Available online at www.sciencedirect.com
ScienceDirect
Information Processing and Management 43 (2007) 1183–1199
INFORMATION PROCESSING & MANAGEMENT
www.elsevier.com/locate/infoproman

Advanced learning algorithms for cross-language patent retrieval and classification

Yaoyong Li [a,*], John Shawe-Taylor [b]

[a] Department of Computer Science, The University of Sheffield, Regent Court, 211, Portobello Street, Sheffield S1 4DP, UK
[b] Department of Computer Science, University College London, Gower Street, London WC1E 6BT, UK

Received 1 September 2005; accepted 29 May 2006
Available online 22 January 2007

Abstract

We study several machine learning algorithms for cross-lan7guage patent retrieval and classification. In comparison with most of other studies involving machine learning for cross-language information retrieval, which basically used learning techniques for monolingual sub-tasks, our learning algorithms exploit the bilingual training documents and learn a semantic representation from them. We study Japanese–English cross-language patent retrieval using Kernel Canonical Correlation Analysis (KCCA), a method of correlating linear relationships between two variables in kernel defined feature spaces. The results are quite encouraging and are significantly better than those obtained by other state of the art methods. We also investigate learning algorithms for cross-language document classification. The learning algorithm are based on KCCA and Support Vector Machines (SVM). In particular, we study two ways of combining the KCCA and SVM and found that one particular combination called SVM_2k achieved better results than other learning algorithms for either bilingual or monolingual test documents.

Keywords: Machine learning; Cross-language patent retrieval; Cross-language document classification

1. Introduction

Automatic processing of patent information is important in industry, business, and law communities, because intellectual property is crucial in knowledge based economies and the number of patent documents is huge and increasing rapidly. One often wants to retrieve information from patents in a language which one is not familiar with. Cross-language patent retrieval and classification is desired in many cases.

Machine learning has been widely used for information processing and achieved state of the art performance in many tasks, such as document retrieval and classification, information filtering, and information extraction. The annual conference TREC[1] presents the latest results of machine learning for a variety of tasks

* Corresponding author. Tel.: +44 114 222 1924; fax: +44 114 222 1810.
E-mail addresses: yaoyong@dcs.shef.ac.uk (Y. Li), J.Shawe-Taylor@cs.ucl.ac.uk (J. Shawe-Taylor).
[1] See http://trec.nist.gov/.

0306-4573/$ - see front matter
doi:10.1016/j.ipm.2006.11.005

from text retrieval. The CLEF[2] in Europe and the cross-language tracks of NTCIR Workshops[3] in Japan are two important evaluation events for cross-language information retrieval. Machine learning has also been applied to cross-language information retrieval. However, most existing machine learning based systems for cross-language application just used machine learning for the monolingual sub-task and employ some machine translation techniques or bilingual directory for translating the query into another language. In this paper we explore learning algorithms for cross-language information retrieval and classification that can learn semantic representation from the given bilingual training data. We test the learning algorithms on cross-language patent retrieval and classification.

Cross-language information retrieval enables us to retrieve information from other languages using a query written in the language we are familiar with. A cross-language information retrieval system can be built up via two approaches. One is to use machine translation to translate the query so that the problem is transformed into a monolingual information retrieval task where a variety of techniques can be employed (e.g. Makita, Higuchi, Fujii, & Ishikawa, 2003). Another way is to first automatically induce a semantic correspondence between two languages by some automatic methods such as machine learning and then use it to project the query into the semantic space to accomplish cross-language information retrieval (e.g. Littman, Dumais, & Landauer, 1998; Vinokourov, Shawe-Taylor, & Cristianini, 2002). In Littman et al. (1998) cross-language latent semantic indexing (CL-LSI) was proposed as a fully automatic method for cross-language information retrieval, which produced results comparable to (and sometimes better than) those obtained with machine translation systems. Vinokourov et al. (2002) established kernel canonical correlation analysis (KCCA) for cross-language information retrieval and achieved significantly better performance than CL-LSI on an English–French corpus. The machine learning based method is interesting because its performance is comparable to the machine translation based methods but its implementation is easier. In this paper we study KCCA for Japanese–English patent retrieval.

In comparison to the previous works in Vinokourov et al. (2002) that also applied KCCA to cross-language information retrieval, we made some modifications on the methods. We used the *Cosine* measure of two projecting vectors instead of the distance measure used in the previous works. Our experiments showed that the Cosine measure gave better retrieval results than the distance measure. In regard of the experiments, we not only repeated their experiments on Japanese–English patent corpus, but also investigated other important issues such as the performance improvement of the learning system when using more training examples and dealing with large training data, which were not discussed in previous papers. More importantly, while the previous works only evaluated their learning algorithms on some synthesized retrieval tasks such as mate retrieval that given a document in one language searches its translation in another language, we also applied the KCCA method to query retrieval based on the NTCIR-3 corpus. Here we are given a query in one language and search the relevant documents in another language. It is therefore a realistic cross-language document retrieval scenario.

We also present learning algorithms and experiments for another interesting topic – cross-language document classification, which is useful for multi-lingual information management. Bel, Koster, and Villegas (2003) studied two different scenarios for cross-language document classification. One scenario assumed that bilingual documents for training and test were available such that one can learn one classifier from bilingual training data. Another scenario was to learn a monolingual classifier for language A and then translate the most important terms from language B to A to classify documents written in B. They used the Winnow learning algorithm and the Rocchio algorithm for classification and evaluated the methods on the English–Spanish ILO corpus. Olsson, Oard, and Hajič (2005) employed a general probabilistic Czech–English dictionary to translate Czech feature data into an English one and then classified Czech documents using the classifier learned from English training data. Gliozzo and Strapparava (2005) extracted a bilingual domain model from comparable corpora using latent semantic indexing method and defined a bilingual kernel based on a bilingual domain model which was used to learn an SVM classifier. Rigutini, Maggini, and Liu (2005) used a machine translation system to translate the English document into Italian and then learned a Naive-Bayes classifier

[2] See http://www.clef-campaign.org/.
[3] See http://research.nii.ac.jp/ntcir/.

based on the Italian translation of training documents and finally classified unlabeled Italian documents using the classifier.

In this paper we studied several learning algorithms which can be used in different scenarios for cross-language document classification. Some of the learning algorithms enable us to learn a classifier from one language and classify documents in other languages. Other learning algorithms require bilingual training documents and bilingual or monolingual test documents. Our learning algorithms are based on KCCA and support vector machines (SVM). SVM is a supervised learning algorithm which has achieved state of the art results for monolingual document classification (e.g. Joachims, 1998). KCCA or other methods were used to obtain the semantic correspondence between the training documents in two languages for cross-language document classification. We are particularly interested in the application of a two-view classification algorithm SVM_2k to our problem. The SVM_2k can be regarded as a combination of the SVM algorithm and the distance minimisation version of KCCA for two-view learning. It obtained better results than single view learning for image recognition (Meng, Shawe-Taylor, Szedmak, & Farquhar, 2004).

The rest of paper is organised as follows. In Section 2 we formulate kernel canonical correlation analysis in the context of cross-language text applications. Section 3 experimentally investigates the KCCA for Japanese–English cross-language patent retrieval and compares with the CL-LSI method. Section 4 presents the KCCA results for the query retrieval of the NTCIR-3 corpus. Section 5 describes several learning algorithms for cross-language document classification and presents the experimental results for the Japanese–English patent corpus. Section 6 concludes.

2. KCCA for cross-language text applications

Canonical correlation analysis (CCA), proposed by Hotelling (1936), aims to find basis vectors for two sets of variables such that the correlation between the projections of the variables onto these basis vectors are mutually maximised. CCA can be seen as using complex labels as a way of guiding feature selection toward the underlying semantics. CCA makes use of two views of the same semantic object to extract the representation of the semantics. Here semantics refers to the content of an object (e.g. document) and different views are the different representations of the object (i.e. the document's text in different languages). In an attempt to increase the flexibility of the feature selection, kernelisation of CCA (KCCA) has been applied to map the data to a higher-dimensional feature space via a kernel function. KCCA is particularly suitable for applications where the semantics of the object with two or more views are crucial. One such problem is cross-language information retrieval where the semantics refers to the content of a document and the texts of document in different languages represent different views.

For cross-language information retrieval, KCCA induces a set of basis vectors in feature space from a collection of bilingual documents. Those vectors can be regarded as a semantic representation of the bilingual corpus. Here the semantic representation means that a basis vector of KCCA corresponds to one theme or several mixed themes of the corpus, which are represented by the typical terms about the themes in the two languages. Fig. 1 shows two examples of the basis vector obtained from a Japanese–English patent collection (see Section 3 for the detailed explanations of this collection). Each such basis vector has more than 150

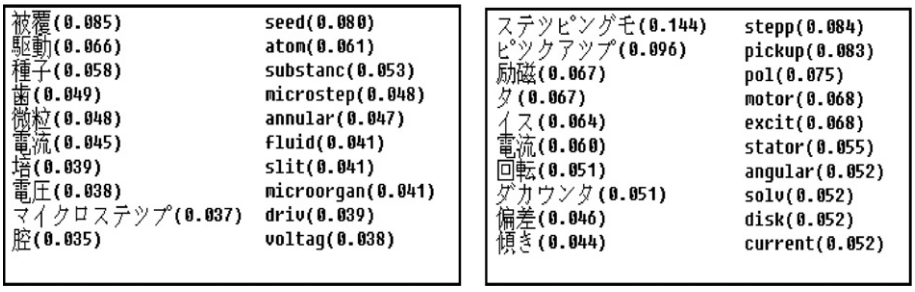

Fig. 1. The semantic representation of KCCA basis vector: 10 terms respectively in Japanese and English which correspond to the largest components of vector. Note that the English terms are the stemmed words.

thousands components, of which we only list the first 10 largest components (the values and the corresponding terms) respectively for Japanese and English. It looks as if the vector in the left part of Fig. 1 represents three mixed themes: a natural farming method, stepping motor and a new device for photo development, while the vector in the right part is mainly for one theme, stepping motor.

Since KCCA extracts distinct themes from a text collection and represents the themes respectively in two languages, we could represent a document in one or another language as some combination of the themes and use this kind of semantic representation for cross-language text applications such as information retrieval and document classification. For example, we can first obtain the semantic representations of a query in one language and some documents in another language by projecting them onto the KCCA basis vectors and then retrieve relevant documents for the query by comparing the semantic representations (see Section 3 for more details). In the following we will show how KCCA infers a set of basis vectors from a bilingual corpus as the semantic representation.

Suppose we are given N pairs of documents in two languages, i.e. every document c_i $(i = 1, \ldots, N)$ in one language is a translation of document d_i in another language. After some preprocessing, we obtain a feature vector $x_i \in \mathscr{X}$ for every document c_i and a feature vector $y_i \in \mathscr{Y}$ for document d_i, where $\mathscr{X}$ and $\mathscr{Y}$ are the feature spaces of the two languages, respectively. Using canonical correlation analysis (CCA), we can find some directions $f_x \in \mathscr{X}$ and $f_y \in \mathscr{Y}$ in the two spaces such that the projections $\{(f_x, x_i)\}_{i=1}^N$ and $\{(f_y, y_i)\}_{i=1}^N$ of the feature vectors of documents from the two languages would be maximally correlated.[4] Then we can find another maximally correlated directions in the two complementary subspaces of the one-dimensional subspaces f_x and f_y in the feature spaces $\mathscr{X}$ and $\mathscr{Y}$, respectively, and so on. If the features consists of content terms (i.e. the stemmed words excluding stop words) from the documents as in the experiments described in Vinokourov et al. (2002) and in this paper (which corresponds to a linear kernel, see the discussions below), then the directions f_x and f_y may represent the terms about the most popular topics in the collection in the two languages, respectively, as these terms are most common in the document pairs $(c, d) \in \mathscr{X} \times \mathscr{Y}$. Therefore, the pair of directions f_x and f_y may represents some of the most distinct themes in the document collection, which could be useful for cross-language applications.

Formally, CCA finds a canonical correlation ρ in the space $\mathscr{X} \times \mathscr{Y}$ which is defined as

$$\rho = \max_{(f_x, f_y) \in \mathscr{X} \times \mathscr{Y}} \operatorname{corr}((f_x, x_i), (f_y, y_i)) = \max_{(f_x, f_y) \in \mathscr{X} \times \mathscr{Y}} \frac{\sum_{i=1}^N (f_x, x_i)(f_y, y_i)}{\sqrt{\sum_i (f_x, x_i)^2 \sum_j (f_y, y_j)^2}} \tag{1}$$

We search for f_x and f_y in the space spanned by the corresponding feature vectors, namely

$$f_x = \sum_l \alpha_l x_l, \quad f_y = \sum_m \beta_m y_m \tag{2}$$

This rewrites the numerator of (1) as

$$\sum_i (f_x, x_i)(f_y, y_i) = \sum_i \sum_{lm} \alpha_l \beta_m (x_l, x_i)(y_m, y_i) = \alpha^{\mathrm{T}} G_x G_y \beta \tag{3}$$

where α is the vector with components α_l $(l = 1, \ldots, N)$ and β the vector with components β_m $(m = 1, \ldots, N)$ and G_x is the Gram matrix of $\{x_i\}_{i=1}^N$ and G_y the Gram matrix of $\{y_j\}_{j=1}^N$. The problem (1) can then be reformulated as

$$\rho = \max_{\alpha, \beta} \frac{\alpha^{\mathrm{T}} G_x G_y \beta}{\sqrt{\alpha^{\mathrm{T}} G_x^2 \alpha \cdot \beta^{\mathrm{T}} G_y^2 \beta}} \tag{4}$$

By introducing regularisation parameters and applying the Lagrangian techniques, the optimisation problem (4) can be reduced into the following generalised eigenvalue problem (for the details about the derivation process, see Li & Shawe-Taylor, 2006 or Hardoon, Szedmark, & Shawe-Taylor, 2004)

[4] In Section 5, in the application for cross-language document classification, together with the SVM, we will consider one kind of non-standard KCCA, in which we minimise the distances between $\{(f_x, x_i)\}$ and $\{(f_y, y_i)\}$ $(i = 1, \ldots, N)$, instead of maximising correlations between them as usual.

$$B\xi = \lambda D\xi \tag{5}$$

where λ is the canonical correlation ρ between the projections (f_x, x_i) and (f_y, y_i) $(i = 1, \ldots, N)$, and

$$B = \begin{pmatrix} 0 & G_x G_y \\ G_y G_x & 0 \end{pmatrix}, \quad D = \begin{pmatrix} G_x^2 + \kappa I & 0 \\ 0 & G_y^2 + \kappa I \end{pmatrix}, \quad \xi = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \tag{6}$$

and κ is regulation parameter.

Therefore, the optimisation problem of the CCA has been transformed into a generalised eigenvalue problem (5), where the eigenvectors with the largest eigenvalues represent the maximally correlated directions in feature space. In other words, the eigenvector $\xi_1 = (\alpha_1^{\mathrm{T}}, \beta_1^{\mathrm{T}})^{\mathrm{T}}$ with the largest eigenvalue λ_1 forms the maximally correlated directions f_x and f_y in the feature spaces $\mathscr{X}$ and $\mathscr{Y}$ by using the Eq. (2). The eigenvector $\xi_2 = (\alpha_2^{\mathrm{T}}, \beta_2^{\mathrm{T}})^{\mathrm{T}}$ with the second largest eigenvalue λ_2 forms the maximally correlated directions in the complementary subspaces of the subspaces f_x and f_y in the feature spaces $\mathscr{X}$ and $\mathscr{Y}$, respectively, and so on.

We can see that, either in the optimisation problem (4) or in the eigenproblem (5), the training points $\{x_i\}_{i=1}^N$ and $\{y_i\}_{i=1}^N$ are involved only through the Gram matrix G_x and G_y. Therefore, the so-called "kernel-trick" can be used to introduce extra flexibility into CCA. Kernelisation of CCA means that the training points $\{x_i\}_{i=1}^N$ and $\{y_i\}_{i=1}^N$ are mapped to another (some high-dimensional) feature space by a kernel function (see e.g. Cristianini & Shawe-Taylor, 2000) and the canonical correlation is then computed in the new feature space. This can be done easily by replacing the Gram matrices with the corresponding kernel matrices in the optimisation formulation (4) and in the eigenproblem (5). A Gaussian kernel was employed in Hardoon et al. (2004) for text-image content based retrieval. The experiments in Vinokourov et al. (2002) showed that the linear kernel was quite good for cross-language applications of KCCA. (As a matter of fact, Joachims (1998) also showed that the linear kernel performed similarly with other types of kernel for the monolingual document categorisation.) Moreover, the linear kernel is simpler and leads to faster learning algorithm than other kernels. Hence, the linear kernel was used in our experiments as well. Using the terms (i.e. stemmed words) as features together with the linear kernel means that the feature space is basically vocabularies. Precisely every dimension of the feature space corresponds to a term (i.e. a stemmed word). Also, we used the same value of regularisation parameter as in Vinokourov et al. (2002), i.e. $\kappa = 1.5$ (also see Vinokourov et al., 2002 for a detailed discussion of the regularisation parameter).

3. Using KCCA for cross-language patent retrieval

3.1. Cross-language patent retrieval with KCCA

In the previous section we have shown that KCCA leads to a generalised eigenvalue problem. The eigenvectors with the largest eigenvalues correspond to the maximally correlating directions in the feature spaces, which constitute some kind of semantic basis vectors. These basis vectors represent a semantic correspondence between the training documents of the two languages, i.e. every vector represents a theme or several mixed themes of training documents in the two languages and a theme is represented by a distribution among the vectors (also see Fig. 1). These basis vectors provide a framework for performing cross-language information retrieval where, given a query in one language, we try to find the relevant documents in another language. Here we adopt the procedure described in Vinokourov et al. (2002) for cross-language information retrieval using KCCA. We first pick a number d of eigenvectors with largest eigenvalues from the solution of (5) for two languages A and B, and compute the corresponding maximally correlated directions in the feature spaces which represent the most distinct themes of the collection in the two language. Then we represent a query in language A as a combination of themes by projecting the query onto the language A part of the basis vectors, and also represent some documents in language B by the same themes by projecting them onto the language B part of the basis vectors. Finally we compare the semantic representations of the query and the documents to select the relevant documents in language B for the query in language A. Formally, to process a query q we represent q as a feature vector $\tilde{q}$ and project it onto the d canonical correlation directions in the feature space

$$\tilde{q}_d = A^{\mathrm{T}} Z^{\mathrm{T}} \tilde{q} \tag{7}$$

where A is an $N \times d$ matrix whose columns are the first or the second half (depending on which language was used in the query) of the eigenvectors of (5) with the largest d eigenvalues, and each column of Z is a training vector in the same language as the query. Similarly, we represent the documents t for retrieval in another language as d-dimensional vectors $\tilde{t}_d$ by projecting them onto the d-dimensional canonical correlation directions. Then the relevance of the document t with query q is measured by the normalised inner product of the two vectors $\tilde{t}_d$ and $\tilde{q}_d$, namely,

$$R(d,t) = \frac{\langle \tilde{q}_d, \tilde{t}_d \rangle}{\|\tilde{q}_d\|_2 \|\tilde{t}_d\|_2} \tag{8}$$

where $\langle \cdot, \cdot \rangle$ represents the inner product and $\| \cdot \|_2$ represents the 2-norm of a vector. Note that in Li and Shawe-Taylor (2006) we measured the relevance of a document for a query by the distances between two KCCA projecting vectors. As the experimental results using inner product (shown below) are clearly better than those using distance (presented in Li & Shawe-Taylor, 2006), the normalised inner product is a more suitable measure for measuring relevance of a document for a query than the distance.

3.2. The dataset used for the experiments

The dataset we used was from the NTCIR-3 patent retrieval test collection.[5] The collection includes about 1.7 million Japanese patent abstracts and their English translations, spanning five years (1995–1999). The corpus has 31 search topics, each of which contains a short description of the topic, some narrative text, the key concept words about the topic, and a news article related to the topic and other metadata. For each topic the corpus also provides some documents (from several hundreds to more than one thousand) which were judged by human assessors and systems in four levels (A, B, C and D) of relevance, where the level A means most relevant of the document to the topic and the level D mean most irrelevant. In our experiment described in this paper, only those documents with the level A were regarded as relevant ones to a particular topic and the documents with level B, C or D were regarded as irrelevant. See Iwayama, Fujii, Kando, and Takano (2003b) for more details about the NTCIR-3 patent corpus.

Only the 336,929 patent abstracts in Japanese as well as in English from the 1995 part of the NTCIT-3 patent corpus (referred to as the 1995 collection hereafter) and the 31 topics and the related judged documents (called judged documents hereafter) in both Japanese and English were used in our experiments. Note that none of the judged documents is in the 1995 collection.

First of all, we collected the terms and computed the *idf* (inverse document frequency) for every term from the 1995 collection and the judged documents. The English terms were collected in the usual way, i.e. downcasing the alphabetic characters, removing the stop words, replacing every non-alphabetic character with a blank space, stemming words using the Porter stemmer, and finally removing the terms which appear less than 3 times in the corpus. We preprocessed the Japanese documents using a Japanese morphological analysis software Chasen[6] version 2.3.3, which was used by many researches such as in Makita et al. (2003), Chen and Gey (2003) and Sahlgren et al. (2003). From the documents processed by the Chasen software, we picked as our terms those words whose part of speech tags were either noun (but not dependent noun, proper noun or number noun), or independent verb, or independent adjective, or unknown. We also removed the Japanese terms appearing less than three times in the documents of the 1995 collection. In this way we obtained 62,506 English terms and 91,451 Japanese terms, respectively.

Then we computed the $tf * idf$ feature vectors for the Japanese patent abstracts and the corresponding English translations in the usual way (see e.g. Joachims, 1998) and finally normalised the feature vectors.

In the following we present our experimental results using KCCA for cross-language information retrieval on the NTCIR-3 corpus. We first discuss the results for so-called mate retrieval and pseudo query retrieval. Our main reason for doing this was because those retrieval scenarios were used to evaluate a state of the art learning method – the cross-language LSI in Littman et al. (1998) and KCCA for cross-language

[5] See http://research.nii.ac.jp/ntcir/permission/perm-en.html.
[6] See http://chasen.aist-nara.ac.jp/.

information retrieval in Vinokourov et al. (2002). In the next section we will discuss the results using the 31 topics to search the judged documents, which is similar to the cross-language patent retrieval task adopted in the NTCIR-3 workshop.

3.3. Mate retrieval

We first conducted experiments for mate retrieval. In mate retrieval a document in one language was treated as a query and only the mate document in another language was considered as relevant. A mate document was considered to be retrieved if it is closest to the query document in the semantic space. We applied KCCA to the first 1000 Japanese documents and the English translations of the 1995 collection. For comparison, we also implemented LSI for cross-language information retrieval (see Littman et al., 1998) under the same experimental settings, since the results of LSI on the NTCIR-3 collection we used were not available from other researchers.

The results presented in the upper part of Table 1 are for 1000 training documents as queries. The lower part of Table 1 shows the results for the 2000 test documents used as queries. These results are consistent with those on the English–French documents (see Vinokourov et al., 2002). That is, KCCA can achieve quite good performance using a fraction of eigenvectors (say 200) while LSI achieved the same results only when using the full 1000 eigenvectors. We also presented the 95% level confidence intervals computed using the bootstrap resampling technique for English to Japanese retrieval for test documents to show the statistical significance of our results in comparison with other results. We can see that the KCCA significantly outperformed LSI on the test documents.

3.4. Pseudo query retrieval

We also carried out experiments for pseudo query retrieval. We generated a short query consisting of the five most probable words for each test document. And the relevant document is the mate of the document in another language. Table 2 shows the relative number of correctly retrieved documents in each experimental setting. Once again, we present the results for the queries from the 1000 training documents and the 2000 test documents, respectively. The retrieval accuracy of KCCA is high and is significantly better than those using LSI when a short query was generated.

The experimental results have shown that KCCA outperformed LSI consistently for cross-language information retrieval. We can also see that similar results were obtained for the English–Japanese bilingual corpus as that reported for English–French documents in Vinokourov et al. (2002), despite English being much more different from Japanese than from French. Therefore, KCCA provides a very encouraging technique for cross-language information retrieval.

Table 1
Mate retrieval (1000 training documents): the accuracy rates averaged over all the training documents and over other 2000 test documents, respectively. The 95% level confidence intervals are also presented for one experiment. Different numbers of eigenvectors were used and KCCA was compared with LSI. E → J means using English query to retrieve Japanese documents and J → E means Japanese document as query to search English documents

# Eigen	5	10	50	100	200	300	400	500	1000
Training docs as queries									
kcca(E → J)	0.427	0.867	0.981	0.988	0.992	0.994	0.993	0.991	0.994
kcca(J → E)	0.424	0.876	0.976	0.981	0.986	0.989	0.987	0.985	0.997
lsi(E → J)	0.093	0.328	0.769	0.898	0.949	0.960	0.965	0.966	0.996
lsi(J → E)	0.091	0.264	0.652	0.827	0.923	0.946	0.952	0.959	0.996
Test docs as queries									
kcca(E → J)	0.046 ± 0.012	0.104 ± 0.026	0.400 ± 0.087	0.515 ± 0.080	0.624 ± 0.062	0.657 ± 0.053	0.676 ± 0.046	0.690 ± 0.045	0.713 ± 0.031
kcca(J → E)	0.041	0.105	0.396	0.530	0.630	0.671	0.694	0.705	0.733
lsi(E → J)	0.037	0.095	0.296	0.376	0.431	0.431	0.417	0.393	0.247
lsi(J → E)	0.029	0.079	0.212	0.294	0.362	0.355	0.329	0.304	0.170

Table 2
Pseudo query retrieval (1000 training documents): the accuracy rates averaged over all the training documents and over 2000 test documents, respectively. The 95% level confidence intervals D are also presented for one experiment. Different numbers of eigenvectors were used and KCCA was compared with LSI

#Eigen	5	10	50	100	200	300	400	500	1000
Training docs as queries									
kcca(E → J)	0.084	0.316	0.747	0.851	0.912	0.930	0.943	0.946	0.964
kcca(J → E)	0.094	0.323	0.727	0.844	0.915	0.932	0.948	0.948	0.976
lsi(E → J)	0.062	0.170	0.415	0.561	0.734	0.785	0.829	0.862	0.911
lsi(J → E)	0.048	0.128	0.244	0.317	0.433	0.495	0.528	0.539	0.548
Test docs as queries									
kcca(E → J)	0.009 ± 0.007	0.046 ± 0.010	0.144 ± 0.025	0.194 ± 0.033	0.212 ± 0.036	0.232 ± 0.049	0.243 ± 0.046	0.247 ± 0.038	0.270 ± 0.025
kcca(J → E)	0.008	0.038	0.150	0.184	0.215	0.237	0.240	0.243	0.265
lsi(E → J)	0.028	0.077	0.152	0.186	0.203	0.212	0.220	0.211	0.172
lsi(J → E)	0.023	0.061	0.114	0.137	0.140	0.140	0.133	0.126	0.093

We can also see from the above results that, while the retrieval accuracy was quite high with training documents as queries, the retrieval accuracy became low when the documents not in the training set were used as queries. This may be due to the small number of training documents. KCCA extracted a semantic correspondence between two languages from the training documents. If the training set is too small to be representative, then the semantic correspondence may not have a good coverage for documents not in the training set.

3.5. More training documents

We expected that KCCA will have better generalisation performance when the training set becomes larger. To verify this, we added another 1000 documents into the training set and then repeated the above experiments with the enlarged training set. In the case of training documents as queries, the results for 2000 training documents were similar to those for 1000 training documents. The results for the 2000 other test documents as queries are presented in Table 3. Comparing with the corresponding results in Tables 1 and 2, we can see from Table 3 that the generalisation performance has indeed improved when using more training documents.

It is possible that the generalisation performance of KCCA will become better if we use yet more training documents. However, we were unable to use a very large training set for KCCA because the computation time becomes very long when using for example 50,000 documents for training. In the following we will discuss several methods to help KCCA deal with a large training set.

3.6. Dealing with large training sets

As shown above, KCCA's performance was improved when we used more training examples. Since KCCA is a kind of unsupervised learning algorithm, we can easily collect a large set of (unlabeled) training data for it.

Table 3
Results of experiments with the 2000 training documents: the accuracy rates averaged over 2000 test documents. Different numbers of the eigenvectors were used

#Eigens	5	10	50	100	200	300	400	500	1000
Mate retrieval									
kcca(E → J)	0.076	0.187	0.556	0.625	0.701	0.743	0.759	0.772	0.781
kcca(J → E)	0.066	0.189	0.567	0.650	0.729	0.759	0.777	0.797	0.811
Pseudo query retrieval									
kcca(E → J)	0.021	0.069	0.183	0.227	0.266	0.286	0.295	0.307	0.334
kcca(J → E)	0.028	0.068	0.193	0.237	0.279	0.299	0.319	0.329	0.354

Hence we can use a large training set for KCCA to achieve better performance for cross-language information retrieval. However, it is difficult to apply KCCA directly to a large training set because of its computational complexity. A naive implementation of KCCA would scale as $O(N^3)$, a computational complexity with cubic growth in the number of data points N. So we have to find more efficient ways for KCCA to handle large training sets.

We have considered two strategies for this. One is to only use the salient examples from the training set. The partial Gram–Schmidt orthogonalisation (PGSO) of the training examples (or equivalently the incomplete Cholesky decomposition of the kernel matrix) is one example using this kind of strategy. The PGSO algorithm projects the data onto a subset spanned by a subset of examples with a pre-defined size. The subset is chosen iteratively by always choosing the example with largest residual norm (see Cristianini, Shawe-Taylor, & Lodhi, 2002 for details). The examples selected by the PGSO determine a subspace used for all the examples and hence the chosen points can be seen as representative examples. By using a subset of (representative) examples instead of a large set containing all the examples, KCCA learning becomes feasible.

Another strategy is to split the training set into small groups. We computed KCCA on each small group and then collected the KCCA basic vectors from all the group to form a set of KCCA vectors for the whole data. In this way the computation time could be reduced by applying the KCCA to many small groups of data rather than one very large set. In Vinokourov et al. (2002) the large training set was randomly split in order to alleviate the problem of large datasets. We have considered clustering the training examples into small groups so that one group consists of documents with similar content.

We have carried out experiments on the NTCIR-3 corpus to compare the following four methods for helping KCCA handle large training data. They were based on the PGSO algorithm, perfect clustering, clustering with 20% noise, and random split, respectively. The experimental results showed that the performance of Gram–Schmidt method was similar to that of perfect clustering and both of them had better results than the random split and the noisy clustering. Since we are currently unable to perform the perfect clustering in most cases, the Gram–Schmidt method appears the most practical way for KCCA to deal with large datasets. See Li and Shawe-Taylor (2006) for more details about the experiments. Note that those experiments used hundreds of documents which were relevant to one of the first 10 queries in the NTCIR-3 corpus in order to do clustering perfectly. In following we present the results using many more documents for evaluating the PGSO algorithm.

As said above, the PGSO algorithm was used to obtain a smaller training set. Then we applied the KCCA to the reduced training set in the usual way. In the experiments we first selected 1000 documents from the first 6000 documents in the 1995 collections by the PGSO method. Then we used the selected 1000 documents to learn the KCCA basic vectors as usual. Finally we used the basic vectors to do cross-language retrieval. Table 4 presents the results of pseudo query retrieval by applying the PGSO KCCA vectors for another 2000 documents. The results are much better than the corresponding results shown in Table 2, which used the first 1000 documents for KCCA learning. They are also comparable to the results using the first 2000 documents for KCCA learning shown in Table 3. In the next section we will show the results using the PGSO selected documents for query searching.

In regard of the computation time, on a Linux machine with 1 G CPU, it took about 15 min to learn the KCCA with 1000 training documents and more than 1 h for 2000 documents as training data. On the other hand, it took less than 1 min to select 1000 documents from 6000 documents using PGSO algorithm. So, using the PGSO to select 1000 documents and then learning KCCA took about fourth time of learning KCCA from 2000 documents. And the results of the two experiments were comparable.

Table 4
Pseudo query retrieval results using 1000 documents selected by the PGSO algorithm for KCCA learning: the accuracy rates averaged over other 2000 test documents. Different numbers of the eigenvectors were used

#Eigens	5	10	50	100	200	300	400	500	1000
kcca(E → J)	0.009	0.063	0.206	0.294	0.311	0.327	0.343	0.339	0.341
kcca(J → E)	0.006	0.037	0.193	0.277	0.315	0.321	0.312	0.316	0.328

Table 5
Results for overlapped bigram index of Japanese text: the accuracy rates for pseudo query retrieval using the first 1000 documents for KCCA learning and other 2000 test documents for testing

#Eigens	5	10	50	100	200	300	400	500	1000
kcca(E → J)	0.007	0.020	0.073	0.099	0.129	0.139	0.140	0.140	0.141
kcca(J → E)	0.024	0.047	0.122	0.156	0.186	0.201	0.207	0.211	0.211

3.7. Two other problems

Finally, we discuss two other problems concerning the application of KCCA to cross-language information retrieval. One problem is concerned with Japanese document preprocessing. We represented a document using all terms of the document. For English it is natural to use stemmed words as terms. However, in Japanese, there is no delimiter between words in a sentence so we have to decide what was regarded as terms in Japanese text. We have employed the Chasen software to segment Japanese sentences into a sequence of words and then to select Japanese terms according to the POS tags. Alternatively, we could have used ngrams of Japanese characters as terms as well. We did experiments by representing Japanese document using the overlapped bigrams of Japanese characters as terms instead of words segmented by the Chasen algorithm. Table 5 presents the results using the overlapped bigram for pseudo query retrieval. The results using overlapped bigram of Japanese characters were lower than the corresponding results using words shown in Table 2, which was consistent with the experimental results presented in Chen and Gey (2003).

Another problem is how to choose the optimal number of KCCA eigenvectors. First, we can see from the above tables that the performance is not very sensitive to the number of KCCA eigenvectors. For example, in most cases, a number of eigenvectors (say 200) close to the optimal number gave similar results. In the application we may use some empirical methods for choosing a good number of eigenvectors. Determining the optimal values of parameters in a learning algorithm is an important research problem in machine learning. Several empirical methods such as *n*-fold cross-validation have been studied and work well in some applications (see e.g. Lewis, Yang, Rose, & Li, 2004). On the other hand, Zha and Simon (1998) proposes a statistical test for choosing the optimal number of dimensions for LSI. We suggest that it is possible to use a similar statistical test method to determine the optimal number of KCCA eigenvectors but this needs more investigation.

4. Query retrieval

Since the NTCIR-3 patent corpus provides 31 topics and the judged documents for each topic in both Japanese and English, we can carry out cross-language document retrieval by creating a query from one topic and using it to search relevant documents in another language. Hence, we have applied KCCA to the bilingual topics and judged documents for cross-language query based document retrieval, which is a more realistic scenario for cross-language information retrieval than the two types of retrieval discussed in Section 3.

Note that for one particular topic our experiments used only those documents which were judged for that particular topic in the NTCIR-3 corpus, while in the NTCIR-3 evaluation scheme all the documents in the two files of 1998 and 1999 years were used where the un-judged documents for one topic were regarded as irrelevant to that topic. As our experiments used much less documents for retrieving than the NTCIR-3 evaluation scheme, our experiment needed much less computing time than the experiment following the NTCIR-3 evaluation scheme, which enabled us to run many experiments for comparing different experimental settings. On the other hand, due to different settings, our results cannot be compared directly to the results of the NTCIR-3 evaluation participating systems.

Another difference between our experiment and the NTCIR-3 evaluation scheme was that our experiments only searched the patent abstracts while evaluation scheme required searching over the full text of the patent documents. It is worth noting that the experiments presented in Iwayama, Fujii, Kando, and Marukawa (2003a) and Chen and Gey (2003) showed that searching over full text resulted in better performance than searching just abstracts.

Regarding the direct applications, our experiments could be useful for data fusion (or called meta-searching) in which a system re-ranks the documents retrieved by several searching engines where each searched document has the judgment from one searching engine. In order to be compatible with the data fusion scenario, in our experiments we excluded those documents which were judged only by human annotators (namely those documents marked by *J* in the judgment file *frel.a* of the NTCIR-3 corpus), which is about 15% of all the judged documents in average for the 31 topics.

In our experiments we first computed the KCCA basic vectors using the first 1000 Japanese and English patent abstracts from the 1995 collection. Then we formed a query using the text in the description field and the narrative field of one topic in one language and used the query to search the judged documents in another language for the particular topic.

According to Iwayama et al. (2003b), the overview paper of the NTCIR-3 workshop, only two groups submitted runs to the NTCIR-3 workshop for the cross-language patent retrieval task. Chen and Gey (2003) computed relevance of document to query using a log-odds of the probability which involved term frequencies in a document as well as in the collection and length of document and query. For cross-language document retrieval, they translated words of English query into Japanese by using an English–Japanese dictionary created automatically from the aligned bilingual patent abstracts of NTCIR-3 corpus. Sahlgren, Hansen, and Karlgren (2003) also automatically created a bilingual thesaurus from the aligned bilingual corpus of NTCIR-3 patent abstracts and used it for word based query translation.

The key difference between our KCCA method and the methods used in previous works is in the way the bilingual corpus is exploited. While both Chen and Gey (2003) and Sahlgren et al. (2003) created an English–Japanese dictionary automatically from the bilingual training corpus and used the dictionary to translate English query into Japanese and then did monolingual retrieval, our KCCA method extracted from the bilingual corpus the semantic correspondences represented by two feature vectors in the two languages, e.g. one or several themes represented respectively by the words in two languages (see Fig. 1).

For applying the KCCA method to query searching, we first computed the KCCA basis vectors using the first 1000 documents in the 1995 collection. Then we projected the feature vectors of query and documents into the basis vectors of the corresponding language and computed the normalised inner product of the projecting vectors as the relevance measure of a document to query. We used the same performance measures as those used in the patent retrieval tasks of NTCIR-3 workshop. In detail, for each query, we ranked the judged documents according to the relevance to the query and computed averaged precision to measure the query results. We also use the mean of averaged precisions (MAP) over the 31 queries for measuring overall performance. However, as explained above, we used a much smaller test set than that used in the evaluation scheme.

Table 6 presents overall performance of our KCCA method with different numbers of basis vectors. The corresponding best overall results from Chen and Gey (2003) was 0.123 and the results in Sahlgren et al. (2003) was much worse than that. Our overall results is better than the previous results. However, as we said above, our experiment settings were different from those participating systems in several aspects. Hence the results presented here cannot be compared with those of the participating systems.

For comparison, we also carried out experiments for monolingual document retrieval, in which one query searched the judged documents in the same language. The method used in our monolingual retrieval experiment was simple – the normalised inner product between the query and document feature vectors was computed as a relevance measure. Table 7 presents the averaged precisions for the first 8 topics and the MAP over the 31 topics of the KCCA cross-language retrieval and the monolingual retrieval. The table also includes the results for random retrieval in which the judged documents for one query were ranked randomly. The random result for each query is the mean of 50 random rankings and the 95% confidence interval was calculated via the *t*-test technique. The retrieval performances were quite different among the queries, reflecting different

Table 6
Results of query searching using KCCA with different numbers of basis vectors: mean averaged precision (MAP)

#Eigens	5	10	50	100	200	300	400	500	1000
kcca(E → J)	0.130	0.152	0.157	0.167	0.178	0.173	0.172	0.177	0.178
kcca(J → E)	0.116	0.142	0.159	0.162	0.176	0.175	0.176	0.179	0.179

Table 7
Results for monolingual retrieval, KCCA cross-language retrieval and random retrieval: averaged precision for the first 8 queries and the MAP over all the 31 queries

Query	1	2	3	4	5	6	7	8	MAP
$E \rightarrow J$									
KCCA	0.043	0.076	0.313	0.039	0.134	0.098	0.293	0.099	0.178
Mono	0.033	0.038	0.253	0.014	0.383	0.092	0.266	0.144	0.179
$J \rightarrow E$									
KCCA	0.041	0.072	0.237	0.030	0.098	0.128	0.355	0.091	0.179
Mono	0.027	0.056	0.363	0.010	0.102	0.154	0.320	0.071	0.179
Random	0.020 ± 0.002	0.022 ± 0.002	0.058 ± 0.004	0.007 ± 0.004	0.031 ± 0.004	0.018 ± 0.004	0.225 ± 0.006	0.057 ± 0.005	0.093 ± 0.005

Table 8
Results of query searching using 2000 documents and the 1000 documents selected by PGSO algorithm: mean averaged precision (MAP)

#Eigens	5	10	50	100	200	300	400	500	1000
The first 2000 documents for kcca learning									
kcca(E → J)	0.123	0.139	0.156	0.167	0.177	0.175	0.170	0.179	0.182
kcca(J → E)	0.098	0.137	0.152	0.161	0.173	0.173	0.178	0.180	0.185
PGSO selected 1000 documents for KCCA learning									
kcca(E → J)	0.119	0.145	0.157	0.166	0.175	0.176	0.171	0.171	0.172
kcca(J → E)	0.084	0.095	0.148	0.156	0.176	0.175	0.175	0.177	0.179

difficulties of retrieval using the feature vector representation. KCCA cross-language retrieval had similar overall performance as monolingual retrieval. The two methods behaved consistently among individual topics. Both of them significantly outperformed random retrieval.

We have also run experiments respectively using the first 2000 documents of the 1995 collection and the 1000 documents selected by the PGSO algorithm from the first 6000 documents for KCCA learning. Table 8 presents the results. Comparing with the corresponding results shown in Table 6, using 2000 documents for learning we obtained a small improvement but not as much as that for mate retrieval or pseudo query retrieval discussed in Section 3. We obtained worse results using the 1000 document selected by the PGSO algorithm than the first 1000 documents of 1995 collection. Certainly it needs further investigation.

5. Cross-language document classification

Cross-language document classification is about using a classifier learned from one language to classify documents in other languages. It is useful in the context of multi-lingual information management because by doing so we need not learn different classifiers for multi-lingual document classification (instead we just learn a single classifier and then use it to classify documents in all languages).

As KCCA extracts the semantic correspondence between two languages, we investigate a learning algorithm based on KCCA for cross-language document classification. We also study other methods for exploiting the semantic correspondence. All the learning algorithms we studied are based on the SVM, which achieved state of the art results for monolingual text classification (Joachims, 1998). Some of the algorithms can learn a classifier from one language and then use the classifier to classify documents in other languages. Other algorithms learn classifier from bilingual training documents and apply the classifier to bilingual or monolingual test document. In the follows we first describe the learning algorithms for cross-language document classification and then test them on the NTCIR-3 patent corpus.

5.1. pSVM and kcca_SVM

As the SVM gives state of the art results for document classification, we used the SVM as cross-language document classifier in our experiments. Fortunately, the SVM learned in one language can be easily used in

another language if we are given pairs of the training documents in two languages – we can first train an SVM using documents in one language and then transform it into a new SVM classifier for another language by substituting the training feature vectors in the dual form of the SVM by the mates in another language, since the SVM in dual form is a weighted sum of the training vectors in the feature space. On the other hand, the semantic correspondence inferred by KCCA between the two languages could also be used as a basis to form the correspondence of feature vectors representing the documents in two languages.

We therefore proposed two methods to use the SVM for cross-language classification. The first one was to just exploit pairs of training documents in two languages, $\{(x_i, y_i) : i = 1, \ldots, N\}$. If an SVM was trained from the training documents $\{x_i : i = 1, \ldots, N\}$ in one language, which can be represented in dual form as

$$h_x(\cdot) = \operatorname{sgn}\left(\sum_{i=1}^{N} \alpha_i K(\cdot, x_i)\right) \tag{9}$$

then we can transform it into an SVM classifier in another language as

$$h_y(\cdot) = \operatorname{sgn}\left(\sum_{i=1}^{N} \alpha_i K(\cdot, y_i)\right) \tag{10}$$

We call the new SVM classifier (10) as *pSVM* since it just employs the semantic correspondence derived directly from the pairings of the training documents in two languages.

Note that this approach can only be applied if the training set is a paired dataset, though one could envisage using the approach by first training an SVM in one language and then only translating the so-called support documents for which the dual variable $\alpha_i > 0$. Typically this only holds for a small subset of the full training set.

Another method exploited the semantic correspondence derived by KCCA. Given a training set containing pairs of documents in both languages, projecting the training documents onto the basic vectors of KCCA resulted in pairs of semantic feature vectors, exactly as we obtained in Section 3 for cross-language information retrieval. These pairs of semantic vectors define a semantic space that documents in either language can be projected to. If we train an SVM in this semantic space using documents from one language, we can use it to classify documents in the other language by projecting them into the semantic space. We call this kind of classifier as *kcca_SVM*. Note that crucially the training set for KCCA may be different from that for the SVM. This implies that a large (unlabeled) training set can be used in KCCA to deduce a good semantic correspondence between the two languages and another labeled document set would be used to train the SVM. However, in the experiments described below, the same training set was used for both KCCA and SVM.

Unfortunately, as shown by the experiments described below, the results for kcca_SVM were not as good as that of pSVM, and both of them performed significantly worse than the monolingual SVM classifier in many cases. This motivated us to study another way of combining SVM with KCCA for two-view classification, *SVM_2k*.

5.2. SVM_2k

Like kcca_SVM, the *SVM_2k* also combines the SVM and KCCA for two-view classification, in order to explore semantic correspondences between the two views of an object. On the other hand, the SVM_2k adopted a distance minimisation version of KCCA, rather than the correlation maximization version in the standard KCCA as shown in Section 2. In detail, the SVM_2k seeks two directions respectively in the two feature spaces (corresponding to the two views of object) such that the distance of the projections of the two feature vectors of one object on the two directions is minimised. Moreover, the hyperplane determining each of the two directions in the corresponding feature space is required to be an SVM classifier for the classification problem of a single view. In other words, the SVM_2k learns two SVM classifiers from the two views of an object. Meanwhile additional constraints were used to make the outputs of the two SVMs on one object (or equivalently, the projections of two feature vectors of one object on the weight vectors of two SVMs) as close as possible by minimising the disagreement rate of the underlying linear functions (i.e. before they are thresholded to created a classification).

Formally, given two views x_i and y_i of one object O_i $(i = 1, \ldots, N)$, the SVM_2k learns two SVM classifiers (W_x, b_x) and (W_y, b_y) through the standard SVM formulation from the training data $\{(x_i, y_i, z_i) : i = 1, \ldots, N\}$, where $z_i = +1$ or -1 is the label of the object O_i for the classification problem considered, and meanwhile minimise the disagreement rate of the two underlying linear functions on the training object O_i $(i = 1, \ldots, N)$. The learning is achieved by solving the following optimisation problem

$$\begin{aligned}
\min_{W_x, b_x, W_y, b_y, \xi_x, \xi_y, \eta} \quad & (\|W_x\|^2 + \|W_y\|^2) + C_x \sum_{i=1}^{N} \xi_{xi} + C_y \sum_{i=1}^{N} \xi_{yi} + D \sum_{i=1}^{N} \eta_i \\
\text{subject to} \quad & |\langle W_x, x_i \rangle + b_x - \langle W_y, y_i \rangle - b_y| \leqslant \eta_i + \epsilon \\
& z_i(\langle W_x, x_i \rangle + b_x) \geqslant 1 - \xi_{xi} \\
& z_i(\langle W_y, y_i \rangle + b_y) \geqslant 1 - \xi_{yi} \\
& \xi_{xi} \geqslant 0; \quad \xi_{yi} \geqslant 0; \quad \eta_i \geqslant 0 \\
& i = 1, \ldots, m
\end{aligned}$$

where the parameter ϵ represents the required closeness of the outputs of the two underlying linear functions for each object and the three regulation parameters C_x, C_y and D are used to achieve the balance between discrimination and tolerance of noise and outliers on training data for the two classifiers. The values of these parameters used in our experiments presented below were as $\epsilon = 0.1$, $C_x = 6$, $C_y = 6$ and $D = 10$. In comparison with the standard SVM problem, SVM_2k learns two SVM classifiers and one constraint is added for each object O_i to force the outputs of two linear functions to be close on the object.

After learning, we obtain two SVM classifiers, each of which is for one view of object. Given one test object with two views (x_0, y_0), the SVM_2k classifies it by using the following Sign function

$$\text{Sign}((\langle W_x, x_0 \rangle + b_x + \langle W_y, y_0 \rangle + b_y)/2) \tag{11}$$

It was shown in Meng et al. (2004) that the results of the two-view learning were better than those of learning from one view or the simple concatenation of two views for image recognition.

In the application of SVM_2k to cross-language document classification, we need a collection of documents c_i $(i = 1, \ldots, N)$ and their translation d_i in another language for learning. We transform documents c_i and d_i respectively into two feature vectors x_i and y_i in the $tf * idf$ representation commonly used for document classification (see e.g. Lewis et al., 2004). Then SVM_2k learns two SVM classifiers from the training data. Finally, given a test document c and its translation d, we transform them into two feature vectors and then used the SVM_2k function (11) to classify it.

Note that the classification function (11) of SVM_2k would appear to need the translation of a test document, which may not be convenient or practical in some applications. On the other hand, we could obtain a classification from a monolingual test document by simply applying the corresponding SVM classifier. For example, if we want to classify document in one language with feature vector y_0, then we can just use the SVM classifier (W_y, b_y) to form the following classification function

$$\text{Sign}(\langle W_y, y_0 \rangle + b_y) \tag{12}$$

We will compare the above monolingual classification with the bilingual classification (11) in the experiments presented below.

5.3. Experiments

We tested the above learning algorithms for the Japanese–English cross-language patent classification on the NTCIR-3 collection.[7] The collection includes 31 topics. For each topic some pairs of documents in

[7] It is desirable to use the same corpus as the previous works so that we can compare our results with others. However, unfortunately, in the previous works about cross-language document classification, different authors used different corpora (see the brief overview in Section 1), and the corpora they used were either not publicly available or were difficult to obtain.

Table 9
Results of cross-language patent classification for the six topics of NTCIR-3 corpus: the *mean* (%) the averaged precisions over 10 runs of the SVM classifiers on Japanese test set. We also present the 95% level confidence intervals computed base on *t*-test technique Results are presented for the six SVM based classifiers as, from left to right, the pSVM and kcca_SVM respectively learned from English training set and tested in Japanese test documents, the monolingual SVM both learned and tested in Japanese documents, the Japanese classifier SVM_2k_j derived from SVM_2k, the SVM classifier learned from the concatenation of the English and Japanese features, and the bilingual SVM_2k classifier

	pSVM	kcca_SVM	SVM	SVM_2k_j	Concat	SVM_2k
01	59.4 ± 8.7	60.3 ± 6.1	66.6 ± 6.1	66.1 ± 5.7	67.5 ± 5.2	67.5 ± 4.7
02	71.1 ± 10.1	68.4 ± 9.8	73.0 ± 9.0	74.8 ± 10.6	73.9 ± 8.9	75.1 ± 9.1
03	16.7 ± 2.7	13.1 ± 2.3	18.8 ± 3.5	20.8 ± 4.3	21.5 ± 4.2	22.5 ± 4.2
07	74.9 ± 3.9	76.0 ± 2.5	76.7 ± 3.0	77.5 ± 3.0	79.0 ± 2.7	80.7 ± 3.2
12	75.0 ± 1.8	73.6 ± 1.8	76.8 ± 2.2	77.6 ± 1.5	76.8 ± 1.3	78.4 ± 1.3
14	76.0 ± 3.7	71.5 ± 3.5	80.9 ± 3.0	82.2 ± 2.9	81.4 ± 3.0	82.7 ± 2.9

Japanese and English were annotated as relevant or irrelevant. The annotated documents for one topic form a dataset for cross-language document classification.

In the experiments we randomly split the dataset into two equal parts, one for training and another for test. We used the English part of the training documents to train an SVM classifier and then induced the pSVM or kcca_SVM classifier for the Japanese documents. We also apply the SVM_2k to learn two SVM classifiers from the English and Japanese training documents and tested them on the test set according to the bilingual classification function (11) and the monolingual classification function (12), respectively. For comparison, we also trained two other related SVM classifiers. One was a monolingual SVM classifier learnt from the Japanese training set and tested on the Japanese test set as well. Another one used a new feature vector obtained by concatenating the two feature vectors respectively from one Japanese document and the corresponding English document.

In the experiments we used averaged precision[8] to evaluate the performances of all the SVM classifiers on the Japanese test set (see e.g. Li, Zaragoza, Herbrich, Shawe-Taylor, & Kandola, 2002 for a detailed explanation of the averaged precision). We ran the experiments 10 times for one topic and then the statistical measures *mean* and *std* were computed for the averaged precisions from the 10 runs.

Table 9 shows the results for six topics, Topic 01, 02, 03, 07, 12 and 14 of the NTCIR-3 patent test collection. These topics were selected such that they were variable with respect of the ratio of relevant and irrelevant documents (the ratios for the six topics varies from 0.019 to 0.84). Hence we can compare the performances of these algorithms on different types of classification problems. For the kcca_SVM we present the results using all the eigenvectors of KCCA.

First, not surprisingly, the two induced classifiers pSVM and kcca_SVM had worse results than the monolingual classifier SVM, since both pSVM and kcca_SVM were trained from English training documents and then induced classifiers for Japanese documents while the SVM was trained directly on Japanese training documents and was tested on the Japanese test documents. On the other hand, the pSVM performed better than kcca_SVM on 4 of the 6 topics.

Secondly, the two classifiers using bilingual documents for training and test, the SVM_2k and the SVM based on concatenation of two languages, performed better than the other four which only used Japanese documents for test. On the other hand, the bilingual classifier need more effort for preparing the document (e.g. every test document should be translated) than the monolingual ones.

Thirdly, the SVM_2k classifier had better results than the SVM classifier using concatenation of English and Japanese feature vectors on 5 out of 6 topics and had similar results on the other one, showing that

[8] We did not use the F_1, a commonly used measure in information retrieval research, to measure the performance. F_1 is dependent on the bias b of the SVM solution but the average precision is not. It is known that the SVM would learn a poor bias if the number of positive training patterns is very small and the bias can be improved by some algorithms (see Li & Shawe-Taylor, 2003; Lewis et al., 2004). But our purpose here is to compare different algorithms rather than achieving high value of F_1. Therefore, we think that the averaged precision is a better measure than F_1 for the experiments.

the SVM_2k provides a better mechanism to learn from two views of an object than the simple concatenation of the two views. Moreover, the monolingual classifier SVM_2k_j derived from SVM_2k performed better than the other monolingual classifier SVM on 5 of the 6 topics. It even performed better than the classifier using the concatenation of two feature vectors on 3 of the 6 topics, showing again the advantage of the SVM_2k for two-view learning. On the other hand, the results of SVM_2k_j were lower than those of the bilingual SVM_2k classifier on all 6 topics, but again the latter classifier requires translated test documents while the former does not need.

Finally, note that the results varied among the six topics but were consistent among the methods. The result for a topic were dependent upon the topic itself (whether it is hard for classification) and was not simply determined by the number of relevant examples. Moreover, if we had used F_1 as the measure instead of the averaged precision, the differences of the results among the topics would have become even bigger (see Footnote 8).

6. Conclusions

We described a method for fully automated cross-language information retrieval in which no query translation was required. The method was based on KCCA, a method of finding the maximally correlated projections of documents in two languages. We used KCCA for cross-language Japanese–English patent retrieval. The experimental results were quite encouraging and were better than those obtained by other state of the art methods such as CL-LSI.

We investigated several methods to help KCCA handle large training data and showed that the PGSO method was a practical method. However, although we obtained quite encouraging results using the PGSO in some experiments, the PGSO application to the query searching was not good and needed further investigation.

We also investigated several methods for cross-language document classification. They were based on the SVM and/or KCCA but may require different kinds of resources for training and application. Both pSVM and kcca_SVM project the SVM classifier learned in one language onto another language directly through the pairs of training documents in two languages. It does not need any translation in the testing phase. We also investigated another way of combining the SVM and KCCA for two-view classification, namely the learning algorithm SVM_2k. The classifier based on the SVM_2k algorithm does need the bilingual version of both training and test documents. It performed much better than the SVM classifier learned from monolingual training documents. It also performed better than the SVM classifier which simply learned from a concatenation of the feature vectors from the two languages. Interestingly, we obtain a monolingual classifier from the SVM_2k learning. It does not need any translation of test documents and performed better than the induced classifiers pSVM and kcca_SVM and the monolingual classifier learned and tested in the same language.

The learning algorithms for cross-language document classification we studied need translations of some training documents or translations of both training and test documents. In our experiments we used manual translation of Japanese patent documents, available in the NTCIR-3 corpus. It is interesting to see what the learning algorithms in particular the SVM_2k could achieve by using machine translation systems to obtain bilingual documents from monolingual documents.

Acknowledgements

We would like to thank Alexei Vinokourov and Nello Cristianini for discussions and technical assistance in implementing KCCA. Thank Sandor Szedmak for providing us the Matlab code solving SVM_2k. Thank Mitsuharu Makita for help in preprocessing Japanese document. Thank National Institute of Informatics (NII) for providing us the NTCIR-3 patent retrieval test collection. We thank anonymous reviewers for detailed comments and valuable suggestions. The work described in this paper has been supported by the European Commission through the IST Programme under Contract IST-2000-25431 (KerMIT). This work is also supported by the EU-funded SEKT project (http://www.sekt-project.org).

References

Bel, N., Koster, C. H. A., & Villegas, M. (2003). Cross-lingual text categorization. In: *Proceedings ECDL 200*, pp. 126–139.

Chen, A., & Gey, F. C. (2003). Experiments on cross-language and patent retrieval at NTCIR-3 workshop. In: *Proceedings of the third NTCIR workshop on research in information retrieval, automatic text summarization and question answering*.

Cristianini, N., & Shawe-Taylor, J. (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press.

Cristianini, N., Shawe-Taylor, J., & Lodhi, H. (2002). Latent semantic kernels. *Journal of Intelligent Information System, 18*(2/3), 127–152.

Gliozzo, A. M., & Strapparava, C. (2005). Cross language text categorization by acquiring multilingual domain models from comparable corpora. In: *Proceedings of the ACL workshop on building and using parallel texts*, pp. 9–16.

Hardoon, D. R., Szedmark, S., & Shawe-Taylor, J. (2004). Canonical correlation analysis: an overview with application to learning methods. *Neural Computation, 16*, 2639–2664.

Hotelling, H. (1936). Relations between two sets of variates. *Biometrika, 28*, 312–377.

Iwayama, M., Fujii, A., Kando, N., & Marukawa, Y. (2003a). An empirical study on retrieval models for different document genres: patents and newspaper articles. In: *Proceedings of the 26th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR2003)*, pp. 251–258.

Iwayama, M., Fujii, A., Kando, N., & Takano, A. (2003b). Overview of patent retrieval task at NTCIR-3. In: *Proceedings of the third NTCIR workshop on research in information retrieval, automatic text summarization and question answering*.

Joachims, T. (1998). Text categorization with support vector machines: learning with many relevant features. In C. Nédellec & C. Rouveirol (Eds.), *Proceedings of ECML-98, 10th European conference on machine learning*. *Lecture Notes in Computer Science* (vol. 1398, pp. 137–142). Heidelberg, DE, Chemnitz, DE: Springer Verlag.

Lewis, D., Yang, Y., Rose, T., & Li, F. (2004). Rcv1: a new benchmark collection for text categorization research. *Journal of Machine Learning Research, 5*(Apr), 361–397.

Li, Y., & Shawe-Taylor, J. (2003). The SVM with uneven margins and Chinese document categorization. In: *Proceedings of The 17th Pacific Asia conference on language, information and computation (PACLIC17), Singapore*, pp. 216–227.

Li, Y., & Shawe-Taylor, J. (2006). Using KCCA for Japanese–English cross-language information retrieval and document classification. *Journal of Intelligent Information System, 27*(2), 117–133.

Li, Y., Zaragoza, H., Herbrich, R., Shawe-Taylor, J., & Kandola, J. (2002). The perception algorithm with uneven margins. In: *Proceedings of the 9th international conference on machine learning (ICML-2002)*, pp. 379–386.

Littman, M. L., Dumais, S. T., & Landauer, T. K. (1998). Automatic cross-language information retrieval using latent semantic indexing. In G. Grefenstette (Ed.), *Cross language information retrieval*. Kluwer.

Makita, M., Higuchi, S., Fujii, A., & Ishikawa, T. (2003). A system for Japanese/English/Korean multilingual patent retrieval. In: *Proceedings of machine translation summit IX*. Online at http://www.amtaweb.org/summit/MTSummit/papers.html.

Meng, H., Shawe-Taylor, J., Szedmak, S., & Farquhar, J. (2004). Support vector machine to synthesise kernels. Machine Learning Workshop, Sheffield.

Olsson, J. S., Oard, D. W., & Hajič, J. (2005). Cross-language text classification. In: *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 645–646.

Rigutini, L., Maggini, M., & Liu, B. (2005). An EM based training algorithm for cross-language text categorization. In: *Proceedings of IEEE/WIC/ACM international conference on web intelligence (WI-05)*.

Sahlgren, M., Hansen, P., & Karlgren, J. (2003). English–Japanese cross-lingual query expansion using random indexing of aligned bilingual text data. In: *Proceedings of the third NTCIR workshop on research in information retrieval, automatic text summarization and question answering*.

Vinokourov, A., Shawe-Taylor, J., & Cristianini, N. (2002). Inferring a semantic representation of text via cross-language correlation analysis. In: *Advances of neural information processing systems, Vol. 15*.

Zha, H., & Simon, H. (1998). A subspace-based model for latent semantic indexing in information retrieval. In: *Proceedings of the thirteenth symposium on the interface*, pp. 315–320.