idea, designed the methodology, revised the manuscript.
Liu Mei: Organized experimental data, assisted in completing experiments.

Conflict of Interest

All authors declare no conflict of interest.

Supporting Data

Supporting data [1-3] are available in the journal' s online version at http://www.infotech.ac.cn. Supporting data [4-8] are stored by the authors and can be requested via email: niweijian@gmail.com.

[1] Liu Tong, Ni Weijian, Liu Mei. top-query.txt. Identified domain queries.
[2] Liu Tong, Ni Weijian, Liu Mei. top-term.txt. Ranked domain terminology.
[3] Liu Tong, Ni Weijian, Liu Mei. convergence.txt. Convergence performance of two-stage manifold ranking.
[4] Liu Tong, Ni Weijian, Liu Mei. querylog-pre.txt. Processed query log data.
[5] Liu Tong, Ni Weijian, Liu Mei. query-user.txt. Query-user association matrix.
[6] Liu Tong, Ni Weijian, Liu Mei. query-url.txt. Query-URL association matrix.
[7] Liu Tong, Ni Weijian, Liu Mei. webcorpus.txt. Domain web corpus dataset.
[8] Liu Tong, Ni Weijian, Liu Mei. url-term.txt. URL-candidate term association matrix.

Received: 2015-08-13
Revised: 2015-12-10

Note: Figure translations are in progress. See original paper for figures.