Designing Semantic Kernels as Implicit Superconcept Expansions

Revised Version to appear in Proceedings of ICDM-2006

Stephan Bloehdorn*, Roberto Basili, Marco Cammisa** and Alessandro Moschitti****
*Institute AIFB, University of Karlsruhe, Germany
{bloehdorn}@aifb.uni-karlsruhe.de
**University of Rome 'Tor Vergata', Italy
{basili,cammisa,moschitti}@info.uniroma2.it

Abstract

Recently, there has been an increased interest in the exploitation of background knowledge in the context of text mining tasks, especially text classification. At the same time, kernel-based learning algorithms like Support Vector Machines have become a dominant paradigm in the text mining community. Amongst other reasons, this is also due to their capability to achieve more accurate learning results by replacing standard linear kernel (bag-of-words) with customized kernel functions which incorporate additional a-priori knowledge. In this paper we propose a new approach to the design of 'semantic smoothing kernels' by means of an implicit superconcept expansion using well-known measures of term similarity. The experimental evaluation on two different datasets indicates that our approach consistently improves performance in situations where (i) training data is scarce or (ii) the bag-of-words representation is too sparse to build stable models when using the linear kernel.

1 Introduction

Finding means for organizing, analyzing and searching the ever growing amounts of textual documents is a challenging task in knowledge management. Text classification systems [Sebastiani, 2002], which aim at automatically classifying text documents into predefined thematic classes are one approach to govern this growing complexity. Their design is mainly based on machine learning methods among which Support Vector Machines (SVMs) [Vapnik *et al.*, 1997] along with other kernel-based algorithms have become a dominant technique during the last years. The popularity of SVMs stems from two vital properties: one the one hand, being firmly grounded in statistical learning theory, they exhibit very high generalization capabilities. On the other hand, they easily incorporate prior knowledge about the target domain by means of a specific choice of the employed *kernel function*. Pioneered by [Joachims, 1998], SVMs have been heavily used for text classification, typically showing good results. The standard feature representation used in text classification settings is the so called *bag-of-words* model originating from Information Retrieval. Here, documents are encoded as vectors whose dimensions correspond to the terms in the overall corpus and the entries correspond to appropriately weighted counts of the terms in the document. Typically, the inner product (or the cosine, i.e. its normalized variant) between two vectors is used as kernel hence making the similarity of two documents dependant only on the amount of terms they share. While this approach has an appealing simplicity, it suffers from data sparseness problems in those cases where reliable distributions of terms are not available in the training documents.

To overcome the above drawback, recently, there has been an increased interest in using prior knowledge about semantic dependencies between terms of different surface form. In text-mining tasks, *semantic smoothing kernels* have emerged as one paradigm to approach this task [Siolas and d'Alché Buc, 2000; Cristianini *et al.*, 2002; Mavroeidis *et al.*, 2005; Basili *et al.*, 2005]. The knowledge encoded by such kernels is derived either from explicit background knowledge in the form of semantic networks or implicitely from statistics about the co-occurrence of terms. The rationale behind these approaches is the observation that the index terms that constitute the feature space cannot be regarded as mutually orthogonal dimensions but rather as dimensions with varying degrees of semantic similarity (with synonymous terms being the most extreme cases where distinct dimensions actually correspond to a single one). In this view, linear kernels within the *bag-of-words* paradigm appear as a rough approximation only. Despite this, literature studies indicate that the *bag-of-words* approach achieves very good results. This is typically explained by the implicit assumption that stable patterns can be detected even in a poor representation as long as sufficient training data is available. However, in those cases where training data is scarce or the representation of individual instances is hampered by extreme sparseness, an a-priori bias in form of a more adequate kernel is likely to boost the overall performance.

In this paper we investigate the use of a new type of semantic smoothing kernels for text classification. We exploit the similarity of two terms within a semantic smoothing matrix which generalizes the standard linear kernel by giving the vector components *across dimensions* a say when evaluating the kernel of two documents. To determine appropriate term similarities, we represent index terms as instances within a seperate *concept space* and determine their mutual similarities by means of their dot product in this space. The term space is indexed by the nodes of a semantic network and the corresponding feature weightings are derived from a number of conceptually well-motivated measures of semantic similarity.

We assess the performance of our approach by means of experiments on the well-known Reuters-21578 corpus using very small subsets of the typically employed 'ModApte' training set partitioning. Additionally, since the benefit of the above similarity metrics is emphasized

when data is highly affected by data sparseness [Basili *et al.*, 2005], we carried out a set of experiments in the domain of question classification (QC). Question classification aims at detecting the type of a question, e.g. whether it asks for a person or for an organization which is critical to locate and extract the right answers in question answering systems. A major challenge of question classification compared to standard text classification settings is that questions typically contain only extremely few words which makes this setting a typical victim of data sparseness.

Our evaluation studies indicate a consistent improvements of results in situations of little training data and data sparseness. The results on Reuters-21578 show that when only few training examples are available our kernels based on semantic similarity outperform one based on *bag-of-words*. The results of our second series of experiments indicate that improvements are even higher in the case of TREC question datasets independently of the size of the training examples.

The remainder of this paper is structured as follows. Section 2 provides preliminary notions on kernels and semantic networks. Section 3 presents a number of measures of lexical semantic relatedness and related notions that will be used for designing semantic kernels. Section 4 describes the design of the semantic kernels whereas Section 5 gives an account on the performance of these in a series of evaluation experiments. We review the related work in section 6 while concluding in section 7 with a summary of the contributions, final remarks and a discussion of envisioned future work.

2 Preliminaries

In this section, we briefly review the basic concepts of SVMs, Kernel Methods (section 2.1) and a few definitions and notions about semantic networks (section 2.2).

2.1 Support Vector Machines and Kernel Methods

Support Vector Machines are state-of-the-art learning methods based on the earlier idea of linear classification. The distinguishing feature of SVMs is the theoretically well motivated and efficient training strategy for determining the separating hyperplane based on the margin maximization principle. The other interesting property of SVMs is their capability of naturally incorporating data-specific notions of item similarity by means of a corresponding kernel function.

Definition 1 (Kernel Function)**.** *Any function κ that for all $x, z \in X$ satisfies $\kappa(x, z) = \langle \phi(x), \phi(z) \rangle$, is a valid kernel, whereby X is the input vector space under consideration and ϕ is a suitable mapping from X to a feature space F.*

Note that the choice of a particular kernel function implies an indirect mapping to a feature space different from the input space x. Kernels can be designed by either choosing an explicit mapping function ϕ and incorporating it into an inner product or by directly defining the kernel function κ while making sure that it complies with the requirement of being a positive semi-definite function. The reader is referred to the rich literature for further information on SVMs and kernel methods, e.g. [Müller *et al.*, 2001; Shawe-Taylor and Cristianini, 2004] for comprehensive introductions.

2.2 Semantic Networks

The target semantic dependencies are encoded in structures which we call, for simplicity, *semantic networks*. These can be seen as *directed graphs*.

Definition 2 (Semantic Network)**.** *A* semantic network *is a tuple $\mathcal{S} := (\mathcal{C}, R)$ consisting of a set $\mathcal{C}$ whose elements are called* concept identifiers*, and a relation $R \subseteq \mathcal{C} \times \mathcal{C}$ called* semantic link*. Often, we call concept identifiers just* concepts *and the semantic links just* links*, for sake of simplicity. For two concepts $c_1, c_2 \in \mathcal{C}$ and $(c_1, c_2) \in R$ we say that c_2 is* superconcept of c_1 *or vice versa that c_1 is* subconcept of c_2.

Our formalization is deliberately generic to capture a wide range of linguistic resources, taxonomies and ontologies. However, in this work, we restrict our attention to WordNet[1], a free lexical reference system and semantic network [Fellbaum, 1998]. WordNet organizes English terms into groups of synonyms (*synsets*) connected by a number of semantic relations. As most of the previous related work, we focus on the hypernym/hyponym relations for nouns that correspond to the superconcept/subconcept relations introduced above.

The measures nextly introduced require three further notions. By *distance* (*d*) of two concepts c_1 and c_2, we refer to the number of superconcept edges between c_1 and c_2. These can be easily computed from the network's adjacency matrix using the Floyd-Warshall algorithm [Floyd, 1962] for all pairs of concepts. The notion of the *depth* (*dep*) of a concept relates to the frequent assumption of a tree-like structure of the semantic network having a unique root element. For an acyclic graph (which we assume in the remainder), a root element can be introduced which becomes superconcept of all concept nodes that are not equipped with outgoing superconcept edges[2]. The depth of a concept is then defined as the distance of the concept to the root. Based on this, the *lowest super ordinate (lso)* of two concepts refers to the concept with maximal depth that subsumes them both.

3 Measuring Semantic Relatedness

The measurement of semantic similarity is a problem that pervades computational linguistics with respect to a large number of applications in natural language processing. A large amount of work has been devoted to defining measures of lexical semantic similarity or its opposite, lexical semantic distance[3] based on semantic networks – in most cases WordNet. In this section, we give a brief review of a number of measures of this type that have been used within this paper with emphasis on a compact description of the measures and the main rationales behind them, pointing the interested reader to [Budanitsky and Hirst, 2006] for a more detailed and most recent survey of the field.

[1] `http://www.cogsci.princeton.edu/~wn/`

[2] This is particularly true for the WordNet noun hierarchy, which up to version 2.0 defined 9 distinct *unique beginner concepts* up to which each concept can be traced.

[3] Note that [Budanitsky and Hirst, 2006] have made a good point in distinguishing the more general concept of *semantic relatedness* from *semantic similarity*. While this distinction is useful in terms of a fine-grained interpretation of the specific type of relation that ties two lexical entities together, it is not critical in the context of our work.

Path Based Measures The *inverted path length* can be seen as an example of particularly simple way to compute semantic similarity between two concepts in a semantic network:

$$sim_{IPL}(c_1, c_2) = \frac{1}{(1 + d(c_1, c_2))^{\alpha}},$$

whereby α specifies the rate of decay. Note that the [Siolas and d'Alché Buc, 2000] have used this measure to define semantic smoothing kernels for the first time. While the similarity of this measure is intriguing, it does not comply with the intuition that concepts closer to the root of the semantic network should have a higher distance compared to concepts far away. Among many others, the measure introduced by Wu&Palmer [Wu and Palmer, 1994] tries to scale the similarity with respect to the depth of the concepts and their lowest super ordinate in the semantic network:

$$sim_{WUP}(c_1, c_2) = \frac{2\, dep(lso(c_1, c_2))}{d(c_1, lso(c_1, c_2)) + d(c_2, lso(c_1, c_2)) + 2\, dep(lso(c_1, c_2))}.$$

Information Content Based Measures A different type of measures tries to incorporate additional knowledge about the information content of a concept besides the structural setup of the semantic network. Resnik [Resnik, 1999] has argued that neither the individual edges nor the absolute depth in a taxonomy can be considered as homogeneous indicators of the semantic content of a concept. To overcome this problem, he introduces the notion of the probability $P(c)$ of encountering a concept c. This probability is typically estimated by the relative frequencies of the lexicalizations of the concept in a corpus relevant for the target domain whereby the counts of subconcepts equally contribute to their respective superconcepts. Resnik follows the argumentation of information theory in quantifying the *information concept (IC)* of an observation as the negative log likelihood. Intuitively, a universal root concept having a probability of 1 carries an information content equal to zero while rare concepts carry high information content values. By means of the argument that *"one key to the similarity of two concepts is the extent to which they share information in common"* he proposes to measure the similarity of two concepts by means of the formula:

$$sim_{RES}(c_1, c_2) = -\log\, P(lso(c_1, c_2)).$$

Based on this proposal, Lin [Lin, 1998] derived a theoretically well motivated similarity measure given by:

$$sim_{LIN}(c_1, c_2) = \frac{2\, \log\, P(lso(c_1, c_2))}{\log\, P(c_1) + \log P\,(c_2)}.$$

As an extension to the original measure proposed by Resnik, the information content of the compared concepts is used as a means for normalization.

4 Designing Semantic Kernels

As motivated in section 1, the aim of our work is to embed the knowledge about the topological relations of the semantic networks in kernel functions. This allows the learning algorithm to relate distinct but similar features during kernel evaluation.

4.1 Semantic Kernels

The general concept of semantic smoothing kernels was for the first time introduced in [Siolas and d'Alché Buc, 2000] and subsequently revisited in [Cristianini *et al.*, 2002; Mavroeidis *et al.*, 2005; Basili *et al.*, 2005], each time based on different design principles.

Definition 3 (Semantic Smoothing Kernel)**.** *The semantic smoothing kernel for two data items (documents) $x, z \in X$ is given by $\kappa(x, z) = x'Qz$ where Q is a square symmetric matrix whose entries represent the semantic proximity between the dimensions of the input space X.*

Note that the definition of a kernel in section 2.1 implies that Q must be a positive semi-definite matrix. Conceptually this means that Q can be decomposed by $Q = PP'$ thus revealing the underlying feature mapping as $\phi(x) = P'x$. The matrix P is a $n \times m$ matrix whereby n corresponds to the dimensionality of the input space X and provides a linear transformation of the input document into a feature space of (possibly far higher) dimensionality m, similar to a query expansion. A first approach to designing semantic kernels would be to embed the pairwise measures of lexical semantic relatedness directly into the matrix Q. However, the requirement of Q being positive semi-definite can typically not be ensured for all measures in general if used directly.

4.2 Semantic Kernels based on Superconcept Expansions

As a way to avoid indefinite similarity matrices, authors like [Siolas and d'Alché Buc, 2000] have enforced the positive-definiteness of Q by exlicitely computing it from $Q = PP'$ whereby the information about the similarities is now encoded in the matrix P. While this approach ensures the validity of the Kernel, the interpretation of the resulting smoothing kernel is less clear. Conceptually, it maps each concept to a number of related concepts and the shared weight of these determines the overall similarity between two terms.

Following own prior work in a differnet setting [Bloehdorn and Hotho, 2004], we follow a different approach for the construction of Q which is, however, also based on an explicit construction of the type $Q = PP'$. We choose a setup of P such that it provides a mapping into the space of all possible *superconcepts* of the input instances, i.e. the terms or concepts in question. That is, the rows of P correspond to vector representations of the concepts of the input space by means of their respective superconcepts. The similarity of two concepts in the resulting smoothing matrix Q is thus the dot product of the vectors of their respective superconcepts. This approach is intuitive as we can typically regard two concepts as similar if they share a large number of superconcepts as opposed to sharing only few superconcepts.

Recently, [Mavroeidis *et al.*, 2005] have proposed this approach motivated by the observation that the dot product of two terms represented as vectors of their respective superconcepts can be shown to be equivalent to a number of popular similarity measures (among them the Resnik measure, but not the Lin and Wup measures) given a particular weighting scheme of the superconcept representation. However, this prior work has focused on the simple case of giving the superconcepts in the mapping P full and equal weight (i.e. restricting P to a 0/1 matrix) while varying the number of superconcepts that are considered. Consistent

with an argument made by the same authors, we argue that the variation of the number of superconcepts yields a high variance and its a-priori choice will always be an ad-hoc decision.

As an alternative approach, we have investigated the use of different weighting schemes for the representation of the superconcepts in P motivated by the following considerations:

1. The weight a superconcept c_j receives in the vectorial description of a concept c_i should be influenced by its distance from c_i.
2. The weight a superconcept c_j receives in the vectorial description of a concept c_i should be influenced by its overall depth in the semantic network.

Based on these rationales and the measures introduced in section 3, we have investigated the following weighting schemes:

full: No weighting, i.e. $P_{ij} = 1$ for all superconcepts c_j of c_i and $P_{ij} = 0$ otherwise.

full-ic: Weighting using information content of c_j, i.e. $P_{ij} = sim_{RES}(c_i, c_j)$.

path-1: Weighting based on inverted path length, i.e. $P_{ij} = sim_{IPL}(c_i, c_j)$ for all superconcepts c_j of c_i and $P_{ij} = 0$ otherwise using the parameter $\alpha = 1$.

path-2: The same but using the parameter $\alpha = 2$.

lin: Weighting using the Lin similarity measure, i.e. $P_{ij} = sim_{LIN}(c_i, c_j)$.

wup: Weighting using the Wu&Palmer similarity measure, i.e. $P_{ij} = sim_{WUP}(c_i, c_j)$.

The different weighting schemes behave differently wrt the above motivations. While full does not implement any of them, full-ic considers rationale 2 while path-1 and path-2 consider rationale 1. The schemes lin and wup reflect combinations of both rationales.

5 Experimental Evaluation

In a series of experiments we aimed at showing that our approach is effective for IR and data mining applications. For this purpose, we experimented with two different datasets related to two different mining tasks: Reuters-21578 for traditional Text Categorization and TREC question classification corpus for advanced retrieval based on the Question Answering paradigm.

5.1 Experimental Setup

We implemented the semantic kernel within a custom kernel module for the current version of SVMlight[4] which is freely available for download[5]. For both Reuters-21578 and TREC datasets, we used the noun hierarchy of WordNet as the underlying semantic network. We first describe the general setup of the smoothing matrices in the following section whereas the results are reported in sections 5.2 and 5.3.

[4] http://svmlight.joachims.org/

[5] http://www.aifb.uni-karlsruhe.de/WBS/sbl/software/semkernel/

Proximity Matrix Setup

The setup of the smoothing matrices used in the evaluation experiments was based on the particular choice of the proximity matrix design, discussed in section 4, as well as on two simplifying assumptions.

Firstly, the existing bag-of-word representation of the documents required the design of a *term proximity matrix* as opposed to the *synset proximity matrix* implicitly assumed so far. We used a simple strategy that maps each term to its most frequent noun sense (if it exists). Note that this approach implies an inherent word sense disambiguation side effect, both with respect to the respective part-of-speech as well as to the chosen noun sense. While this effect is likely to have a negative impact on the results, the error introduced by this approach is systematic. In the light of these considerations, the results can also be seen as a pessimistic estimate of the potential effectiveness given a perfectly disambiguated input.

Secondly in the case of the Reuters-21578 experiments, we restricted the entries in the term proximity matrix to those terms having document frequencies of at least five. This speeds up the computation during kernel evaluation while we used the full term similarity matrix in the case of the question classification experiments. Entries that were undefined in the term proximity matrix – be it because a missing mapping to a noun synset or because of low document frequency – were implicitly assumed to take the default values (i.e. zero and one for off-diagonal and diagonal entries respectively) during kernel evaluation[6]. Frequency counts needed for the calculation of the measures making use of information content were obtained from (i) the complete Reuters-21578 collection in the case of the Reuters-21578 experiments or (ii) from the Brown corpus in the case of the experiments on the TREC question dataset[7].

5.2 Experiments on Reuters-21578

As basis for our experiments on Reuters-21578 we used the 'ModApte' split which divides the Reuters-21578 collection into 9,603 training documents, 3,299 test documents and 8,676 unused documents. We prepared the bag-of-words representation of the documents based on the standard preprocessing steps, namely tokenization, removal of the standard stopwords for English defined in the SMART stopword list and lemmatization, resulting in a total number of 32,443[8] distinct features which were all weighted using the standard TFIDF scheme.

Based on results of previous work, we were well aware of the fact that the introduction of prior semantic knowledge typically has a small effect when sufficient training data is available and in some cases may even degrade the performance compared to the linear kernel. In our experiments, we thus primarily aimed at quantifying performance

[6] Technically, this can be seen as defining the overall kernel κ as the sum of two individual kernels: $\kappa(x, y) = \kappa_s(x_s, z_s) + \kappa_t(x_t, z_t)$ whereby κ_s is the semantic smoothing kernel as introduced above and κ_t is the conventional linear kernel, defined on the vectors formed by restriction to the dimensions indexed in the smoothing matrix and the remaining dimensions respectively.

[7] This decision was motivated by the fact that word frequency estimations on the dataset itself would be rather unreliable due to its far smaller overall size.

[8] The high number of features compared to other published work is an effect of the preprocessing scheme that we used which includes sequences of digits as features. This is, however, unlikely to have a significant effect on the results.

gains in those cases where very little training data was available. For this purpose, we prepared small subsets of the ModeApte training set by randomly choosing 2%, 3%, 4% and 5% of the available training data. To account for the high inherent sampling variance, this approach was repeated 10 times for each of the 4 subset sizes resulting in a total number of 40 subsets. Note that we checked that at least one positive training document for each of the aforementioned 10 categories was present in every created training subset. Binary classification experiments were then conducted for each category and each subset resulting in a total number of 400 experiments in each run. While in each of these experiments the SVM classifier was trained using the respective subset, the corresponding testing was conducted using the full ModeApte test set.

Evaluation Results Table 1 summarizes the absolute macro F_1 values obtained over the different subsets of Reuters-21578 as explained above. The 'soft margin' parameter c that controls the influence of misclassified examples was set to $c = 0.1$ in all experiments. The results indicate a consistent improvement of the F_1 values for all of the smoothing kernels based on superconcept representations.

	Subset Size			
kernel	**02p**	**03p**	**04p**	**05p**
linear	0.45	0.51	0.54	0.57
full	0.50	0.53	0.57	0.58
full-ic	0.53*	0.55	0.60	0.61
path-1	0.50	0.54	0.59	0.61
path-2	0.48	0.53	0.57	0.59
lin	0.53*	0.57*	0.61*	0.62*
wup	0.52	0.55	0.59	0.61

Table 1: Absolute macro F1 results for Reuters-21578 subsets and different semantic smoothing kernels. The best result per subset is highlighted.

The extent of the improvement for the smoothing kernels based on superconcept representations relative to the linear kernel can be seen more clearly in figure 1. According to prior findings in [Mavroeidis *et al.*, 2005] the improvement gradually diminishes as more training data becomes available. Among the different weighting schemes for superconcept representations, the lin weighting scheme consistently outperforms the other measures. This finding confirms our assumptions on the desired structure of the weighting scheme as the Lin measure respects both the overall depth of the respective superconcept by virtue of the information content as well as the distance from the base concept by means of the difference in information content. On the contrary, the default scheme (full) that does not employ any weighting schemes tends to be inferior to other models that use them.

While we were not primarily interested in the application of our approach in those cases where sufficient training data is available, we have nevertheless investigated the effect of a selection of superconcept smoothing kernels when the full ModeApte training set is used. Figure 2 summarizes the results in terms of per-category F_1 values. The results indicate only little shifts in performance, typically degrading performance compared to the linear kernel to a small extent. This finding supports our assumption that

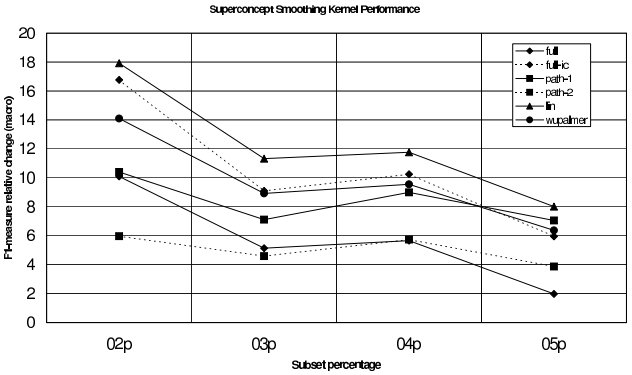

Figure 1: Relative improvements of the macro F1 results for Reuters-21578 subsets and different superconcept-based semantic smoothing kernels.

the semantic smoothing kernels are not particularly useful in scenarios where training data isn't scarce. Also note that the comparatively high complexity of semantic kernels limits the practical application for training based on large amounts of training datas.

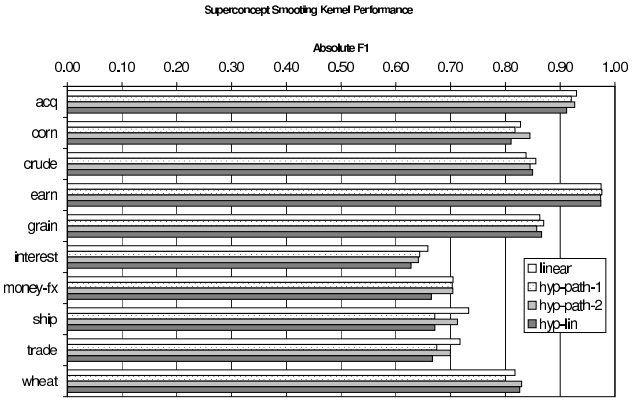

Figure 2: Absolute F1 results for 10 Reuters-21578 categories and selected superconcept-based semantic smoothing kernels using the full training sets.

5.3 Experiments on the TREC Question Classification Dataset

The long tradition of QA in TREC has produced a large question set used by several researchers which can be exploited for experiments on question classification (QC). Such questions are categorized according to different taxonomies of different *grains*. We consider the *coarse grained* classification scheme described in [Zhang and Lee, 2003; Li and Roth, 2002]: Abbreviations, Descriptions (e.g. *definition* and *manner*), Entity (e.g. *animal*, *body* and *color*), Human (e.g. *group* and *individual*), Location (e.g. *city* and *country*) and Numeric (e.g. *code* and *date*).

We used a set of questions labeled according to the above taxonomy. This dataset has also been employed in [Zhang and Lee, 2003; Li and Roth, 2002] and is freely available[9]. It is divided into 5,500 questions[10] for training and the 500 TREC 10 questions for testing. Similarly to the

[9] http://l2r.cs.uiuc.edu/~cogcomp/Data/QA/QC/

[10] These are selected from the 4500 English questions published by USC (Hovy et al., 2001), 500 questions annotated for rare classes and the 894 questions from TREC 8 and TREC 9.

first experiment, we preprocessed the questions using the usual steps leading to a total number of 8,075 distinct features weighted according to standard TFIDF. Again, we performed binary classification experiments on each of the 9 question types.

Evaluation Results In this experiment, we additionally applied several values of the 'soft margin' parameter c since our preliminary tests showed that its variation has an important influence on the overall results. Starting from $c = 0.1$ and $c = 1.0$ as typical default choices, we varied these in three steps to $c = 0.1 \ldots 0.3$ and $c = 1 \ldots 3$. Table 2 summarizes the absolute macro F_1 as well as the micro F_1 values obtained in the question classification setting. The best values per setting of c are highlighted.

macro-averaging						
	soft margin parameter *c*					
kernel	**0.1**	**0.2**	**0.3**	**1.0**	**2.0**	**3.0**
linear	0.21	0.38	0.47	0.62	0.63	0.64
full	0.38	0.49	0.55	0.61	0.61	0.68
full-ic	0.53*	0.53*	0.53	0.62	0.55	0.55
path-1	0.25	0.42	0.51	0.64*	0.64	0.64
path-2	0.22	0.39	0.47	0.63	0.65*	0.64
lin	0.36	0.49	0.56*	0.64*	0.62	0.70*
wup	0.34	0.49	0.54	0.62	0.61	0.69

macro-averaging						
	soft margin parameter *c*					
kernel	**0.1**	**0.2**	**0.3**	**1.0**	**2.0**	**3.0**
linear	0.09	0.25	0.34	0.55	0.57	0.58
full	0.27	0.38	0.45	0.55	0.56	0.68
full-ic	0.47*	0.46*	0.47*	0.60*	0.49	0.48
path-1	0.14	0.32	0.40	0.57	0.58	0.59
path-2	0.08	0.28	0.37	0.57	0.59*	0.58
lin	0.27	0.37	0.47*	0.57	0.57	0.69*
wup	0.23	0.37	0.45	0.56	0.56	0.68

Table 2: Absolute macro and micro F1 results for QC, for different values of *c* and different semantic smoothing kernels. The best results per setting of *c* are highlighted

The results indicate a consistent superior accuracy of the semantic smoothing kernels over the linear kernel baseline. With the exception of the full-ic setup, which shows good results for small values of c but detoriates later on, all semantic smoothing kernels improve performace in both the macro- as well as micro-averaged setting. According to the results on the Reuters-21578 experiments, the lin scheme achieves the best overall performance with a relative improvement of 9.32% for the macro F_1 value in the case of $c = 3$ (i.e. the setting for which the linear kernel achieves its maximum). We generally note that the improvements are more extreme for the case of small values of c while they appear more stable for larger values.

6 Related Work

To date, the work on integrating prior knowledge about feature similarities into text classification or other related tasks is quite scattered. Much of the early work in this direction was done in the context of *query expansion* techniques as e.g. reported in [Bodner and Song, 1996]. Early work in the direction of incorporation semantic background knowledge in combination with the Ripper classification algorithm was reported in [Scott and Matwin, 1999]. However, this early work showed negative results on two independent data sets. An alternative approach motivated by the idea of letting terms and higher level semantic features (including fixed depths of hypernyms) compete within the boosting algorithm paradigm was reported in [Bloehdorn and Hotho, 2004].

Semantic kernels were initially introduced in [Siolas and d'Alché Buc, 2000] using inverted path length as a similarity measure and subsequently explored in [Basili *et al.*, 2005] using conceptual density as a similarity measure among others. An alternative approach reported in [Cristianini *et al.*, 2002] aimed at incorporating the well-established technique of Latent Semantic Indexing (LSI) into the semantic kernel paradigm. As [Cristianini *et al.*, 2002] have pointed out, a similar framework has been used in [Jiang and Littman, 2000], although without the explicit notion of kernel functions. Recently [Mavroeidis *et al.*, 2005] reported on experiments with semantic smoothing kernels defined on superconcept representations such that it forms a natural basis for our work. In contrast to our approach, the authors used extensive word sense disambiguation (WSD) machinery which also formed a core contribution. Similar to [Bloehdorn and Hotho, 2004], the superconcept representations of terms were built upon fixed numbers of superconcepts without further weighting.

7 Conclusion

In this paper, we have investigated the design of semantic smoothing kernels. We similar framework to the one used in [Mavroeidis *et al.*, 2005] which expresses the similarity of term features by means of the shared superconcepts. In contrast to earlier work in this direction, we employed theoretically well motivated measures of semantic similarity between the base concepts under consideration and their corresponding superconcepts.

We conducted a series of experiments on the Reuters-21578 corpus using different sizes of training subsets and on the TREC question classification data. Our results indicate a consistent improvement in performance for superconcept semantic smoothing kernels in those cases where little training data is available or the feature representations are extremely sparse. Especially the lin scheme as proved to be a weighting scheme with stable improvements.

As both [Mavroeidis *et al.*, 2005] and [Bloehdorn and Hotho, 2004] have pointed out, the success of the introduction of semantic background knowledge in text-mining tasks critically depends on the employed word sense disambiguation strategy. Our experiments were deliberatly kept simple and thus did not use a decent word sense disambiguation step. We thus expect a further improvement in results when a powerful WSD technique (e.g. the one explored in [Mavroeidis *et al.*, 2005]) is applied. We aim at investigating this issue together with experiments on other corpora and the exploitation of semantic relations different from those based on superconcepts. We also aim at employing semantic kernels in scenarios different from text classification where the target background knowledge may take the form of arbitrary ontological structures. As a different trail we will investigate the combination of our semantic kernels with other types of kernels that also exploit more input structure.

Acknowledgements

This research was partially supported by the European Commission under contract IST-2003-506826 SEKT. The expressed con-

tent is the view of the author(s) but not necessarily the view of the SEKT consortium.

References

[Basili *et al.*, 2005] Roberto Basili, Marco Cammisa, and Alessandro Moschitti. A semantic kernel to classify texts with very few training examples. In *In Proceedings of the Workshop on Learning in Web Search, at the 22nd International Conference on Machine Learning (ICML 2005), Bonn, Germany*, 2005.

[Bloehdorn and Hotho, 2004] Stephan Bloehdorn and Andreas Hotho. Text classification by boosting weak learners based on terms and concepts. In *Proceedings of the 4th IEEE International Conference on Data Mining (ICDM 2004), 1-4 November 2004, Brighton, UK*, pages 331–334. IEEE Computer Society, NOV 2004.

[Bodner and Song, 1996] R. C. Bodner and F. Song. Knowledge-Based Approaches to Query Expansion in Information Retrieval. In *Advances in Artificial Intelligence*. Springer, New York, NY, USA, 1996.

[Budanitsky and Hirst, 2006] Alexander Budanitsky and Graeme Hirst. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47, March 2006.

[Cristianini *et al.*, 2002] Nello Cristianini, John Shawe-Taylor, and Huma Lodhi. Latent semantic kernels. *Journal of Intelligent Information Systems*, 18(2-3):127–152, 2002.

[Fellbaum, 1998] Christiane Fellbaum, editor. *WordNet - An Electronic Lexical Database*. MIT Press, 1998.

[Floyd, 1962] Robert W. Floyd. Algorithm 97: Shortest path. *Commun. ACM*, 5(6):345, 1962.

[Jiang and Littman, 2000] Fan Jiang and Michael L. Littman. Approximate dimension equalization in vector-based information retrieval. In *Proceedings of the 7th International Conference on Machine Learning. Stanford University June 29-July 2, 2000*, pages 423–430, 2000.

[Joachims, 1998] Thorsten Joachims. Text categorization with suport vector machines: Learning with many relevant features. In Claire Nedellec and Céline Rouveirol, editors, *Proceedings of ECML 1998*, volume 1398 of *Lecture Notes in Computer Science*, pages 137–142. Springer, 1998.

[Li and Roth, 2002] Xin Li and Dan Roth. Learning question classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING'02)*, 2002.

[Lin, 1998] Dekang Lin. An information-theoretic definition of similarity. In Jude W. Shavlik, editor, *Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconson, USA, July 24-27, 1998*, pages 296–304. Morgan Kaufmann, 1998.

[Mavroeidis *et al.*, 2005] Dimitrios Mavroeidis, George Tsatsaronis, Michalis Vazirgiannis, Martin Theobald, and Gerhard Weikum. Word sense disambiguation for exploiting hierarchical thesauri in text classification. In Alípio Jorge, Luís Torgo, Pavel Brazdil, Rui Camacho, and João Gama, editors, *Knowledge Discovery in Databases: Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2005), Porto, Portugal, October 3-7, 2005*, pages 181–192. Springer, 2005.

[Müller *et al.*, 2001] K.-R. Müller, S. Mika, G. Rätsch, S. Tsuda, and B Schölkopf. An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–202, 2001.

[Resnik, 1999] Philip Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130, 1999.

[Scott and Matwin, 1999] Sam Scott and Stan Matwin. Feature engineering for text classification. In Ivan Bratko and Saso Dzeroski, editors, *Proceedings of the Sixteenth International Conference on Machine Learning (ICML 1999), Bled, Slovenia, June 27 - 30, 1999.*, pages 379–388. Morgan Kaufmann, 1999.

[Sebastiani, 2002] F. Sebastiani. Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[Shawe-Taylor and Cristianini, 2004] John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[Siolas and d'Alché Buc, 2000] Georges Siolas and Florence d'Alché Buc. Support vector machines based on a semantic kernel for text categorization. In *IJCNN '00: Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN'00)-Volume 5*, page 5205, Washington, DC, USA, 2000. IEEE Computer Society.

[Vapnik *et al.*, 1997] V. Vapnik, S. Golowich, and A. Smola. Support vector method for function approximation, regression estimation, and signal processing. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 281–287, Cambridge, MA, 1997. MIT Press.

[Wu and Palmer, 1994] Zhibiao Wu and Martha Stone Palmer. Verb semantics and lexical selection. In *Proceedings of the 32nd. Annual Meeting of the Association for Computational Linguistics (ACL 1994)*, pages 133–138, 1994.

[Zhang and Lee, 2003] Dell Zhang and Wee Sun Lee. Question classification using support vector machines. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 26–32, New York, NY, USA, 2003. ACM Press.