Postprint: Hierarchical Path Generation for Chinese Text Based on Wikipedia

Authors: summer

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

Abstract

【Objective】To generate hierarchical semantic paths for free text utilizing the Wikipedia knowledge base. 【Method】For Chinese export data from Wikipedia, a hierarchical tree-structured graph is constructed; subsequently, free text is represented as article concept vectors through explicit semantic analysis, and mapped onto the tree graph via article-category associations to constitute seed category nodes. Hierarchical paths are then generated through information diffusion initiated from these seed nodes, followed by top-down path selection and optimization. 【Results】The average relevance of the first hierarchical path achieves 54.10% on the test set, and the top 20 paths are globally ranked in descending order of relevance. 【Limitations】The impact of retaining different numbers of concepts in explicit concept vectors on the quality of generated paths has not been analyzed. 【Conclusion】The hierarchical paths generated based on the Wikipedia knowledge base are capable of reflecting the primary semantic information of the text.

Full Text

Generating Hierarchical Paths of Chinese Text from Wikipedia

Xia Tian
(Key Laboratory of Data Engineering and Knowledge Engineering of Ministry of Education, Renmin University of China, Beijing 100872, China)
(School of Information Resource Management, Renmin University of China, Beijing 100872, China)

Abstract

[**Objective**] This study aims to generate hierarchical semantic paths for free texts by leveraging Wikipedia as a knowledge base. [**Methods**] We first constructed a hierarchical tree-like graph from Chinese Wikipedia export data. Free texts were then represented as article concept vectors through Explicit Semantic Analysis (ESA) and mapped onto the tree-like graph as seed category nodes via article-category associations. Hierarchical paths were subsequently generated through information diffusion starting from seed nodes combined with top-down path selection and optimization. [**Results**] On the test dataset, the average relevance of the first hierarchical path reached 54.10%, and the top 20 paths were overall sorted in descending order of relevance. [**Limitations**] We did not analyze how retaining different numbers of explicit concepts in the concept vector affects the quality of generated paths. [**Conclusions**] The hierarchical paths generated from Wikipedia can effectively reflect the main semantic information of texts.

Keywords: Semantic path; Explicit semantic analysis; Hierarchical classification; Wikipedia

1. Introduction

Semantic description of text is a common task in text analysis. Based on granularity, representation methods can be divided into three levels: fine-grained representation such as bag-of-words, which treats text as a collection of independent terms with different weights calculated using Boolean logic, TF-IDF, and other methods; coarse-grained representation such as classification, which automatically selects the most relevant categories from a predefined set by constructing classifiers like Naive Bayes, SVM, or decision trees; and intermediate-level representation, most commonly represented by graph structures and topic models. The former represents text as a semantic graph composed of concept nodes and edges [1], while the latter, typified by LDA [2], views text as generated by several topics following certain distributions, where topics themselves are generated from terms according to specific distributions.

Among these three granularities, classification provides the most concise semantic description of text with the strongest human readability. However, traditional classification techniques operate on a fixed number of categories at the same semantic level without hierarchical relationships, limiting their ability to capture deep semantic information. Introducing multi-level classification to describe text through hierarchical semantic paths would facilitate better and faster acquisition of main semantic content.

This study focuses on identifying hierarchical semantic paths for free texts. Based on Chinese Wikipedia export data, we constructed a large-scale hierarchical tree-like graph. Using Explicit Semantic Analysis, we mapped the semantic information of arbitrary texts onto this graph, then generated hierarchical classification paths through node information diffusion and top-down path selection and optimization. Hierarchical semantic description of text can be achieved through hierarchical classification, which assigns unknown objects to categories within a massive hierarchy [3]. Hierarchical classification requires a well-structured hierarchy and sufficient training data, typically implemented by converting the problem into traditional classification and applying conventional algorithms [4-5]. However, manually maintaining a rigorous large-scale hierarchical tree is difficult, and the large number of category nodes makes classification algorithms inefficient, limiting the application scope of hierarchical classification.

Compared to hierarchical classification, leveraging Wikipedia' s existing article and category networks offers advantages for identifying hierarchical semantics: Wikipedia has formed an open, dynamically growing category system, and the linking relationships between categories and articles provide abundant explicit semantic information, upon which several effective text semantic analysis techniques have been built [6].

Muchnik et al. [7] utilized Wikipedia' s article link network to automatically construct latent hierarchical structures of terms but did not use Wikipedia' s category information. Gabrilovich et al. [8] proposed Explicit Semantic Analysis (ESA), a classic method for text semantic representation based on Wikipedia. ESA uses Wikipedia articles and their interlinks to represent text as vectors of concepts (article titles), which has been widely applied in word relatedness computation [9], query expansion [10], text classification [8], and other tasks. ESA expresses statistical correlations between text and Wikipedia concepts, maintaining the independence assumption among vector elements like bag-of-words methods. Consequently, ESA' s intuitive interpretability of actual text semantics remains weak. For example, using the ESA model built from our dataset for the text "Sina Weibo," the top 5 most relevant articles are "Tencent Weibo," "Long Weibo," "Weibo AIR," "Free Weibo," and "Controversies about Sina Weibo." In contrast, our path identification technique outputs the first two hierarchical paths as "Society/Mass media/World Wide Web/Web 2.0" and "Society/Culture/Internet culture/Virtual community," which clearly better describe the text' s semantic information.

Overall, while representing text semantics through concept terms has seen good research progress, no publicly effective method exists for generating hierarchical semantic paths for arbitrary texts. This paper directly targets Wikipedia' s open category system, extracting hierarchical paths that express main semantic information from the massive network composed of articles, categories, and inter-category relationships. Unlike hierarchical classification methods, our approach handles a huge number of categories in a complex network without building sophisticated classifiers. Instead, it dynamically completes path identification through information diffusion in a simplified network, ultimately generating highly readable hierarchical semantic paths.

3. Construction of Hierarchical Classification Tree-like Graph from Wikipedia

Wikipedia provides an open hierarchical classification system for multi-dimensional annotation of world knowledge represented as articles. The Chinese classification system primarily follows the Dewey Decimal Classification, supplemented by references to the *Chinese Library Classification* and Lai Yongxiang' s *Chinese Book Classification*. Wikipedia data, compiled through collective intelligence, is relatively rich and complete, and freely accessible, making it suitable for constructing large-scale hierarchical classification systems.

The Wikipedia category system can be represented as a directed graph $WG = \langle V, E\rangle$, where node set $V = \{v_1, v_2, ..., v_m\}$ represents categories, and arc set $E = \{e_1, e_2, ..., e_n\}$ represents category relationships. For arc $e_k = \langle v_i, v_j\rangle$, node v_i is the parent category of v_j. A subgraph of Wikipedia category graph WG is shown in [Figure 1: see original paper].

Wikipedia' s Chinese classification system uses "Category:Contents" as the main entry, which has 22 direct subcategories serving as the first-level categories with practical significance, as listed in .

For convenience, we define the following for GW:

(1) Let "Category:Contents" be the root node of GW, denoted as $root(G)$.

(2) For an arc $\langle v_i, v_j\rangle$, v_i is the parent node of v_j, and v_j is the child node of v_i. Let $parents(v)$ denote the set of all parent nodes of v, and $children(v)$ denote the set of all child nodes of v. For example, in Figure 1: $children(\text{"Natural science"}) = \{\text{"Physical science"}, \text{"Mathematics"}, ...\}$ and $parents(\text{"Physical science"}) = \{\text{"Scientific disciplines"}, \text{"Natural science"}, ...\}$.

(3) Each node v is assigned a depth attribute relative to the root node. When v is the root node, its depth is 0. Otherwise, it is recursively defined as: $depth(v) = \min_{v_i \in parents(v)}(depth(v_i) + 1)$. For example, in Figure 1: $depth(\text{"Science"}) = depth(\text{"Natural science"}) = 1$ and $depth(\text{"Scientific disciplines"}) = depth(\text{"Physical science"}) = 2$.

(4) Any simple path from a first-level node to category node v is called a hierarchical classification path of v, denoted as p_v, with $|p_v|$ representing the path length (number of category nodes). In Figure 1, "Natural science → Physical science" is a classification path of node "Physical science" with length 2.

The complete Wikipedia category graph GW has several aspects that hinder automatic algorithmic analysis. First, Wikipedia contains numerous categories for navigation purposes, such as "Main topic classifications → Categories → Categories named after people → Categories named after people by occupation → Categories named after businesspeople → Bill Gates." Removing these navigation-oriented categories can improve automatic path identification. Second, topological sorting reveals numerous cycles in GW, which hinder recursive processing, such as "Social sciences → Criminology → Crimes → Human rights abuses → Religious persecution → Religious pluralism → Religious persecution." Third, some nodes exhibit path inclusion, where a node has multiple paths of different lengths and the longer path contains all categories of the shorter path. For example, in Figure 1, both "Science → Natural science → Physical science" and "Natural science → Physical science" are paths of "Physical science." Typically, retaining only the shorter path preserves the main semantic information while simplifying the graph. Additionally, GW contains minor anomalies such as missing category references, invalid paths, and duplicate categories. For instance, category "Buddhist ritual implements" points to the non-existent parent "Ritual implements"; category "Types of wisdom" has no parent category (invalid path); and "Russian explorers" and "俄羅斯探險家" represent the same concept but correspond to two independent category pages.

To address these issues, we propose a hierarchical classification graph construction algorithm that prunes GW by removing certain nodes and edges to eliminate cycles and path inclusion, resulting in the simplified hierarchical classification graph GH:

Input: Original Wikipedia category relation graph GW
Output: Tree-like graph GH for hierarchical path identification

1. $R = root(GW)$
2. $V_H = \{R\}, E_H = \emptyset$
3. Initialize queue Q and enqueue R into Q
4. While Q is not empty:
5. v = dequeue from Q
6. If $v \in \{$"Interdisciplinary fields", "Main topic classifications", "Lists"$\}$ then continue
7. For each $child$ in $children(v)$:
8. `If $depth(child) = depth(v) + 1$:`
9. `  $V_H = V_H \cup \{child\}$`
10. `$E_H = E_H \cup \{edge(v \rightarrow child)\}$`
11. `Enqueue $child$ into $Q$`
12. End if
13. End while
14. Return $GH = \langle V_H, E_H \rangle$

The algorithm performs breadth-first traversal of GW starting from the root node using a queue structure. For the currently visited node v (line 5), it eliminates side classifications by ignoring three first-level categories: "Interdisciplinary fields,""Main topic classifications,"and "Lists"(line 6). It then processes each child node: when the child' s depth equals the current node' s depth plus 1, the child node and edge $v \to child$ are added to node set V_H and edge set E_H (lines 9-10); otherwise, the edge $v \to child$ is ignored, indicating another node in GW points to *child* with a shorter distance to the root. Finally, V_H and E_H constitute the simplified hierarchical classification graph GH.

The algorithm ensures GH has a root node with no incoming edges, and each node v receives incoming edges only from upper-level nodes (depth v' s depth minus 1) and outgoing edges only to lower-level nodes (depth v' s depth plus 1). GH exhibits most tree properties: root node, child nodes, leaf nodes, and hierarchical structure. However, since nodes in GH may have multiple parent nodes, we call it a tree-like graph.

4. Hierarchical Classification Path Identification Method

Hierarchical semantic classification path identification based on Wikipedia consists of three components: representing free text as explicit concepts composed of Wikipedia articles; mapping explicit concepts to the tree-like graph and solving the hierarchical classification path set; and optimizing path selection by comprehensively considering relevance and novelty.

We make the following assumption: any text t can be described by Wikipedia categories, starting from the root node and proceeding top-down through intermediate categories until reaching fine-grained categories recognized by Wikipedia editors. Let $p(c_i|t)$ denote the semantic relevance between category c_i and text t. After assigning relevance values to each category, we can select the most relevant paths from root to terminal nodes as the semantic hierarchical paths for text t.

4.1 Explicit Concept Representation of Text ESA represents free text as a vector of concepts using a general knowledge base, typically trained from Wikipedia. Given a concept set $\{a_1, a_2, ..., a_m\}$ (corresponding to Wikipedia article titles) and associated documents (Wikipedia article content) $\{d_1, d_2, ..., d_m\}$, ESA constructs a sparse matrix T where each column represents a concept and each row corresponds to a term appearing in the documents. Each element $T[i, j]$ represents the TF-IDF value of term t_i appearing in document d_j [8].

As noted in [8], not all documents are equally effective for ESA. We filter Wikipedia articles based on content and link relationships. Content-wise, articles that are redirect pages, disambiguation pages, list pages, or contain fewer than 200 words are filtered as non-significant. Link relationship-wise, articles with fewer than 20 total incoming and outgoing links are filtered.

To build the ESA model, we scan the filtered Wikipedia data to compute TF-IDF values for each "term $\to$ article" pair, forming the final ESA matrix T. We also maintain article-to-category membership relationships to bridge free texts and hierarchical paths.

After constructing matrix T, given text t, its explicit semantic concept vector is calculated as:

$$V_t = \sum_{w \in terms(t)} tf(w,t) \times idf(w) \times T[w]$$

where $tf(w,t)$ denotes term frequency of w in text t, $idf(w)$ denotes inverse document frequency of w across all Wikipedia data, and $T[w]$ represents the row vector corresponding to w in matrix T.

To capture main semantic concepts, we sort vector V_t by element scores in descending order and select the top n elements as the final explicit semantic analysis result, formally represented as $ESA = \{p(a_1|t), p(a_2|t), ..., p(a_n|t)\}$, where $p(a_i|t)$ represents the semantic relevance between article a_i and text t.

4.2 Semantic Association and Diffusion of Category Nodes and Path Solving For text t and category c_j in HG, if there exists a_i such that $c_j \in CS(a_i)$, then category c_j is called an initial seed category node (seed node) of text t on tree-like graph HG. Let $w(c_j|t)$ denote the relevance weight of seed node c_j to text t, calculated as:

$$w(c_j|t) = \sum_{a_i \in ESA} \frac{p(a_i|t)}{|CS(a_i)|}$$

where $|CS(a_i)|$ represents the size of the category set associated with article a_i. This formula shows that a seed node' s weight is proportionally accumulated from ESA scores of its associated articles. To ensure seed node weights sum to 1, we normalize them:

$$w'(c_j|t) = \frac{w(c_j|t)}{\sum_{c_i \in \Delta} w(c_i|t)}$$

where Δ denotes the set of all seed nodes. For completeness, if $c_i \notin \Delta$, we set $w'(c_i|t) = 0$.

In representation, only seed categories directly associated with articles are observed, while intermediate nodes from root to seed nodes remain hidden in hierarchical tree GH. To solve for these intermediate nodes and their relevance values, we propose a backward diffusion method that propagates each node's relevance value to its parent nodes until reaching the root, with $p(root(G)|t) = 1$, meaning all seed node information ultimately converges at the root, and any text belongs to Wikipedia' s root category.

Let $p(c_i \leftarrow c_j|t)$ denote the information diffused from node c_j to node c_i, defined as:

$$p(c_i \leftarrow c_j|t) = p(c_j|t) \times \frac{count(c_j)}{\sum_{c_k \in parents(c_j)} count(c_k)}$$

where $count(c_i)$ represents the number of articles belonging to node c_i or its descendants. Then $p(c_i|t)$ is solved as:

$$p(c_i|t) = w'(c_i|t) + \sum_{c_j \in children(c_i)} p(c_i \leftarrow c_j|t)$$

This shows that node c_i' s semantic relevance to text t is determined by both directly associated articles' ESA weights and information transmitted from all child nodes. Starting from seed nodes and computing bottom-up yields relevance values for all intermediate and root nodes.

Then, starting from the root node and passing through all intermediate nodes with relevance > 0 until reaching seed nodes, we obtain all possible classification paths. For any classification path $path_k$, its semantic relevance to text t is:

$$PR(path_k|t) = \sum_{c_i \in path_k} p(c_i|t)$$

According to formula (8), we sort hierarchical classification paths by relevance in descending order and select the top N as candidate results for text t' s semantic path identification.

4.3 Optimized Selection of Hierarchical Classification Paths To ensure novelty and diversity of generated paths, we prune candidate paths by removing highly similar redundant paths, referencing the method in [11]. First, we build an undirected weighted graph $IG = \langle V, W \rangle$ from candidate path set based on text t, where each node $v_i \in V$ corresponds to a classification path $path_i$, and weight $w_i \in W$ is $PR(path_i|t)$. For any two nodes v_i, v_j and their corresponding paths $path_i$, $path_j$, if similarity $sim(path_i, path_j)$ exceeds a threshold, graph IG contains undirected edge $(v_i, v_j) \in E$.

We select independent paths using the following greedy strategy: (1) Select the node v with maximum weight from GI as an effective path, mark it, delete its adjacent nodes and edges, and add v to queue Q; (2) Repeat until all nodes in GI are selected or deleted.

Queue Q then stores all independent hierarchical paths sorted by semantic relevance. Calculating similarity between any two paths is crucial:

$$sim(p_1, p_2) = \frac{\sum_{k=1}^{L}(L-k+1) \times sim(c(p_1,k), c(p_2,k))}{\max(|p_1|, |p_2|)}$$

where $L = \min(|p_1|, |p_2|)$, $c(p,k)$ denotes the k-th node in path p, and $sim(c_1, c_2)$ denotes similarity between two categories. For simplicity, we define $sim(c_1, c_2) = 1$ when $c_1 = c_2$, and 0 otherwise.

5. Experiments

5.1 Experimental Data We selected Chinese Wikipedia export data from June 2015, "zhwiki-20150602-pages-articles-multistream.xml.bz" , containing 2,648,029 pages: 55.93% article pages, 7.47% category pages, and 36.60% other resource pages (attachments, images, etc.). After data cleaning, we retained 184,968 article pages and 176,484 category pages for building the ESA model and hierarchical tree-like graph, respectively.

The cleaned Wikipedia category graph GW contains 176,484 nodes and 335,329 edges. After processing with the tree-like graph construction algorithm to remove cycles and isolated nodes, the final hierarchical classification graph GH contains 171,681 nodes and 220,861 edges—97.28% and 65.86% of GW, respectively—essentially preserving original category names while eliminating numerous redundant paths.

To construct the test set, we removed 184,968 training articles from Wikipedia and randomly sampled 1.5% from the remaining 637,911 non-redirect articles, excluding those with fewer than 50 characters. The final test set contains 6,629 articles saved in XML format, each containing page ID, title, text (with tags removed), and associated categories.

5.2 Evaluation Metrics For a test article a_i, let $CS(a_i)$ denote its original category set in Wikipedia, and $PS(a_i) = \{path_1, path_2, ..., path_m\}$ denote the hierarchical classification path set generated by our method. We define relevance R between article a_i and any path $path_j$ as:

$$R(a_i, path_j) = \max_{c \in CS(a_i)} rel(path_j, c)$$

where $rel(path, c)$ measures relevance between category c and path $path$:

$$rel(path, c) = \frac{mnp(path, c)}{dis(path, c)}$$

$dis(path, c)$ represents the distance from category node c to path $path$ in Wikipedia' s category graph, and $mnp(path, c)$ denotes the matched node position in $path$ when the distance is minimized.

Further, let $R(a_i, k)$ denote the average relevance between article a_i and the top k generated paths, abbreviated as $R@k$:

$$R(a_i, k) = \frac{1}{k} \sum_{j=1}^{k} R(a_i, path_j)$$

5.3 Results and Analysis Using article titles and the first 300 characters of main text as free text, we computed explicit concept vectors, retaining the top 20 concepts to generate seed category nodes and hierarchical semantic path sets. shows two examples from the test set ("Chinese Classical Texts" and "Adjacency Matrix") with manually assigned categories and automatically generated top 5 paths, including each path' s relevance R and k-average relevance ($k \in [1, 5]$).

The results demonstrate that our method can appropriately locate text content within the hierarchical classification knowledge system. The generated hierarchical paths can reflect main semantic information from different perspectives, showing high correlation with human-annotated fine-grained categories and providing effective references for article categorization.

To reflect overall performance, we computed k-average relevance for top k paths against test articles' original categories. Results for different k values are shown in [Figure 2: see original paper]. The right side shows $R@k$ for $k = 1$ to 5 across the entire test set; the left curve shows the overall trend for $k = 1$ to 20. The mean relevance decreases significantly as k increases, indicating that identification results are properly sorted by semantic relevance. When $k = 1$, the average relevance reaches 0.541, meaning the first path matches human-annotated categories in over half of cases. Lower relevance for some test articles stems from method limitations and data quality issues—ESA representation introduces noise, and manually annotated categories may be incomplete (see), causing highly semantically relevant paths to receive low test scores.

6. Conclusion

This paper proposes a semantic hierarchical path identification method based on Wikipedia. The method first represents free text as Wikipedia concept vectors using Explicit Semantic Analysis, then associates these vectors with the hierarchical tree-like graph through article-category membership relationships. Through semantic diffusion from seed category nodes to the root and top-down path solving and optimization, it achieves hierarchical path labeling for arbitrary texts. Experimental results show high correlation between automatically generated paths and human-annotated categories.

Future research includes: (1) Exploring new information diffusion calculation methods for category nodes to further improve hierarchical path identification; (2) Applying hierarchical path identification technology to text mining tasks such as similarity computation and classification.

References

[1] Wu Jiangning, Liu Qiaofeng. Research on Graph Structure Based Method for Chinese Text Representation [J]. Journal of the China Society for Scientific and Technical Information, 2010, 29(4): 618-624.

[2] Blei D M, Ng A Y, Jordan M I. Latent Dirichlet Allocation [J]. Journal of Machine Learning Research, 2003, 3: 993-1022.

[3] He Li, Jia Yan, Han Weihong, et al. Research and Development of Large Scale Hierarchical Classification Problem [J]. Chinese Journal of Computers, 2012, 35(10): 2101-2115.

[4] Silla C N, Freitas A A. A Survey of Hierarchical Classification Across Different Application Domains [J]. Data Mining and Knowledge Discovery, 2011, 22(1-2): 31-72.

[5] Zhang C, Xue G R, Yu Y, et al. Web-scale Classification with Naive Bayes [C]. In: Proceedings of the 18th International Conference on World Wide Web, Madrid, Spain. 2009.

[6] Medelyan O, Milne D, Legg C, et al. Mining Meaning from Wikipedia [J]. International Journal of Human-Computer Studies, 2009, 67(9): 716-754.

[7] Muchnik L, Itzhack R, Solomon S, et al. Self-emergence of Knowledge Trees: Extraction of the Wikipedia Hierarchies [J]. Physical Review E, 2007, 76(1): 1-12. DOI: http://dx.doi.org/10.1103/PhysRevE.76.016106.

[8] Gabrilovich E, Markovitch S. Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis [C]. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence. 2007: 1606-1611.

[9] Aggarwal N, Asooja K, Buitelaar P. Exploring ESA to Improve Word Relatedness [C]. In: Proceedings of the 3rd Joint Conference on Lexical and Computational Semantics. 2014: 51-56.

[10] Milne D N, Witten I H. A Knowledge-Based Search Engine Powered by Wikipedia [C]. In: Proceedings of the 23rd ACM International Conference on Information and Knowledge Management. 2007.

[11] Chakrabarti D, Mehta R. The Paths More Taken: Matching DOM Trees to Search Logs for Accurate Webpage Clustering [C]. In: Proceedings of the 19th International Conference on World Wide Web. 2010.

Conflict of Interest Statement: The author declares no conflict of interest.

Supporting Data:
[1] Xia Tian. wiki6629.zip. A test dataset of 6,629 Wikipedia articles in XML format, each containing title, ~300-character excerpt, and category information.

[2] Xia Tian. generated_{paths}.zip. Test dataset with semantic paths generated by our method.
[3] Xia Tian. data_{{{code}}_{{url}}}.txt. Download links for Wikipedia dataset and code.

Received: 2015-11-16
Revised: 2015-12-21

Summon Discovery Service Now Provides Altmetric Information

ProQuest subsidiary Ex Libris recently announced the integration of Altmetric into the Summon discovery service, significantly enriching user experience and improving content discovery. This collaboration between ProQuest and Altmetric enables researchers to learn about online sharing, commenting, and discussion of research outputs with a single click.

Libraries can activate the Altmetric badge in Summon discovery service, which displays an Altmetric badge for each search result (e.g., an article). Users can click the badge to explore related discussions sourced by Altmetric from mainstream media, Wikipedia, blogs, social networks, reference managers, post-publication peer review forums, and other online communities.

Regarding this integration, Shlomi Kringel, VP of Discovery and Delivery Solutions at Ex Libris, stated: "Improving the research experience by increasing exposure of scholarly content and enhancing search result value is an important goal for all our services. Adding Altmetric badges to Summon discovery service helps users more easily assess a research output' s impact in academia and among readers, and understand the reasons behind this impact."

Altmetric founder Euan Adie added: "We are delighted that ProQuest has integrated Altmetric into Summon discovery service. We hope that as more users see online activities related to research outputs, they will also engage more actively in ongoing academic discussions within their fields."

Libraries can activate Altmetric badges without subscribing to Altmetric.com, and the badges will appear in search results across ProQuest platforms including 360 Links, Ex Libris Primo, and Summon discovery service.

(Compiled from: http://www.proquest.com/about/news/2016/Altmetric-data-now-available-in-the-Summon-Discovery-Service.html)
(Journal News)

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

Source: ChinaXiv —Machine translation. Verify with original.