Knowl Inf Syst (2009) 19:265–281
DOI 10.1007/s10115-008-0152-4

REGULAR PAPER

Using Wikipedia knowledge to improve text classification

Pu Wang · Jian Hu · Hua-Jun Zeng · Zheng Chen

Received: 29 October 2007 / Revised: 24 December 2007 / Accepted: 29 January 2008 /
Published online: 17 September 2008

Abstract Text classification has been widely used to assist users with the discovery of useful information from the Internet. However, traditional classification methods are based on the "Bag of Words" (BOW) representation, which only accounts for term frequency in the documents, and ignores important semantic relationships between key terms. To overcome this problem, previous work attempted to enrich text representation by means of manual intervention or automatic document expansion. The achieved improvement is unfortunately very limited, due to the poor coverage capability of the dictionary, and to the ineffectiveness of term expansion. In this paper, we automatically construct a thesaurus of concepts from Wikipedia. We then introduce a unified framework to expand the BOW representation with semantic relations (synonymy, hyponymy, and associative relations), and demonstrate its efficacy in enhancing previous approaches for text classification. Experimental results on several data sets show that the proposed approach, integrated with the thesaurus built from Wikipedia, can achieve significant improvements with respect to the baseline algorithm.

Keywords Text classification · Wikipedia · Thesaurus

1 Introduction

The exponential growth of online documents in the World Wide Web has raised an urgent demand for efficient and high-quality text classification algorithms to achieve fast navigation and browsing of web pages based on a reliable document organization. Traditional document classification algorithms are based on the "Bag of Words" (BOW) approach, which represents a document as a vector of weighted occurrence frequencies of individual terms. However, the BOW representation is limited, as it only accounts for term frequency in the documents,

P. Wang (✉)
Department of Computer Science, George Mason University, Fairfax, VA 22030, USA
e-mail: pwang7@gmu.edu

J. Hu · H.-J. Zeng · Z. Chen
Machine Learning Group, Microsoft Research Asia, Beijing, China

and ignores important semantic relationships between key terms. To break through this limitation, work has been done to exploit ontologies for content-based classification of large document corpora. Hotho et al. [1] utilized a term ontology structured from WordNet [18] to improve the BOW text representation. The authors adopted various strategies to enrich text document representation with synonyms and hyponyms from WordNet. Although experimental results have shown some improvement in clustering performance, WordNet has limited coverage, since it is a manually constructed dictionary, and therefore laborious to maintain. To deal with the ontology coverage and maintenance problem, other research explored the usage of a knowledge-base derived from the Internet, such as Open Directory Project [20], and Wikipedia. The authors in [2,3] applied feature generation techniques to text processing using ODP and Wikipedia. Their application on text classification has confirmed that background-knowledge-based features generated from ODP and Wikipedia can facilitate text categorization. Furthermore, their results show that Wikipedia is less noisy than ODP when used as knowledge-base. However, ODP and Wikipedia are not structured thesauri as WordNet, and therefore they cannot resolve synonymy and polysemy (two fundamental problems in text classification) directly. The authors in [2,3] claim that their multi-resolution approach performs implicit word sense disambiguation. However, their approach ignores the abundant structural relations within Wikipedia, such as hyperlinks and hierarchical categories, and the retrieval-based feature generation process inevitably brings a lot of noise. In this paper, we tackle these issues.

In our work, we first build an informative and easy-to-use thesaurus from Wikipedia, which explicitly derives concept relationships based on the profuse structural knowledge of Wikipedia, including synonymy, polysemy, hyponymy, and associative relations. The generated thesaurus serves as a controlled vocabulary that bridges the variety of idiolects and terminologies present in the corpus of documents. It facilitates the integration of the rich knowledge of Wikipedia into text documents, by resolving synonyms and introducing more general and associative concepts, which may assist the identification of related topics among text documents. Furthermore, the coverage of the resulting thesaurus is much broader than manually constructed thesauri like WordNet, and it provides richer contexts for sense disambiguation of polysemous concepts. We then propose a unified framework to explicitly integrate the hierarchical relations, synonymy, and associative semantic relations within our Wikipedia thesaurus. This allows to improve the performance of text classification by enriching the traditional text document similarity measure with semantic information. To evaluate the performance of the proposed method, we have performed an empirical evaluation on several real data sets. The experimental results show that our proposed framework, which integrates hierarchical relations, synonym and associative relations with traditional text similarity measures based on the BOW model, does improve text classification performance significantly with respect to the baseline algorithm.

The rest of the paper is organized as follows. Section 2 describes related work. In Sect. 3, our method for building a thesaurus from Wikipedia is discussed. We outline the algorithm of utilizing Wikipedia relations to improve text classification in Sect. 4. The experimental setting and results are discussed in Sect. 5. We conclude our paper in Sect. 6.

2 Related work

To date, the work on integrating semantic background knowledge into text representation is quite limited, and the classification or clustering results are not satisfactory. The authors in [14,17] successfully integrated the WordNet resource for a document categorization task.

They evaluated their methods on the Reuters corpus [9], and showed improved classification results with respect to the Rocchio and Widrow-Hoff algorithms. In contrast to our approach, Rodriguez et al. [14] and Urena-Lopez et al. [17] utilized WordNet in a supervised scenario without employing WordNet relations such as hypernyms and associative relations. Furthermore, they built the term vectors manually. The authors in [15] utilized WordNet synsets as features for document representation, and subsequent clustering. Word sense disambiguation was not performed, and WordNet synsets actually decreased clustering performance. Hotho et al. [1] integrated WordNet knowledge into text clustering, and investigated word sense disambiguation strategies and feature weighting schema by considering the hyponym relations derived from WordNet. Experimental results on the Reuters Corpus have shown improvements in comparison with the best baseline. However, due to the limited coverage of WordNet, the word sense disambiguation effect is quite limited. In addition, WordNet does not provide associative terms as Wikipedia.

Gabrilovich et al. [2,3] proposed and evaluated a method to render text classification systems with encyclopedic knowledge, namely Wikipedia and ODP. They first built an auxiliary text classifier that could match documents with the most relevant articles in Wikipedia. Then, they augmented the conventional BOW representation with new features, corresponding to the concepts (mainly the titles) represented by the relevant Wikipedia articles. Empirical results showed that this representation improved text categorization performance across a diverse collection of data sets. However, the authors did not make full use of the rich relations of Wikipedia, such as hyponyms, synonyms and associated terms. In addition, as pointed out by the authors, the feature generation process can introduce a lot of noise, although the feature selection step can mitigate this problem.

3 Wikipedia

Launched in 2001, Wikipedia (http://www.wikipedia.org) is a multilingual, web-based, free content encyclopedia written collaboratively by more than 75,000 regular editing contributors. Its articles can be edited by anyone with access to its website. Wikipedia is a very dynamic and fast growing resource: articles about newsworthy events are often added within few days of their occurrence [19]. Each article in Wikipedia describes a single topic; its title is a succinct, well-formed phrase that resembles a term in a conventional thesaurus [4]. Each article must belong to at least one category of Wikipedia. Hyperlinks between articles keep many of the same semantic relations as defined in the international standard for thesauri [1], such as equivalence relations (synonymy), hierarchical relations (hyponymy), and associative relations. However, as an open resource, it inevitably includes a lot of noise. To generate a clean and easy-to-use thesaurus from Wikipedia, we first preprocess Wikipedia data to collect concepts, and then explicitly derive relationships using the profuse structural knowledge of Wikipedia.

3.1 Wikipedia as a Thesaurus

The title of each Wikipedia articles describes a topic, and we denote it as a concept. However, some of the titles, such as “1980s”, “List of newspapers”, and so on, are meaningless (they are only used for Wikipedia management and administration). Hence, we first filter the Wikipedia titles according to the rules described below (titles satisfying at least one of the rules will be filtered):

- The article belongs to categories related to chronology, i.e. “Years”,“Decades” and “Centuries”.
- The first letter is not capitalized.
- The title is a single stopword.
- For a multiword title, not all words other than prepositions, determiners, conjunctions, or negations are capitalized.
- The title occurs less than three times in its article.

3.1.1 Synonymy

Wikipedia ensures the existence of only one article for each concept by using “Redirect” hyperlinks to group equivalent concepts to the preferred one. A redirect page, which only contains a redirect link, exists for each alternative name of a concept that can be used to refer to the preferred one in Wikipedia. Thus, synonymy in Wikipedia is handled through redirect pages. A “Redirect” link also copes with capitalization, spelling variations, abbreviations, colloquialisms, and scientific terms. For example, an entry with a considerably high number of redirect pages is “United States” [5]. Its redirect pages correspond to acronyms (U.S.A., U.S., USA, US), Spanish translations (Los Estados Unidos, Estados Unidos), misspellings (Untied States), or synonyms (Yankee land).

In addition, Wikipedia articles often mention concepts, which already have corresponding articles in Wikipedia. For such a concept, a Wikipedia article usually links at least its first mention in the corresponding article by using a hyperlink. The anchor text on each hyperlink may be different from the title of the linked article. Thus, anchor texts can be used as synonyms of the linked article concepts.

3.1.2 Polysemy

Disambiguation pages are created for ambiguous terms, i.e. terms that denote two or more entities. For example, the term “Puma” may refer to either a kind of animal, a kind of racing car, or a famous sportswear brand. Wikipedia provides disambiguation pages which contain various possible meanings, from which users could select articles corresponding to the intended concept. For example, the disambiguation page for the term “Puma” lists 22 associated concepts, from persons, to vehicles and sport clubs.

3.1.3 Hyponymy (hierarchical relations)

In Wikipedia, both articles and categories themselves can belong to more than one category, e.g. the article about “Puma” belongs to two categories: “Cat stubs” and “Felines”. These categories can be further categorized by associating them with one or more parent categories. Thus, the category structure of Wikipedia does not form a simple tree-structured taxonomy, but a directed acyclic graph, in which multiple categorization schemes co-exist simultaneously [4], making Wikipedia categories not a taxonomy with a fully-fledged subsumption hierarchy, but only a thematically organized thesaurus. To extract real “is a” relations from Wikipedia categories, we apply the method proposed in [16] to derive generic “is a” relations from category links. In this way, we can derive hyponyms for each Wikipedia concept.

3.1.4 Associative relations

Each Wikipedia article contains many hyperlinks, which express different degrees of relatedness. Milne et al. [4] observed that links often occur between articles that are only tenuously related. For example, links exist between the articles “Cougar” and “South America”, and between the articles “Cougar” and “Puma”. The former two articles are not as closely related as the latter pair. Thus, it is important to quantify the strength of hyperlinks. To this end, we introduce three kinds of measures to rank links between Wikipedia articles.

Content-based measure. This measure is based on the BOW representation of Wikipedia articles. The relatedness of two articles is modeled as the extent to which they share terms. Each article is represented as a $tf\text{-}idf$ vector: the value associated to a given term reflects its frequency of occurrence within the corresponding article (tf) and within the entire corpus (idf). The associative relation between two articles is then measured by computing the cosine similarity between the corresponding vectors. Intuitively, if two text documents address a similar topic, they share many terms and therefore their cosine similarity will be high. On the other hand, if two documents address different topics, they share fewer terms, and their cosine similarity will be smaller. Clearly, this measure (denoted as $S_{tf\text{-}idf}$) has the same drawbacks of the BOW approach, since it only considers terms that appear in the text documents. Other measures need to be synthesized.

Out-link category-based measure. The out-link category-based measure compares the out-link categories of the associative articles. The out-link categories of a given article are the categories to which out-link articles from the original one belong. It has been observed that the larger the number of shared out-link categories between two articles, the stronger the associative relation between them. To capture this notion of similarity, articles are represented as vectors of out-link categories, where each component corresponds to a category, and the value of the ith component is the number of out-link articles which belong to the ith category. Table 1 shows a fraction of the out-link categories shared by the associative concepts

Table 1 Out-link categories of the articles “Data Mining”, “Machine Learning”, and “Computer Network”

Category name	Data Mining	Machine Learning	Computer Network
Information Technology	2	3	1
Artificial Intelligence	2	6	0
Computer Science	2	6	4
Applied Mathematics	2	2	0
Classification Algorithms	5	7	0
Artificial Intelligence researchers	2	2	0
Neural Networks	1	3	0
Statistics	9	10	0
Information Technology Management	3	2	5
Machine Learning	4	14	0
Business Intelligence	4	2	1
Data Management	6	0	21
Computer Networks	1	2	1
Networks	1	3	3
Intelligent Document	3	0	1

The values correspond to the numbers of out-link articles which belong to the corresponding category

"Data Mining", "Machine Learning", and "Computer Network". Obviously, the category distributions of "Data Mining" and "Machine Learning" are more similar to each other than those corresponding to "Data Mining" and "Computer Network".

Let f_i be the number of out-link articles which belong to the c_i category. Then, the out-link category-based representation of an article $\vec{c}$ is given by $\vec{c} = (f_1, \ldots, f_n)$, where each component corresponds to a category name. Given two linked articles $\vec{c}_1$ and $\vec{c}_2$, their similarity (denoted as S_{OLC}) is measured by computing the cosine similarity between the corresponding vectors:

$$S_{\mathrm{olc}} = \frac{\vec{c}_1 \cdot \vec{c}_2}{|\vec{c}_1| \cdot |\vec{c}_2|} \tag{1}$$

Distance-based measure. This measure is a distance measure (rather then a similarity measure). The simplest distance-based measure can be computed according to the straightforward edge count method, which measures the semantic distance as the number of nodes in the taxonomy along the shortest path between two conceptual nodes [6]. Accordingly, given the acyclic graph formed by the Wikipedia hierarchical category structure, we define the distance (denoted as D_{cat}) between two articles as the length of the shortest path connecting the two categories they belong to. The distance measure is normalized by taking into account the depth of the taxonomy.

A linear combination of the three measures allows to quantify the overall strength of an associative relation between concepts:

$$S_{\mathrm{overall}} = \lambda_1 S_{tf\text{-}idf} + \lambda_2 S_{\mathrm{OLC}} + (1 - \lambda_1 - \lambda_2)(1 - Dis_{\mathrm{cat}}), \tag{2}$$

where $\lambda_1, \lambda_2 \in (0, 1)$ are parameters to weigh the individual measures. In Sect. 5 we will explain how to adjust these parameters. Equation (2) allows to rank all the associative articles linked to any given concept.

4 Compiling Wikipedia knowledge into document representation

The "Bag of Words" (BOW) approach only leverages the terms explicitly mentioned in text documents, thus failing to reflect relationships between important terms that do not co-occur in the given corpus. Integrating background knowledge in text documents may overcome the shortage of the BOW approach. Moreover, Wikipedia is well-known for its extensive encyclopedic coverage, not available through other electronic resources. Therefore, we build a general thesaurus from Wikipedia to exploit the background knowledge for text corpora. In the following, we first describe the text document representation, and then introduce a framework to integrate hierarchical, synonym and associative relations with the traditional text similarity measure to improve text classification.

4.1 Text document representation

Text documents are traditionally represented as bags of weighted terms. Let D be a set of documents, and T the set of all different terms occurring in D. Typically, T is constructed after removing stop words, and after stemming terms to their roots [7]. Let $d \in D$ represent a document, and $t \in T$ a term. We define $tf(d, t) = \frac{n(d,t)}{|d|}$, where $n(d, t)$ is the absolute frequency of term t in document d, and $|d|$ the length of d. The $tf\text{-}idf$ value of a term t in document d is defined as

$$tf\text{-}idf(d,t) = tf(d,t) \cdot \log\left(\frac{|D|}{df(t)}\right), \tag{3}$$

where $df(t)$ is the document frequency of term t. It counts the number of documents in D where term t appears. $|D|$ is the total number of documents in the corpus. The idf term has the effect of down weighting terms which appear in many documents of the corpus. The resulting term vector representation of a document d is given by $\vec{t}_d = (tf\text{-}idf(d,t_1),\ldots,tf\text{-}idf(d,t_m))$ (assuming $|T| = m$). To compute the semantic relatedness of a pair of text fragments d_1 and d_2 based on their content, one computes the cosine similarity of their corresponding term vectors:

$$S_{\text{content}} = \frac{\vec{t}_{d_1} \cdot \vec{t}_{d_2}}{|\vec{t}_{d_1}| \cdot |\vec{t}_{d_2}|}. \tag{4}$$

4.2 Wikipedia-based text document similarity measure

Our objective is to extend the measure defined in Eq. (4) with a Wikipedia-based similarity measure. The overall enrichment procedure is illustrated in Fig. 1. Using a concept index built from Wikipedia, we search candidate concepts mentioned in each text document, and then add synonyms, hyponyms and associative concepts of these candidate concepts into documents. As a result, guided by the original content, new concepts are added as features of documents, with the purpose of enriching their representation. As such, related documents, which originally do not share common terms, are enriched with the same concepts, and therefore shifted closer to each other in the new representation. For example, consider two documents concerning the concepts "Puma" and "Cougar", respectively. Assuming that the

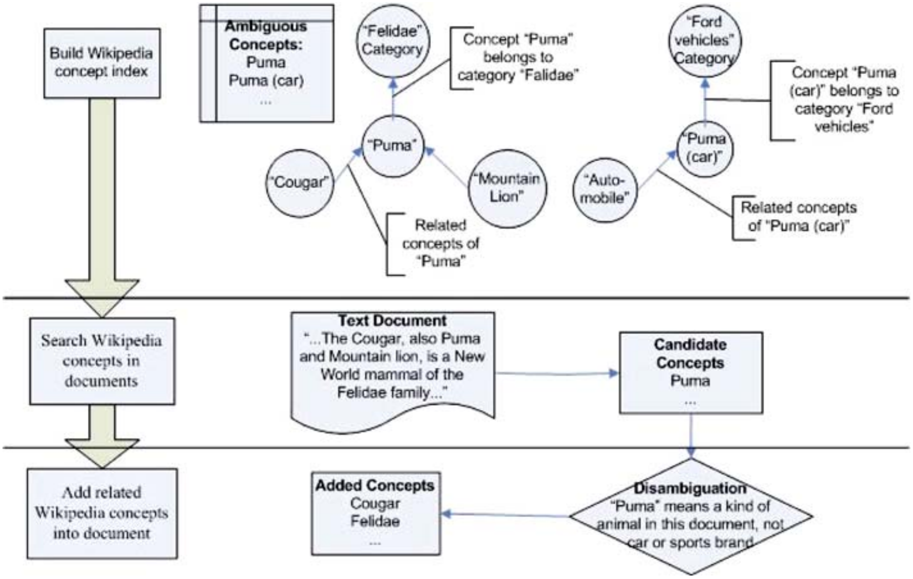
Build Wikipedia concept index
Ambiguous Concepts: Puma Puma (car) ...
"Felidae" Category
Concept "Puma" belongs to category "Falidae"
"Ford vehicles" Category
Concept "Puma (car)" belongs to category "Ford vehicles"
"Puma"
"Cougar"
"Mountain Lion"
Related concepts of "Puma"
"Puma (car)"
"Auto-mobile"
Related concepts of "Puma (car)"
Search Wikipedia concepts in documents
Text Document "...The Cougar, also Puma and Mountain lion, is a New World mammal of the Felidae family..."
Candidate Concepts Puma ...
Add related Wikipedia concepts into document
Added Concepts Cougar Felidae ...
Disambiguation "Puma" means a kind of animal in this document, not car or sports brand

Fig. 1 Document enrichment procedure

two documents do not share many terms, a classifier based on (4) may deem the two documents as different in their content. Nevertheless, according to our thesaurus, the concepts "Puma" and "Cougar" belong to the same category "Felines". Thus, by enriching both documents with the more general concept "Felines", their semantic relationship can be revealed. Below we describe in detail the steps involved in this process.

4.2.1 Index of Wikipedia concepts

We define the titles (filtered according to the rules presented in Sect. 3.1) of Wikipedia articles as concepts. We then build a concept index: given a word, the index finds all Wikipedia concepts containing the specified word. We also gather all polysemous concepts, which are concepts holding multiple meanings, into an *ambiguous concept set*.

4.2.2 Search of Wikipedia concepts in documents

Using the Wikipedia concept index, we search for the Wikipedia concepts which are mentioned in documents. Such concepts are called *candidate concepts*. We search candidate concepts in documents according to the following steps:

- Split the document into vectors of term sequences using punctuations such as semicolons, question and exclamation marks, and periods.
- Find candidate concepts in each term sequence via a window filtering condition described below.
- Filter candidate concepts to remove the concepts subsumed by other candidate concepts.

For each term sequence, the window filtering condition searches candidate concepts using the Front Maximum Matching algorithm. It requires that every word of a concept must appear in the sequence within a window of certain length. For concepts of less than three words, the window's length is the number of words in the concept; for concepts of more than (or equal to) three words, the window's length is the number of words in the concept plus one:

$$LEN_W = \begin{cases} WordCount(Concept) & \text{if } WordCount(Concept) \leq 2 \\ WordCount(Concept) + 1 & \text{if } WordCount(Concept) > 2 \end{cases}$$

Here is an example of an application of the window filtering condition. Let us consider the concept "Ford Puma", and the term sequences listed in Table 2. The length of the filtering window for the concept "Ford Puma" is two. Although all four term sequences in Table 2 contain both terms "Ford" and "Puma", only the first sequence will introduce the concept "Ford Puma" as a candidate concept. This is because only the first sequence satisfies the window filtering condition, i.e., the two words "Ford" and "Puma" appear within a window of length two, and their order is consistent with their arrangement in the concept "Ford Puma".

This windowing procedure guarantees that all the words of a concept appear in a sequence within a certain distance from each other, to ensure that the concept with the intended semantic

Table 2 Application of the window filtering condition

Sent. 1	The Ford Puma was a small coupe produced by the Ford Motor Company.
Sent. 2	The Ford Racing Puma was created in a limited run of just 500 by Tickford.
Sent. 3	Stylistically, the Puma followed Ford's New Edge design strategy.
Sent. 4	The Puma was only sold in Europe and was supposedly replaced by the Ford StreetKa, which is based on the Fiesta just as the Puma was.

is indeed contained in the document. As another example, consider the sentence “Harrison Ford, a famous actor, was in a suit of Puma sportswear.” Although this sentence contains the words “Ford” and “Puma”, it is not about the car “Ford Puma”.

4.2.3 Adding Wikipedia concepts into documents

After searching for candidate concepts in documents, we add the related concepts of each candidate concept into documents. The related concepts of a candidate concept include its synonyms, hyponyms and associative concepts. If a candidate concept belongs to the ambiguous concept set, i.e., the candidate concept is polysemous, it is necessary to perform word sense disambiguation to identify the intended meaning in the document. We adopt two strategies to do word sense disambiguation: the first one is based on text similarity; the second one uses context.

Disambiguation with text similarity. This method performs word sense disambiguation by measuring document similarity based on the amount of overlapping text. For instance, the Reuters document #15264 talks about copper mining, but the concept “Copper” in Wikipedia refers to several different meanings, as listed in Table 3. The correct meaning of a polysemous concept can be determined by comparing the cosine similarity between the tf-idf term vector of the text document and the term vector of Wikipedia articles describing the different meanings of the polysemous concept. As discussed in Sect. 3.1.4, the higher the cosine similarity between two tf-idf term vectors, the closer related the two corresponding text documents are. Thus, the meaning described by the article which gives the highest tf-idf cosine similarity is considered to be the most appropriate one. As shown in Table 3, the Wikipedia article describing “Copper” has the highest similarity with the Reuters document #15264. A manual inspection of the Reuters document confirms that the topic discussed therein indeed corresponds to the Wikipedia concept “Copper”.

Disambiguation with context. Disambiguation with context is based on the notion of conceptual distance [8]. To understand this approach, we start with an example. Let us consider the sentence in Table 4. The sentence contains the polysemous concept “Puma”, which can refer

Table 3 tf-idf similarities between the Reuters document #15264 and Wikipedia articles corresponding to different meanings of the term “Copper”

Meanings of “Copper”	tf-idf similarity with Reuters #15264
Copper	0.339733115
Copper (color)	0.203722805
Copper (comic)	0.197735235
Copper(I) iodide	0.133150464
Copper(I) oxide	0.169211264
Copper(I)-thiophene-2- carboxylate	0.064311508
Copper(II) acetate	0.162892376
Copper(II) carbonate	0.20138639
Copper(II) fluoride	0.159417748
Copper(II) hydroxide	0.178744084
Copper(II) nitrate	0.158279119
Copper(II) oxide	0.208643492
Copper(II) sulfate	0.172548312

Table 4 Disambiguation with context

The cougar, also known as the **puma** or mountain lion, is a New World mammal of the Felidae family.

to a kind of car or to an animal. The sentence also contains other Wikipedia concepts: "Cougar", "Mountain Lion" and "Felidae", of which "Cougar" and "Felidae" are also polysemous concepts. However, Wikipedia has a redirect link from the concept "Mountain Lion" to the concept "Cougar". This fact suggests that "Cougar" in this sentence refers to a kind of animal. Furthermore, the Wikipedia concepts "Cougar", "Puma" and "Felidae" belong to the same category "Felines", which reinforces the fact that "Cougar", "Puma" and "Felidae" all refer to a kind of animal in the given context. As a result, the sense of "Puma" is disambiguated by means of other concepts which appear in the same context.

This method performs word sense disambiguation by leveraging the relations between the concepts appearing in the same context. The relations between concepts are represented by the structural information of Wikipedia, which is captured in our thesaurus. Thus, using the thesaurus, we define the conceptual distance function between any two concepts C_1 and C_2 as follows:

$$Dis_{\text{Concept}}(C_1, C_2) = \begin{cases} 1 & \text{if } C_1 \text{ links to } C_2 \\ Dis_{\text{Category}}(C_1, C_2) & \text{otherwise} \end{cases} \tag{5}$$

where "C_1 links to C_2" means that C_1 is either a synonym or an associative concept of C_2. In other words, if there is a link between C_1 and C_2, their conceptual distance is 1, otherwise it is equal to their category distance. In Table 4, the conceptual distance between "Cougar" and "Mountain Lion" is 1.

If a sentence contains a polysemous concept, we calculate the conceptual distance of each meaning of the concept with other non-polysemus concepts mentioned in the sentence. We then compute the average conceptual distance of each meaning. The meaning with the smallest average conceptual distance is considered to be the most appropriate one.

However, many sentences only contain at most one Wikipedia concept. In such cases, disambiguation with context is not applicable, and disambiguation with text similarity is performed. When disambiguation with context is available, both methods are applied, and the average of the two disambiguation results is considered as combined result.

Here is an example of applying this disambiguation strategy. The document #15264 from Reuters-21578 discusses a joint mining venture by a consortium of companies, and belongs to the category "copper". This document mentions several concepts as "copper", "mining" and "Teck Cominco" (which is a Canadian mining company). Table 5 shows the hyponyms, associative concepts and synonyms introduced in the document by these concepts.

When adding synonyms, associative concepts and hyponyms of a candidate concept into a text document, we need to address the question of how many concepts should be added. Direct hyponyms (which are the category names a concept directly belongs to) are typically strongly related to a concept, while ancestor categories are far too general (the larger the distance between ancestor categories and a Wikipedia concept, the weaker their relations). For example, the concept "Puma" belongs to the category "Felines", and the category "Felines" belongs to the category "Carnivores", which belongs to the category "Mammals". The relation between "Puma" and "Felines" is much stronger than those between "Puma" and "Carnivores" or "Mammals". Section 5.3 shows the results corresponding to the addition of different levels of synonyms and hyponyms.

Table 5 Hyponyms, associative concepts, and synonyms added to Reuters document #15264

Term	Hyponyms	Associative concepts	Synonyms
Mining	"mining companies"	"open-pit mining"	"mine planning"
	"mining companies"	"hard rock mining"	"miner"
		"sub-surface mining"	"metal mining"
		"surface mining"	"mine (industry)"
			"mineral engineering"
			"mineral extraction"
			"miners"
			"mining industry"
			"ore body"
Copper	"chemical elements"	"copper(II) carbonate"	"copper (element)"
	"transition metals"	"copper(II) oxide"	"copper band"
		"copper extraction"	"copper sheet"
		"native copper"	"copper sheet metal"
			"cuprous"
			"cuprum"
			"copper mine"
			"cupper"
			"cupric"
Teck Cominco	"mining companies of Canada"	"mining"	"Teck Cominco Ltd."
	"s&p/tsx composite index"	"Canadian Pacific Railway"	
		"Kirkland Lake"	
		"Ontario"	

5 Empirical evaluation

The evaluation was done with the Wikipedia snapshot dumped on November 30, 2006. After decompression, the resulting XML file was 8.6 GB in size.

5.1 Wikipedia data

As an open source project, the content of Wikipedia can be downloaded from http://download.wikipedia.org. It is available under the form of database dumps that are released periodically, from several days to several weeks apart. The full content and revision history at this point in time occupy 90 GB of compressed data. We only use the link structure and articles' content.

We identified over four million distinct entities (articles and redirections) that constitute the vocabulary of the thesaurus. They are organized into 127,325 categories with an average of two subcategories and 26 articles each. The articles themselves are highly inter-linked; each article links to an average of 25 other articles.

After filtering Wikipedia concepts as described in Sect. 4.2, we obtained 627,255 concepts. Table 6 breaks down the numbers of the different types of data.

5.2 Data sets

We used three real data sets in our experiments: Reuters-21578 [9], OHSUMED [10], and 20 Newsgroups (20NG) [11].

For the Reuters-21578, following common practice, we used the ModApte split (9603 training and 3299 testing documents), and two category sets: the 10 largest categories, and 90 categories with at least one training example and one testing example. OHSUMED is a subset of MEDLINE, which contains 348,566 medical documents. Each document contains a title, and about two-thirds (233,445) also contain an abstract. Each document is labeled with an average of 13 MeSH3 categories (out of 14,000 total). Following [12], we used a subset of documents from 1991 that have abstracts, taking the first 10,000 documents for training and the next 10,000 for testing. To limit the number of categories for the experiments, we randomly generated 5 sets of 10 categories each. 20 Newsgroups (20NG) is a well-balanced data set containing 20 categories of 1000 documents each.

5.3 Experimental results

A linear Support Vector Machine (SVM) [12] is used to learn a model to classify documents. We measured text categorization performance using the precision-recall break-even point (BEP). For the Reuters and OHSUMED data sets, we report both the micro-averaged and the macro-averaged BEP, since their categories differ in size substantially. The micro-averaged BEP operates at the document level, and is primarily affected by the categorization performance on larger categories, whereas the macro-averaged BEP averages results over categories, and thus small categories have large impact on the overall performance. Following established practice, we used a fixed data split for the Reuters and OHSUMED data sets, and consequently used macro-sign test (S test) [13] to assess the statistical significance of differences in classifier performance. For the 20NG data set, we performed fourfold cross-validation, and used paired t test to assess the significance.

5.3.1 Parameter tuning

As described in Sect. 3.1, we perform a linear combination (see Eq. (2)) of three different measures to compute the degree of relatedness between Wikipedia concepts. Here we discuss how to tune λ_1 and λ_2 in Eq. (2).

First, we randomly select ten Wikipedia concepts, and then extract all the out-link concepts of the articles corresponding to the ten concepts. To obtain high quality ground truth for tuning,

Table 6 Content of Wikipedia

Terms in Wikipedia	2250000
Concepts	1110111
Redirected concepts	1020000
Categories	120000
Relations in Wikipedia	33060000
Redirect to concept	1020000
Category to subcategory	240000
Category to concepts	3050000
Concept to concept	28750000

we asked three assessors to manually label all the linked concepts of the ten articles using three relevance levels (“relevant” = 2, “neutral” = 1, and “not relevant” = 0). The labeling process was carried out independently among assessors. No one among the three assessors (graduate students with good command of English) could access the labeling results of others. After labeling, each out-link concept in the ten articles is labeled with three relevance tags, and we use the corresponding average value as the final relatedness value. For example, if one user labels two linked concepts as neutral and the other two users label them as relevant, then the final relatedness of the pair of linked concepts is $(1 + 2 + 2)/3 = 1.67$. We then calculate the content-based, out-link category-based, and distance-based measures between the ten selected concepts and related ones. Thus, we tune λ_1 and λ_2 to values between 0.1 and 1.0 (at steps of 0.1) so that the resulting relatedness measure (2) matches the users’ evaluations as close as possible. The resulting values are $\lambda_1 = 0.4$ and $\lambda_2 = 0.5$.

5.3.2 The effect of document enrichment

As described in Sect. 4.2, when enriching documents, we first identify the candidate concepts mentioned in a text document, and then enrich the documents with new concepts introduced by the candidate concepts. We have considered different strategies: adding synonyms, adding hyponyms, and adding associative concepts. Here we demonstrate the effect of adding different kinds of concepts on classification.

Table 7 shows the performance obtained when augmenting the documents with hyponyms. We first added the direct hyponyms (which are the category names a candidate concept directly belongs to) for each candidate concepts; then we added the hyponyms of both first and second levels (which are the parents’ category names of the direct category a candidate concept belongs to), and so on up to the fifth level. In Table 7, the “Baseline” algorithm performs no text augmentation; “H1” corresponds to the addition of direct hyponyms; “H2” corresponds to the addition of hyponyms of first and second levels, and so on. The results show that adding direct hyponyms only, or hyponyms of first and second levels give the best results. Adding hyponyms of higher levels deteriorates the performance.

Table 8 shows the results of enriching documents with associative concepts. For each candidate concepts, we append the documents corresponding to the top 5, 10, 15, 20 and 25 most similar associative concepts. Again, “Baseline” means that no concepts are added; “A5” corresponds to the addition of the top five associative concepts, and so on. Similarly to the first set of experiments, we found that adding the top five or ten associative concepts gives the best classification results, whereas the addition of more associative concepts gives

Table 7 The effect of adding hyponyms

Dataset	Reuters		20NG		OHSUMED	
	Micro	Macro	Micro	Macro	Micro	Macro
Baseline	0.877	0.605	0.868	0.865	0.602	0.548
H1	**0.891**	**0.623**	**0.904**	**0.892**	**0.658**	**0.585**
H2	0.883	0.619	0.898	0.886	0.642	0.574
H3	0.878	0.607	0.881	0.879	0.631	0.568
H4	0.871	0.601	0.875	0.868	0.617	0.553
H5	0.868	0.593	0.869	0.857	0.604	0.540

The classification performance is measured using precision-recall break-even point

Table 8 The effect of adding associative concepts

Dataset	Reuters		20NG		OHSUMED	
	Micro	Macro	Micro	Macro	Micro	Macro
Baseline	0.877	0.605	0.868	0.865	0.602	0.548
A5	**0.907**	**0.629**	**0.915**	**0.896**	**0.667**	**0.590**
A10	0.899	0.621	0.908	0.887	0.656	0.578
A15	0.884	0.617	0.896	0.875	0.639	0.561
A20	0.879	0.608	0.889	0.868	0.628	0.553
A25	0.871	0.599	0.878	0.859	0.611	0.542

The classification performance is measured using precision-recall break-even point

worse results than the baseline algorithm. Furthermore, in general, augmenting documents with associative concepts is more effective than adding hyponyms.

Table 9 shows the results of adding synonyms. The addition of synonyms does not provide the improvement achieved by the previous two strategies. Since we can not rank synonyms of a given candidate concept, we just add all its synonyms into documents, which inevitably brings in noise. As mentioned in Sect. 3.1.1, “Redirect” links also cope with capitalization and spelling variations, abbreviations, colloquialisms, and scientific terms. For instance, the document #15264 of Reuters-21578 talks about copper mining, and the synonyms of the word “copper” are “copper (element)”, “copper band”, “copper sheet”, “copper sheet metal”, “cuprous”, “cuprum”, “copper mine”, “cupper”, “cupric” and “element 29”. We noticed that “cupper” maybe a misprint of “copper”, and therefore should not be added.

Finally, we experimented with the addition of both hyponyms and associative concepts into documents. The results (Table 10) show that, when adding direct hyponyms and the top five associative concepts for each candidate concept, we achieve the best results.

Table 9 The effect of adding synonyms

Dataset	Reuters		20NG		OHSUMED	
	Micro	Macro	Micro	Macro	Micro	Macro
Baseline	0.877	0.605	0.868	0.865	0.602	0.548
Add synonyms	0.854	0.597	0.852	0.858	0.524	0.515

Table 10 The effect of adding both hyponyms and associative concepts

Dataset	Reuters		20NG		OHSUMED	
	Micro	Macro	Micro	Macro	Micro	Macro
Baseline	0.877	0.605	0.868	0.865	0.602	0.548
Combined	**0.912**	**0.631**	**0.917**	**0.904**	**0.672**	**0.593**

The classification performance is measured using precision-recall break-even point

6 Conclusions

We introduced a methodology to build a thesaurus from Wikipedia, and to leverage the thesaurus to facilitate text categorization. Wikipedia is a rich and extensive resource of encyclopedic knowledge; our approach makes use of its structure to build the thesaurus. To improve text classification, we enrich documents with related concepts, and perform explicit disambiguation to determine the proper meaning of each polysemous concept expressed in documents. By doing so, background knowledge can be introduced into documents, which overcomes the limitations of the BOW approach. The experimental results demonstrate the effectiveness of our approach.

In our future work, we plan to ameliorate the effect of adding synonyms by filtering "Redirect" links. After removing meaningless redirect links such as spelling variations, and keeping significant ones such as synonyms and abbreviations, we expect that the addition of synonyms will indeed be beneficial.

Furthermore, our disambiguation strategy can be improved. The thesaurus builds a relation graph for each concept, which includes synonyms, hyponyms and associative concepts. The use of such graph can be useful to achieve an improved disambiguation process.

Our thesaurus only explores part of Wikipedia resources. Additional information can be mined. For example, currently our thesaurus does not take advantage of anchor texts. The anchor text of a link provides synonyms for the titles of the linked articles. Moreover, Wikipedia includes articles written in many languages, and therefore cross language information retrieval can be supported.

Acknowledgments We thank the anonymous reviewers for their very useful comments and suggestions. Part of this work was done while Pu Wang was an intern at Microsoft Research Asia.

References

1. Hotho A, Staab S, Stumme G (2003) Wordnet improves text document clustering. In: Proceedings of the semantic web workshop at SIGIR'03
2. Gabrilovich E, Markovitch S (2005) Feature generation for text categorization using world knowledge. In Proceedings of the 19th international joint conference on artificial intelligence (IJCAI'05)
3. Gabrilovich E, Markovitch S (2006) Overcoming the brittleness bottleneck using Wikipedia: enhancing text categorization with encyclopedic knowledge. In: Proceedings of the 21nd AAAI conference on artificial intelligence (AAAI'06)
4. Milne D, Medelyan O, Witten IH (2006) Mining domain-specific Thesauri from Wikipedia: a case study. In: Proceedings of 2007 IEEE/WIC/ACM international conference on web intelligence (WI'06)
5. Bunescu R, Pasca M (2006) Using encyclopedic knowledge for named entity disambiguation. In: Proceedings of the 11th conference of the european chapter of the association for computational linguistics (EACL'06)
6. Strube M, Ponzetto SP (2006) WikiRelate! computing semantic relatedness using Wikipedia. In: Proceedings of the 21nd AAAI conference on artificial intelligence (AAAI'06)
7. Porter MF (1980) An algorithm for suffix stripping. Program 14(3):130–137
8. Agirre E, Rigau G (1995) A proposal for word sense disambiguation using conceptual distance. In: Proceedings of the 1st international conference on recent advances in natural language processing (RANLP'95)
9. Reuters-21578 text categorization test collection, Distribution 1.0. Reuters. 1997. http://www.daviddlewis.com/resources/testcollections/reuters21578/
10. Hersh W, Buckley C, Leone T, Hickam D (1994) OHSUMED: an interactive retrieval evaluation and new large test collection for research. In: Proceedings of the 17th annual international ACM-SIGIR conference on research and development in information retrieval (SIGIR'94), pp 192–201
11. Lang K (1995) Newsweeder: learning to filter netnews. In: Proceedings of the 12th international conference on machine learning (ICML'95), pp 331–339

12. Joachims T (1998) Text categorization with support vector machines: learning with many relevant features. In: Proceedings of the 10th european conference on machine learning (ECML'98), pp 137-142
13. Yang Y, Liu X (1999) A re-examination of text categorization methods. In: Proceedings of the 22th annual international ACM-SIGIR conference on research and development in information retrieval (SIGIR'99), pp 42–49
14. de Buenaga Rodriguez M, Gomez Hidalgo JM, Agudo BD (1999) Using WordNet to complement training information in text categorization. In: The 2nd international conference on recent advances in natural language processing (RANLP'97)
15. Dave K, Lawrence S, Pennock DM (2003) Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: Proceedings of the 12th international World Wide Web conference (WWW'03)
16. Ponzetto SP, Strube M (2007) Deriving a large scale taxonomy from Wikipedia. In: Proceedings of the 22nd AAAI conference on artificial intelligence (AAAI'07)
17. Urena-Lopez LA, Buenaga M, Gomez JM (2001) Integrating linguistic resources in TC through WSD. Comput Hum 35:215C230
18. Miller G (1995) WordNet: a lexical database for english. Communications of the ACM
19. Wikipedia (2001). http://en.wikipedia.org/wiki/Wikipedia:About
20. Open Directory Project (1998). http://dmoz.org

Author Biographies

Pu Wang received a BE degree from Beihang University, Beijing, China, in 2004 and an MS degree from Peking University, Beijing, China, in 2007. From 2005 to 2007, he was an intern in the Machine Learning Group at Microsoft Research Asia, Beijing, China. He is currently a PhD student at the Department of Computer Science, George Mason University, USA. His research interests focus on machine learning and data mining.

Jian Hu is currently an Assistant Researcher at Microsoft Research Asia, Beijing, China. He received master and bachelor degrees in the Department of Computer Science and Technology, at Shanghai Jiao Tong University in 2006 and 2003, respectively. His current research interests include information retrieval, natural language processing and Web usage data mining.

Hua-Jun Zeng is a researcher in the Machine Learning Group at Microsoft Research Asia, Beijing, Peopel's Republic of China. He received his Master degree in Computer Science from the Shanghai Jiao-Tong University, and Bachelor degree in Industrial Automation from Shanghai Jiao-Tong University. His research interests include information retrieval, text mining, log mining and machine learning.

Zheng Chen joined Microsoft Research, China in March 1999 to pursue his wide-ranging research interests in Machine Learning, Information Retrieval, Speech Recognition, Natural Language Processing, Multimedia information retrieval, personal information management, and Artificial intelligence. He received his bachelors and PhD engineering degrees in Computer Science Department of Tsinghua University in 1994 and 1999.