Building Semantic Kernels for Text Classification using Wikipedia

Pu Wang and Carlotta Domeniconi
Department of Computer Science
George Mason University
pwang7@gmu.edu, carlotta@cs.gmu.edu

ABSTRACT

Document classification presents difficult challenges due to the sparsity and the high dimensionality of text data, and to the complex semantics of the natural language. The traditional document representation is a word-based vector (Bag of Words, or BOW), where each dimension is associated with a term of the dictionary containing all the words that appear in the corpus. Although simple and commonly used, this representation has several limitations. It is essential to embed semantic information and conceptual patterns in order to enhance the prediction capabilities of classification algorithms. In this paper, we overcome the shortages of the BOW approach by embedding background knowledge derived from Wikipedia into a semantic kernel, which is then used to enrich the representation of documents. Our empirical evaluation with real data sets demonstrates that our approach successfully achieves improved classification accuracy with respect to the BOW technique, and to other recently developed methods.

Categories and Subject Descriptors

I.5.3 [**Pattern recognition**]: Clustering—*algorithms, similarity measure*; I.7.0 [**Document and Text Processing**]: General

General Terms

Algorithms

Keywords

Text Classification, Wikipedia, Kernel Methods, Semantic Kernels

1. INTRODUCTION

Text categorization represents a challenging problem to data mining and machine learning communities due to the growing demand for automatic information retrieval systems. Traditionally, document classification is based on a "Bag of Words" approach (BOW): each document is represented as a vector with a dimension for each term of the dictionary containing all the words that appear in the corpus. The value associated to a given term reflects its frequency of occurrence within the corresponding document (Term Frequency, or *tf*), and within the entire corpus (Inverse Document Frequency, or *idf*). This technique has three major drawbacks: (1) it breaks multi-word expressions, like "Text Classification", into independent features; (2) it maps synonymous words into different components; and (3) it considers polysemous words (i.e., words with multiple meanings) as one single component. Although traditional preprocessing of documents, such as eliminating stop words, pruning rare words, stemming, and normalization, can improve the representation, its effect is still limited. It is therefore essential to further embed semantic information and conceptual patterns to be able to enhance the prediction capabilities of classification algorithms.

We overcome the shortages of the BOW approach by embedding background knowledge constructed from Wikipedia into a semantic kernel, which is used to enrich the representation of documents. Our semantic kernel is able to keep multi-word concepts unbroken, it captures the semantic closeness of synonyms, and performs word sense disambiguation for polysemous terms.

Attempts have been made in the literature to construct semantic kernels from ontologies, such as *WordNet*. Alhough empirical results have shown improvements in some cases, the applicability of WordNet to improve classification accuracy is very limited. This is because the ontology is manually built, and its coverage is far too restricted. For this reason, we make use of Wikipedia, the world largest electronic encyclopedia to date. In Wikipedia, a concept is illustrated with an article, and each concept belongs to at least one category (e.g., the concept "jaguar" belongs to the category "felines"). A concept may redirect to another concept if the two are synonyms. If a concept is polysemous, Wikipedia provides a disambiguation page, which lists all possible meanings of the polysemous concept. Each meaning is again illustrated with an article.

Thus, Wikipedia is a rich source of linguistic information. However, Wikipedia is not a structured thesaurus like Wordnet. In [21], the authors constructed an informative thesaurus from Wikipedia, which explicitely derives synonymy, polysemy, hyponymy, and associative relations between concepts. The resulting thesaurus offers a much broader coverage than any manually built one, such as WordNet, and surpasses them in accuracy it can achieve [15]. In this pa-

KDD'08, August 24–27, 2008, Las Vegas, Nevada, USA.

per, we leverage the thesaurus derived from Wikipedia [21] to embed semantic information in our document representation, and therefore achieve improved classification accuracy based on documents' content.

The paper is organized as follows. Section 2 discusses related work. Section 3 briefly describes the structure of Wikipedia, and how the authors in [21] build a thesaurus from Wikipedia. In Section 4, we introduce our technique for building semantic kernels. Experimental results are presented and discussed in Section 5. Finally, Section 6 provides conclusions and ideas for future work.

2. RELATED WORK

Research has been done to exploit ontologies for content-based categorization of large corpora of documents. In particular, WordNet has been widely used. Siolas et al. [19] build a semantic kernel based on WordNet. Their approach can be viewed as an extension of the ordinary Euclidean metric. Jing et al. [12] define a term similarity matrix using WordNet to improve text clustering. Their approach only uses synonyms and hyponyms. It fails to handle polysemy, and breaks multi-word concepts into single terms. Hotho et al. [11] integrate WordNet knowledge into text clustering, and investigate word sense disambiguation strategies and feature weighting schema by considering the hyponymy relations derived from WordNet. Their experimental evaluation shows some improvement compared with the best baseline results. However, considering the restricted coverage of WordNet, the effect of word sense disambiguation is quite limited. The authors in [6, 20] successfully integrate the WordNet resource for document classification. They show improved classification results with respect to the Rocchio and Widrow-Hoff algorithms. Their approach, though, does not utilize hypernyms and associate terms (as we do with Wikipedia). Although [5] utilized WordNet synsets as features for document representation and subsequent clustering, the authors did not perform word sense disambiguation, and found that WordNet synsets actually decreased clustering performance.

Gabrilovich et al. [8, 9] propose a method to integrate text classification with Wikipedia. They first build an auxiliary text classifier that can match documents with the most relevant articles of Wikipedia, and then augment the BOW representation with new features which are the concepts (mainly the titles) represented by the relevant Wikipedia articles. They perform feature generation using a multi-resolution approach: features are generated for each document at the level of individual words, sentences, paragraphs, and finally the entire document. This feature generation procedure acts similarly to a retrieval process: it receives a text fragment (such as words, a sentence, a paragraph, or the whole document) as input, and then maps it to the most relevant Wikipedia articles. This method, however, only leverages text similarity between text fragments and Wikipedia articles, ignoring the abundant structural information within Wikipedia, e.g. internal links. The titles of the retrieved Wikipedia articles are treated as new features to enrich the representation of documents [8, 9]. The authors claim that their feature generation method implicitly performs words sense disambiguation: polysemous words within the context of a text fragment are mapped to the concepts which correspond to the sense shared by other context words. However, the processing effort is very high, since each document needs to be scanned many times. Furthermore, the feature generation procedure inevitably brings a lot of noise, because a specific text fragment contained in an article may not be relevant for its discrimination. Furthermore, implicit word sense disambiguation processing is not as effective as explicit disambiguation, as we perform in our approach.

Milne et al. [15] build a professional, domain-specific thesaurus of agriculture from Wikipedia. Such thesaurus takes little advantage of the rich relations within Wikipedia articles. On the contrary, our approach relies on a general thesaurus, which supports the processing of documents concerning a variety of topics. We investigate a methodology that makes use of such thesaurus, to enable the integration of the rich semantic information of Wikipedia into a kernel.

3. WIKIPEDIA AS A THESAURUS

Wikipedia (started in 2001) is today the largest encyclopedia in the world. Each article in Wikipedia describes a topic (or concept), and it has a short title, which is a well-formed phrase like a term in a conventional thesaurus [15]. Each article belongs to at least one category, and hyperlinks between articles capture their semantic relations, as defined in the international standard for thesauri [11]. Specifically, the represented semantic relations are: equivalence (*synonymy*), hierarchical (*hyponymy*), and associative.

Wikipedia contains only one article for any given concept (called *preferred term*). *Redirect* hyperlinks exist to group equivalent concepts with the preferred one. Figure 1 shows an example of a redirect link between the synonyms "puma" and "cougar". Besides synomyms, redirect links handle capitalizations, spelling variations, abbreviations, colloquialisms, and scientific terms. For example, "United States" is an entry with a large number of redirect pages: acronyms (U.S.A., U.S., USA, US); Spanish translations (Los Estados, Unidos, Estados Unidos); common mispellings (Untied States); and synonyms (Yankee land) [2].

Disambiguation pages are provided for a polysemous concept. A disambiguation page lists all possible meanings associated with the corresponding concept, where each meaning is discussed in an article. For example, the disambiguation page of the term "puma" lists 22 associated concepts, including animals, cars, and a sportswear brand.

Each article (or concept) in Wikipedia belongs to at least one category, and categories are nested in a hierarchical organization. Figure 1 shows a fragment of such structure. The resulting hierarchy is a directed acyclic graph, where multiple categorization schemes co-exist [15].

Associative hyperlinks exist between articles. Some are one-way links, others are two-way. They capture different degrees of relatedness. For example, a two-way link exists between the concepts "puma" and "cougar", and a one-way link connects "cougar" to "South America". While the first link captures a close relationship between the terms, the second one represents a much weaker relation. (Note that one-way links establishing strong connections also exist, e.g., from "Data Mining" to "Machine Learning".) Thus, meaningful measures need to be considered to properly rank associative links between articles. Three such measures have been introduced in [21]: *Content-based*, *Out-link category-based*, and *Distance-based*. We briefly describe them here. In Section 4.2 we use them to define the proximity between associative concepts.

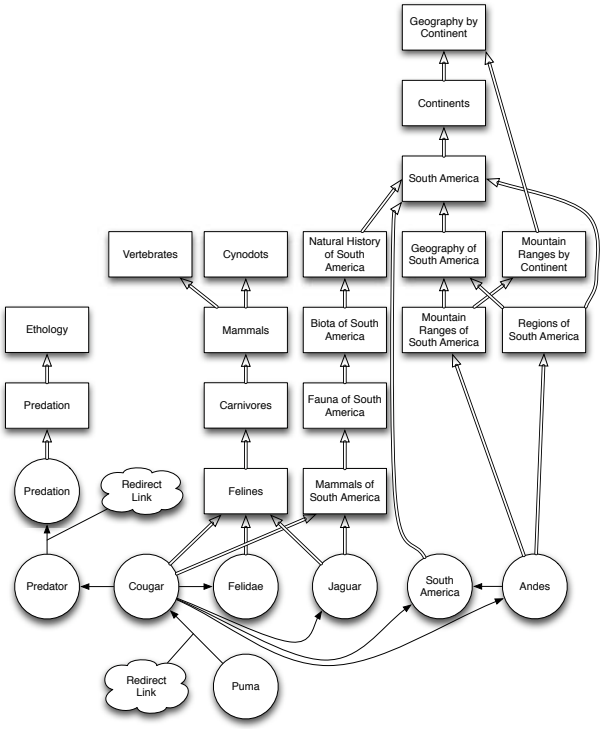

Figure 1: A fragment of Wikipedia's taxonomy

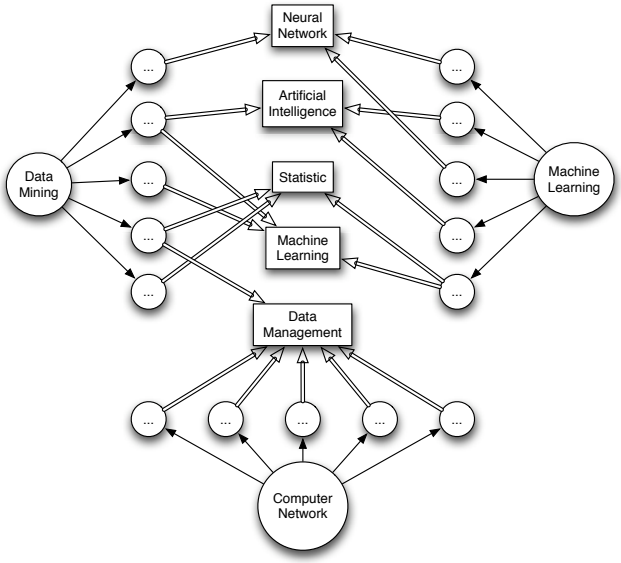

Figure 2: Out-link categories of the concepts "Machine Learning", "Data Mining", and "Computer Network"

The content-based measure is based on the bag-of-words representation of Wikipedia articles. Each article is modeled as a *tf-idf* vector; the associative relation between two articles is then measured by computing the cosine similarity between the corresponding vectors. Clearly, this measure (denoted as S_{BOW}) has the same limitations of the BOW approach.

The out-link category-based measure compares the out-link categories of two associative articles. The out-link categories of a given article are the categories to which out-link articles from the original one belong. Figure 2 shows (a fraction of) the out-link categories of the associative concepts "Data Mining", "Machine Learning", and "Computer Network". The concepts "Data Mining" and "Machine Learning" share 22 out-link categories; "Data Mining" and "Computer Network" share 10; "Machine Learning" and "Computer Network" share again the same 10 categories. The larger the number of shared categories, the stronger the associative relation between the articles. 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 i-th component is the number of out-link articles which belong to the i-th category. The cosine similarity is then computed between the resulting vectors, and denoted as S_{OLC}. The computation of S_{OLC} for the concepts illustrated in Figure 2 gives the following values, which indeed reflect the actual semantic of the corresponding terms: $S_{OLC}(\textit{Data Mining}, \textit{Machine Learning}) = 0.656$, $S_{OLC}(\textit{Data Mining}, \textit{Computer Network}) = 0.213$, $S_{OLC}(\textit{Machine Learning}, \textit{Computer Network}) = 0.157$.

The third measure is a distance measure (rather then a similarity measure like the first two). The distance between two articles is measured as the length of the shortest path connecting the two categories they belong to, in the acyclic graph of the category taxonomy. The distance measure is normalized by taking into account the depth of the taxonomy. It is denoted as D_{cat}.

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

$$S_{overall} = \lambda_1 S_{BOW} + \lambda_2 S_{OLC} + (1 - \lambda_1 - \lambda_2)(1 - D_{cat}) \quad (1)$$

where $\lambda_1, \lambda_2 \in (0, 1)$ are parameters to weigh the individual measures. Equation (1) allows to rank all the associative articles linked to any given concept.

4. CONCEPT-BASED KERNELS

As mentioned before, the "Bag of Words" (BOW) approach breaks multi-word expressions, maps synonymous words into different components, and treats polysemous as one single component. Here, we overcome the shortages of the BOW approach by embedding background knowledge into a semantic kernel, which is then used to enrich the representation of documents.

In the following, we first describe how to enrich text documents with semantic kernels, and then illustrate our technique for building semantic kernels using background knowledge constructed from Wikipedia.

4.1 Kernel Methods for Text

The BOW model (also called Vector Space Model, or VSM) [18] of a document d is defined as follows:

$$\phi : d \mapsto \phi(d) = (tf(t_1, d), tf(t_2, d), \ldots, tf(t_D, d)) \in \mathcal{R}^D$$

where $tf(t_i, d)$ is the frequency of term t_i in document d, and D is the size of the dictionary.

The basic idea of kernel methods is to embed the data in a suitable feature space, such that solving the problem (e.g.,

Table 1: Example of document term vectors

	Puma	Cougar	Feline	...
d_1	2	0	0	...
d_2	0	1	0	...

classification or clustering) in the new space is easier (e.g., linear). A kernel represents the similarity between two objects (e.g., documents or terms), defined as dot-product in this new vector space. The kernel trick [17] allows to keep the mapping implicit. In other words, it is only required to know the inner products between the images of the data items in the original space. Therefore, defining a suitable kernel means finding a good representation of the data objects.

In text classification, semantically similar documents should be mapped to nearby positions in feature space. In order to address the omission of semantic content of the words in VSM, a transformation of the document vector of the type $\tilde{\phi}(d) = \phi(d)S$ is required, where S is a semantic matrix. Different choices of the matrix S lead to different variants of VSM. Using this transformation, the corresponding vector space kernel takes the form

$$\begin{aligned} \tilde{k}(d_1, d_2) &= \phi(d_1)SS^\top\phi(d_2)^\top \\ &= \tilde{\phi}(d_1)\tilde{\phi}(d_2)^\top \end{aligned} \quad (2)$$

Thus, the inner product between two documents d_1 and d_2 in feature space can be computed efficiently directly from the original data items using a kernel function.

The semantic matrix S can be created as a composition of embeddings, which add refinements to the semantics of the representation. Therefore, S can be defined as:

$$S = RP \quad (3)$$

where R is a diagonal matrix containing the term weightings or relevance, and P is a *proximity matrix* defining the semantic similarities between the different terms of the corpus. One simple way of defining the term weighting matrix R is to use the inverse document frequency (idf).

P has non-zero off diagonal entries, $P_{ij} > 0$, when the term i is semantically related to the term j. Embedding P in the vector space kernel corresponds to representing a document as a less sparse vector, $\phi(d)P$, which has non-zero entries for all terms that are semantically similar to those present in document d. There are different methods for obtaining P [22, 1]. Here, we leverage the external knowledge provided by Wikipedia.

Given the thesaurus built from Wikipedia, it is straightforward to build a proximity (or similarity) matrix P. Here is a simple example. Suppose the corpus contains one document d_1 that talks about pumas (the animal). A second document d_2 discusses the life of cougars. d_1 contains instances of the word "puma", but no occurrences of "cougar". Vice versa, d_2 containts the word "cougar", but "puma" does not appear in d_2. Fragments of the BOW representations of d_1 and d_2 are given in Table 1, where the feature values are term frequencies. The two vectors may not share any features (e.g., neither document contains the word "feline"). Table 2 shows a fragment of a proximity matrix computed from the thesaurus based on Wikipedia. The similarity between "puma" and "cougar" is one since the two terms are synonyms. The similarity between "puma" and "feline" (or "cougar" and "feline") is 0.4, as computed according to equation (1). Table 3 illustrates the updated term vectors of documents d_1 and d_2, obtained by multipling the original term vectors (Table 1) with the proximity matrix of Table 2. The new vectors are less sparse, with non-zero entries not only for terms included in the original document, but also for terms semantically related to those present in the document. This enriched representation brings documents which are semantically related closer to each other, and therefore it facilitates the categorization of documents based on their content. We now discuss the enrichment steps in detail.

Table 2: Example of a proximity matrix

...	Puma	Cougar	Feline	...
Puma	1	1	0.4	...
Cougar	1	1	0.4	...
Feline	0.4	0.4	1	...
...				...

Table 3: Example of "enriched" term vectors

	Puma	Cougar	Feline	...
d_1'	2	2	0.8	...
d_2'	1	1	0.4	...

4.2 Semantic Kernels derived from Wikipedia

The thesaurus derived from Wikipedia provides a list of concepts. For each document in a given corpus, we search for the Wikipedia concepts mentioned in the document. Such concepts are called *candidate concepts* for the corresponding document. When searching for candidate concepts, we adopt an exact matching strategy, by which only the concepts that explicitly appear in a document become the candidate concepts. (If an m-gram concept is contained in an n-gram concept (with $n > m$), only the last one becomes a candidate concept.) We then construct a vector representation of a document, which contains two parts: terms and candidate concepts. For example, consider the text fragment "Machine Learning, Statistical Learning, and Data Mining are related subjects". Table 4 shows the traditional BOW term vector for this text fragment (after stemming), where feature values correspond to term frequencies. Table 5 shows the new vector representation, where boldface entries are candidate concepts, and non-boldface entries correspond to terms.

We observe that, for each document, if a word only appears in candidate concepts, it won't be chosen as a term feature any longer. For example, in the text fragment given above, the word "learning" only appears in the candidate concepts "Machine Learning" and "Statistical Learning". Therefore, it doesn't appear as a term in Table 5. On the other hand, according to the traditional BOW approach, after stemming, the term "learn" becomes an entry of the term vector (Table 4). Furthermore, as illustrated in Table 5, we keep each candidate concept as it is, without performing stemming or splitting multi-word expressions, since multi-word candidate concepts carry meanings that cannot be captured by the individual terms.

Table 4: Traditional BOW term vector

Entry	*tf*
machine	1
learn	2
statistic	1
data	1
mine	1
relate	1
subject	1

Table 5: Vector of candidate concepts and terms

Entry	*tf*
machine learning	1
statistical learning	1
data mining	1
relate	1
subject	1

When generating the concept-based vector representation of documents, special care needs to be given to polysemous concepts, i.e., concepts that have multiple meanings. It is necessary to perform word sense disambiguation to find the specific meaning of ambiguos concepts within the corresponding document. For instance, the concept "puma" is an ambiguous one. If "puma" is mentioned in a document, its actual meaning in the document should be identified, i.e., whether it refers to a kind of animal, or to a sportswear brand, or to something else. In Section 4.2.1 we explain how we address this issue.

Once the candidate concepts have been identified, we use the Wikipedia thesaurus to select synonyms, hyponyms, and associative concepts of the candidate ones. The vector associated to a document d is then enriched to include such related concepts: $\phi(d) = (<terms>,<candidate\ concepts>,<related\ concepts>)$. The value of each component corresponds to a *tf-idf* value. The feature value associated to a related concept (which does not appear explicitely in any document of the corpus) is the *tf-idf* value of the corresponding candidate concept in the document. Note that this definition of $\phi(d)$ already embeds the matrix R as defined in equation (3).

We can now define a proximity matrix P for each pair of concepts (candidate and related). The matrix P is represented in Table 6. For mathematical convenience, we also include the terms in P. P is a symmetrical matrix whose elements are defined as follows. For any two terms t_i and t_j, $P_{ij} = 0$ if $i \neq j$; $P_{ij} = 1$ if $i = j$. For any term t_i and any concept c_j, $P_{ij} = 0$. For any two concepts c_i and c_j:

$$P_{ij} = \begin{cases} 1 & \text{if } c_i \text{ and } c_j \text{ are synonyms;} \\ \mu^{-depth} & \text{if } c_i \text{ and } c_j \text{ are hyponyms;} \\ S_{overall} & \text{if } c_i \text{ and } c_j \text{ are associative concepts;} \\ 0 & \text{otherwise.} \end{cases}$$

$S_{overall}$ is computed according to equation (1). *depth* represents the distance between the corresponding categories of two hyponym concepts in the category structure of Wikipedia. For example, suppose c_i belongs to category A and c_j to category B. If A is a direct subcategory of B, then $depth = 1$. If A is a direct subcategory of C, and C is a direct subcategory of B, then $depth = 2$. μ is a back-off factor, which regulates how fast the proximity between two concepts decreases as their category distance increases. (In our experiments, we set $\mu = 2$.)

Table 6: Proximity matrix

	Terms				Concepts			
Terms	1	0	$\cdots$	0	0	0	$\cdots$	0
	0	1	$\cdots$	0	0	0	$\cdots$	0
	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	0	0	$\cdots$	1	0	0	$\cdots$	0
Concepts	0	0	$\cdots$	0	1	a	$\cdots$	b
	0	0	$\cdots$	0	a	1	$\cdots$	c
	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	0	0	$\cdots$	0	b	c	$\cdots$	1

Table 7: Cosine similarity between the Reuters document #9 and the Wikipedia's articles corresponding to the different meanings of the term "Stock"

Meanings of "Stock"	*Similarity with Reuters #9*
Stock (finance)	**0.2037**
Stock (food)	0.1977
Stock (cards)	0.1531
Stocks (plants)	0.1382
Stock (firearm)	0.0686
Livestock	0.0411
Inventory	0.0343

By composing the vector $\phi(d)$ with the proximity matrix P, we obtain our extended vector space model for document d: $\tilde{\phi}(d) = \phi(d)P$. $\tilde{\phi}(d)$ is a less sparse vector with non-zero entries for all concepts that are semantically similar to those present in d. The strength of the value associated with a related concept depends on the number and frequency of occurrence of candidate concepts with a close meaning. An example of this effect can be observed in Table 3. Let us assume that the concept "feline" is a related concept (i.e., did not appear originally in any of the given documents). "feline" appears in document d_1' with strength 0.8, since the original document d_1 contains two occurrences of the synonym concept "puma" (see Table 1), while it appears in d_2' with a smaller strength (0.4), since the original document d_2 contains only one occurrence of the synonym concept "cougar" (see Table 1). The overall process, from building the thesaurus from Wikipedia, to constructing the proximity matrix and enriching documents with concepts, is depicted in Figure 3.

4.2.1 Disambiguation of Concept Senses

If a candidate concept is polysemous, i.e. it has multiple meanings, it is necessary to perform word sense disambiguation to find its most proper meaning in the context where it appears, prior to calculating its proximity to other related concepts. We utilize text similarity to do explicit word sense disambiguation. This method computes document similarity by measuring the overlapping of terms. For instance, the Reuters-21578 document #9 talks about stock splits, and the concept "stock" in Wikipedia refers to sev-

Table 8: The hyponym, associative, and synonym concepts introduced in Reuters document #9

Candidate Concepts	Hyponyms	Associative Concepts	Synonyms
Stock	Stock market Equity securities Corporate finance	House stock Bucket shop Treasury stock Stock exchange Market capitalization	Stock (finance)
Shareholder	Stock market	Board of directors Business organizations Corporation Fiduciary Stock	Shareholders
Board of directors	Business law Corporate governance Corporations law Management	Chief executive officer Shareholder Fiduciary Corporate governance Corporation	Boards of directors

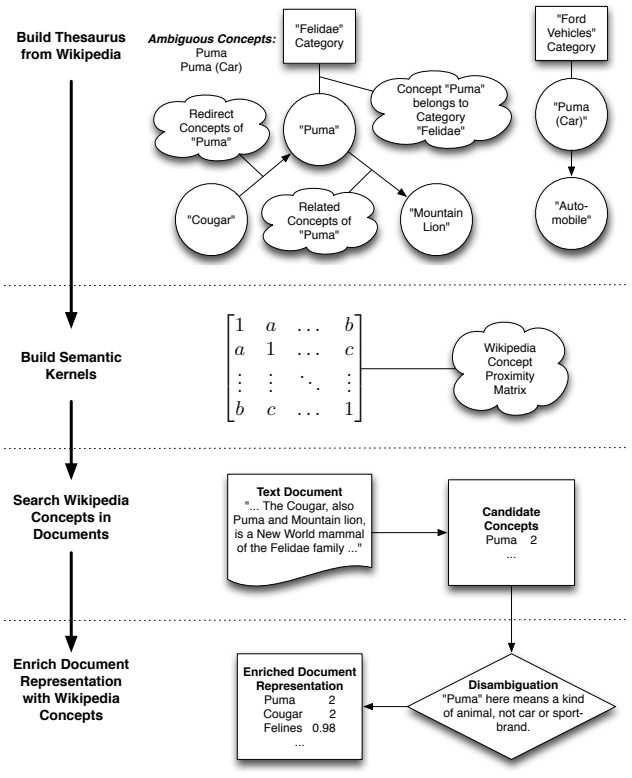

Figure 3: The process that derives semantic kernels from Wikipedia

eral different meanings, as listed in Table 7. The correct meaning of a polysemous concept is determined by comparing the cosine similarities between the *tf-idf* term vector of the text document (where the concept appears), and each of Wikipedia's articles (corresponding *tf-idf* vectors) describing the different meanings of the polysemous concept. The larger the cosine similarity between two *tf-idf* term vectors is, the higher the similarity between the two corresponding text documents. Thus, the meaning described by the article with the largest cosine similarity is considered to be the most appropriate one. From Table 7, the Wikipedia article describing "stock" (finance) has the largest similarity with the Reuters document #9, and this is indeed confirmed to be the case by manual examination of the document (document #9 belongs to the Reuters category "earn").

As mentioned above, document #9 discusses the stock split of a company, and belongs to the Reuters category "earn". The document contains several candidate concepts, such as "stock", "shareholder", and "board of directors". Table 8 gives an example of the corresponding related concepts identified by our method, and added to the vector representation of document #9 of the Reuters data set.

5. EMPIRICAL EVALUATION

5.1 Processing Wikipedia XML data

The evaluation was performed using the Wikipedia XML Corpus [7]. The Wikipedia XML Corpus contains processed Wikipedia data parsed into an XML format. Each XML file corresponds to an article in Wikipedia, and maintains the original ID, title and content of the corresponding Wikipedia article. Furthermore, each XML file keeps track of the linked article ID, for every redirect link and hyperlink contained in the original Wikipedia article.

5.1.1 Filtering Wikipedia Concepts

We do not include all concepts of Wikipedia in the thesaurus. Some concepts, such as "List of ISO standards" or "1960s", do not contribute to the achievement of improved discrimination among documents. Thus, before building the thesaurus from Wikipedia, we remove concepts deemed not useful. To this end, we implement a few heuristics. First,

Table 9: Number of terms, concepts, and links after filtering

Terms in Wikipedia XML corpus	**659,388**
Concept After Filtering	495,214
Redirected Concepts	413
Categories	113,484
Relations in Wikipedia XML corpus	**15,206,174**
Category to Subcategory	145,468
Category to Concept	1,447,347
Concept to Concept	13,613,359

all concepts of Wikipedia which belong to categories related to chronology, such as "Years", "Decades", and "Centuries", are removed. Second, we analyze the titles of Wikipedia articles to decide whether they correspond to useful concepts. In particular, we implement the following rules:

1. If the title of an article is a multi-word title, we check the capitalization of all the words other than prepositions, determiners, conjunctions, and negations. If all the words are capitalized, we keep the article.
2. If the title is one word title, and it occurs in the article more than three times [2], we keep the article.
3. Otherwise, the article is discarded.

After filtering Wikipedia concepts using these rules, we obtained about 500,000 concepts to be included in the thesaurus. Table 9 provides a break down of the resulting number of elements (terms, concepts, and links) used to build the thesaurus, and therefore our semantic kernels. In particular, we note the limited number of redirected concepts (413). This is due to the fact that redirect links in Wikipedia often refers to the plural version of a concept, or to misspellings of a concept, and they are filtered out in the XML Corpus. Such variations of a concept, in fact, should not be added to the documents, as they would contribute only noise. For example, in Table 8, the synonyms associated to the candidate concepts "Shareholder" and "Board of visitors" correspond to their plural versions. Thus, in practice they are not added to the documents.

5.2 Data Sets

We used four real data sets (Reuters-21578, OHSUMED, 20 Newsgroups, and Movies) to evaluate the performance of our approach for document classification. In the following, a description of each data set is provided.

1. Reuters-21578 [3][1]. This collection of documents is one of the most widely used for text categorization research. The documents in the Reuters-21578 collection appeared on the Reuters newswire in 1987. The documents were assembled and indexed with categories by personnel from Reuters Ltd. and Carnegie Group, Inc. Following common practice, we used the ModApte split for training and testing purposes. Namely, after removing non-labeled data and documents without a body, we used 7,775 documents for training, and 3,019 documents for testing, for a total of more than 100 categories.
2. OHSUMED [10][2]. The OHSUMED test collection is a set of 348,566 references from MEDLINE, the online medical information database, consisting of titles and/or abstracts from 270 medical journals over a five-year period (1987-1991). The available fields are title, abstract, MeSH indexing terms, author, source, and publication type. About two-thirds (233,445) of the references contain an abstract. Each document is labeled with a subset of 500 of the MeSH terms. Following Joachims [13], we used a subset of documents from 1991 that have abstracts, taking the first 10,000 labeled documents for training, and the following 10,000 labeled documents for testing.
3. 20 Newsgroups (20NG) [14][3]. The 20 Newsgroups data set is a popular collection of approximately 20,000 newsgroup documents, partitioned nearly evenly across 20 different newsgroups (about 1,000 documents per class). For training and testing, a 4-fold cross-validation was implemented.
4. Movie Reviews (Movies) [16][4]. This collection contains 2,000 reviews of movies from the Internet Movie Database archive. Half of the reviews express a positive sentiment (opinion) about the movie, and half express a negative opinion. For the purpose of text categorization, we concentrated in discriminating between positive and negative ratings. For training and testing, we again performed 4-fold cross-validation.

5.3 Methods and Evaluation Measures

We used a Support Vector Machine (SVM) to learn models for the classification of documents. This choice was driven by the fact that SVMs have provided state-of-the-art results in the literature for text categorization. We conducted all the experiments using the software LIBSVM [4][5], and a linear kernel. We compare the performance of three methods:

1. *Baseline*. The traditional BOW (*tf-idf*) representation of documents is used in this case. We preprocessed the documents by eliminating stop words, pruning rare words (i.e., words with document frequency equal to one), and stemming the terms.
2. *Wiki-Enrich.* This is the method proposed in [21]. This technique makes use of the same thesaurus used in our approach, but the enrichment process is fundamentally different. The vector space model of a document is extended with direct hyponyms, synonyms,

[1]Available at http://www.daviddlewis.com/resources/testcollections/reuters21578/

[2]Available via anonymous ftp from medir.ohsu.edu in the directory /pub/ohsumed

[3]Available at http://people.csail.mit.edu/jrennie/20Newsgroups/

[4]Available at http://www.cs.cornell.edu/people/pabo/movie-review-data

[5]Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm

Table 10: Micro-ageraged and Macro-averaged precision results

Data sets	*Baseline*		*Wiki-Enrich.*		*Wiki-SK*	
	Micro	Macro	Micro	Macro	Micro	Macro
Reuters-21578	0.8557	0.5936	0.8747	0.6025	0.8976	0.6039
OHSUMED	0.5146	0.5032	0.5473	0.5110	0.5967	0.5227
20NG	0.8351	-	0.8597	-	0.8992	-
Movies	0.8124	-	0.8512	-	0.8637	-

and the 5 closest associative concepts. The feature representation of such related concepts is again at the single term level (i.e., multi-word concepts are broken into individual terms), and their values are simply the resulting *tf-idf* values.

3. *Wiki-SK*. This is our proposed approach. *SK* is for Semantic Kernels.

We measured the performance of each method using the *precision ratio*, as defined in [23]:

$$precision = \frac{categories\ found\ and\ correct}{total\ categories\ found}$$

For the Reuters and OHSUMED data sets, we report both the *micro-averaged* and the *macro-averaged* precisions, since these data sets are multi-labeled, and the categories differ in size substantially. The micro-averaged precision operates at the document level, and is primarily affected by the categorization accuracy of the larger categories. On the other hand, the macro-averaged precision averages results over all categories; thus, small categories have more influence on the overall performance.

5.4 Experimental Settings

For both methods *Wiki-Enrich.* and *Wiki-SK* the parameters λ_1 and λ_2 of equation (1) were tuned according to the methology suggested in [21]. As a result, the values $\lambda_1 = 0.4$ and $\lambda_2 = 0.5$ were used in our experiments.

When retrieving the related concepts in our approach (*Wiki-SK*), for building the proximity matrix P (and the vector $\tilde{\phi}(d)$), we consider the direct hyponyms, the synonyms, and the 10 closest associative concepts for each Wikipedia (candidate) concept found in a document. However, not all candidate concepts present in a document are allowed to introduce related concepts. In order to identify the *eligible* candidate concepts, we calculate the cosine similarity between the *tf-idf* vector representation of the document containing the candidate concept, and the *tf-idf* vector representation of the Wikipedia article describing the same concept. Such similarity is computed for each candidate concept in a document. Only the top three to five candidate concepts that provide the highest similarities become the eligible ones. The specific number (between three and five) is chosen based on the length of the document. This process can be seen as an extension, to all concepts, of the procedure for the disambiguation of concept senses (introduced in Section 4.2.1). This refinement proved to be effective in pruning concepts which do not express the focus of the topic being discussed in a document. Thus, it is successful in avoiding the expansion of the vector of terms with noisy features.

5.5 Results

Table 10 shows the micro-averaged and the macro-averaged precision values obtained for the three methods on the four data sets. Our method *Wiki-SK* provides higher micro and macro precision values on all data sets. The improvement with respect to the *Baseline* is significant for all four data sets. The largest improvements with respect to *Wiki-Enrich.* are obtained for the OHSUMED and 20NG data sets. In comparison with *Wiki-Enrich.*, our kernel-based method is capable of modeling relevant multi-word concepts as individual features, and of assigning meaningful strength values to them via our proximity matrix. Furthermore, our heuristic to filter concepts, and our selection mechanism to identify eligible candidate concepts successfully avoid the introduction of noisy features. Overall, our results demonstrate the benefit and potential of embedding semantic knowledge into document representation by means of Wikipedia-based kernels.

6. CONCLUSIONS AND FUTURE WORK

To the best of our knowledge, this paper represents a first attempt to improve text classification by defining concept-based kernels using Wikipedia. Our approach overcomes the limitations of the bag-of-words approach by incorporating background knowledge derived from Wikipedia into a semantic kernel, which is then used to enrich the content of documents. This methodology is able to keep multi-word concepts unbroken, it captures the semantic closeness to synonyms, and performs word sense disambiguation for polysemous terms.

We note that our approach to highlight the semantic content of documents, from the definition of a proximity matrix, to the disambiguation of terms and to the identification of eligible candidate concepts, is totally unsupervised, i.e. makes no use of the class labels associated to documents. Thus, the same enrichment procedure could be extended to enhance the clustering of documents, when indeed class labels are not available, or too expensive to obtain.

On the other hand, for classification problems where class labels are available, one could use them to facilitate the disambiguation process, and the identification of crucial concepts in a document. Furthermore, class labels can be exploited to measure the correlation between terms and classes, and consequently define proper term weightings for the matrix R in equation (3). We plan to explore these avenues in our future work.

7. ACKNOWLEDGMENTS

This work was in part supported by NSF CAREER Award IIS-0447814.

8. REFERENCES

[1] L. AlSumait and C. Domeniconi. Local Semantic Kernels for Text Document Clustering. In *Workshop on Text Mining, SIAM International Conference on Data Mining*, Minneapolis, MN, 2007. SIAM.

[2] R. Bunescu and M. Pasca. Using encyclopedic knowledge for named entity disambiguation. In *Conference of the European Chapter of the Association for Computational Linguistics*, Trento, Italy, 2006.

[3] Carnegie Group, Inc. and Reuters, Ltd. *Reuters-21578 text categorization test collection*, 1997.

[4] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001.

[5] K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *International World Wide Web Conference*, Budapest, Hungary, 2003.

[6] M. de Buenega Rodriguez, J. M. Gomez-Hidalgo, and B. Diaz-Agudo. Using wordnet to complement training information in text categorization. In *International Conference on Recent Advances in Natural Language Processing*, 1997.

[7] L. Denoyer and P. Gallinari. The Wikipedia XML Corpus. *SIGIR Forum*, 2006.

[8] E. Gabrilovich and S. Markovitch. Feature generation for text categorization using world knowledge. In *International Joint Conference on Artificial Intelligence*, Edinburgh, Scotland, 2005.

[9] E. Gabrilovich and S. Markovitch. Overcoming the brittleness bottleneck using wikipedia: enhancing text categorization with encyclopedic knowledge. In *National Conference on Artificial Intelligence (AAAI)*, Boston, Massachusetts, 2006.

[10] W. Hersh, C. Buckley, T. Leone, and D. Hickam. Ohsumed: An interactive retrieval evaluation and new large test collection for research. In *Conference on Research and Development in Information Retrieval*, Dublin, Ireland, 1994. ACM/Springer.

[11] A. Hotho, S. Staab, and G. Stumme. Wordnet improves text document clustering. In *Semantic Web Workshop, SIGIR Conference*, Toronto, Canada, 2003. ACM.

[12] L. Jing, L. Zhou, M. K. Ng, and J. Z. Huang. Ontology-based distance measure for text clustering. In *Workshop on Text Mining, SIAM International Conference on Data Mining*, Bethesda, MD, 2006. SIAM.

[13] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *European Conference on Machine Learning*, Chemnitz, Germany, 1998. Springer.

[14] K. Lang. Newsweeder: Learning to filter netnews. In *International Conference on Machine Learning*, Tahoe City, California, 1995. Morgan Kaufmann.

[15] D. Milne, O. Medelyan, and I. H. Witten. Mining domain-specific thesauri from wikipedia: A case study. In *International Conference on Web Intelligence*, Hong Kong, 2006.

[16] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Conference on Empirical Methods in Natural Language Processing*, Pennsylvania, Philadelphia, 2002.

[17] J. Shawe-Taylor and N. Cristianini. *Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[18] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[19] G. Siolas and F. d'Alché Buc. Support vector machines based on a semantic kernel for text categorization. In *International Joint Conference on Neural Networks (IJCNN'00)*, pages 205–209, Como, Italy, 2000. IEEE.

[20] L. A. Urena-Lopez, M. Buenaga, and J. M. Gomez. Integrating linguistic resources in TC through WSD. *Computers and the Humanities*, 35:215–230, 2001.

[21] P. Wang, J. Hu, H.-J. Zeng, L. Chen, and Z. Chen. Improving text classification by using encyclopedia knowledge. In *International Conference on Data Mining*, pages 332–341, Omaha, NE, 2007. IEEE.

[22] S. K. M. Wong, W. Ziarko, and P. C. N. Wong. Generalized vector space model in information retrieval. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 18–25, Montreal, Canada, 1985.

[23] Y. Yang and J. Pedersen. A comparative study on feature selection in text categorization. In *International Conference on Machine Learning*, Nashville, Tennessee, 1997. Morgan Kaufmann.