

Semantic Relatedness Estimation using the Layout Information of Wikipedia Articles

Patrick Chan, Osaka University, Japan
Yoshinori Hijikata, Osaka University, Japan
Toshiya Kuramochi, Osaka University, Japan
Shogo Nishida, Osaka University, Japan

Abstract

Computing the semantic relatedness between two words or phrases is an important problem in fields such as information retrieval and natural language processing. Explicit Semantic Analysis (ESA), a state-of-the-art approach to solve the problem uses word frequency to estimate relevance. Therefore, the relevance of words with low frequency cannot always be well estimated. To improve the relevance estimate of low-frequency words and concepts, we apply regression to word frequency, its location in an article, and its text style to calculate the relevance. The relevance value is subsequently used to compute semantic relatedness. Empirical evaluation shows that, for low-frequency words, our method achieves better estimate of semantic relatedness over ESA. Furthermore, when all words of the dataset are considered, the combination of our proposed method and the conventional approach outperforms the conventional approach alone.

Keywords: semantic relatedness estimation, ESA, Wikipedia article, word frequency, layout information

INTRODUCTION

Semantic relatedness has a wide range of applications such as search, text summarization, and word sense disambiguation. It generally represents how much a word or phrase has a logical or causal connection to another word or phrase. To compute semantic relatedness, previous works made use of various linguistic resources such as WordNet and Wikipedia. They used the information about the graph built from a data source or the word frequency in a text corpus. This paper describes the result obtained using a new type of information, page layout information of Wikipedia, to improve the estimation of semantic relatedness.

Semantic relatedness applications take words or phrases as input, extract the highly semantically related words, and use the related words for their own needs. For example, a search engine generates a limited selection of results with the search terms alone, but if it uses the related words of the search terms as well, it can produce a diverse set of results.

Many approaches have been used to estimate semantic relatedness. Among these methods, Explicit Semantic Analysis (ESA) (Gabrilovich & Markovitch, 2007) is a Wikipedia-mining-based method that has recently become popular. It models a word as a vector of concepts, each of which is represented by a Wikipedia article. Each vector element shows the relevance between the word and the concept, which is the word's normalized TFIDF (Karen, 1972) value in the corresponding Wikipedia article. Finally, it calculates the semantic relatedness from the cosine similarity between two concept vectors. Not only word frequency but also layout information, such as the word text style and its location in an article, are probably related to the relevance between a word and a concept. For example, the topmost section of a Wikipedia article, regarded as the summary, usually contains carefully chosen, descriptive words explaining the concept. Bold words, normally used for emphasis, might be related more to the concept than other words. Therefore, we aim at obtaining a better relevance estimate using TFIDF and an article's layout information.

The following contributions are made by this paper.

- For words with low frequency, our proposed method achieves a higher correlation than that of ESA. Moreover, for all word pairs on the benchmark, the use of both our proposed method and ESA together results in a higher correlation than that of ESA.
- This report is the first of research work analyzing the page layout information of Wikipedia and using it to solve a research problem. The research problem we solve is semantic relatedness.
- We apply a more suitable statistical significance test to our result than our closely related work (Gabrilovich & Markovitch, 2007). Whereas Gabrilovich and Markovitch (2007) applied the test of statistically significant difference between two Pearson correlation coefficients on two Spearman's rank correlation coefficients and claimed statistically significant difference between

the Pearson correlations as the point of superiority of their method, we apply the statistical significance test designed for Spearman's rank correlation coefficients.

The rest of the paper is organized as described below. **Firstly, we present** a description of the related work. **Then, we give** an overview of Wikipedia layout information and **explain** the preprocessing method of Wikipedia articles and our extraction method of layout information. **The next section** includes an overall description and details of our proposed method. **We** elaborate the experimental dataset, procedure, and results. **Finally, we present some conclusions and future works.**

RELATED WORK

This section presents a review of previously established approaches to semantic relatedness problems. **Firstly,** we specifically examine recent approaches that use Wikipedia to compute semantic relatedness. **Then, we review** the approaches that use search queries as a source to compute semantic relatedness. **We also introduce** approaches that use other knowledge bases to compute semantic relatedness. **Lastly, we explain** our position in these research fields.

Wikipedia Mining Approaches

Previous approaches to computing semantic relatedness by Wikipedia mining have measured relatedness from two perspectives. One perspective uses a Wikipedia article as an independent concept. Another perspective constructs a graph with nodes connected when a Wikipedia link exists from one article to another or when the articles share a category. The respective approaches pursued by Gabrilovich & Markovitch (2007) and Radinsky *et al.* (2011) treat a Wikipedia article as a concept, whereas those by Ito *et al.* (2008), Strube & Ponzetto (2006) and Hecht & Witten (2008) build a graph from Wikipedia. The former approaches map a word to a set of concepts and ascertain the number of shared concepts. The latter approaches use graph distance to estimate the semantic relatedness. The computed semantic relatedness is then compared with WordSimilarity-353 (Finkelstein *et al.*, 2002), a dataset containing semantic relatedness rated by humans. Hereinafter, we present details of the research works introduced above.

Gabrilovich & Markovitch (2007) proposed a method called Explicit Semantic

Analysis (ESA) for computing semantic relatedness between words, which transforms each word into a vector of concepts where each concept is represented by a distinct Wikipedia article. It then sets the relevance between a word and a concept to be the normalized TFIDF value of the word in the Wikipedia article. Finally it computes the semantic relatedness between two words using the cosine similarity of the two corresponding concept vectors. This simple yet powerful method markedly outperforms all prior methods.

Radinsky *et al.* (2011) proposed a method called Temporal Semantic Analysis (TSA), which requires two datasets. The first is the Wikipedia database. The second is the newspaper articles of The New York Times. To compute the semantic relatedness between two words, each word is converted into a set of Wikipedia articles containing the word. Subsequently, for each of the article titles of the two sets, the number of their appearances in The New York Times over time is found. Finally, semantic relatedness between two words is decided by the number of article titles that correlates highly over time. At the time of this writing, this approach achieves the highest performance on the benchmark dataset.

Strube & Ponzetto (2006) proposed a method called WikiRelate! for computing the semantic relatedness of two words. This method first extracts the set of articles in which the words appear. Subsequently, it retrieves the categories of the pages. The computation of semantic relatedness is based on pages extracted and the paths found in the category tree. This approach is the first to use Wikipedia for computing semantic relatedness. However, it does not have high correlation with human ratings.

Milne & Witten (2008) proposed a method using links between articles of Wikipedia. They targeted only words that have a corresponding article in Wikipedia. Although ESA counts the number of occurrences of the target word in a Wikipedia article, their method counts the number of link occurrences. They measured the relatedness between any two Wikipedia articles using the articles linking to the two articles independently. The experimental result showed that their method outperforms WikiRelate! in terms of estimation accuracy. However, the accuracy of ESA is better than their method.

The last Wikipedia mining method proposed by Ito *et al.* (2008) matches a title of a Wikipedia article of the target word. It transforms two Wikipedia articles as two vectors of words and calculates their vector similarity. Then it builds a graph from the Wikipedia links and computes their graph distance. The semantic relatedness is judged by the vector similarity and the graph distance. This method is an improved version of a prior work (Nakayama *et al.*, 2007).

Recently, these methods for computing semantic relatedness have been applied to state-of-the-art search tasks. For example, Hecht *et al.* (2012) used semantic relatedness for realizing explanatory search task. They proposed a computing method of semantic relatedness including user aspects (identified relation to end users). They used WikiRelate!, Milne and Witten's method and ESA as basic methods for computing semantic relatedness.

Search-query-based Approaches

Two related works (Metzler *et al.*, 2007; Sahami & Heilman, 2006) specifically obtain semantic relatedness in the research area of search engine.

Metzler *et al.* (2007) attempted to find related queries when a search query is given. This research work has applied five lexical, probabilistic, and hybrid methods for extracting related search queries from a given search query. Their method requires search query logs in a search engine and compares them. The experiment uses MSN search query logs.

An approach by Sahami & Heilman (2006) also uses a set of search queries, and determines which search query pairs relate to one another. During the experiment, raters select queries from the dataset called 2003 Google Zeitgeist (<http://www.google.com/intl/en/press/zeitgeist.html>). Then the system calculates similarity between the selected query and all the existing queries from their designed kernel function that uses the returned documents of the given Google query. The calculated similarity is validated by human rating.

Other Knowledge-based Approach

Some studies use knowledge bases other than Wikipedia. Existing approaches are roughly classifiable as either graph-based or content-based. The former usually uses

graph-based lexical database such as WordNet (Budanitsky & Hirst, 2006). The latter usually uses text corpus on the Web (Reisinger & Raymond, 2010).

Agirre *et al.* (2009) used WordNet and text corpus on the Web for computing the semantic relatedness of words. They compared graph-based approaches and content-based approaches. For graph-based approaches, they computed the personalized PageRank over WordNet for each word, thereby obtaining a probability distribution over WordNet synsets. They created vectors using the probability distribution and calculated the similarity between vectors. For content-based approaches, they collected Web-based corpus consisting of four billion pages. They set a window around the target word and collected surrounding words. They calculated the number of occurrences of surrounding words and created vectors for the target word. Although they found that the combination of these two approaches improves the performance, that performance is not high, as that of Wikipedia-mining approaches.

Yih & Qazvinian (2012) proposed a hybrid method of text corpus, Web search results, and thesauruses for computing semantic relatedness. They created vectors using text corpus, Web search results, and thesauruses independently. The prediction is made by calculating the average cosine scores derived from these vector space models. For creating vectors using the text corpus, they used English Wikipedia and used a window for extracting surrounding 20 words of the target word. For creating vectors using Web search results, they used a commercial search engine, Bing, and retrieved the set of relevant snippets. For creating vectors using thesauruses, they used WordNet and Encarta. A word is represented in a synset vector. The experimental result showed that their hybrid method achieves high performance. However, it is not compared to pure Wikipedia-based methods.

Halawi *et al.* (2012) proposed a method using a text corpus. Unlike other studies using text corpora, the method represents a word in a low-dimensional space. The space is the latent space that reflects meanings of words within sentences. The method also incorporates the known relatedness of pairs of words as constraints. Wikipedia is used for obtaining the known relation of words. Their method achieves high accuracy in their experiments.

Our Research Position

Among research works using the Wikipedia dataset, ESA achieves the highest performance. Although TSA outperformed ESA on the estimation accuracy, TSA requires data from The New York Times, which is not freely available online. The association thesaurus construction method by Ito *et al.* only works on words that exactly match Wikipedia article titles. Research works using search queries use search query logs in search engines that are unobtainable by anybody but search engine administrators. Our method uses only the Wikipedia dataset and works on any word combination. We therefore propose an improved version of ESA and compare it with the original ESA.

Wikipedia Layout Information

This section offers an introduction to Wikipedia and its page layout information. It also explains the preprocessing procedure for Wikipedia articles and our method of extracting layout information.

Figure 1. Example of a Wikipedia article.


Koala's March

From Wikipedia, the free encyclopedia

Koala's March is a bite-sized cookie snack with a sweet filling inside. It is made by Lotte, and the product was first released in Japan, and was released as *Koala Yummies* in the [United States](#).

Contents

- 1 Safety
 - 1.1 Scandal
- 2 Target Consumers
- 3 See Also



Chocolate and Roast Almond flavor
Koala's March

Safety

There are rare cases of health related issues.

Scandal

In the [2008 Chinese milk scandal](#), the Koala's March cookies were recalled in both [Macau](#) and [Hong Kong](#).

Target Consumers

Consumers are mainly kids, but many adults enjoy it.

See Also

- [Pocky](#)
- [List of Japanese snack food](#)

Wikipedia and Its Page Layout

Wikipedia is a free, online encyclopedia that anybody can edit. The Wikipedia dataset, which contains all the Wikipedia articles in XML format, is freely available online. Articles of the dataset are written in Wiki code, which expresses how text should be displayed by the browser.

We introduce layout information of Wikipedia articles. Figure 1 portrays a sample Wikipedia article. The article title is shown at the top in large text. In Figure 1, the phrase “Koala's March” at the top of the page is the article title. Under the picture of the right side of the article is a file caption. In Figure 1, it is the sentence “Chocolate and Roast Almond flavor Koala's March”. Anchor text of a Wikipedia link is shown in blue. Some examples are “Lotte”, “United States”, “Macau”, etc. in the figure. Another presentation of layout information is a list. Two list items exist: “Pocky” and “List of Japanese snack foods”. Text styles of two kinds exist. The phrase “Koala's March”, which is the first two words of the summary (the first paragraph) is bold text. “Koala Yummies” at the third line is in italic. The section number is a numerical value that denotes the section in which a word appears. The smaller the section number, the closer the word is to the top of the article. The section level is a numerical value that denotes the depth of the section. The sections “Safety”, “Target Consumers”, and “See Also” are in section level 1, whereas the section “Scandal” is in section level 2.

Preprocessing Details

In our study, the raw Wikipedia dataset undergoes the same preprocessing procedure that was used in earlier research (Gabrilovich & Markovitch, 2007). Infrequent words and poorly developed articles are filtered out to yield a cleaner set of data. The process discards unnecessary or immature articles such as helper articles for editing the encyclopedia articles and articles with titles containing only numbers.

It uses the white space character and the characters /t, /n, /r, ` , ~, !, @, #, \$, /, %, ^, &, *, (,), _, =, +, |, [,] , ;, {, }, ,, ., /, ?, <, >, :, ‘, and “ to tokenize. It also applies Porter stemming (Porter, 1980) to do stemming of the acquired words.

Layout Information Extraction

Preprocessing is followed by the phase of extracting layout information. We extracted the headers, lists, text styles (bold/italic), inter-article links, and file links

from Wikipedia articles. Headers are extracted for tracking the section number and the section level. The extraction details are explained below.

Headers, lists, and text style can be extracted easily by regular expressions. The titles of a header, a subheader, a subsubheader, and a subsubsubheader are wrapped respectively by ‘==’, ‘===’, ‘====’, and ‘=====’. One can use the regular expression “==(.*?)==”, “===(.*?)===”, “====(.*?)====”, and “===== (.*?)=====” to extract the titles. While parsing the Wiki code, the number of headers encountered thus far is recorded for determining the section number that implies the word position. A list is begun by ‘*’. The regular expression “*(.*?)\$” is applied to extract the text of a list.

Any string that is surrounded by two single quotes is rendered as bold. Similarly, any string that is surrounded by three single quotes is rendered as italic. When a string is surrounded by five single quotes, it is both bold and italic. Regular expressions that extract these three scenarios are similar to those that extract the headers. Instead of the equal signs, single quotes are used for matching. Inter-article links have two Wiki code patterns. They are parsed separately. The first pattern is in the form of ‘[[<article name>|<anchor text>]]’ whereas the second pattern is ‘[[<article name>]]’. Because the Wikipedia parser, by design, has the browser display only the anchor text of the former case, the anchor text, without the article name, is extracted. For the latter case, the browser displays the article name itself. Therefore, we extract the article name. The regular expressions are, respectively, “[.+?|(.*?)” and “[((^|)+?)”.

Next, file links are extracted. First, file links are entered in three Wiki code formats, which are ‘[[File:<file name>|...|<caption>]]’, ‘[[Image:<file name>|...|<caption>]]’, and ‘[[Media:<file name>|...|<caption>]]’. The labels “File”, “Image”, and “Media” are programming functions that the Wikipedia parser uses to find out how to process the parameters. The three functions are interchangeable and behave similarly, so we specifically describe how extraction is done of the “File” label. The ‘...’ of ‘[[File:<file name>|...|<caption>]]’ stands for the numerous parameters fed to the File programming function. As a result, for the pattern ‘[[File:<file name>|...|<caption>]]’, we extract the last parameter and treat it as a file caption.

OUR PROPOSED METHOD

Method Outline

We use layout information for estimating the relevance between a word and a concept. However, it is not possible to estimate relevance if we do not know the degree to which each type of layout information is related to the relevance. To ascertain the relation between the layout information type and the relevance, we ask assessors to rate the relevance between a given word and a given concept and apply regression. The resultant regression formula enables us to use the layout information of a word in a Wikipedia article to compute the relevance between the word and the article's corresponding concept.

Figure 2. Flow diagram of our proposed method.

The three-step process of ESA is the conversion of a word to a concept vector, the calculation of the relevance value of the vector, and the computation of cosine similarity between two concept vectors. Our regression-based proposed method, shown in Figure 2, changes only step 2 of ESA. The relevance calculation of step 2 is done using a regression formula built from the training set:

$$\begin{aligned} \text{Relevance} = & \beta_0 + \beta_1 * \text{BOLD} + \beta_2 * \text{ITALIC} + \beta_3 * \text{ANCHOR} + \beta_4 * \text{CAPTION} + \beta_5 \\ & * \text{LIST} + \beta_6 * \text{HEIGHT} + \beta_7 * \text{DEPTH} + \beta_8 * \text{TFIDF} \end{aligned}$$

In this formula, *Relevance* stands for the dependent variable, capitalized words represent independent variables, and β are their respective weights. Whereas ESA sets the relevance as the word's normalized TFIDF value, which means setting β_8 as one and the rest of β s as zero, our method estimates the relevance as the trained regression formula.

We are unsure about which particular regression method fits our problem best, so we try three different methods and simply use the one providing the best result. We try out ordinary least squares linear regression (OLS), ordinal logistic regression (OLR), and support vector regression (SVR).

Independent Variables

Eight different independent variables and their definitions are listed below.

- **BOLD**: Returns 1 if the word is bold, and 0 otherwise.

- **ITALIC**: Returns 1 if the word is italic, and 0 otherwise.
- **ANCHOR**: Returns 1 if the word is part of the anchor text of an inter-article link, and 0 otherwise.
- **CAPTION**: Returns 1 if the word is part of a file caption, and 0 otherwise.
- **LIST**: Returns 1 if the word is part of a list, and 0 otherwise.
- **DEPTH**: Returns the section level of where the word is. In detail, returns 1, 2, 3, and 4 for words that are respectively under a main header, a subheader, a subsubheader, and a subsubsubheader .
- **HEIGHT**: Returns the section number of where the word is. In detail, returns 1 if the word is in the summary section and $(n + 1)$ if the word is under the n -th main header.
- **TFIDF**: Returns the value of the normalized TFIDF. It is the same value used by ESA. Eight different independent variables and their definitions are listed below.

We use the normalized TFIDF like ESA used (Gabrilovich & Markovitch, 2007). Its calculation method is explained below. Let n be the number of articles of Wikipedia, i be the index of a term, t_i be the i -th term, df_i be the document frequency (Karen, 1972), j be the index of a Wikipedia article, and a_j be the j -th Wikipedia article. The TFIDF of the i -th word at j -th article is defined below.

$$TFIDF(i, j) = tf(t_i, a_j) * \log \frac{n}{df_i}$$

Unlike the normal TFIDF, the function $tf(t_i, a_j)$ is defined here as $1 + \log(\text{num}(t_i, a_j))$ when t_i exists at least one time at a_j and 0 otherwise and $\text{num}(t_i, a_j)$ is the number of times t_i exists in an article a_j .

The normalized TFIDF of the i -th word in the j -th article is defined as

$$NormalizedTFIDF(i, j) = \frac{TFIDF(i, j)}{\sqrt{\sum_{i=1}^r TFIDF(i, j)^2}}$$

where r is the number of unique terms in a_j .

Independent Variable Settings

We apply regression in two different settings to address the case in which a word has more than one instance (a case when the same word occurs more than one time in a Wikipedia article), and each instance possesses different layout information. The first setting considers the layout information of all instances, whereas the second

setting uses the layout information of the most representative instance, which is the instance appearing first in the article. In the first setting, HEIGHT returns the section number of the topmost word. DEPTH returns either the deepest section level or the most shallow section level. BOLD, ITALIC, CAPTION, ANCHOR, and LIST returns 1 if at least one instance satisfies the respective property. The second setting considers the top word (the instance occurring first in the Wikipedia article), so all independent variable values are obtained from the top word.

Dependent Variable

We obtained human assessors' ratings by choosing 60 articles randomly from the Wikipedia dataset (the actual dataset is explained in [Section “Objectives and Experimental Settings”](#)) containing at least one bold word, one italic word, three words from the file caption, three inter-article links, and three words from list to ensure that layout information of various types is included.

The relevance of a small subset of words was then evaluated. It was costly to evaluate all words of a Wikipedia article. Therefore, we asked a human assessor to evaluate 30 words for each of the 60 articles. Again, the words to be evaluated were chosen in a way that layout information of all types were covered. We first randomly chose at least one, but up to three, words for bold words, italic words, words from file caption, words from inter-article links, words from list, and words from each available header level. Subsequently, we randomly chose words until we had collected 30 words.

Three human assessors, all graduate students, evaluated how relevant a concept was to a word on a seven-point scale. To obtain a good rating, assessors were obligated to look up the meaning of the evaluated word if they did not know its meaning. Finally, they were not permitted to assign any rating to a word or phrase that they were not confident about evaluating.

To increase the training set reliability, we deleted any word that was given no rating by any human assessor. Finally, 1,535 words remained. The average of the three (or two) users' ratings was used as the gold standard of relatedness between the word and the Wikipedia article.

We calculate β_i using the gold standard of relatedness and the actual layout

information (independent variables) of many pairs of word and Wikipedia article by regression (the regression algorithms we used are explained in [Subsection “Regression Method Comparison”](#)).

Combination with ESA

For computing the semantic relatedness of words that have low word frequency, we check to verify that our method works better than ESA. For words having high word frequency, however, ESA might perform better. We try a hybrid method in which our method is applied for the former case and ESA for the latter case. This method changes its applied method according to these words’ word frequency. We do experiments to find out the performance of our method and ESA when the words of different word frequencies are used. The experimental result shows which method the hybrid method should apply for words with a certain word frequency.

Actual Calculation

We give an example of words and a Wikipedia article to show how the relatedness between them is calculated. Ordinary least squares regression method (OLS) is used to explain the actual calculation (details of the results are presented in [Subsection “Regression Method Comparison”](#)). The β_i obtained by OLS are shown in the first line (“ β_i ” line) in Table 1. All words are used. The section level is set as the deepest here.

We use “Fujifilm X-series” (shown in Figure 3) as Wikipedia article and “camera” and “launch” as words for the explanation. Both “camera” and “launch” is included in this article. Also, “camera” is used in ANCHOR, CAPTION, LIST in this article. “Launch” is used in “LIST”. The numbers of occurrences of these words are shown in the second and third line in Table 1. Setting β_i as values in the first line in Table 1 and independent variables as values in the second or third line in Table 1 yields relevance values to this article (last column in Table 1). “Camera” is used many times and in many layout types. Therefore, it acquires higher relevance than “launch”.

Figure 3. Example of a Wikipedia article used for calculation.

Fujifilm X-series

From Wikipedia, the free encyclopedia

The **Fujifilm X-Series** range of **digital cameras** consists of the company's high-end digital cameras^[1] and is aimed professional and keen enthusiast photographers. It is part of the larger range of Fujifilm's **digital cameras**.

Models [[edit](#)]

- Fujifilm X-Pro1^[2] Mirrorless interchangeable-lens camera that uses the "X-Trans CMOS" sensor and the Fujifilm X-mount system of lenses. It was announced in January 2012 and launched in March 2012. At the time of launch, 3 prime lenses were available:
 - FUJINON XF18mmF2 R^[3] 18mm focal length (27mm 135 equivalent) F2.0-F16 aperture
 - FUJINON XF35mmF1.4 R^[4] 35mm focal length (53mm 135 equivalent) F1.4-F16 aperture
 - FUJINON XF60mmF2.4 R Macro^[5] 60mm focal length (91mm 135 equivalent) F2.4-F22 aperture
- On Sep. 6, 2012, Fujifilm announced two additional lenses for its X-mount:
 - FUJINON XF18-55mmF2.8-4 R LM OIS^[6] 18-55mm focal length (27-83mm 135 equivalent) F2.8-F4-F22 aperture
 - FUJINON XF14mmF2.8 R^[7] 14mm focal length (21mm 135 equivalent) F2.8-F22 aperture
- Fujifilm FinePix X100^[8] prime lens digital camera that uses a custom APS-C sized CMOS sensor and Hybrid Viewfinder. Announced at Photokina 2010, the X100 launched globally in March 2011.
- Fujifilm FinePix X100S^[3] A redesign of the X100, announced in 2013.
- **Fujifilm X10**^[10], **Fujifilm X20**^[11]
- Fujifilm X-S1^[12]
- **Fujifilm X-E1**^[13]
- Accessories:
 - M Mount Adaptor^[14] allows the use of a wide variety of M Mount lenses on X Mount camera bodies.
 - Wide Conversion Lens WCL-X100^[15] converts the X100 fixed lens from 23mm (35mm in 135 equivalent) fixed focal length to a 19mm wide angle (28mm in 135 equivalent).



See also [[edit](#)]

- Fujifilm digital cameras
- Fujifilm

Table 1. Coefficient β_i obtained using ordinary least squares regression (OLS) with all words used and the section level set as the deepest (first line). The number of occurrences of “camera” and “launch” (second or third line) in the Wikipedia article “Fujifilm X-series”.

	BOLD	ITALIC	ANCHOR	CAPTION	LIST	DEPTH	HEIGHT	TFIDF	relevance
β_i	0.372	0.151	0.094	0.048	0.063	0.001	-0.003	1.6	
“camera”	0	0	5	1	4	1	1	0.144	0.998
“launch”	0	0	0	0	3	1	1	0.028	0.233

EXPERIMENTS

This report describes all results obtained using our proposed method. **Firstly, we describe** the experiment objectives and our experimental settings. **Then, we describe** experiment results comparing our method and ESA. **We also present** results obtained from a combination of our method and ESA in this experiment. Finally, **we present** results of investigation of the layout information.

Objectives and Experimental Settings

We seek to ascertain the relation between Wikipedia layout information and word relatedness better by answering the following questions.

1. What is the best means of tuning our method for it to outperform ESA (the baseline method)? We test various regression methods and independent variable settings. Then we compare our method and ESA under various settings.
2. If our proposed method can outperform ESA for a subset of word pairs (low-frequency word pairs) in the dataset, what will be the performance of combining the proposed method and ESA together? Can the combined method outperform ESA on the full set of the word pairs?
3. How effective is layout information for predicting the relevance between a word and a concept? We examine the statistical significance and the coefficient of each independent variable in the regression formula.

We used Perl 5.12.3 for text manipulation and R 2.13.2 for regression and statistical analysis. The programs were run with a 64-bit Windows 7 OS (Microsoft Corp.) on a computer equipped with 32 GB RAM and a dual 3 GHz processor.

We downloaded the English version of Wikipedia dump of October 11, 2010. The data were over 27GB, containing over three million articles. We followed the preprocessing procedure written in [Section “Wikipedia Layout Information”](#) and obtained 793,687 concepts (Wikipedia articles) after preprocessing. Statistics of the extracted layout information after preprocessing are presented in Table 2. Each element of the layout information follows power-law distribution, so the standard deviation is greater than the mean.

Table 2. Layout information statistics.

Word attribute	Mean number per article	Standard deviation
Word frequency	590	703
Bold word	6	20
Italic word	26	65
Part of a Wikipedia link	98	156
Part of a file caption	6	18
Part of a list	96	256
At section level 1	127	183

At section level 2	54	139
At section level 3	9	58
At section level 4	1	20

The benchmark dataset is called WordSimilarity-353 (Finkelstein *et al.*, 2002), which has been used in many previous research efforts (Gabrilovich & Markovitch, 2007; Ito *et al.*, 2008; Radinsky *et al.*, 2011). It comprises 353 pairs of words, along with the relatedness judged by at least 10 people. The dataset is regarded as reliable because people generally agree on the relatedness of words (Gabrilovich & Markovitch, 2007). We want to ascertain how much closer the estimation method can simulate the human-rated relatedness, as indicated by Spearman's rank correlation coefficient.

Experimental Results of Method Tuning

We first provide the empirical evaluation of the three regression methods. Then we compare our method and ESA under the three independent variable settings.

Regression Method Comparison

We used the layout information of all word instances and the deepest header level as the independent variable setting to find out which regression method performed the best. The result is presented in Table 3. Results show that ordinary least squares linear regression (OLS) outperformed the other two methods. In addition, ordinal logistic regression (OLR) performed the worst because the concept vectors contained many zero entries. OLR returned one of the seven values from 0 to 1 and a lot of the relevance that was close to 0 was estimated to be 0. Support vector regression (SVR) performed slightly worse than OLS, but the difference of the results was not huge.

Table 3. Spearman's correlation for each regression method using all words of WordSimilarity-353.

Regression method	Spearman's correlation
Ordinary least square linear regression	0.696
Support vector regression	0.689
Ordinal logistic regression	0.454

Evaluating the Proposed method under Different Independent Variable

Settings

We compared the performance of the three settings of independent variables and investigated the differences between the proposed method and ESA. The comparison and the investigation were conducted under different settings of word frequency and using OLS as the regression method because OLS yielded the best result in the previous experiment.

We perceived the average normalized TFIDF per concept of a word as its word frequency. Experiments were done in two scenarios. The first scenario used WordSimilarity-353 word pairs, both of which were in the 25 percentile, 50 percentile, 75 percentile, and 100 percentile of the word frequency. The second scenario used WordSimilarity-353 word pairs, either of which was in the four levels of percentile. The number of remaining word pairs of the dataset in both scenarios is shown in Table 4.

Table 4. Number of remaining words if only n percentile of words was considered. Both: both words of a word pair in WordSimilarity-353 were under the n percentile. Either: either word of a word pair in WordSimilarity-353 was under the n percentile.

n percentile	Remaining pairs (Both)	Remaining pairs (Either)
25 percentile	25	147
50 percentile	103	254
75 percentile	219	323
100 percentile	353	353

Figure 4. Result obtained from the experiment when either word of the word pairs is under the n percentile. PMAD: Proposed method (All words used. Section level set as the deepest), PMAS: Proposed method (All words used. Section level set as the most shallow), PMT: Proposed method (Top word used).

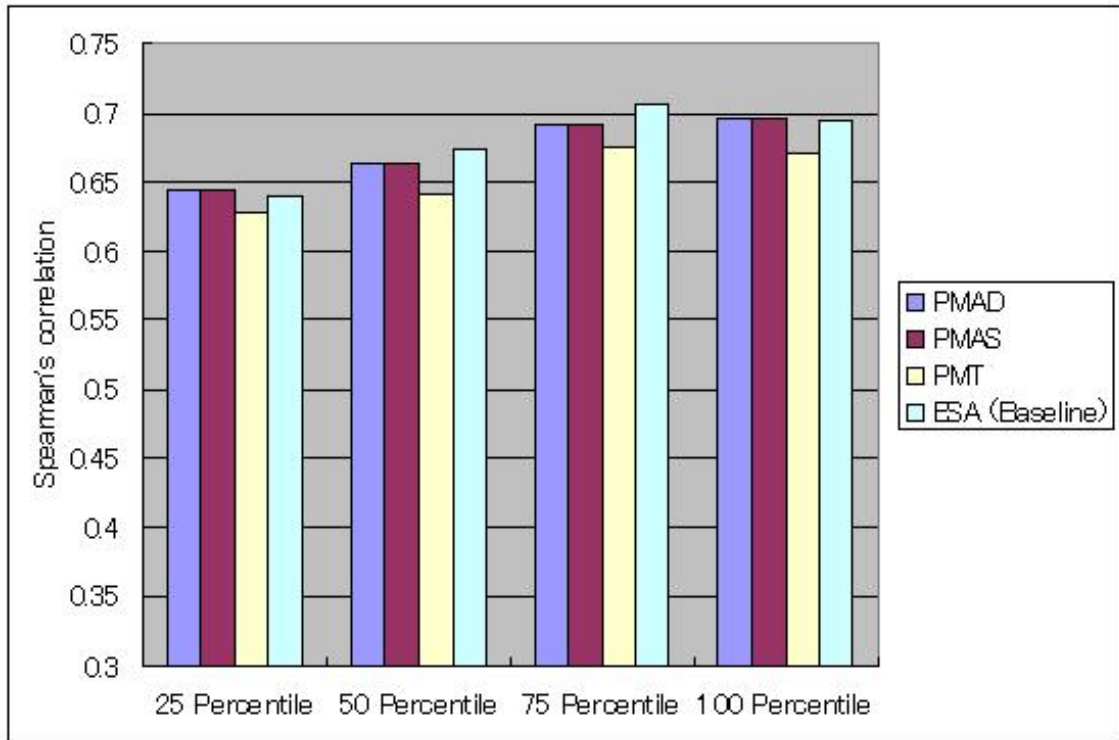


Figure 5. Result obtained from the experiment when both words of the word pairs are under the n percentile. PMAD: Proposed method (All words used. Section level set as the deepest), PMAS: Proposed method (All words used. Section level set as the shallowest), PMT: Proposed method (Top word used).

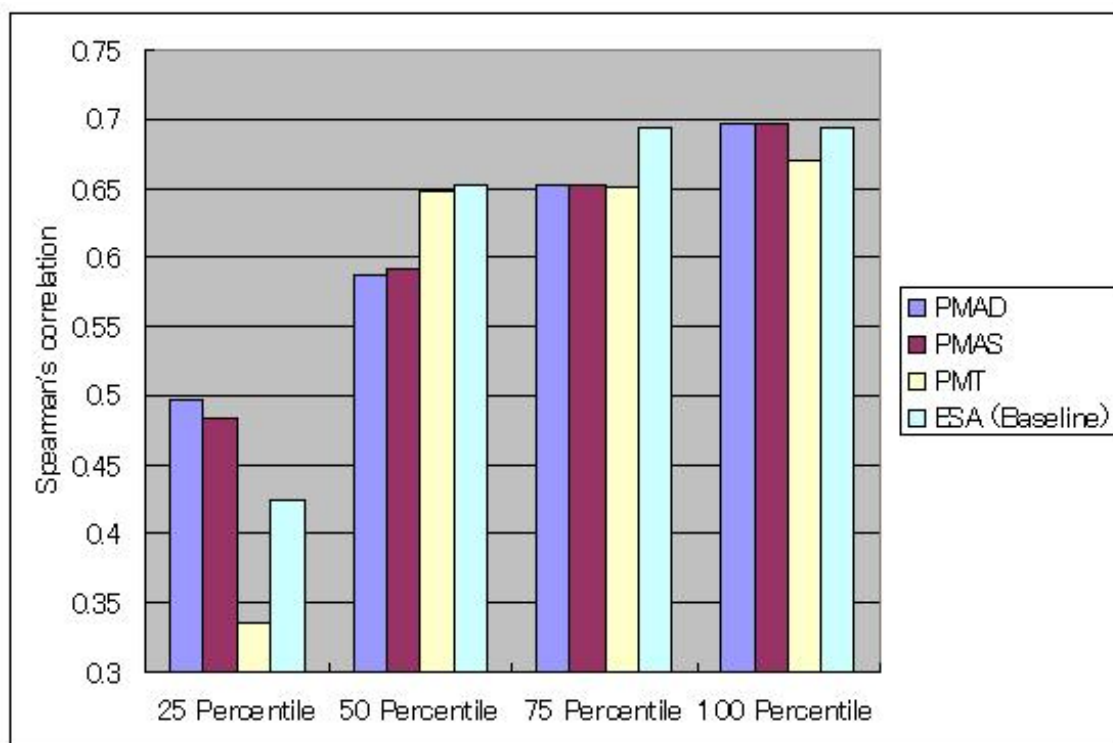


Figure 4 and Figure 5 show how well our method estimated the semantic relatedness of word pairs with various frequencies (the case in which either word of the word pairs is under the n percentile in Figure 4 and the case in which both word pairs are under the n percentile in Figure 5).

Our method is implemented in three versions that diverge from the independent variable settings. PMAD is our proposed method that uses the layout information of all words and the deepest section level. PMAS is our proposed method that uses the layout information of all words and the most shallow section level. PMT is our proposed method, which uses the layout information of only the top word.

Figure 4 and Figure 5 show that using the layout information of all word instances tends to generate a higher correlation with the human rating than using the top word only. There was little difference between the most shallow header level setting and the deepest header level setting.

For the case in which either word of the word pairs was under the n percentile (see Figure 4), the difference between ESA and the propose method with various settings was not that huge. However, in the case in which both words of the word pairs had to

be under the n percentile (see Figure 5), more interesting results arose. When n was equal to 25, the best setting of our proposed method had a 0.497 correlation, whereas ESA resulted in a 0.424 correlation. For low-frequency words, the high reliability of layout information improved the relevance estimate. When n was 50 and over, the usefulness of word frequency outweighed that of the combination of layout information and word frequency, which means that TFIDF gives sufficient information to calculate the relatedness between words.

Combination of Proposed method and Baseline Method

Ranking Combination

The ranking obtained using our proposed method functioned considerably better for the word pairs for which the normalized TFIDF values of both words were in the lower 25 percentile. Nevertheless, ESA outperformed in other word frequencies. To get the best of both worlds, we used our proposed method to calculate the word relatedness for the 25 word pairs for which both words were in the lower 25 percentile. Then we calculated the relatedness of the remaining 328 word pairs using ESA alone. Finally, we calculated Spearman's correlation for all word frequency ranges. The setting of our proposed method used all words and the deepest level as the section level. The result, shown in Table 5, demonstrates that the correlation of the combination method increased, although the increase was small.

Table 5. Spearman's correlation with the human rating obtained from the proposed method and the combination of the proposed method and ESA.

Method	Spearman's correlation
ESA	0.696
Proposed Method and ESA combined	0.708

Difference assessed using Spearman's Correlation

We assessed the statistical significance of the difference between the proposed method and ESA. The ESA paper calculated the Spearman's correlation between the human rating and estimated semantic relatedness by ESA. It applied a test of statistical significance to the difference between the Pearson correlations. Using a Spearman's correlation value as a Pearson's correlation value is inappropriate.

We find the Spearman's correlation between the rank generated by ESA and the

rank obtained using our proposed method and ascertain if the resultant rank correlation differs significantly (Maritz, 1981). When comparing words that were both in the 25 percentile of word frequency, the test revealed that our proposed method ($\rho = 0.497$) and ESA ($\rho = 0.424$) differed to a statistically significant degree ($\rho < 0.01$).

Layout Information Statistical Analysis

Table 6. Linear regression summary showing the relevance relation between types of layout information and a concept when the setting uses all the words and the section level is set as the deepest.

Word attributes	Coefficient	Standard error	Significance
BOLD	0.357	0.024	$\rho < 0.001$
ITALIC	0.151	0.020	$\rho < 0.001$
ANCHOR	0.094	0.016	$\rho < 0.001$
CAPTION	0.048	0.014	$\rho = 0.003$
LIST	0.063	0.016	$\rho < 0.001$
DEPTH	0.001	0.014	$\rho = 0.916$
HEIGHT	-0.003	0.010	$\rho = 0.174$
TFIDF	1.60	0.168	$\rho < 0.001$

We examined the relation between the layout information and the relevance of a word and a concept. Table 6 shows the regression summary when OLS was run in the setting of all word instances being used and section level being the deepest.

The normalized TFIDF, a significant attribute, had the greatest weight. Significance was verified using a *t*-test. The text styles (bold and italic) were also significant attributes. Bold words are mostly used for emphasis, so it is understandable that it was related to relevance the most among all layout information. Italic words are used for multiple purposes. Some people like to use them for emphasis as well, but names, citation sources, and so on are marked as italic. The noisier characteristic of italic words makes it a weaker attribute than bold words to deduce the relevance.

Words that are in file captions, lists, and Wikipedia links are not as useful as indicators of relevance, but these three attributes were all significant parameters.

In file captions, some text explanations are intended only for uploaded photograph data (such as “Samurai in armor, 1860s. Hand-colored photograph by Felice Beato”

attached to photograph of a samurai warrior, where “hand-colored” and “photograph” are not closely related to the concept “samurai”). Wikipedia links need not be highly relevant to the Wikipedia article because some article writers merely add a Wikipedia link simply because certain words include their own Wikipedia articles.

The depth of a word's section level and the top word's position were not significant parameters. The top word's position has small relevance with the concept (Because the attribute HEIGHT increased for each header, the negative weight showed that words nearer the top of an article had increasing relevance), although the depth of a word's section level has no relevance with the concept. This layout information is obtained from the headers. The text body exists below a header. The size of its text is greater than that of bold/italic text, anchor text, file caption, and list text. Every word in it is assigned the same section level and word position. Some words are related to the concept. Other words are not. Therefore, this layout information is not related closely to the relevance.

Comparison with Keyword Recommendations of Commercial Search Engines

Finally, the characteristics of words with high relevance are examined by comparing them with keyword recommendations in commercial search engines. Commercial search engines such as Google and Yahoo! provide service of keyword recommendation based on the current input search keywords. For example, when inputting ‘tiger’, ‘seafood’ and ‘planet’ in Google and Yahoo!, the recommendation results became as shown in Table 7.

As shown in Table 7, most of the recommended keywords to ‘tiger’ and ‘planet’ are commercial products, shops, and other proper names. Recommended keywords to ‘seafood’ are for searching recipes using seafood. These keywords are pragmatic ones in users' real search activities rather than a semantic relation. When we consider the semantic relation, hyperonyms or synonyms should be obtained. For example, ‘animal’, ‘cat’ and ‘mammal’ should be shown for the word ‘tiger’ as a semantically related word.

In fact, ‘tiger’, ‘seafood’ and ‘planet’ are included in WordSimilarity-353. For 17 words obtained randomly by manual selection from WordSimilarity-353, we calculated the semantic relatedness using our method (OLS with all words and

deepest section level are used). The top five and worst five words are shown in Table 8.

As shown in Table 8, the related words obtained using our method include hyperonyms and synonyms. For example, ‘zoo’, ‘cat’ and ‘animal’ are obtained to word ‘tiger’. They are hyperonyms and words strongly related to ‘tiger’. These relations are useful for intelligent computation, such as that used for agent systems.

Table 7. Examples of keyword recommendation in Google and Yahoo!. The top ten recommended keywords of ‘tiger’, ‘seafood’ and ‘planet’ are shown here.

(a) Google

tiger	seafood	planet
tiger woods	seafood recipes	planet fitness
tigerdirect	seafood restaurants	planet minecraft
tiger woods net worth	seafood city	planet of the apes
tiger woods girlfriend	seafood city	planet hollywood
tiger woods pga tour 14	seafood chowder	planet Hollywood las vegas
tiger balm	seafood lasagna	planet x
tiger beat	seafood salad	planet fitness locations
tiger woods wife	seafood pasta	planet fitness hours
tiger lily	seafood stew	planet money

(b) Yahoo!

tiger	seafood	planet
tiger woods	seafood restaurant	movie star planet
tigerdirect	seafood recipes	planet fitness
tiger airways	seafood gumbo	planet Hollywood las vegas
tiger mom	seafood city	planet minecraft
Detroit tiger	seafood paella recipe	planet tyche
eye of the tiger	seafood salad	planet of the apes
tiger beat	legal seafood	animal planet
marshals tiger lied	gulf seafood concerns	cheat planet

tiger blames fatigue	pappas seafood	prison planet
tiger lily	seafood buffet	lonely planet

Table 8. Examples of keyword recommendation in Google and Yahoo!. The top ten recommended keywords of `tiger', `seafood' and `planet' are shown here.

(a) `tiger'

Top five		Worst five	
zoo	0.0316	seafood	0.0012
cat	0.0133	new	0.0010
animal	0.0113	food	0.0008
sun	0.0074	lobster	0.0005
money	0.0073	dish	0.0003

(b) `seafood'

Top five		Worst five	
dish	0.1130	sun	0.0011
food	0.0492	money	0.0012
lobster	0.0920	word	0.0008
sea	0.0140	new	0.0005
coast	0.0040	planet	0.0003

(c) `planet'

Top five		Worst five	
star	0.0465	tiger	0.0017
sun	0.0389	sea	0.0013
animal	0.0184	food	0.0007
cat	0.0044	word	0.0006
radio	0.0041	seafood	0.0003

REFERENCES

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Pasca, M., & Soroa, A. (2009). A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of NAACL HLT'09* (pp.19-27).

Budanitsky, A., & Hirst, G. (2006) Evaluating WordNet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1), 13-47.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., & Ruppin, E. (2002). Placing search in context: The concept revisited. *ACM TOIS*, 20(1), 116-131.

Gabrilovich, E., & Markovitch, S. (2007). Computing semantic relatedness using Wikipedia-based explicit semantic analysis, In *Proceedings of IJCAI '07*.

Halawi, G., Dror, G., Gabrilovich, E., & Koren, Y. (2012). Large-scale learning of word relatedness with constraints. In *Proceedings of KDD'12* (pp.1406-1414).

Hecht, B., Carton, S. H., Quaderi, M., Schoning, J., Raubal, M., Gergle, D., & Downey D. (2012). Explanatory semantic relatedness and explicit spatialization for exploratory search, In *Proceedings of SIGIR'12* (pp.415-424).

Ito, M, Nakayama, K., Hara, T., & Nishio, S. (2008) Association thesaurus construction methods based on link co-occurrence analysis for Wikipedia. In *Proceedings of CIKM'08*.

Karen, S. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28, 11-21.

Maritz, J. S. (1981). *Distribution-free statistical methods*. Chapman & Hall.

Metzler, D., Dumais, S., & Meek, C. (2007). Similarity measures for short segments of text. In *Proceedings of ECIR'07* (pp.16-27).

Milne, D., & Witten, I. H. (2008). An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In *Proceedings of AAAI'08*.

Nakayama, K., Hara, T., & Nishio, S. (2007). Wikipedia mining for an association web thesaurus construction. In *Proceedings of IEEE International Conference on Web Information Systems Engineering'07* (pp.322-334).

Porter, M. F. (1980). An algorithm for suffix stripping. *Program (Automated Library and Information Systems)*, 14(3), 130-137.

Radinsky, K., Agichtein, E., Gabrilovich, E., & Markovitch, S. (2011). A word at a time: Computing word relatedness using temporal semantic analysis. In *Proceedings of WWW '11*.

Reisinger, J. & Raymond, J. M. (2010) Multi-prototype vector-space models of word meaning. In *Proceedings of NAACL HLT'10* (pp.109-117).

Sahami, M., & Heilman, T. (2006). A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of WWW '06* (pp.377-386).

Strube, M., & Ponzetto, S. (2006). WikiRelate! Computing semantic relatedness using Wikipedia. In *Proceedings of AAAI'06* (pp.1419-1424).

Yih, W. & Qazvinian, V. (2012). Measuring word relatedness using heterogeneous vector space models. In *Proceedings of NAACL HLT'12* (pp.616-620).

Authors

Patrick Chan

Patrick Chan received the B.A. degree from UC Berkeley in 2003. He received the M.E. degree from Osaka University in 2012. He joined Microsoft Japan afterwards. In Osaka University, his interests were knowledge acquisition and text mining.

Yoshinori Hijikata

Yoshinori Hijikata received the B.E. and M.E. degrees from Osaka University in 1996 and 1998, respectively. In 1998, he joined IBM Research, Tokyo Research Laboratory. After working on Web technologies there, he received Ph.D. degree from Osaka University in 2002. Currently, he is an associate professor in Osaka University. His research interests are on Web intelligence, recommender systems and text mining. He received the best paper awards from IPSJ Interaction'05, ACM IUI'06, IEICE DEWS'06, IPSJ WebDB Forum'11, WebDB Forum'12 and IPSJ Interaction'13. He also received IPSJ Yamashita SIG Research Award in 2013. He is a member of the IPSJ, IEICE, JSAI, HIS, DBSJ.

Toshiya Kuramochi

Toshiya Kuramochi received the B.E. degrees from Osaka University in 2012. His interests are graph mining and text mining.

Shogo Nishida

Shogo Nishida received the B.S. M.S. and Ph.D. degrees in Electrical Engineering from the University of Tokyo, in 1974, 1976 and 1984, respectively. From 1976 to 1995, he worked for Mitsubishi Electric Corporation, Central Research Laboratory. From 1984 to 1985, he visited MIT Media Laboratory, Boston, Massachusetts, as a visiting researcher. In 1995, He moved to Osaka University as a Professor of Graduate School of Engineering Science. He was the dean of Graduate School of Engineering Science from 2004 to 2007, and the Trustee and Vice President of Osaka University from 2007 to 2011. His research interests include CSCW, Media Technology, Human Interfaces and Human Communication. He is a Fellow of IEEE (1998), a Fellow of IEICE in Japan (2005), a Fellow of IEE in Japan (2008) and Honorary Member of Human Interface Society of Japan (2007).