Hyponym Extraction from the Web based on Property Inheritance of Text and Image Features

Shun Hattori
College of Information and Systems
Muroran Institute of Technology
27-1 Mizumoto-cho, Muroran, Hokkaido 050-8585, Japan
Email: hattori@csse.muroran-it.ac.jp

***Abstract*—Concept hierarchy knowledge, such as hyponymy and meronymy, is very important for various Natural Language Processing systems. While WordNet and Wikipedia are being manually constructed and maintained as lexical ontologies, many researchers have tackled how to extract concept hierarchies from very large corpora of text documents, such as the Web, not manually, but automatically. However, their methods are mostly based on lexico-syntactic patterns as not necessary but sufficient conditions of hyponymy and meronymy, so they can achieve high precision but low recall when using stricter patterns or they can achieve high recall but low precision when using looser patterns. Therefore, we need necessary conditions of hyponymy and meronymy to achieve high recall and not low precision. The previous papers have assumed "Property Inheritance" from a target concept to its hyponyms and/or "Property Aggregation" from its hyponyms to the target concept to be necessary and sufficient conditions of hyponymy, and proposed several methods to extract hyponymy relations from the Web, based on property inheritance and/or property aggregation of text features such as meronyms and behavior. This paper proposes a method to acquire hyponymy relations from the Web, based on property inheritance of not only text features, but also image features for each conceptual word.**

Keywords-hyponymy; meronymy; concept hierarchy; Web mining; image analysis; property inheritance; typical image.

I. INTRODUCTION

Concept hierarchies, such as hyponymy (is-a) and meronymy (has-a) relations, are very fundamental for various Natural Language Processing (NLP) systems. For example, query expansion in information retrieval [1–4] or image retrieval [5], question answering [6], machine translation, object information extraction by text mining [7], Sense-based Object-name Search (SOS) [8], etc. Our appearance information extraction [7] is based on the heuristics that an appearance description about a target object-name (e.g, "kingfisher") often has a pair of an appearance descriptor and its hypernym (e.g., "blue bird" and "beautiful bird") or its meronym (e.g., "blue wings" and "long beak").

While WordNet [9] and Wikipedia [10] are being manually constructed and maintained as lexical ontologies at the cost of much time and effort, many researchers have tackled how to extract concept hierarchies from very large corpora of text documents, such as the Web, not manually, but automatically [11–14]. However, their methods are mostly based on lexico-syntactic patterns as sufficient but not necessary conditions of concept hierarchies. Therefore, they can achieve high precision but low recall when using stricter patterns (e.g., "x such as y" and "y is a kind of x") or they can achieve high recall but low precision when using looser patterns (e.g., "y is a/an x").

To achieve high recall and not low precision, our previous works [15–18] have assumed "Property Inheritance" from a target concept to its hyponyms (i.e., subordinate concepts for the target concept) and/or "Property Aggregation" from its hyponyms to the target concept to be necessary and sufficient conditions of hyponymy, and proposed several methods to extract hyponymy relations from the Web by text mining techniques, based on property inheritance and/or property aggregation of text features such as meronyms and behavior-words. The former assumption is to utilize the other semantic relations surrounding the subordinate (hyponymy) relation between a target concept and its hyponym candidate, i.e., superordinate relationships (hypernymy) and coordinate relationships (including synonymy and antonymy), and to improve a weighting of hyponymy extraction by using multiple property inheritances not only from the target concept to its hyponym candidate, but also between the other pairs of concepts (e.g., from a hypernym of the target concept to its hyponym candidate and/or from the target concept to a coordinate concept of its hyponym candidate). The latter assumption is to improve a weighting of property extraction by using property aggregation to each target concept from its typical hyponyms.

To make our previous method more robust, this paper utilizes not only Web text, but also Web images, and proposes a method to acquire hyponymy relations from the Web, based on property inheritance of not only text features, but also image features for each conceptual word.

The remainder of the paper is organized as follows. Section II proposes a method to extract hyponymy relations from the Web, based on property inheritance of not only text features, but also image features. Section III shows some experimental results to validate the proposed method. Finally, we conclude this paper in Section IV.

 ISBN: 978-1-61208-240-0

II. METHOD

This section introduces our previously published basic method [15] to extract hyponymy relations from the Web by using not only lexico-syntactic patterns with a target word and its hyponym candidate as sufficient but not necessary conditions of hyponymy, but also "Property Inheritance" (of text features such as meronyms and behavior-words) from the target word to its hyponym candidate as their necessary and sufficient conditions. To make the basic method more robust, this section proposes a method to acquire hyponymy relations from the Web, based on property inheritance of not only text features, but also typical image features for each concept by using not only Web text, but also Web images.

Our methods for automatic hyponym extraction from the Web are based on the following basic assumption of "Property Inheritance". Let C be the universal set of concepts (conceptual words). This paper assumes that if and only if a concept $x \in C$ is a hypernym (superordinate) of a concept $y \in C$, in other words, the concept y is a hyponym (subordinate) of the concept x, then the set of properties that the concept y has, $P(y)$, completely includes the set of properties that the concept x has, $P(x)$, and the concept y is not equal (equivalent) to the concept x.

$$\text{isa}(y, x) = 1 \Leftrightarrow P(y) \supseteq P(x) \text{ and } y \neq x,$$

$$P(c) = \{p \in P \mid \text{has}(p, c) = 1\},$$

where P stands for the universal set of properties and $\text{has}(p, c) \in \{0, 1\}$ indicates whether or not a concept $c \in C$ has a property $p \in P$,

$$\text{has}(p, c) = \begin{cases} 1 & \text{if a concept } c \text{ has a property } p, \\ 0 & \text{otherwise.} \end{cases}$$

In other words, if and only if a concept y is a hyponym of a concept x, then the number of properties that both concepts x and y share is equal to the number of properties that the superordinate concept x has (and is less than the number of properties that the subordinate concept y has).

$$\text{isa}(y, x) = \begin{cases} 1 \text{ if } \sum_{p \in P} \text{has}(p, y) \cdot \text{has}(p, x) = \sum_{p \in P} \text{has}(p, x), \\ 0 \text{ if } \sum_{p \in P} \text{has}(p, y) \cdot \text{has}(p, x) < \sum_{p \in P} \text{has}(p, x). \end{cases}$$

It is essential for automatic hyponym extraction from the Web based on the above basic assumption to calculate the binary value $\text{has}(p, c) \in \{0, 1\}$ for any pair of a property $p \in P$ and a concept $c \in C$ accurately. However, it is not easy, and we can calculate only the continuous value $\text{has}^*(p, c) \in [0, 1]$ by using Web text and/or Web images in this paper. Therefore, we suppose that the ratio of the number of properties that a concept $y \in C$ inherits from a target concept $x \in C$ to the number of properties that the target concept x has,

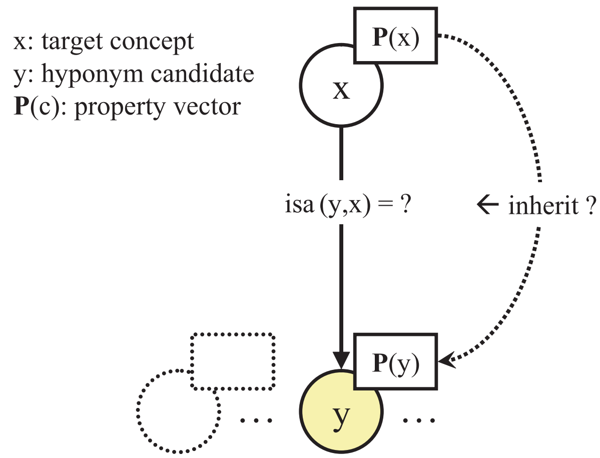


Figure 1. Hyponym Extraction based on Property Inheritance.

$$\frac{\sum_{p \in P} \text{has}^*(p, y) \cdot \text{has}^*(p, x)}{\sum_{p \in P} \text{has}^*(p, x) \cdot \text{has}^*(p, x)},$$

can measure how suitable the concept y is for a hyponym of the target concept x, $\text{isa}^*(y, x)$, as an approximation of whether or not the concept y is a hyponym of the target concept x, $\text{isa}(y, x)$. Then, the concept y would be considered to be a hyponym of the target concept x when the ratio is enough near to one (or greater than a threshold value), while the concept y would be considered to be not a hyponym of the target concept x when the ratio is not near to one (or less than a threshold value).

When a target concept $x \in C$ is given, our proposed method based on property inheritance executes the following four steps to extract its hyponyms from the Web. First, a set of candidates for its hyponyms of the target concept x, $C(x)$ is collected from the Web as exhaustively as possible. Second, the continuous value $\text{has-txt}^*(p, c)$ or $\text{has-img}^*(p, c)$ for each pair of a property (text or image feature) $p \in P$ and a concept $c \in C$ (the target concept x or its hyponym candidate $y \in C(x)$) is calculated by analyzing not only Web text, but also Web images. Last, the continuous value $\text{isa-PI}^*_n(y, x)$ for each pair of the target concept x and its hyponym candidate $y \in C(x)$ is calculated based on property inheritance of the top n typical properties of the target concept x to its hyponym candidate y, and then a set of its top k hyponym candidates ordered by their weight would be outputted to the users.

Step 1. Hyponym Candidate Collection

A set of hyponym candidates of the target concept x, $C(x)$ needs to be collected from the Web as exhaustively as possible and enough precisely. If $C(x)$ should be set to the universal set of concepts, C, its recall could equal to 1.0 (the highest) but its precision would nearly equal to 0.0 (too low). Meanwhile, if $y \in C(x)$ is collected from some sort of corpus of text documents by using too strict lexico-syntactic pattern (e.g., "y is a kind of x"), its precision is enough high but its recall is too low in most cases. Therefore, this paper uses not too strict but enough strict lexico-syntactic pattern of hyponymy to collect the set from the Web as exhaustively as possible and enough precisely. Any noun phrase y whose lexico-syntactic pattern "y is a/an x" exists at least once in the title and/or summary text of the top 1000 search results by submitting a phrase "is a/an x" as a query to Yahoo! Web Search API [19] is inserted into $C(x)$ as a hyponym candidate of the target concept x.

Step 2. Text Property Extraction

In our previous papers [15–18], typical properties p such as meronyms and behavior-words of each concept (the target concept x or its hyponym candidate $y \in C(x)$) are extracted from only Web text as precisely as possible by using an enough strict lexico-syntactic pattern "c's p" as a sufficient condition of meronymy. The continuous value has-txt$^*(p, c)$ of a text property p for each concept c is defined as follows:

$$\text{has-txt}^*(p, c) := \frac{\text{if}(\texttt{["}c\text{'s } p\texttt{"]})}{\text{if}(\texttt{["}c\text{'s}\texttt{"]})} \in [0, 1],$$

where if$(\texttt{[}q\texttt{]})$ stands for the number (frequency) of Web images that meet a query condition q in such a corpus as the Web. This paper calculates it by submitting each query to Yahoo! Image Search API [20]. Note that has-txt$^*(p, c)$ is not a binary value $\{0, 1\}$ but a continuous value $[0, 1]$, so it cannot indicate whether or not a concept c has a property p but how typical the property p is of the concept c.

Step 3. Image Property Extraction

This paper considers not only Web text, but also Web images, and extracts not only text features such as meronyms and behavior-words, but also image features of typical images as typical properties for each concept c. The top 100 search results by submitting a phrase "c" as a query to Yahoo! Image Search API are reranked based on the VisualRanking algorithm [21] to acquire more typical images of the target concept c. The continuous value has-img$^*(p, c)$ of an image feature p for each concept c is defined as follows by using the top k $(= 10)$ reranked images $I_k(c)$:

$$\text{has-img}^*(p, c) := \frac{\sum_{i \in I_k(c)} \text{prop}(p, i)}{k} \in [0, 1],$$

where prop(p, i) stands for the proportion of a HSV or SIFT [22] color-feature p in a Web image i.

Step 4. Candidate Weighting by Property Inheritance

To filter out noisy hyponym candidates of the target concept x, each hyponym candidate $y \in C(x)$ is assigned the weight isa-PI$^*_n(y, x)$, based on not only the inheritance inherit-txt$^*_n(y, x)$ of the top n typical text features, but also the inheritance inherit-img$^*_n(y, x)$ of the top n typical image features from the target concept x:

$$\begin{aligned}\text{isa-PI}^*_n(y, x) \;&:=\; (1-\alpha)\cdot \text{inherit-txt}^*_n(y, x)\\ &\quad\;+\; \alpha \cdot \text{inherit-img}^*_n(y, x),\end{aligned}$$

$$\text{inherit-txt}^*_n(y, x) := \frac{\sum_{p \in P^t_n(x)} \text{has-txt}^*(p, y) \cdot \text{has-txt}^*(p, x)}{\sum_{p \in P^t_n(x)} \text{has-txt}^*(p, x) \cdot \text{has-txt}^*(p, x)},$$

$$\text{inherit-img}^*_n(y, x) := \frac{\sum_{p \in P^i_n(x)} \text{has-img}^*(p, y) \cdot \text{has-img}^*(p, x)}{\sum_{p \in P^i_n(x)} \text{has-img}^*(p, x) \cdot \text{has-img}^*(p, x)},$$

where $\alpha \in [0, 1]$ stands for a certain combination parameter.

III. Experiment

This section shows some experimental results to validate the proposed method to extract hyponymy relations from the Web, based on "Property Inheritance" of not only typical text features, but also typical image features for each concept, compared with a traditional lexico-syntactic pattern based hyponym extraction.

Figure 2 compares the average Precison-Recall curves by the proposed hybrid hyponym extraction $(\alpha = 0.5, n = 10)$ by using not only Web text, but also Web images, the previous hyponym extraction $(\alpha = 0, n = 10)$ by using only Web text, and a lexico-syntactic pattern based hyponym extraction for several kinds of target conceptual words such as "bird" and "flower". The MAP (Mean Average Precision) of the proposed hybrid hyponym extraction is the best.

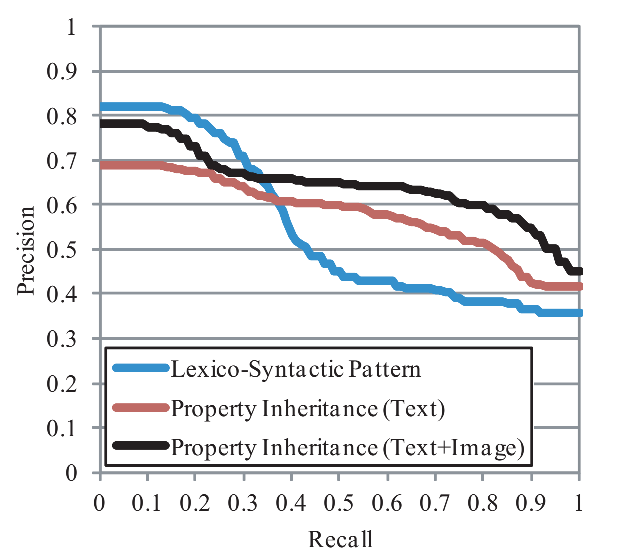


Figure 2. Precison-Recall of Hyponym Extraction based on Property Inheritance of Text and/or Image Features.

Table I
TOP 18 HYPONYMS EXTRACTED FROM THE WEB FOR "PENGUIN".

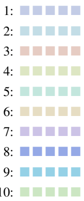
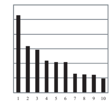
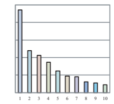
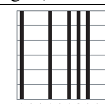
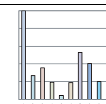
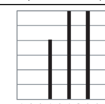
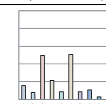
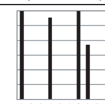
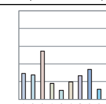
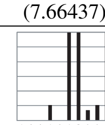
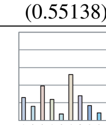
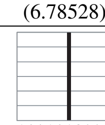
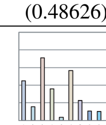
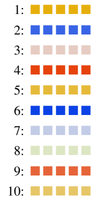
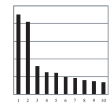
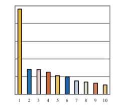
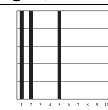
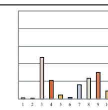
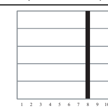
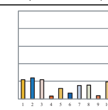
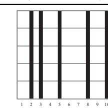
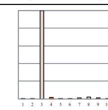
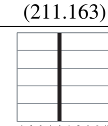
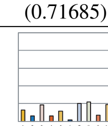
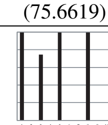
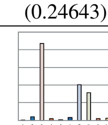
		1: photostream 2: iceberg 3: revenge 4: beak 5: poems 6: head 7: feet 8: nest 9: lair 10: eye		penguin (——)	Top 10 Typical Text Features  (——)	Top 10 Typical Color Features  (——)
Rank	Syntactic Pattern	Text ($\alpha = 0.0$)	Image ($\alpha = 1.0$)	Text+Image ($\alpha = 0.5$)		
1	animal (196)	**gentoo penguin** (16.1158)	**gentoo penguin** (1.02559)	**gentoo penguin** (8.57070)	 (16.1158)	 (1.02559)
2	favorite animal (128)	**yellow-eyed penguin** (11.0503)	**emperor penguin** (1.02353)	**yellow-eyed penguin** (5.72191)	 (11.0503)	 (0.39347)
3	**tux** (86)	**little blue penguin** (7.66437)	**baby penguin** (0.94967)	**little blue penguin** (4.10788)	 (7.66437)	 (0.55138)
4	book (50)	**king penguin** (6.78528)	**chinstrap penguin** (0.89687)	**king penguin** (3.63577)	 (6.78528)	 (0.48626)
5	character (48)	**magellanic penguin** (6.53255)	pc (0.86006)	**magellanic penguin** (3.61665)	 (6.53255)	 (0.70074)
6	**hoiho** (43)	**emperor penguin** (4.74698)	**african penguin** (0.85294)	**emperor penguin** (2.88526)	(4.74698)	(1.02353)
7	pablo (43)	**baby penguin** (3.65535)	sutter (0.78754)	**baby penguin** (2.30251)	(3.65535)	(0.94967)
8	friend (37)	**chinstrap penguin** (2.67442)	inch serving platter (0.784431)	**chinstrap penguin** (1.78565)	(2.67442)	(0.89687)
9	**spheniscus mendiculus** (28)	mr. flibble (2.37420)	google (0.77023)	mr. flibble (1.31628)	(2.37420)	(0.25837)
10	avatar (27)	**macaroni penguin** (2.08840)	**adelie penguin** (0.76570)	**macaroni penguin** (1.24987)	(2.08840)	(0.41134)
11	hot dog (24)	favorite animal (1.25312)	political activist banksy (0.75514)	**royal penguin** (0.91535)	(1.17650)	(0.65420)
12	uguin (22)	**royal penguin** (1.17650)	ty avalanche (0.75316)	favorite animal (0.86913)	(1.25312)	(0.48515)
13	**galapagos penguin** (18)	**little penguin** (0.93420)	video (0.73873))	**adelie penguin** (0.84118)	(0.91665)	(0.76570)
14	god (18)	**adelie penguin** (0.91665)	**tux** (0.73620)	**little penguin** (0.74092)	(0.93420)	(0.54764)
15	**snares islands penguin** (17)	vigilance (0.86808)	**antarctic penguin** (0.73326)	**tux** (0.66230)	(0.58840)	(0.73620)
16	heart (15)	misaki (0.79266)	**linux mascot tux** (0.71541)	**african penguin** (0.65259)	(0.45224)	(0.85294)
17	poet (10)	wentworth miller (0.78618)	free pablo (0.70746)	vigilance (0.63249)	(0.86808)	(0.39691)
18	**gentoo penguin** (9)	enemies (0.64338)	abbath (0.70085)	misaki (0.61684)	(0.79266)	(0.44102)

Table II
TOP 18 HYPONYMS EXTRACTED FROM THE WEB FOR “SUNFLOWER”.

		1: love 2: garden 3: field 4: seeds 5: life 6: smile 7: seed 8: head 9: leaves 10: spiral		sunflower (——)	Top 10 Typical Text Features  (——)	Top 10 Typical Color Features  (——)
Rank	Syntactic Pattern	Text ($\alpha = 0.0$)	Image ($\alpha = 1.0$)	Text+Image ($\alpha = 0.5$)		
1	seed (208)	jill jack (480.541)	yellow (1.22165)	jill jack (240.390)	 (480.541)	 (0.23893)
2	favorite flower (52)	**tall sunflower** (213.538)	**girasol** (1.05447)	**tall sunflower** (106.943)	 (213.538)	 (0.34733)
3	district (42)	present invention (211.163)	**marigold** (0.86360)	present invention (105.940)	 (211.163)	 (0.71685)
4	navy blue field (23)	independent person (75.6619)	second parent sunflower plant (0.85420)	independent person (37.9542)	 (75.6619)	 (0.24643)
5	favorite thing (22)	**mirasol** (48.8920)	pairwise disjoint sets (0.83355)	**mirasol** (24.5911)	 (48.8920)	 (0.29011)
6	logo (21)	larva (42.6859)	sol (0.81621)	larva (21.4258)	(42.6859)	(0.16564)
7	yellow (12)	**common sunflower** (40.2172)	known prior art (0.75949)	**common sunflower** (20.4846)	(40.2172)	(0.75199)
8	hell (11)	favorite flower (35.4822)	**common sunflower** (0.75199)	favorite flower (17.8299)	(35.4822)	(0.17753)
9	sunbutter (11)	lead singer (19.1564)	inflorescence (0.73851)	lead singer (9.71413)	(19.1564)	(0.27188)
10	seal (10)	species (15.7655)	present invention (0.71685)	species (8.03862)	(15.7655)	(0.31178)
11	happiness (9)	aliya (13.6240)	imidazolinone herbicide (0.66606)	aliya (6.97572)	(13.6240)	(0.32740)
12	flower variation (8)	g-dragon (11.7593)	silver necklace (0.61189)	g-dragon (6.00615)	(11.7593)	(0.25297)
13	friend (7)	**jerusalem artichoke** (11.6205)	**maximilian’s sunflower** (0.60568)	**jerusalem artichoke** (5.93293)	(11.6205)	(0.24531)
14	colour (6)	happiness (10.4684)	sunbutter (0.60099)	happiness (5.39702)	(10.4684)	(0.32564)
15	disjoint sets (6)	arapahoe (9.35538)	**helianthus annuus** (0.59916)	arapahoe (4.89790)	(9.35538)	(0.44043)
16	**jerusalem artichoke** (6)	mommy (6.20476)	size (0.59646)	mommy (3.25753)	(6.20476)	(0.31031)
17	pervenets (6)	fabric (5.60841)	disjoint sets (0.55639)	fabric (2.94874)	(5.60841)	(0.28907)
18	g-dragon (4)	larry (3.25074)	crepe back satin (0.55406)	larry (1.82185)	(3.25074)	(0.39296)

IV. Conclusion

To achieve high recall and not low precision in automatic hyponym extraction from the Web, our previous work has assumed "Property Inheritance" from a target concept to its hyponyms and/or "Property Aggregation" from its hyponyms to the target concept to be necessary and sufficient conditions of hyponymy, and proposed several methods to extract hyponymy relations from the Web, based on property inheritance and/or property aggregation of text features such as meronyms and behavior-words. To make our previous method more robust, this paper has utilized not only Web text, but also Web images, proposed a method to acquire hyponymy relations from the Web, based on property inheritance of not only text features, but also image features for each conceptual word, and validated the proposed method by showing some experimental results.

Acknowledgment

This work was supported in part by JSPS Grant-in-Aid for Young Scientists (B) "A research on Web Sensors to extract spatio-temporal data from the Web" (#23700129, Project Leader: Shun Hattori, 2011-2012).

References

[1] Mandala, R., Tokunaga, T., and Tanaka, H.: "The Use of WordNet in Information Retrieval," Proceedings of the COLING ACL Workshop on Usage of WordNet in Natural Language Processing, pp. 31–37 (1998).

[2] Hattori, S., Tezuka, T., and Tanaka, K.: "Activity-based Query Refinement for Context-aware Information Retrieval," Proceedings. of the 9th International Conference on Asian Digital Libraries (ICADL'06), LNCS vol. 4312, pp. 474–477 (2006).

[3] Hattori, S., Tezuka, T., Hiroaki, O., Oyama, S., Kawamoto, J., Tajima, K., and Tanaka, K.: "ReCQ: Real-world Context-aware Querying," Proceedings of the 6th International and Interdisciplinary Conference on Modeling and Using Context (CONTEXT'07), LNAI vol. 4635, pp. 248–262 (2007).

[4] Hattori, S.: "Alternative Query Discovery from the Web for Daily Mobile Decision Support," Proceedings of the 5th IADIS International Conference on Wireless Applications and Computing (WAC'11), pp. 67–74 (2011).

[5] Hattori, S.: "Hyponymy-Based Peculiar Image Retrieval," Int'l Journal of Computer Information Systems and Industrial Management (IJCISIM), MIR Labs, vol. 5, pp. 79–88 (2012).

[6] Fleischman, M., Hovy, E. and Echihabi, A.: "Offline Strategies for Online Question Answering: Answering Questions Before They Are Asked," Proc. 41st Annual Meeting of the Association for Computational Linguistics, pp. 1–7 (2003).

[7] Hattori, S., Tezuka, T., and Tanaka, K.: "Mining the Web for Appearance Description," Proceedings of the 18th International Conference on Database and Expert Systems Applications (DEXA'07), LNCS vol. 4653, pp. 790–800 (2007).

[8] Hattori, S. and Tanaka, K.: "Object-Name Search by Visual Appearance and Spatio-Temporal Descriptions," Proc. of the 3rd Int'l Conference on Ubiquitous Information Management and Communication (ICUIMC'09), pp. 63–70 (2009).

[9] Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. J.: "Introduction to WordNet: An On-line Lexical Database," International Journal of Lexicography, vol. 3, no. 4, pp. 235–312 (1993).

[10] Völkel, M., Krötzsch, M., Vrandecic, D., Haller, H., and Studer, R.: "Semantic Wikipedia," Proc. 15th Int'l Conference on World Wide Web (WWW'06), pp. 585–594 (2006).

[11] Hearst, M. A.: "Automatic Acquisition of Hyponyms from Large Text Corpora," Proc. 14th Int'l Conference on Computational Linguistics (COLING'92), vol. 2, pp. 539–545 (1992).

[12] Morin, E. and Jacquemin, C.: "Automatic Acquisition and Expansion of Hypernym Links," Computers and the Humanities, vol. 38, no. 4, pp. 363–396 (2004).

[13] Kim, H., Kim, H., Choi, I., and Kim, M.: "Finding Relations from a Large Corpus using Generalized Patterns," International Journal of Information Technology, vol. 12, no. 7, pp. 22–29 (2006).

[14] Ruiz-Casado, M., Alfonseca, E., and Castells, P.: "Automatising the Learning of Lexical Patterns: An Application to the Enrichment of WordNet by Extracting Semantic Relationships from Wikipedia," Data & Knowledge Engineering, vol. 61, no. 3, pp. 484–499 (2007).

[15] Hattori, S., Ohshima, H., Oyama, S., and Tanaka, K.: "Mining the Web for Hyponymy Relations based on Property Inheritance," Proceedings of the 10th Asia-Pacific Web Conference (APWeb'08), LNCS vol. 4976, pp. 99–110 (2008).

[16] Hattori, S. and Tanaka, K.: "Extracting Concept Hierarchy Knowledge from the Web based on Property Inheritance and Aggregation," Proc. of the 7th IEEE/WIC/ACM Int'l Conference on Web Intelligence (WI'08), pp. 432–437 (2008).

[17] Hattori, S. and Tanaka, K.: "Extracting Concept Hierarchy Knowledge from the Web by Property Inheritance and Recursive Use of Term Relationships," IPSJ Transactions on Databases (TOD40), vol. 1, no. 3, pp. 60–81 (2008).

[18] Hattori, S.: "Object-oriented Semantic and Sensory Knowledge Extraction from the Web," Web Intelligence and Intelligent Agents, In-Tech, pp. 365–390 (2010).

[19] Yahoo! Web Search API, http://search.yahooapis.jp/PremiumWebSearchService/V1/webSearch (2012).

[20] Yahoo! Image Search API, http://search.yahooapis.jp/PremiumImageSearchService/V1/imageSearch (2012).

[21] Jing, Y. and Baluja, S.: "VisualRank: Applying PageRank to Large-Scale Image Search," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 11, pp. 1877–1890 (2008).

[22] Lowe, D. G.: "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, vol. 60, no. 2, pp. 91–110 (2004).

 ISBN: 978-1-61208-240-0