Distributional Correspondence Indexing for Cross-Lingual and Cross-Domain Sentiment Classification (Extended Abstract)*

Alejandro Moreo Fernández, Andrea Esuli and Fabrizio Sebastiani
Istituto di Scienza e Tecnologie dell’Informazione, Consiglio Nazionale delle Ricerche
{alejandro.moreo,andrea.esuli,fabrizio.sebastiani}@isti.cnr.it

Abstract
Domain Adaptation (DA) techniques aim at enabling machine learning methods learn effective classifiers for a “target” domain when the only available training data belongs to a different “source” domain. In this extended abstract we briefly describe a new DA method called Distributional Correspondence Indexing (DCI) for sentiment classification. DCI derives term representations in a vector space common to both domains where each dimension reflects its distributional correspondence to a pivot, i.e., to a highly predictive term that behaves similarly across domains. The experiments we have conducted show that DCI obtains better performance than current state-of-the-art techniques for cross-lingual and cross-domain sentiment classification.

1 Introduction
Automated text classification methods usually rely on a training set of labelled examples in order to learn a classifier that will predict the classes of unlabelled documents. Deploying a model for a new target domain in the absence of high-quality annotated examples thus entails a substantial human labelling effort. Transfer learning (TL) [Pan and Yang, 2010] focuses on alleviating this problem by leveraging training examples from a different, although related, source domain for which the amount of available labelled examples is higher. TL thus operates in applicative scenarios in which the so-called “iid assumption” (the training and the test data are randomly drawn from the same distribution) no longer holds.

One such scenario of the utmost importance is sentiment classification [Liu, 2012], the task of classifying opinion-laden documents as conveying a positive or a negative sentiment towards a given entity (e.g., a product, a policy, a political candidate). In many contexts the amount of available, pre-labelled opinions is scarce, or even null, in the case of new entities (e.g., products, policies, or political candidates). In such cases, promptly generating a sentiment classifier might become difficult, due to the considerable cost and time involved in producing a representative set of training documents.

In sentiment classification, TL finds a natural application in domain adaptation (DA), i.e., the task of adapting a sentiment classifier to operate on a new domain. For example, we might want to use a training set of book reviews written in English to classify movie reviews written in English, or to classify book reviews written in German. The former case is typically known as cross-domain adaptation, while the second one is instead known as cross-lingual adaptation [Pan et al., 2012].

Central to text classification is the representation of words and documents as numerical vectors that reflect, through their relative distances, the semantic relations among them, i.e., the smaller the distance between a pair of word vectors, the more semantically similar the words are assumed to be. While distributional semantic models like Latent Semantic Analysis (LSA) [Deerwester et al., 1990] and Latent Dirichlet Allocation (LDA) [Blei et al., 2003] have proven useful to represent similar words (e.g., beautiful and nice) close to each other (e.g., in terms of cosine similarity) in the vector space, one main difficulty in domain adaptation is to discover semantic correspondences that may exist across domains (e.g., writer and director when adapting from a source domain of books to a target domain of films).

Structural Correspondence Learning (SCL) was first adapted for the cross-domain case [Blitzer et al., 2007] and then extended to the cross-lingual case in [Prettenhofer and Stein, 2011]. SCL discovers correspondences among terms from different domains that show similar distributions with respect to a set of pivot terms (highly predictive terms expected to behave in a similar way in both domains). Each pivot defines in SCL an auxiliary classification problem that brings to bear “structural” information of the adaptation problem. The intuition according to which the semantics of words is somehow determined by its distribution in text with respect to other terms is generally referred to as the distributional hypothesis [Harris, 1954]. The Distributional Correspondence Indexing (DCI) [Moreo Fernández et al., 2016] method we propose also builds upon the distributional hypothesis, but it formalizes the notion of correspondence through the Distributional Correspondence Functions (DCF), which are much lighter, in terms of computational cost, than the auxiliary problems of SCL.

We experimentally show that DCI compares favourably to the state of the art in two popular sentiment datasets covering both cross-domain and cross-lingual adaptation, and at a

*This paper is an extended abstract of an article appeared as [Moreo Fernández et al., 2016].
substantially smaller computational cost.

2 Distributional Correspondence Functions

DCFs are a family of real-valued functions that quantify the degree of correspondence between two features (terms, in our case) \( f^1 \) and \( f^2 \) by comparing their \emph{context distribution vectors} \( F^1 \) and \( F^2 \) from any (unlabelled) collection \( U \). A context distribution vector is a \( n \)-dimensional vector that models how a feature relates to a set of contexts (documents, in our case); \( f^1_u \) denotes the value of the vector for term \( f^1 \) in context \( c \), with \( f^1_u = 0 \) if \( f^1 \) does not appear in context \( c \). The cases in which \( f^1_u > 0 \) are determined by the weighting function in use (e.g., \( tfidf \)), and might lead to different interpretations of the DCF, e.g., as a probability function in an event space (Section 2.1), or as a kernel in a vector space (Section 2.2).

A DCF is thus a function \( \eta : \mathcal{R}^n \times \mathcal{R}^n \rightarrow \mathcal{R} \), where the sign of \( \eta(f^1, F^2) \) indicates the polarity of the correspondence, i.e., positive values indicate positive correlation and negative values indicate negative correlation. We force DCFs to be centered at 0 for \emph{null correspondence}, i.e., \( \eta(f^1, F^2) = 0 \) means the correspondence between \( f^1 \) and \( F^2 \) is not different than the correspondence between any pair of context vectors which randomly distribute the same number of events as \( f^1 \) and \( F^2 \).

2.1 Probability-Based DCF

Probability-based DCFs derive from information theory, and build upon the joint probability distributions of two features in the binomial event space.

The first part of Table 1 shows the probability-based DCFs we investigate. We consider \emph{Pointwise Mutual Information} (PMI – the ratio between the joint distribution and the product of the marginal distributions), and a simple probabilistic function (here called \emph{Linear}) that contrasts the probabilities of \( f^2 \) conditioned on \( f^1 \) and \( F^2 \), respectively. We also consider \emph{Mutual Information} (MI – the reduction in entropy of a distribution due to the observation of another distribution) in an asymmetric version called AMI, where in order to distinguish between positive and negative correspondence we swap the sign of MI by means of a function \( \rho \) if \( tpr + tnr < 1 \).

2.2 Kernel-Based DCFs

Kernel-based DCFs are rooted in the kernel functions typically used, e.g., within Support Vector Machines (SVM). In this case the values in the context vector can be numeric, thus indicating the relative importance of a term in a given context (we use \( tfidf \) as the weighting function). We consider normalized context vectors, i.e., after weighting the document-by-term matrix we normalize the term vectors to unit length.

However (and differently from the probability-based DCFs we have considered above), not all kernels satisfy the \emph{null correspondence} property. In general, a valid DCF \( \eta_K \) can be defined from a kernel \( K(\cdot, \cdot) \) as:

\[
\eta_K(u, v) = K(u, v) - \mathbb{E}_{v' \sim P(v)} [K(u', v')] 
\]

\[(1)\]

where \( u' \) and \( v' \) are any two random context vectors with the same prevalences as \( u \) and \( v \) (random variables represented by \( U \) and \( V \), respectively).

The second part of Table 1 shows the Kernel-based DCFs we have considered, based on the cosine, the polynomial and the radial basis function kernels.

<table>
<thead>
<tr>
<th>DCF</th>
<th>Mathematical form</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>( P(f^1</td>
</tr>
<tr>
<td>PMI</td>
<td>( \log_2 \frac{P(f^1,f^2)}{P(f^1)P(f^2)} )</td>
</tr>
<tr>
<td>AMI</td>
<td>( \sum_{i \in [f^1]} \sum_{j \in [f^2]} P(x,y) \log_2 \frac{P(x,y)}{P(x)P(y)} )</td>
</tr>
<tr>
<td>Cosine</td>
<td>( \frac{(c^1) \cdot (c^2)}{|c^1| |c^2|} )</td>
</tr>
<tr>
<td>Polynomial</td>
<td>( (a + |F^1|)^k - (a + |F^2|)^k )</td>
</tr>
<tr>
<td>RBF</td>
<td>( \exp(-\gamma |F^1 - F^2|^2) - \exp(-4\gamma (1 - \sqrt{\rho})^2) )</td>
</tr>
</tbody>
</table>

Table 1: Mathematical forms of DCFs discussed in this work. We use \( p_k \) to denote the prevalence of vector \( F^k \).

3 Distributional Correspondence Indexing

The working hypothesis of DCI is that words that play similar roles in their respective domains may present approximately invariant correspondences to the pivots. For example, consider the source domain of book reviews and the target domain of movie reviews. Consider also the list of pivots [intriguing, annoying, captivating, . . .] and a DCF \( \eta \). We might thus expect to have:

\[
\begin{align*}
\eta(\text{book}; \text{intriguing}) & \approx \eta(\text{movie}; \text{intriguing}) \\
\eta(\text{book}; \text{annoying}) & \approx \eta(\text{movie}; \text{annoying}) \\
\eta(\text{book}; \text{captivating}) & \approx \eta(\text{movie}; \text{captivating})
\end{align*}
\]

etc., since \text{book} and \text{movie} terms play analogous roles in the source and target domains. Therefore, by embedding each term with respect to its DCF values to the pivots, similar terms across domains might end up being represented by similar vectors in a common vector space. The cross-lingual adaptation is achieved by presenting each pivot by a translation equivalent in the target language.

3.1 Pivot Selection

[Blitzer et al., 2006; 2007] defined pivots as highly predictive terms which occur frequently in the source and target domains and behave similarly in both domains. A good pivot should be highly task-dependent, and also present a similar degree of domain-dependence in the two domains. We thus look for pivots by selecting the top \( n \) terms (where \( n \) is a parameter) ranked by their pivot strength \( \Psi(f^i) \) defined as:

\[
\Psi(f^i) = MI_s (f^i) \min \left\{ \frac{p_i^1}{p_i^2}, \frac{p_i^2}{p_i^1} \right\} \max \left\{ p_i^1, p_i^2 \right\}
\]

\[(2)\]

where \( MI_s (f^i) \) is the mutual information (as previously done in [Prettenhofer and Stein, 2011]) of the term \( f^i \) to the label (to be estimated on the training set \( T_{tr} \)), and the rightmost factor is the cross-consistency, measured as the drift in prevalence (\( p_i \)) of the term \( f^i \) across the two domains.
3.2 Term Profiles
Given $n$ pivots and a DCF $\eta$, each source and target term (including the pivot terms) is embedded as an $n$-dimensional profile vector

$$f = (\eta(f, p^1), \eta(f, p^2), \ldots, \eta(f, p^n))$$

(3)

where $f$ and $p^i$ are the context distribution vectors (typically estimated in unlabelled collections) of the term $f$ being profiled and the $i^{th}$ pivot, respectively.

In order to avoid pivots with high prevalence to generate high DCF values which could lead to dominant dimensions, we center each profile dimension on its expected value and then rescale by the standard deviation, so that the values for all profile dimensions are approximately normally distributed in $\mathcal{N}(0, 1)$, and then rescale them to unit length.

As we assume pivot terms behave similarly in the two domains, we unify their term profiles by averaging the source and the target profiles in order to correct the possible misalignment between the source and target views of the pivot.

3.3 Document Indexing
Finally, train and test documents are indexed in the profile space via a weighted sum of all profile vectors associated to their terms. That is, document $d_j$ is represented as the $m$-dimensional vector

$$d_j = \sum_{f_i \in d_j} w_{ij} \cdot f_i$$

(4)

where $w_{ij}$ is the weight of term $f_i$ in document $d_j$ according to any weighting function (in our experiments we used the standard cosine-normalized $\text{tfidf}$), and $f_i^j$ is the normalized term profile vector for $f_i$.

4 Experiments
In this section we experimentally compare DCI, equipped with different DCFs, to other state-of-the-art methods proposed in the literature.

We test these methods on two popular, publicly available sentiment datasets: Multi-Domain Sentiment Dataset\(^1\) (MDS) [Blitzer et al., 2007] and Webis-CLS-10\(^2\) [Prettenhofer and Stein, 2011]. MDS is frequently used for evaluating cross-domain adaptation approaches and consists of English product reviews taken from Amazon.com for the four domains Books, DVDs, Electronics, and Kitchen appliances. The dataset comprises 1000 positive reviews and 1000 negative reviews for each of the four domains, and a set of unlabelled documents ranging from 3,586 to 5,945 documents for each domain. According to the same evaluation procedure followed by the proposers of other methods we compare against, we randomly split each labelled dataset into a training set of 1600 instances and a test set of 400 instances. In order to facilitate reproducibility and to allow for a fair comparison with the results reported in previous literature, we use the same pre-processed version of the dataset and the same experimental protocol used in [Blitzer et al., 2007].

\(1^\text{http://www.cs.jhu.edu/ mdredze/datasets/sentiment/}
\(^2\text{https://goo.gl/eppCro}\)

The Webis-CLS-10 dataset, frequently used for evaluating cross-lingual methods, consists of Amazon product reviews written in four languages (English, German, French, and Japanese), covering three product domains (Books, DVDs, and Music). For each language-domain pair there are 2,000 training documents, 2,000 test documents, and from 9,000 to 50,000 unlabelled documents depending on the language-domain combination. We adhere to the exact experimental protocol described in [Prettenhofer and Stein, 2011] and used by many researchers in the field.

As the evaluation measure we use standard accuracy (the proportion of correctly classified documents over the total number of outcomes), following the practice common in the related literature.

For this extended abstract, we have updated the list of baseline methods by adding new approaches (TCT [Huang et al., 2017], TrAdaB [Huang et al., 2017], DANN [Ganin et al., 2016], CL-TS [Zhou et al., 2015], Bi-PV [Xu and Wan, 2017], BiDRL [Zhou et al., 2016b], WSDNNs, [Zhou et al., 2016a], CLDFA [Xu and Yang, 2017]) which have been published in the cross-domain and cross-lingual arena after our original work [Moreo Fernández et al., 2016], and kept those which performed best in our original evaluation (SCL-MI [Blitzer et al., 2007], SFA [Pan et al., 2010], SDA [Glorot et al., 2011], and SSMC [Xiao and Guo, 2014]). We also consider an upper bound that trains the classifier on the training set of the target domain (“Upper”), and a lower bound that trains the classifier on the source domain and then applies the trained classifier directly in the target domain, i.e., without carrying out any sort of knowledge transfer (“No-Trans”). For cross-lingual adaptation we also report the machine translation baseline (“MT”), which first translates all target documents into the source language (English, in our experiments) before giving them as input to the classifier; we use the pre-translated documents provided by [Prettenhofer and Stein, 2011].

We implemented our method as part of the JaTeCs [Esuli et al., 2017] framework using SVMs as the learning device. In all experiments we set the number of pivots to 100. To emulate the word oracle that translates a source pivot word to the target language we used the bilingual dictionaries created by [Prettenhofer and Stein, 2011].

Table 2 and 3 report the results for cross-domain adaptation and cross-lingual adaptation, respectively (bold indicates the best score for each dataset, while greyed-out cells indicate the DCI variants which outperform all other competitors). We refer the interested reader to our original paper [Moreo Fernández et al., 2016], where other scenarios (such as simultaneously tackling cross-domain and cross-lingual) are explored.

Most variants of DCI perform comparably or better to all compared methods, even to the new approaches that have been published after our original paper. In this regard, it is interesting to note that many configurations of DCI outperform on average all the competitors. Particularly, the Cosine and Polynomial DCFs have shown to stand the test of time, delivering the best averaged results both in cross-domain and cross-lingual experiments.
### 4.1 Embeddings

The intuitions behind DCI bear strong resemblance to those behind “word embeddings” [Mikolov et al., 2013], as from deep learning. Although neural models typically require large quantities of text data and expensive resources in terms of computation, DCI delivers meaningful representations in a fraction of the time they require (see [Moreo Fernández et al., 2016] for a more detailed discussion on efficiency).

Table 4 illustrates the semantic properties captured by our term profiles; it lists the most similar (via cosine similarity) target terms to a given source term.

<table>
<thead>
<tr>
<th>Source</th>
<th>Target</th>
<th>No/Trans</th>
<th>Upper</th>
<th>SCL-MI</th>
<th>SFA</th>
<th>TCT</th>
<th>SDA</th>
<th>TraAdB</th>
<th>DANN</th>
<th>Linear</th>
<th>PMI</th>
<th>AMI</th>
<th>Cos</th>
<th>Poly</th>
<th>RBF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Books</td>
<td>DVD</td>
<td>.772</td>
<td>.847</td>
<td>.758</td>
<td>.814</td>
<td>.818</td>
<td>.804</td>
<td>.796</td>
<td>.829</td>
<td>.808</td>
<td>.811</td>
<td>.806</td>
<td>.817</td>
<td>.829</td>
<td>.815</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>.708</td>
<td>.869</td>
<td>.759</td>
<td>.725</td>
<td>.757</td>
<td>.806</td>
<td>.749</td>
<td>.804</td>
<td>.810</td>
<td>.822</td>
<td>.793</td>
<td>.822</td>
<td>.826</td>
<td>.821</td>
</tr>
<tr>
<td></td>
<td>Kitchen</td>
<td>.745</td>
<td>.902</td>
<td>.789</td>
<td>.788</td>
<td>.789</td>
<td>.844</td>
<td>.778</td>
<td>.843</td>
<td>.834</td>
<td>.839</td>
<td>.822</td>
<td>.835</td>
<td>.844</td>
<td>.835</td>
</tr>
<tr>
<td>DVD</td>
<td>Books</td>
<td>.728</td>
<td>.844</td>
<td>.797</td>
<td>.775</td>
<td>.792</td>
<td>.724</td>
<td>.747</td>
<td>.825</td>
<td>.825</td>
<td>.821</td>
<td>.811</td>
<td>.824</td>
<td>.830</td>
<td>.825</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>.730</td>
<td>.869</td>
<td>.741</td>
<td>.767</td>
<td>.778</td>
<td>.872</td>
<td>.759</td>
<td>.809</td>
<td>.822</td>
<td>.832</td>
<td>.812</td>
<td>.824</td>
<td>.833</td>
<td>.826</td>
</tr>
<tr>
<td>Electronics</td>
<td>Books</td>
<td>.707</td>
<td>.844</td>
<td>.754</td>
<td>.757</td>
<td>.759</td>
<td>.768</td>
<td>.691</td>
<td>.774</td>
<td>.766</td>
<td>.763</td>
<td>.753</td>
<td>.764</td>
<td>.776</td>
<td>.765</td>
</tr>
<tr>
<td></td>
<td>DVD</td>
<td>.706</td>
<td>.847</td>
<td>.762</td>
<td>.772</td>
<td>.773</td>
<td>.902</td>
<td>.718</td>
<td>.781</td>
<td>.768</td>
<td>.779</td>
<td>.765</td>
<td>.774</td>
<td>.799</td>
<td>.774</td>
</tr>
<tr>
<td>Kitchen</td>
<td>Books</td>
<td>.709</td>
<td>.844</td>
<td>.686</td>
<td>.748</td>
<td>.748</td>
<td>.807</td>
<td>.706</td>
<td>.718</td>
<td>.783</td>
<td>.783</td>
<td>.769</td>
<td>.790</td>
<td>.791</td>
<td>.784</td>
</tr>
<tr>
<td></td>
<td>DVD</td>
<td>.727</td>
<td>.847</td>
<td>.769</td>
<td>.766</td>
<td>.785</td>
<td>.835</td>
<td>.744</td>
<td>.789</td>
<td>.788</td>
<td>.789</td>
<td>.781</td>
<td>.799</td>
<td>.807</td>
<td>.798</td>
</tr>
<tr>
<td></td>
<td>Electronics</td>
<td>.827</td>
<td>.869</td>
<td>.868</td>
<td>.851</td>
<td>.856</td>
<td>.802</td>
<td>.831</td>
<td>.856</td>
<td>.855</td>
<td>.851</td>
<td>.843</td>
<td>.858</td>
<td>.863</td>
<td>.857</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>.745</td>
<td>.866</td>
<td>.780</td>
<td>.786</td>
<td>.794</td>
<td>.812</td>
<td>.759</td>
<td>.813</td>
<td>.815</td>
<td>.818</td>
<td>.804</td>
<td>.820</td>
<td>.828</td>
<td>.819</td>
</tr>
</tbody>
</table>

Table 2: Cross-domain adaptation on the MDS dataset.

<table>
<thead>
<tr>
<th>Target Language</th>
<th>Domain</th>
<th>No/Trans</th>
<th>Upper</th>
<th>SCL-MI</th>
<th>SFA</th>
<th>TCT</th>
<th>SDA</th>
<th>T-JV</th>
<th>Bi-JV</th>
<th>BiDRL</th>
<th>WSINSN</th>
<th>CLDFA</th>
<th>Linear</th>
<th>PMI</th>
<th>AMI</th>
<th>Cos</th>
<th>Poly</th>
<th>RBF</th>
</tr>
</thead>
<tbody>
<tr>
<td>German</td>
<td>Books</td>
<td>.541</td>
<td>.867</td>
<td>.808</td>
<td>.833</td>
<td>.819</td>
<td>.799</td>
<td>.796</td>
<td>.841</td>
<td>.813</td>
<td>.774</td>
<td>.796</td>
<td>.798</td>
<td>.714</td>
<td>.794</td>
<td>.827</td>
<td>.837</td>
<td>.829</td>
</tr>
<tr>
<td></td>
<td>DVD</td>
<td>.556</td>
<td>.835</td>
<td>.800</td>
<td>.809</td>
<td>.823</td>
<td>.819</td>
<td>.786</td>
<td>.841</td>
<td>.824</td>
<td>.831</td>
<td>.826</td>
<td>.819</td>
<td>.800</td>
<td>.822</td>
<td>.833</td>
<td>.788</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>.539</td>
<td>.859</td>
<td>.790</td>
<td>.829</td>
<td>.813</td>
<td>.796</td>
<td>.825</td>
<td>.847</td>
<td>.807</td>
<td>.790</td>
<td>.844</td>
<td>.850</td>
<td>.837</td>
<td>.856</td>
<td>.844</td>
<td>.804</td>
<td></td>
</tr>
<tr>
<td>French</td>
<td>Books</td>
<td>.589</td>
<td>.811</td>
<td>.821</td>
<td>.813</td>
<td>.811</td>
<td>.826</td>
<td>.843</td>
<td>.844</td>
<td>.834</td>
<td>.746</td>
<td>.761</td>
<td>.768</td>
<td>.842</td>
<td>.819</td>
<td>.844</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DVD</td>
<td>.519</td>
<td>.871</td>
<td>.794</td>
<td>.804</td>
<td>.827</td>
<td>.827</td>
<td>.796</td>
<td>.836</td>
<td>.836</td>
<td>.826</td>
<td>.823</td>
<td>.823</td>
<td>.801</td>
<td>.827</td>
<td>.806</td>
<td>.846</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>.551</td>
<td>.889</td>
<td>.764</td>
<td>.805</td>
<td>.805</td>
<td>.802</td>
<td>.801</td>
<td>.826</td>
<td>.810</td>
<td>.833</td>
<td>.816</td>
<td>.827</td>
<td>.818</td>
<td>.844</td>
<td>.840</td>
<td>.803</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DVD</td>
<td>.471</td>
<td>.834</td>
<td>.722</td>
<td>.764</td>
<td>.776</td>
<td>.771</td>
<td>.754</td>
<td>.768</td>
<td>.783</td>
<td>.805</td>
<td>.822</td>
<td>.768</td>
<td>.797</td>
<td>.801</td>
<td>.795</td>
<td>.761</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Music</td>
<td>.535</td>
<td>.842</td>
<td>.714</td>
<td>.773</td>
<td>.775</td>
<td>.684</td>
<td>.733</td>
<td>.788</td>
<td>.775</td>
<td>.765</td>
<td>.826</td>
<td>.816</td>
<td>.807</td>
<td>.839</td>
<td>.832</td>
<td>.826</td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>.532</td>
<td>.852</td>
<td>.767</td>
<td>.797</td>
<td>.801</td>
<td>.794</td>
<td>.786</td>
<td>.814</td>
<td>.803</td>
<td>.811</td>
<td>.809</td>
<td>.790</td>
<td>.793</td>
<td>.824</td>
<td>.818</td>
<td>.809</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Cross-lingual adaptation on the Webis-CLS-10 dataset.

### 5 Conclusions and Future Work

Distributional Correspondence Indexing is an efficient method for domain adaptation that represents terms in a vectorial space based on their distributional correspondence with respect to a small, fixed set of terms. This representation is motivated by the distributional hypothesis [Harris, 1954] and the notion of a “pivot term” [Blitzer et al., 2006]; the method indexes documents from different domains into a common vector space based on their semantic correspondence. Unlike other distributional semantic methods, DCI gives a lighter interpretation to the hypothesis through the distributional correspondence functions, resulting in a computationally cheap approach to domain adaptation.

Empirical evaluation we have carried out on two popular sentiment analysis benchmarks shows that our method outperforms several state-of-the-art approaches in different domain adaptation settings, including cross-domain and cross-lingual sentiment adaptation. DCI has remained unbeaten since its appearance in 2016.

In future research, we plan to put to test DCI in other domains and settings, including multi-class multi- and single-label datasets, highly imbalanced classes, and transductive problems.
References


