Lost in Transduction: Transductive Transfer Learning in Text Classification

ALEJANDRO MOREO, ANDREA ESULI, and FABRIZIO SEBASTIANI, Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche

Obtaining high-quality labelled data for training a classifier in a new application domain is often costly. *Transfer Learning* (a.k.a. "Inductive Transfer") tries to alleviate these costs by transferring, to the "target" domain of interest, knowledge available from a different "source" domain. In transfer learning the lack of labelled information from the target domain is compensated by the availability at training time of a set of unlabelled examples from the target distribution. *Transductive Transfer Learning* denotes the transfer learning setting in which the only set of target documents that we are interested in classifying is known and available at training time. Although this definition is indeed in line with Vapnik's original definition of "transduction", current terminology in the field is confused. In this article we discuss how the term "transduction" has been misused in the transfer learning literature, and propose a clarification consistent with the original characterization of this term given by Vapnik. We go on to observe that the above terminology misuse has brought about misleading experimental comparisons, with inductive transfer learning methods that have been incorrectly compared with transductive transfer learning methods. We then give empirical evidence that the difference in performance between the inductive version and the transductive version of a transfer learning method can indeed be statistically significant (i.e., that knowing at training time the only data one needs to classify indeed gives an advantage). Our clarification allows a reassessment of the field, and of the relative merits of the major, state-of-the-art algorithms for transfer learning in text classification.

CCS Concepts: • **Computing methodologies** → **Machine learning**; *Semi-supervised learning settings*; *Classification and regression trees.*

Additional Key Words and Phrases: Transduction, Induction, Transfer Learning, Text Classification, Distributional Hypothesis

ACM Reference Format:
Alejandro Moreo, Andrea Esuli, and Fabrizio Sebastiani. 2021. Lost in Transduction: Transductive Transfer Learning in Text Classification. *ACM Trans. Knowl. Discov. Data.* 1, 1 (February 2021), 21 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

When performed via supervised machine learning, text classification (TC) requires the availability of a substantive amount of accurately annotated training data sampled from the distribution of interest. When enough labelled data are not available, it is necessary to annotate unlabelled data, and this requires time and money. Many research efforts have thus been devoted to devising methods that, in the presence of little or no labelled data, allow to leverage other resources, such as unlabelled

Authors' address: Alejandro Moreo, alejandro.moreo@isti.cnr.it; Andrea Esuli, andrea.esuli@isti.cnr.it; Fabrizio Sebastiani, fabrizio.sebastiani@isti.cnr.it, Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche, Via Giuseppe Moruzzi, 1, 56124, Pisa, Italy.

1556-4681/2021/2-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

data (when at least some labelled data are available we then speak of *semi-supervised learning* [12]), or data labelled for tasks somehow different from the one of interest (*transfer learning* [38], a.k.a. *inductive transfer* [52]). Transfer learning thus relaxes a core assumption of supervised machine learning, usually referred to as the *iid assumption*, according to which the training examples and the unlabelled examples to be classified are drawn from the same distribution.[1]

A typical instantiation of transfer learning (and the only one we are going to consider in this paper) is the one in which labelled data are available only for a so-called *source* domain (or *out-domain*) $\mathcal{S}$, and any available data from the *target* domain of interest (or *in-domain*) $\mathcal{T}$ are unlabelled. For example, consider the case in which we want to create a sentiment classifier for book reviews, for which no labelled examples are available. Instead of incurring the cost of manually annotating book reviews, we might attempt to reuse labelled *movie* reviews we may already have, in combination with a set of *unlabelled* book reviews. Such a transfer learning setting displays characteristics of induction (the learner is asked to infer a general rule h from the observation of data) and semi-supervision (some of the observed data are labelled and some are not). Arguably, a meaningful term to describe this setting would thus be "semi-supervised inductive transfer learning". However, this term would clash with the definitions proposed in [1], where "inductive transfer learning" is used to refer to the case in which labelled data exist also for the target domain $\mathcal{T}$, and with that of [38], where "inductive transfer learning" instead denotes the case in which in the source domain $\mathcal{S}$ and in the target domain $\mathcal{T}$ the data are from the same distribution but the sets of classes are different.

The root of these discrepancies in terminology may be explained by the fact that transfer learning has evolved in parallel with research on *dataset shift* [45], a strongly related area devoted to the more general problem of dealing with various types of distributional difference between the labelled and the unlabelled data; different instantiations of dataset shift are, e.g., *covariate shift* [48], *prior probability shift* [50], and *concept drift* [54]. Indeed, when a field emerges from the joint effort of different scientific communities (as is the case for transfer learning), it is common to find both terminological inconsistencies and attempts to unify and clarify such terminology (e.g., [33]). As a further related example, some authors use the term "domain adaptation" (DA) to denote a special case of transfer learning (e.g., [38]), others consider DA and transfer learning as two separate problems (e.g., [40]), and yet others consider the two terms as synonyms (e.g., [52]).[2]

We think that one of these terminological inconsistencies is becoming particularly problematic, because it may completely mislead the reader about the applicative context to which a given transfer learning method can be applied, and because it may lead (and has indeed led) to flawed experimental comparisons among different transfer learning methods. We are speaking about the use of the term "transduction", originally introduced by Vladimir Vapnik [51], and about how its meaning has drifted in the transfer learning literature.

In machine learning, the term *transduction* as introduced by Vapnik means "inference from particular to particular" [16], i.e., describes the inference carried out by learning methods that (i) are given access not only to a labelled training set but also *to the only set of unlabelled data we are interested in classifying* (in this paper we will call the latter *the object set*, and (ii) do not label the

[1] More precisely, the iid assumption states that, if the training set and the unlabelled set are viewed as random variables, these two random variables exhibit the same probability distribution and are mutually independent.

[2] This is in line with what Lipton and Steinhardt [31] call "a troubling scientific trend" in machine learning, a trend of misuse of language caused by overloading technical terminology, which "consists of taking a term that holds precise technical meaning and using it in an imprecise or contradictory way."

documents in the object set by means of a general-purpose classifier.[3] In other words, transduction is meant to be applied to settings in which we exactly know, before any learning has taken place, that we will not be interested in classifying any unlabelled data other than those belonging to a finite, specific set that is already available to us at training time.[4] Scenarios such as these are common, e.g., in market research [6], in e-discovery [36], or when assisting the production of systematic reviews [26].

The main contributions of this article can be summarized as follows. In Section 2 we propose a clarification of terminology that restores the original sense of the term "transductive inference", as proposed by Vapnik, in the context of transfer learning, while in Section 3 we discuss how the meaning of "transduction" has shifted in recent literature. In Section 4 we then identify cases in which the misuse of terminology has led to confusion and incorrect experimental comparisons. In Section 5 we go on to provide empirical evidence that the differences in performance between the inductive and transductive variants of a given transfer learning method can be statistically significant, which implies that experimental comparisons that confuse the two variants (among which the ones identified in Section 4) are *seriously* flawed. For doing so, we provide examples of these statistically significant differences, which we obtain by (i) comparing the performance of previously published transfer methods belonging to the inductive group and the transductive group, and (ii) by comparing the performance of two inductive transfer learning methods (Structural Correspondence Learning (SCL) [7, 43] and Distributional Correspondence Indexing (DCI) [34]) with corresponding transductive variants that we have generated. Section 6 presents some concluding remarks.

2 A TAXONOMY OF LEARNING METHODS

In this section we formalize the difference between methods for inductive learning, semi-supervised learning, transductive learning, inductive transfer learning, and transductive transfer learning.

Let us first define some basic concepts. A *domain* is a triple $\mathcal{D} = (X, F, \phi)$, where X is a random variable taking values on documents, F is a feature space (e.g., a vector space $\mathcal{R}^m$), and ϕ is the representation function $\phi : X \rightarrow F$ which maps documents into the feature space. Note that the image of ϕ is also a random variable, that we call the *domain distribution* and denote as $P_{\mathcal{D}}$. A *sample* σ of a domain $\mathcal{D}$ is an empirical distribution containing random variates of $P_{\mathcal{D}}$, i.e., a set $\sigma = \{\mathbf{x}_i\}_{i=1}^{n} \subset P_{\mathcal{D}}$ of feature vectors drawn from the domain distribution. We will use $\sigma_{\mathcal{D}}$ to indicate that sample σ originates from domain $\mathcal{D}$.

For ease of discussion, in this paper we restrict our attention to binary classification; however, everything we say can straightforwardly be extended to other types of classification, such as single-label multiclass classification, multi-label multiclass classification, and ordinal classification. A binary classifier is a function $h : F \rightarrow Y$, with $Y = \{-1, +1\}$ the label space. We use $\sigma_{\mathcal{D}}^{L}$ to denote any labelled sample $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n} \subset P_{\mathcal{D}} \times Y$, where document $\mathbf{x}_i$ has label y_i.

The following instantiations of the aforementioned concepts will prove useful in our subsequent definitions: in the rest of the paper $\mathcal{S}$ and $\mathcal{T}$ will denote the source and target domains, while $Tr_{\mathcal{S}}^{L}$, $Tr_{\mathcal{S}}^{U}$, $Tr_{\mathcal{T}}^{U}$, $Ob_{\mathcal{S}}^{U}$, $Ob_{\mathcal{T}}^{U}$, $Te_{\mathcal{S}}^{U}$, $Te_{\mathcal{T}}^{U}$, will denote samples, where $Tr_{\mathcal{S}}^{L}$ is a labelled training set, $Tr_{\mathcal{S}}^{U}$ and $Tr_{\mathcal{T}}^{U}$ are unlabelled training sets, $Ob_{\mathcal{S}}^{U}$ and $Ob_{\mathcal{T}}^{U}$ are unlabelled object sets, and $Te_{\mathcal{S}}^{U}$ and $Te_{\mathcal{T}}^{U}$ are unlabelled test sets, all drawn from $\mathcal{S}$ and $\mathcal{T}$ as indicated. As we will see in Definition 2.2, the notion

[3] As Vapnik puts it, "The direct estimation of values of a function only at points of interest using a given set of functions forms a new type of inference which can be called transductive inference. In contrast to the inductive solution that derives results in two steps, from particular to general (the inductive step) and then from general to particular (the deductive step), the transductive solution derives results in one step, directly from particular to particular (the transductive step)." [51, p. 12]

[4] Put it another way, should we later become interested in classifying another set of unlabelled data, the learning phase should be carried out anew.

of an "unlabelled training set" is justified, since in semi-supervised learning also unlabelled data play a role in training a classifier. We recall from Section 1 that an object set is a set of unlabelled documents such that (a) it is available at training time, and (b) the unlabelled data it contains are the only unlabelled data that we are interested in classifying.

Definition 2.1. *An* inductive learning (IL) *method is a method that, given a labelled training set* $Tr^L_{\mathcal{S}}$*, learns a general hypothesis* $h : P_{\mathcal{S}} \rightarrow Y$*.* □

The adequacy of h must be measured according to an evaluation function that measures the agreement between the predicted labels $h(\mathbf{x}_i)$ and the true labels y_i for a test set $Te^U_{\mathcal{S}}$ of documents. ($Te^U_{\mathcal{S}}$ is viewed as "unlabelled" because the true labels y_i are hidden from h.) The purpose of $Te^U_{\mathcal{S}}$ is to support this evaluation, which means that $Te^U_{\mathcal{S}}$ must not be seen at training time (that this practice has not always been adhered to in past transfer learning literature is, as we will see, a central claim of our article). Unlike $Ob^U_{\mathcal{S}}$ in transductive learning (see below), $Te^U_{\mathcal{S}}$ is expected to be sufficiently representative of $P_{\mathcal{S}}$, since the goal of the evaluation is to estimate the accuracy of h at classifying any possible unlabelled sample from the domain. Note that the training and test documents are assumed to be drawn iid from the same (and only) domain $\mathcal{S}$.

Definition 2.2. *A* semi-supervised learning (SSL) *method is a method that, given a labelled training set* $Tr^L_{\mathcal{S}}$ *and an unlabelled training set* $Tr^U_{\mathcal{S}}$*, learns a general hypothesis* $h : P_{\mathcal{S}} \rightarrow Y$*.* □

This case is also inductive, with the only difference that the learning device has access not only to labelled data $Tr^L_{\mathcal{S}}$ but also to unlabelled data $Tr^U_{\mathcal{S}}$ drawn from the same domain $\mathcal{S}$.

Definition 2.3. *A* transductive learning (TL) *method is a method that, given a labelled training set* $Tr^L_{\mathcal{S}}$ *and an unlabelled object set* $Ob^U_{\mathcal{S}}$*, generates predicted labels* $h(\mathbf{x}_i)$ *for all documents* $\mathbf{x}_i$ *in* $Ob^U_{\mathcal{S}}$ *directly, i.e., without using a general rule* $h : P_{\mathcal{S}} \rightarrow Y$*.* □

Note that in this case there is no requirement that the method also returns a general rule $h : P_{\mathcal{S}} \rightarrow Y$, i.e., the method might just learn a function $h' : Ob^U_{\mathcal{S}} \rightarrow Y$ that takes binary decisions *only for the elements of* $Ob^U_{\mathcal{S}}$.

It is important to distinguish a TL *method* (or *algorithm*) from a TL *problem* (or *setting*): in a nutshell, a problem is characterized by what we have and by what we want to achieve, while a method is characterized by how we achieve it. A TL problem is a situation in which, given a labelled training set $Tr^L_{\mathcal{S}}$ and an unlabelled object set $Ob^U_{\mathcal{S}}$, we need to generate predicted labels $h(\mathbf{x}_i)$ for all documents $\mathbf{x}_i$ in $Ob^U_{\mathcal{S}}$. In principle, IL *methods* are also applicable to TL *problems*, since $Ob^U_{\mathcal{S}}$ is just a specific sample from $P_{\mathcal{S}}$; in other words, we can generate predicted labels $h(\mathbf{x}_i)$ for all documents $\mathbf{x}_i$ in $Ob^U_{\mathcal{S}}$ *indirectly*, i.e., by learning a general rule $h : P_{\mathcal{S}} \rightarrow Y$ and using it to generate these predicted labels. Adopting such a solution might be called a "TLP-via-ILM approach", solving a TL problem via an IL method. Similarly, a "TLP-via-SSLM approach" would consist of solving a TL problem via a SSL method, and could be achieved by using an additional unlabelled training set $Tr^U_{\mathcal{S}}$ (with $Tr^U_{\mathcal{S}} \cap Ob^U_{\mathcal{S}} = \{\}$).

While legitimate, these solutions are suboptimal according to what is now known as "Vapnik's principle" [51], which suggests that[5]

[5]Vapnik's is a common-sense principle, one of the many "laws of parsimony" that guide scientific development. Another instance of Vapnik's principle in machine learning is represented by "quantification" (a.k.a. "supervised prevalence estimation" – see [19]), the task of predicting the distribution across the classes of a set of unlabelled items: while this can be achieved by classifying each item and counting how many items have been assigned to which class, it is more effective (in keeping with Vapnik's principle) to solve this problem directly, without resorting to classification.

> "If you possess a restricted amount of information for solving some problem, try to solve the problem directly and never solve a more general problem as an intermediate step. It is possible that the available information is sufficient for a direct solution but is insufficient for solving a more general intermediate problem."

In other words, Vapnik suggests that the optimal way (i.e., the one conducive to higher accuracy) of solving a TL problem is *directly, via a TL method*, and not indirectly, via (fully supervised or semi-supervised) IL methods.

Definition 2.4. *An* inductive transfer learning (ITL) *method is a method that, given a labelled training set $Tr^L_{\mathcal{S}}$ (plus, optionally, an unlabelled training set $Tr^U_{\mathcal{S}}$), and an unlabelled training set $Tr^U_{\mathcal{T}}$,* from two different but related domains $\mathcal{S}$ and $\mathcal{T}$, learns a general hypothesis $h : P_{\mathcal{T}} \rightarrow Y$. □

Note that this approach includes aspects from induction (the requirement that a general hypothesis is generated) and semi-supervision (the optional presence of the unlabelled training set). In this case the iid assumption no longer holds since $\mathcal{S} = (X_{\mathcal{S}}, F_{\mathcal{S}}, \phi_{\mathcal{S}})$ and $\mathcal{T} = (X_{\mathcal{T}}, F_{\mathcal{T}}, \phi_{\mathcal{T}})$ are different. This difference might be of type $X_{\mathcal{S}} \neq X_{\mathcal{T}}$ (with $F_{\mathcal{S}} = F_{\mathcal{T}}$), which is usually known as *cross-domain adaptation*, or of type $F_{\mathcal{S}} \neq F_{\mathcal{T}}$ (with $X_{\mathcal{S}} \sim X_{\mathcal{T}}$)[6], in which case the problem is typically known as *cross-lingual* adaptation[7]. Therefore, in both cases $\phi_{\mathcal{S}} \neq \phi_{\mathcal{T}}$ holds.

Definition 2.5. *A* transductive transfer learning (TTL) *method is a method that, given a labelled training set $Tr^L_{\mathcal{S}}$ and an unlabelled object set $Ob^U_{\mathcal{T}}$ (and optionally two unlabelled training sets $Tr^U_{\mathcal{S}}$ and $Tr^U_{\mathcal{T}}$, with $Tr^U_{\mathcal{T}} \cap Ob^U_{\mathcal{T}} = \{\}$) from two different but related domains $\mathcal{S}$ and $\mathcal{T}$, generates predicted labels $h(\mathbf{x}_i)$ for all documents $\mathbf{x}_i$ in $Ob^U_{\mathcal{T}}$ directly, i.e., without using a general rule $h : P_{\mathcal{T}} \rightarrow Y$.* □

The main differences of a TTL algorithm with respect to an ITL one thus lie in the facts that in the former, unlike in the latter, (i) there is an object set $Ob^U_{\mathcal{T}}$ which is observed at training time, and (ii) we generate no general hypothesis $h : P_{\mathcal{T}} \rightarrow Y$ but only predicted labels $h(\mathbf{x}_i)$ for documents $\mathbf{x}_i$ in $Ob^U_{\mathcal{T}}$.[8] The main difference of a TTL algorithm with respect to a TL one is instead that in the former, unlike in the latter, the training set and the object set are not iid, since they originate from two different domains $\mathcal{S}$ and $\mathcal{T}$.

Similarly to what we said for TL methods and TL problems, we should distinguish between TTL methods and TTL problems, the latter being the settings in which we need to generate predicted labels $h(\mathbf{x}_i)$ for all documents $\mathbf{x}_i$ in an object set $Ob^U_{\mathcal{T}}$, given a labelled training set $Tr^L_{\mathcal{S}}$ (and optionally two unlabelled training sets $Tr^U_{\mathcal{S}}$ and $Tr^U_{\mathcal{T}}$, with $Tr^U_{\mathcal{T}} \cap Ob^U_{\mathcal{T}} = \{\}$). A TTL problem may also be solved via an ITL method (which might be called a "TTLP-via-ITLM approach"), i.e., by labelling the documents $\mathbf{x}_i$ in $Ob^U_{\mathcal{T}}$ indirectly by learning a general-purpose rule $h : P_{\mathcal{T}} \rightarrow Y$, but this would be yet another violation of Vapnik's principle.

The definitions above are concisely summarized in Table 1.

It is possible to characterize the learning methods described above with respect to the stand they take according to three basic dichotomies:

- *Fully Supervised (FS) vs. Semi-Supervised (SS)*: the training data that a fully supervised method uses only consist of a labelled set Tr^L, while the training data that a semi-supervised method uses consist of a labelled set Tr^L and an unlabelled set Tr^U;

[6] In set theory, two sets A and B are said to be equivalent, denoted $A \sim B$ or $A \equiv B$, if there exists a bijection between the two, i.e., if they have the same cardinality. In cross-lingual adaptation, this comes down to assuming that a one-to-one correspondence between the documents in the source language and the documents in the target language is always possible (using, e.g., a translation oracle), since the documents in $X_{\mathcal{S}}$ and $X_{\mathcal{T}}$ are conceptually equivalent.

[7] Other instantiations exist, in which the cross-domain and cross-lingual adaptations are tackled simultaneously; see e.g., [34].

[8] In this respect, it is worth mentioning that Vapnik and his co-authors [16] suggested that transductive inference might still be attained in scenarios in which the iid assumption is relaxed.

Table 1. A taxonomy of learning methods and learning problems. An IL / SSL / TL / ITL / TTL *problem* (or setting) is characterised by the sets indicated in the middle five columns in rows 1 / 2 / 3 / 4 / 5, respectively. An IL / SSL / TL / ITL / TTL *method* (or algorithm) is characterised by the fact that, in the presence of the sets indicated in the middle five columns, the only output is the one indicated in the last column in rows 1 / 2 / 3 / 4 / 5, respectively.

	Labelled Training Set (source)	Unlabelled Training Set (source)	Unlabelled Object Set (source)	Unlabelled Training Set (target)	Unlabelled Object Set (target)	Output
Inductive Learning (IL)	$Tr_{\mathcal{S}}^{L}$	–	–	–	–	$h : P_{\mathcal{S}} \rightarrow Y$
Semi-Supervised Learning (SSL)	$Tr_{\mathcal{S}}^{L}$	$Tr_{\mathcal{S}}^{U}$	–	–	–	$h : P_{\mathcal{S}} \rightarrow Y$
Transductive Learning (TL)	$Tr_{\mathcal{S}}^{L}$	–	$Ob_{\mathcal{S}}^{U}$	–	–	$h(\mathbf{x}_i)$ for all $\mathbf{x}_i$ in $Ob_{\mathcal{S}}^{U}$
Inductive Transfer Learning (ITL)	$Tr_{\mathcal{S}}^{L}$	$(Tr_{\mathcal{S}}^{U})$	–	$Tr_{\mathcal{T}}^{U}$	–	$h : P_{\mathcal{T}} \rightarrow Y$
Transductive Transfer Learning (TTL)	$Tr_{\mathcal{S}}^{L}$	$(Tr_{\mathcal{S}}^{U})$	–	$(Tr_{\mathcal{T}}^{U})$	$Ob_{\mathcal{T}}^{U}$	$h(\mathbf{x}_i)$ for all $\mathbf{x}_i$ in $Ob_{\mathcal{T}}^{U}$

- *Inductive (IN) vs. Transductive (TR)*: an inductive method learns a general hypothesis $h : P_{\mathcal{D}} \rightarrow Y$, while a transductive method only generates predicted labels $h(\mathbf{x}_i)$ for all documents $\mathbf{x}_i$ in an object set $Ob_{\mathcal{D}}^{U}$;
- *Transfer (TF) vs. No-Transfer (NT)*: a transfer learning method needs to issue predictions for a target domain $\mathcal{T}$ different from the source domain $\mathcal{S}$ from which its labelled training data $Tr_{\mathcal{S}}^{L}$ come from, while for a no-transfer learning method $\mathcal{S}$ and $\mathcal{T}$ are the same domain.

The cases addressed by Definitions 2.1 to 2.5 are thus characterized by the triples (FS, IN, NT), (SS, IN, NT), (FS, TR, NT), (*, IN, TF), (*, IN, TF), (*, TR, TF), respectively.

Note that in this section, and in the rest of the paper, we have assumed that the learning problem is one of *classification*. However, everything we say in this paper straightforwardly applies to other supervised learning tasks, such as regression.[9]

3 THE SHIFTING MEANING OF "TRANSDUCTION"

The definition of "transduction" given in Section 2 is the one given by Vapnik [51, p. 12] (see also Footnote 3), and refers, according to the terminology we have introduced in Section 2, to transductive learning *methods*. However, in the context of transfer learning, that definition would partially clash with that of Arnold et al. [1], where the term "transductive transfer learning" appeared for the first time. In these authors' definition, the term "transductive learning" encompasses all scenarios where all the data we want to classify are already available at training time, and has nothing to do with the type of method used for classifying these data. In other words, Arnold et al. [1] seem to be thinking of transductive learning *problems* rather than of transductive learning *methods*: what in Section 2 we have called a "TTLP-via-ITLM" approach would squarely count, according to [1], as a transductive learning method. Indeed, some among the models that [1] proposed solve a transductive learning problem via an inductive learning method. Several works that followed (e.g., [3, 44]) adopted this definition of "transductive transfer learning". To the best of our knowledge, the only works about

[9]We have only dealt with text classification and not with regression, for three main reasons. The first reason is that classification is usually considered the "mother" of all supervised tasks. The second reason is that the terminological confusion that this paper addresses has arisen for classification, and has never involved, to the best of our knowledge, regression or other supervised tasks. The third reason is that, in the realm of text, classification is by far the most popular task, while text regression is a very infrequently tackled task, which also entails a difficulty to find datasets for experimentation; for example, when querying DBLP, at the time of writing query "text classification" returns 5,400 matches while query "text regression" just returns 110 matches.

transfer learning which use the term "transduction" in Vapnik's original sense are [46, 47] (although they presents no *text* classification experiments) and [2, 23] (which will be discussed in Section 4.2).

Years after [1] was published, the term "transduction" was used in the widely cited survey by Pan and Yang [38], which has henceforth become a standard reference for transfer learning.[10] However, these authors altered again the meaning of "transductive transfer learning", which they used to describe the more general setting in which "no labelled data in the target domain are available while a lot of labelled data in the source domain are available" [38], thus removing the constraint that all the unlabelled data we are interested in classifying must be available at training time.[11] In their terminology, cross-domain adaptation and cross-lingual adaptation (see Section 2) become two subproblems of transductive transfer learning, regardless of whether there is or not a set of unlabelled data available at training time that is the only data we are interested in classifying (i.e., an object set). Probably, [5, 20] were the first works that adopted this altered definition.

The lack of a clear distinction between induction and transduction, in the terms defined by Vapnik, in the field of transfer learning, makes it difficult for readers to understand whether a transfer learning *method* as applied to a transductive *problem* is actually an inductive transfer learning method (i.e., it labels the items in the object set by using a classifier that can be applied to any future set of unlabelled data) or is instead a transductive transfer learning method (i.e., it labels the unlabelled data seen at training time directly); we show examples of the two types of methods in Section 4. This aspect is worth taking into account since, despite the fact that a transductive transfer learning method could well be applied to different unlabelled sets by rerunning the method from scratch every time, this additional cost is avoided in inductive transfer learning. On the other side, on a transductive transfer learning problem one should expect better accuracy from a transductive transfer learning method than from an inductive transfer learning method, since the former is solving a less general (hence easier) problem than the latter, and thus might be preferred, given that generalization is not needed (see [11] for a broader discussion).

One consequence of the above-mentioned terminological confusion is the existence of "unfair" comparisons in the field, where some transductive transfer learning methods have been claimed to be superior to inductive transfer learning methods when tested on *inductive* transfer learning problems, i.e., *in problems in which the methods were not assumed to be learning from the unlabelled documents, and in which transductive methods were not meant to be applied at all.* This will be the topic of the next two sections.

4 INDUCTIVE AND TRANSDUCTIVE TRANSFER PROBLEMS

In this section we give a general view of previous efforts in the field on the basis of the distinctions discussed before, i.e., we will classify the methods according to whether they have been tested on an inductive setting or on a transductive setting, and according to whether they are actually inductive transfer learning methods or transductive transfer learning methods. In doing so, we do not describe each method in detail; we refer the interested reader to [38, 39] for a more detailed discussion, or to the original papers.

[10] At the time of writing, this paper has 11,989 citations on Google Scholar.

[11] This reformulation of the problem was deliberate and acknowledged in their survey, and was thus not due to a mistake. In their own words, "Note that the word "transductive" is used with several meanings. In the traditional machine learning setting, transductive learning (...) refers to the situation where all test data are required to be seen at training time, and that the learned model cannot be reused for future data. (Thus, when some new test data arrive, they must be classified together with all existing data. (...) In our categorization of transfer learning, in contrast, we use the term *transductive* to emphasize the concept that in this type of transfer learning, the tasks must be the same and there must be some unlabelled data available in the target domain." [38, p. 1352]

The goal of this section is not to offer a review of past literature, but rather to show the need for a clear distinction between induction and transduction. On the basis of this, we also identify cases in which the lack of such a clear distinction has led to unfair experimental comparisons and, in turn, to unreliable conclusions on the relative merits of different methods.

4.1 Transductive Transfer Problems

In a transductive transfer learning problem the learner is given access to the unlabelled object set $Ob^U_{\mathcal{T}}$ right from the beginning. The best-known benchmarks that have been used in order to test solutions to this problem are adaptations of the Reuters-21578, SRAA, and 20Newsgroups datasets (all well-known datasets for text classification by topic) proposed by Dai et al. [13, 14] for cross-domain adaptation.

The adaptation that [13, 14] propose leverages the hierarchical structure of the set of classes that characterise these datasets in order to generate new benchmarks for testing transfer learning systems. This procedure consists of picking two top-level classes, say, A and B, with subclasses $A_1, \ldots, A_x$ and $B_1, \ldots, B_y$, respectively, where the task is defined as a binary classification problem in which one needs to discriminate class A from class B. Then, two disjoint "folds" are extracted to form the source data ($\mathcal{S}$) and target data ($\mathcal{T}$); for instance, $A_{\mathcal{S}} = \bigcup_{i=1}^{\alpha} A_i$ and $A_{\mathcal{T}} = \bigcup_{i=\alpha+1}^{x} A_i$ will represent the source and target parts for class A, while $B_{\mathcal{S}} = \bigcup_{i=1}^{\beta} B_i$ and $B_{\mathcal{T}} = \bigcup_{i=\beta+1}^{x} B_i$ will represent the source and target parts for class B, for some $1 < \alpha < x$ and $1 < \beta < y$. Note that the documents in $\mathcal{S}$ and those in $\mathcal{T}$ are indeed related (they belong to the same top-level class) but different (they belong to different subclasses of the same top-level class), as requested in transfer learning. The training (source) set and the test (target) set are defined as $Tr^L_{\mathcal{S}} = A_{\mathcal{S}} \bigcup B_{\mathcal{S}}$ and $Te^U_{\mathcal{S}} = A_{\mathcal{T}} \bigcup B_{\mathcal{T}}$, respectively. Note that what we have described here is a setup for testing *inductive* transfer learning methods; if we want to test *transductive* transfer learning methods, $A_{\mathcal{T}} \bigcup B_{\mathcal{T}}$ must play the role of the object set $Ob^U_{\mathcal{T}}$ and of the test set $Te^U_{\mathcal{T}}$ at the same time, i.e., the documents in $A_{\mathcal{T}} \bigcup B_{\mathcal{T}}$ are available to the algorithm at training time, and the accuracy of the algorithm is measured in terms of how good it is at labelling them. Note also that there is no other unlabelled set, either from the source domain or from the target domain.

Datasets structured like this were first used by Dai et al. [13, 14] to test two different approaches: CoCC [13], which co-clusters domains and words as a means to propagate the class structure from the source domain to the target domain; and TrAdaBoost [14], an extension of AdaBoost that implements transfer learning. Since then, many authors have adopted experimental settings with the same structure, in order to test transfer learning systems based on topic models (e.g., Topic-Bridged PLSA (TPLSA – [60]), Topic-Bridged LDA (TLDA – [55]), and Partially Supervised Cross-Collection LDA (PSCCLDA –[4])), non-negative matrix factorization (e.g., MTrick [65]), probabilistic models (e.g., Topic Correlation Analysis (TCA – [27])), and clustering techniques (e.g., Cross-Domain Spectral Classification (CDSC – [30])).

However, although these methods have been tested on transductive transfer *problems* (i.e., by having $A_{\mathcal{T}} \bigcup B_{\mathcal{T}}$ play the role of $Ob^U_{\mathcal{T}}$ and $Te^U_{\mathcal{T}}$ at the same time), not all of them are transductive transfer *methods* as defined in Section 2. Indeed, TrAdaBoost [14], TLDA [55], and TCA [27] are *inductive* transfer methods; i.e., when applied to a transductive problem, a "TTLP-via-ITLM approach" must be followed. When inductive transfer learning methods are tested on an inductive transfer learning problem, they are meant to be tested on a test set $Te^U_{\mathcal{T}}$ *different* from the unlabelled set $Tr^U_{\mathcal{T}}$ on which they have been trained, in order to show that they generalize. Analogously, when these methods are tested on a transductive problem, the unlabelled training set $Tr^U_{\mathcal{T}}$ and the object set $Ob^U_{\mathcal{T}}$ must be different too. It is one of the central observations of this paper that this caveat has

not always been adhered to in comparative experimentations, and this has brought about flawed comparative results that are still being relied upon today.

4.2 Inductive Transfer Problems

In an inductive transfer learning problem the learner has access to the labelled set $Tr_{\mathcal{S}}^{L}$ from the source domain and the unlabelled set $Tr_{\mathcal{T}}^{U}$ from the target domain (an unlabelled set $Tr_{\mathcal{S}}^{U}$ from the source domain might be available as well). There is no object set $Ob_{\mathcal{T}}^{U}$ since the goal is to generate (induce) a general-purpose classifier for the entire target domain. The test set $Te_{\mathcal{T}}^{U}$ is thus only meant to be used for evaluation purposes, i.e., for estimating the effectiveness of the classifier in classifying any document from the target domain.

The most popular benchmarks for testing solutions to these problems are MDS [7], which was proposed for cross-domain adaptation, and its cross-lingual extension Webis-CLS-10 [42]. Both datasets consist of Amazon product reviews for different product categories, and include 2,000 labelled reviews per product category and a number of unlabelled reviews, ranging from 3,586 (DVD reviews in MDS) to more than 50,000 (in Webis-CLS-10). Neutral reviews have been filtered out, and the task is thus defined as a binary sentiment classification problem (Positive vs. Negative).

This has promoted a (somehow unmotivated) partition of transfer learning methods, according to which most of the methods tested on transductive transfer problems deal with classification by topic, while most of the methods tested on inductive transfer problems deal instead with classification by sentiment. The net result is that inductive transfer problems have received comparatively more attention than their transductive transfer counterparts. In what follows we give a comprehensive overview of the most important methods in the area, and show that some of them are actually transductive transfer methods, something that was not to be expected given the characteristics of the datasets they have been tested on.

Arguably, the most important methods proposed for the inductive transfer problem are Structural Correspondence Learning (SCL) [7] for cross-domain adaptation, and its cross-lingual version (CL-SCL) [43]. SCL bridges the gap between the source and target domains by solving intermediate structural problems defined upon the notion of *pivot features* (frequent and predictive features that behave approximately similarly in both domains). Pivots are typically discovered by inspecting the supervised source set (e.g., by measuring the mutual information between a feature and the class labels); their distributional properties are mined by inspecting the unlabelled source and target training sets $Tr_{\mathcal{S}}^{U}$ and $Tr_{\mathcal{T}}^{U}$. Other methods that follow similar principles have been described since then, including further pivot-based approaches like Spectral Feature Alignment (SFA) [37] for cross-domain adaptation, and Distributional Correspondence Indexing (DCI) [34] for cross-domain and cross-lingual adaptation. Other methods that similarly rely on mutual information as a means to quantify semantic correlations among words have been described, as e.g., Sentiment-Sensitive Thesaurus (SST) [10] does in order to expand a sentiment thesaurus.

Although the concept of "pivot" concerns, strictly speaking, pairs of related words, the same concept is still present behind many non-negative matrix factorization (NMF) techniques, though blurred under the notion of "latent topic". Examples of NMF techniques include Topical Correspondence Transfer (TCT) [63] for cross-domain adaptation, Semi-supervised Matrix Completion (SSMC) [57], Two-Step Learning (TSL) [56], and the Subspace Learning Framework (CL-SLF) [62] for cross-lingual adaptation. Very recently, [23] has proposed TKC, a transductive method based on string kernels that was also evaluated on the MDS dataset.

Yet another group of approaches tested on inductive transfer problems has emerged, fostered by the recent upsurge of deep learning. We distinguish between deep architectures and word

embeddings-based approaches. The first approach based on deep architectures was Stacked Denoising Autoencoders (SDA) [18], a method that exploited the autoencoding architecture to enforce a consistent representation between source and target in cross-domain adaptation. This was followed by other SDA-based approaches such as Cross-Domain Feature Learning (CDFL) [61], approaches based on adversarial neural networks such as Domain Adversarial Neural Network (DANN) [17] and a transductive variant (TransDANN) [2], Cross-Lingual Distillation with Feature Adaptation (CLDFA) [59], and combinations of adversarial training with attention mechanisms, such as Adversarial Memory Network (AMN) [29] and Hierarchical Attention Transfer Network (HATN) [28]. Finally, methods for learning (monolingual) word embeddings (Sentiment-Sensitive Embeddings – SSE) [9] for cross-domain adaptation, bilingual word embeddings (Bilingual Model – BM) [58], bilingual phrase embeddings [41], or for jointly learning bilingual word and document embeddings (Bilingual Document Representation Learning – BiDRL) [64] for cross-lingual adaptation, have also been proposed.

Some of the aforementioned methods make use of parallel data (generated via automatic translation tools as in SSMC [57], CL-SLF [62], BiDRL [64], CLDFA [59], or inspecting already existing parallel resources as in BM [58]) or counted with a fraction of labelled data from the target domain (as is the case of SSMC [57]). Somehow surprisingly, it turns out that most of these methods are actually of the transductive transfer type (and this is something the reader might not expect, considering the datasets those methods have been tested on, and the baselines they have compared against); concretely, this affects the methods SSMC [57], CL-SLF [62], BiDRL [64], and CLDFA [59]. The reason is that the parallel data the authors considered in their experiments are the translations that Prettenhofer and Stein made available for the non-English *test* documents in Webis-CLS-10. This means that, even assuming the approaches could have been trained on a different set of parallel documents (and this is something which incidentally remains unclear), the truth is that the results they reported are inevitably optimized for the specific test documents (unfairly taken to be the *object* set), and can thus not be granted to be representative of the more general inductive transfer problem. TransDANN [2] and TKC [23] also fall in the "transductive group", though in these cases the incursion was deliberated and openly acknowledged.

Methods like SSMC [57], CL-SLF [62], BiDRL [64], and CLDFA [59] thus follow a controversial approach that, in line with the definitions of Section 2, we could call "ITL-via-TTL". That is, the authors of these papers have applied a TTL method to a dataset for testing the accuracy of ITL methods by (unfairly) assuming the test set $Te^U_{\mathcal{T}}$ to be an object set $Ob^U_{\mathcal{T}}$. From a methodological point of view, the comparison against ITL methods is unfair since the performance of a TTL method is tailored to (i.e., optimized for) the object set $Ob^U_{\mathcal{T}}$, which is assumed to be unavailable for a proper ITL method. From a conceptual point of view, the goals that ITL and TTL methods pursue are not comparable either, since a TTL method does not necessarily learn a general hypothesis, as a true ITL method is instead expected to.

5 FROM INDUCTION TO TRANSDUCTION: TWO EMPIRICAL CASES

Up to now we have commented on the fundamental differences between ITL methods and TTL methods. In order to quantify the impact of these differences in terms of effectiveness, we generate transductive variants of two representative inductive transfer learning methods, *Structural Correspondence Learning* (SCL) [7, 43] and *Distributional Correspondence Indexing* (DCI) [34] (Sections 5.1 and 5.2), and we empirically evaluate the difference in performance between the inductive and the transductive versions (Section 5.3). We have chosen SCL and DCI for several reasons. First, SCL and DCI cater for both cross-domain adaptation and cross-lingual adaptation, which allows us to evaluate the impact of the above differences on a variety of transfer learning scenarios. Second, the code implementing SCL and DCI has been made publicly available by their authors, which eases

our task. (Implementation details are given in Section 5.3.) Third, SCL and DCI are among the most representative inductive transfer learning methods in the text classification literature.

While the former method relies on the Structural Correspondence Learning paradigm already discussed in Section 4.2, DCI relies on the "distributional hypothesis"[12] to generate a vector space specifically devised for knowledge transfer. In this vector space, words that play similar roles across domains are close to each other (e.g., word "read" from the book domain is close to word "listen" from the music domain, as both play analogous roles in their respective domains) since word vectors are defined with respect to the *pivot* words (frequent and highly predictive words that behave similarly across domains; example pivot words are "excellent" or "poor" in *any* domain having to do with product reviews). Both methods consist of two main phases: representation (Section 5.1) and classification (Section 5.2), which we describe in the next sections.

The transductive variants we generate for the (originally inductive) SCL and DCI methods serve the sole purpose of evaluating whether the differences in performance between inductive and transductive versions is significant or not; these transductive variants are rather obvious, and should not be considered part of our original contribution.

5.1 Document Representation

SCL and DCI bridge the gap between the source domain $\mathcal{S} = (\mathbf{x}_{\mathcal{S}}, F_{\mathcal{S}}, \phi_{\mathcal{S}})$ and target domain $\mathcal{T} = (\mathbf{x}_{\mathcal{T}}, F_{\mathcal{T}}, \phi_{\mathcal{T}})$, where $F_{\mathcal{S}} = \mathcal{R}^m$ and $F_{\mathcal{T}} = \mathcal{R}^n$ are two vector spaces (into which documents are mapped via, e.g., tf-idf weighting), by working out additional representation functions $\phi'_{\mathcal{S}} : \mathcal{R}^m \rightarrow \mathcal{R}^k$ and $\phi'_{\mathcal{T}} : \mathcal{R}^n \rightarrow \mathcal{R}^k$ that generate document representations in a shared vector space $\mathcal{R}^k$, whose dimensions are the above-mentioned pivot words. Here, m and n are the number of distinct features (i.e., the vocabulary sizes) in the source and target domains, respectively, and k is a user-defined parameter which specifies the number of dimensions of the shared space, i.e., the number of pivot words.

The representation functions are implemented as linear mappings

$$\begin{aligned} \phi'_{\mathcal{S}}(\mathbf{x}) &= \mathbf{x}^{\top} \cdot \mathbf{Z}_{\mathcal{S}} && \mathbf{x} \in \mathcal{R}^m,\ \mathbf{Z}_{\mathcal{S}} \in \mathcal{R}^{m,k} \\ \phi'_{\mathcal{T}}(\mathbf{x}) &= \mathbf{x}^{\top} \cdot \mathbf{Z}_{\mathcal{T}} && \mathbf{x} \in \mathcal{R}^n,\ \mathbf{Z}_{\mathcal{T}} \in \mathcal{R}^{n,k} \end{aligned}$$

where $\mathbf{Z}_{\mathcal{S}}$ and $\mathbf{Z}_{\mathcal{T}}$ are the projection matrices whose rows are the k-dimensional word profiles (or embeddings).[13] In a domain $\mathcal{D}$, entry $\mathbf{Z}_{ij}$ of projection matrix $\mathbf{Z}$ quantifies the degree of correlation between the i-th word in the original vector space and the j-th pivot word.

SCL and DCI implement different criteria for computing this correlation. In SCL, the correlations between the words and a given pivot in a domain $\mathcal{D}$ are measured by solving a structural (classification) problem in which all words are used as features to predict the presence or absence of the pivot in a sample of documents from the domain distribution $P_{\mathcal{D}}$. The correlation of each word with respect to the pivot is thus taken to be the corresponding coefficient of the hyperplane that defines the separation. The projection matrix $Z_{\mathcal{D}} \in \mathcal{R}^{n,k}$ is defined as the k principal components of a matrix in $\mathcal{R}^{n,p}$ containing, as its columns, all p hyperplanes, with p the number of pivots. When the feature spaces $F_{\mathcal{S}}$ and $F_{\mathcal{T}}$ are not disjoint (that is, when we are not tackling cases of cross-lingual adaptation), SCL replaces the original vector with a concatenation of the vector and the projection [7], i.e., $\mathbf{x}' \leftarrow [\mathbf{x}; \phi'_{\mathcal{D}}(\mathbf{x})]$. However, we have obtained much better results by normalising each component before concatenating them. Specifically, we reduce the dimensionality

[12] The distributional hypothesis states that words with similar meanings tend to co-occur in the same contexts [21].

[13] Word profiles that SCL and DCI generate are indeed essentially word embeddings (low-dimensional and dense vectorial representations of words). However, they are generated by means of simple operations on the co-occurrence matrices, and are not the products of any neural procedure.

of $\mathbf{x}$ from n to k, in order to match that of $\phi'_{\mathcal{D}}(\mathbf{x})$, via principal component analysis, and we then L2-normalize each component before concatenating them.

In DCI, the correlation $\mathbf{Z}_{ij}$ is defined in terms of "distributional correspondence" between the i-th word and the j-th pivot, and is computed via a *distributional correspondence function*[14] (DCF) f using a sample of documents from the domain distribution $P_{\mathcal{D}}$. Each profile dimension is standardized so that the columns of $\mathbf{Z}$ have zero mean and unit variance. Note that, differently from SCL, in DCI it holds that $k = p$, since the dimensionality of the matrix is not reduced. In this work we adopt cosine as the DCF since it outperformed all other DCFs in the experiments reported in [34, 35].

We also use the same pivot selection strategy used in SCL [7, 43] and DCI [34] (a strategy that has its roots in the principles exposed in [8]), i.e., we select pivots by first filtering out words that are not frequent enough, and then removing from the remaining words the ones that are not discriminating enough (according to the mutual information between the word and the label, as estimated on the training set $Tr^L_{\mathcal{S}}$). In the cross-lingual case, pivot selection involves a word-translation oracle, i.e., a mapping from source words to target words (see [43]).

The projections $\mathbf{Z}_{\mathcal{S}}$ and $\mathbf{Z}_{\mathcal{T}}$ are learnt from documents-by-words matrices of tf-idf normalised weights. These matrices should be as large as possible in order to effectively capture the distributional properties of the words. This means that, in scenarios in which the unlabelled sets $Tr^U_{\mathcal{S}}$ and $Tr^U_{\mathcal{T}}$, of sizes q and r, are available, we first represent them as matrices $\mathbf{Tr}^U_{\mathcal{S}} \in \mathcal{R}^{q,m}$ and $\mathbf{Tr}^U_{\mathcal{T}} \in \mathcal{R}^{r,n}$ and then compute

$$\mathbf{Z}_{\mathcal{S}} = \psi(\mathbf{Tr}^U_{\mathcal{S}}, \mathbf{Tr}^U_{\mathcal{S}}, \vec{p})$$
$$\mathbf{Z}_{\mathcal{T}} = \psi(\mathbf{Tr}^U_{\mathcal{T}}, \mathbf{Tr}^U_{\mathcal{T}}, \vec{p})$$

where ψ is either SCL or DCI, and where $\vec{p}$ is the list of pivot words (properly translated to the target language in cases of cross-lingual adaptation). In transductive settings where unlabelled sets are not available, $\mathbf{Z}_{\mathcal{S}}$ and $\mathbf{Z}_{\mathcal{T}}$ are directly modelled on the training samples in $Tr^L_{\mathcal{S}}$ and in the object samples in $Ob^U_{\mathcal{T}}$, of sizes q' and r' (properly converted into matrices $\mathbf{Tr}^L_{\mathcal{S}} \in \mathcal{R}^{q',m}$ and $\mathbf{Ob}^L_{\mathcal{T}} \in \mathcal{R}^{r',n}$), as

$$\mathbf{Z}_{\mathcal{S}} = \psi(\mathbf{Tr}^L_{\mathcal{S}}, \mathbf{Tr}^L_{\mathcal{S}}, \vec{p})$$
$$\mathbf{Z}_{\mathcal{T}} = \psi(\mathbf{Ob}^U_{\mathcal{T}}, \mathbf{Ob}^U_{\mathcal{T}}, \vec{p})$$

5.2 Learning and Classification

In the transductive modality both $\phi'_{\mathcal{S}}$ and $\phi'_{\mathcal{T}}$ have to be invoked on $Tr^L_{\mathcal{S}}$ and $Ob^U_{\mathcal{T}}$ in order to generate (labelled and unlabelled) representations in the shared space before training the transductive classifier. This is required because the transductive classifier directly outputs labels for the elements in $Ob^U_{\mathcal{T}}$ as part of the learning procedure (the transductive step).

In the inductive settings, SCL and DCI first use $\phi'_{\mathcal{S}}$ to represent the training documents in $Tr^L_{\mathcal{S}}$ to train the classifier (the inductive step), while $\phi'_{\mathcal{T}}$ is invoked only at testing time in order to classify the documents in $Te^U_{\mathcal{T}}$ (the deductive step).

5.2.1 Transductive SVMs. The underlying machine learning algorithm we use for the transductive versions of SCL and DCI[15] is Transductive Support Vector Machines (TSVM) [25] with soft margins, that assign labels for elements in the object set as part of the learning process. TSVMs implement

[14] DCFs are real-valued functions that quantify the deviation in "correspondence" between two words with respect to the correspondence that is expected due to chance.

[15] The non-transductive learners used in experiments are detailed in Section 5.3.

transduction by attempting to maximize the margin of the hyperplane that separates both the training and the unlabelled data (instead of the training data alone, as for inductive SVMs). For the TSVMs we have used the linear kernel, which has consistently delivered good accuracy in text classification applications so far [24].

The transductive SVM classification problem is stated as the structural risk minimization problem

$$\begin{aligned} \text{Minimize over} \quad & y_1^*, \ldots, y_k^*, \vec{w}, b, \xi_1, \ldots, \xi_n, \xi_1^*, \ldots, \xi_k^* \\ & \frac{1}{2}||\vec{w}||^2 + C \sum_{i=1}^{n} \xi_i + C^* \sum_{j=1}^{k} \xi_j^* \\ \text{subject to} \quad & \forall_{i=1}^{n} : y_i(\vec{w} \cdot \vec{x}_i + b) \geq 1 - \xi_i \\ & \forall_{j=1}^{k} : y_j^*(\vec{w} \cdot \vec{x}_j^* + b) \geq 1 - \xi_j^* \\ & \forall_{i=1}^{n} : \xi_i > 0 \\ & \forall_{j=1}^{k} : \xi_j^* > 0 \end{aligned}$$

where y_i^* are the binary decisions for the object documents $\vec{x}_j^*$; $\vec{w}, b$ are the parameters (hyperplane and bias) of the separation functional; ξ_i are the slack variables for the labelled examples; ξ_i^* are the slack variables for the unlabelled examples; and C and C^* are two hyperparameters controlling the trade-off between training error and margin for the labelled and unlabelled examples, respectively.

Note that y_j^* are the predicted labels for the object documents and, though the algorithm actually produces a classifier, defined as $h(x) = sign(\vec{w} \cdot \vec{x} + b)$, this classifier is not used to (re)classify the object documents. Indeed, there is no guarantee that the label attributed in the transductive step coincides with the label that classifier h would assign, that is, $h(\mathbf{x}_j) = y_j^*$ does not necessarily hold; specifically, it is not true for the documents $\mathbf{x}_j$ for which $\xi_j^* > 1$.

The implementation of TSVMs we have used is the one made available by Thorsten Joachims in his SVM^{light} package[16] [25].

5.3 Experiments

In this section, we desribe the results of experiments that compare the transductive versions of SCL (hereafter: TSCL) and DCI (hereafter: TDCI) against the original inductive ones (hereafter: ISCL and IDCI). The experimental settings we explore account for (i) classification by sentiment and by topic, (ii) inductive settings and transductive settings, and (iii) cross-domain and cross-lingual adaptation. In doing so, we deliberately apply TSCL and TDCI also in environments in which the use of transductive techniques is questionable: the aim of this experimentation is thus that of providing empirical evidence that confounding the inductive and transductive paradigms can indeed bring unfair benefits to transductive approaches in terms of performance against their inductive competitors, and that this improvement is statistically significant. Somehow unconventionally, this experimentation does not aim at setting a new best performance for a given dataset since, as will become clear, some of the current best results from the literature have been obtained, as we argue, unfairly.

The datasets we consider include:

- Reuters-21578[17] : a set of news stories produced by Reuters in 1987. Documents in the collection are assigned to 5 top-level classes; among these, classes *orgs*, *people*, *places* have been considered for transfer learning experiments in previous work, leading to three binary distinctions: *orgs vs. people*, *orgs vs. places*, and *people vs. places*.

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

[17] http://www.cse.ust.hk/TL/dataset/Reuters.zip

Table 2. Characteristics of the datasets used for *transductive* transfer learning. Tr^U indicates the sample that, in the experiments, sometimes plays the role of $Tr^U_{\mathcal{S}}$ and sometimes plays the role of $Tr^U_{\mathcal{T}}$. When the cardinality of a sample is indicated as an interval, this indicates how this cardinality varies across the various tasks (indicated in column "Tasks"). The last column indicates the works where this dataset was first used.

Dataset	Classification	Adaptation	Tasks	$\lvert Tr^L_{\mathcal{S}}\rvert$	$\lvert Tr^U\rvert$	$\lvert Ob^U_{\mathcal{T}}\rvert$	Ref.
Reuters-21578	by topic	cross-domain	3	[1046,1210]	0	[1016,1239]	[13, 14]
SRAA	by topic	cross-domain	2	8000	0	8000	[13, 14]
20Newsgroups	by topic	cross-domain	6	[3561,4900]	0	[3374,4904]	[13, 14]

Table 3. Characteristics of the datasets used for *inductive* transfer learning. Notational conventions are as in Table 2.

Dataset	Classification	Adaptation	Tasks	$\lvert Tr^L_{\mathcal{S}}\rvert$	$\lvert Tr^U\rvert$	$\lvert Te^U_{\mathcal{T}}\rvert$	Ref.
MDS	by sentiment	cross-domain	12	1600	[3586,5945]	400	[7]
Webis-CLS-10	by sentiment	cross-lingual	9	2000	[9358,50000]	2000	[42]

- SRAA[18] : a set of Usenet posts about *simulated autos*, *simulated aviation*, *real autos*, and *real aviation*. The pairs of classes *real vs. simulated*, and *auto vs. aviation* define two cross-domain transfer learning tasks.
- 20Newsgroups[19] : a set of posts from 20 Usenet discussion groups. Previous transfer learning experiments reported for this dataset considered all binary combinations for the 4 most frequent top-level classes in the dataset (*comp*, *sci*, *rec*, *talk*). We adopted the setup proposed by [13, 14]), in which six datasets are defined by selecting a pair of top categories for each dataset. One top category of the pair acts as the positive category and the other as the negative category (e.g., *comp vs. sci*, *rec vs. talk*). The subcategories of a top category are then considered as the different domains on which the transfer learning process is applied (e.g., *sci.crypt*, *sci.med* for the top category *sci*).
- MDS[20] : a set of Amazon product reviews for the four domains *Books*, *DVD*, *Electronics*, and *Kitchen appliances*. The preprocessed version contains bags of uni- and bi-grams, and is labelled according to binary sentiment polarity. There are 2,000 labelled instances for each domain, which are to be split in 5 folds according to [7] for performance evaluation. This means that each reported accuracy value is an average across 5 experiments, each of which considers 1600 training examples from the source domain and 400 test examples from the target domain. This is the only dataset in which accuracy scores are computed via k-fold cross-validation.
- Webis-CLS-10[21] : a cross-lingual collection for sentiment classification consisting of positive and negative Amazon product reviews for three domains (*Books*, *DVD*, *Music*) in four languages (*English*, *German*, *French*, *Japanese*). English is always used as the source language, following [42].

Tables 2 and 3 display additional characteristics of the datasets; see also Section 4.2 for further details.

[18] http://people.cs.umass.edu/~mccallum/data/sraa.tar.gz
[19] http://qwone.com/~jason/20Newsgroups/
[20] http://www.cs.jhu.edu/~mdredze/datasets/sentiment/
[21] https://www.uni-weimar.de/en/media/chairs/computer-science-department/webis/data/corpus-webis-cls-10/

As the evaluation measure we adopt "vanilla" accuracy, i.e.,

$$A = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

where TP, FP, FN, TN are the numbers of true positives, false positives, false negatives, true negatives, as from the standard 2×2 contingency table. Adopting vanilla accuracy (the metric of choice in previous related work) as the evaluation measure is perfectly reasonable since the datasets are balanced.

We have implemented TSCL by adapting the publicly available implementation of CL-SCL [43] made available as part of the NUT package.[22] Apart from bypassing the translation of pivot words when the source and target languages are the same, and apart from implementing the normalised concatenation described in Section 5.1, the main change we have made concerns the replacement of the original learning device in charge of the final predictions with SVM^{light}.[23] TSCL is thus obtained by having SVM^{light} operate in transductive mode and making the object set (with labels omitted) available at training time. In previous literature, SCL has been tested on Webis-CLS-10 [43] and on MDS [7]. For Webis-CLS-10 we thus adopt the configuration proposed in [43] for this dataset, that uses $p = 450$ pivots, $k = 100$ principal components of the shared space, and discards pivot candidates appearing in fewer than $\phi = 30$ support documents. We explore this and other configurations that have been proposed in past literature for MDS. In particular, we also test the configuration of this dataset proposed in [7] (that consists of setting $p = 100$, $k = 50$, and $\phi = 5$), but we also explore other configurations that worked well for DCI (see [35]) and that consider a higher number of pivots (up to $p = 1000$), and thus a higher dimensionality (up to $k = 1000$). As done in [43] for Webis-CLS-10, we choose the configuration that works best for the first task of the dataset (Books-DVD, as typically encountered in most papers); we end up using $p = 1000$, $k = 1000$, and $\phi = 5$. We also report results for ISCL and TSCL on the Reuters-21578, SRAA, and 20Newsgroups datasets, for which, to the best of our knowledge, no published results for SCL existed so far. Similarly, we choose the configuration that yielded the best result in one of the tasks (we choose *comp vs. sci* – the first task from the dataset with more tasks), which results in setting $p = 1000$, $k = 100$, and $\phi = 5$. We do not consider configurations involving $p > 450$ in Webis-CLS-10 since translating pivots is assumed to incur a cost; $p = 450$ has been agreed upon in past literature as a reasonable cost-effective tradeoff, and setting $k > 100$ did not yield any better results.

We have implemented TDCI by adapting the PyDCI [35] package[24] to use SVM^{light} as the learning device, in place of the *scikit-learn* implementation of SVMs (which does not cater for transduction). Those modifications are now integrated as part of the PyDCI package. We set the number of pivots to $p = 450$ for Webis-CLS-10 following [42], and to $p = 1000$ for the other datasets as proposed in [35].

Since we have adopted a different learner, the accuracy values we report here do not coincide with those previously reported for SCL in [7, 43], nor with those reported for DCI in [35]. Although no significant variations exist in the latter case, the differences between SCL and ISCL turn out to be more pronounced.

We set the parameters C and C^* controlling the trade-off between training error and margin, to the SVM^{light} default values in all cases.

[22] https://github.com/pprett/nut

[23] Note that the modification we have made to the NUT software only affects the final classification, and not the generation of the vector representations in the shared space. These representations depend on the predictions of a set of classifiers that are tasked to solve the structural problems. The learners we used for solving these intermediate structural problems still rely on the implementation of Prettenhofer and Stein's truncated stochastic gradient descent variant made available at https://github.com/pprett/bolt.

[24] https://github.com/AlexMoreo/pydci

Table 4. Cross-domain adaptation on the Reuters-21578 (rows 1–3), SRAA (rows 4–5), and 20Newsgroups (rows 6–11) datasets. Symbol "♦" indicates that the method in the corresponding column has access to the object set.

		♦		♦	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦	♦		♦	
Dataset	**ISVM**	**TSVM**	**Upper**	**SFA** [37]	**CoCC** [13]	**TrAdaB** [14]	**TrAdaB(T)** [14]	**TPLSA** [60]	**TLDA** [55]	**CDSC** [30]	**MTrick** [65]	**TCA** [27]	**PSCCLDA** [4]	**ISCL**	**IDCI**	**TSCL**		**TDCI**	
orgs vs. places	.714	.742	.924	.683	.680	.720	.752	.653	-	.682	**.768**	.730	.742	.695	.739	.756	(+8.7%)	.767	(+3.8%)
orgs vs. people	.742	.792	.921	.671	.764	.685	.696	.763	-	.768	.808	.792	.807	.766	.802	.789	(+3.0%)	**.817**	(+1.9%)
people vs. places	.592	.614	.923	.506	**.826**	.784	.821	.805	-	.798	.690	.626	.690	.601	.604	.636	(+5.9%)	.668	(+10.6%)
real vs. simulated	.684	.828	.962	-	.880	.881	.898	.889	-	.812	-	-	-	.719	.862	.811	(+12.9%)	**.912**	(+5.8%)
auto vs. aviation	.752	.880	.969	-	.932	.904	**.962**	.947	-	.880	-	-	-	.780	.930	.880	(+12.9%)	.941	(+1.2%)
comp vs. sci	.713	.832	.982	.830	.870	-	-	**.989**	.939	.902	-	.891	.900	.704	.784	.773	(+9.7%)	.869	(+10.8%)
rec vs. talk	.778	.967	.995	.854	.965	.920	**.979**	.977	.925	.908	.950	.962	.962	.746	.940	.868	(+16.3%)	.966	(+2.8%)
rec vs. sci	.807	.937	.994	.885	.945	.903	**.987**	.951	.912	.876	.955	.879	.955	.785	.926	.833	(+6.0%)	.969	(+4.6%)
sci vs. talk	.790	.905	.990	.854	.946	.875	.925	**.962**	.907	.956	.937	.940	.947	.776	.894	.830	(+6.9%)	.915	(+2.3%)
comp vs. rec	.869	.904	.992	.939	.958	-	-	.951	.882	.958	-	.940	.958	.904	**.966**	.885	(-2.2%)	.905	(-6.3%)
comp vs. talk	.914	.885	.994	.971	**.980**	-	-	.977	.948	.976	-	.967	.967	.953	.979	.884	(-7.3%)	.885	(-9.6%)
Average	.784	.875	.979	-	.886	-	-	.897	-	.865	-	-	-	.766	.888	.813	(+6.1%)	**.898**	(+1.2%)

We compare the performance of TDCI with most of the baselines discussed in Section 4.[25] For TrAdaBoost [14] we report results for TrAdaB (that uses SVM as the learner) and TrAdaB(T) (that uses instead TSVM). Note that ISCL acts as an alternative implementation of CL-SCL in Webis-CLS-10 [43], and of SCL in MDS [7]. The accuracy scores for the baseline methods are taken from the original publications. In all cases, we also report results for (i) ISVM, an (inductive) SVM that simply classifies the target documents without carrying out any sort of adaptation; (ii) TSVM, a transductive SVM that trains on the source domain using the target object set as unlabelled examples (again, without any adaptation); and (iii) Upper, a SVM that trains and tests in the target domain; we report the accuracy of a 5-fold cross validation in the object set. In Webis-CLS-10, we also report (iv) CL-MT, an inductive SVM that trains on the source English documents and tests on translations of the non-English target documents (we used the translations made available by [42]). SVM^{light} is used to generate the classifier in all these baselines

Tables 4, 5, 6 report the accuracy scores of the methods discussed across the various datasets. Boldface indicates the best score for each dataset; the accuracy scores of the transductive variants TSCL and TDCI are listed together with the (percentage of) relative accuracy improvement with respect to the inductive ISCL and IDCI counterparts (positive is better). Methods that access (thus, optimize for) the object set are marked with the "♦" symbol. This symbol is thus used to establish which systems can be legitimately compared with each others.

Table 6 is the one mixing more transductive and inductive methods. It looks clear that methods belonging to the transductive group (those marked with the "♦" symbol) tend to obtain higher scores than methods from the inductive group (the difference in performance is indeed statistically significant according to a two-sided t-test for means of two independent samples at a confidence level of $\alpha = 0.05$ – the trivial baselines ISVM, TSVM, Upper and CL-MT were left out of the test for obvious reasons).

Unsurprisingly, the transductive variants of SCL and DCI bring about a considerable gain in most cases (up to a relative improvement of 16.5% of accuracy in Japanese-DVD for SCL and 10.9% in *comp vs. sci* for DCI). There are a few exceptions though, which in some cases (*comp vs. rec* and *comp vs. talk* in 20Newsgroups, and Japanese-Books in Webis-CLS-10) are particularly pronounced.

[25] We have left TransDANN [2] out since their results on MDS display much lower figures, likely because the authors have used a different version of the dataset.

Table 5. Cross-domain adaptation on the MDS dataset.

			♦											♦			♦		♦	
Source	**Target**	**ISVM**	**TSVM**	**Upper**	**SCL** [7]	**SFA** [37]	**TCT** [63]	**SDA** [18]	**CDFL** [61]	**TrAdaB** [22]	**DANN** [17]	**AMN** [29]	**TKC** [23]	**ISCL**	**IDCI**	**TSCL**		**TDCI**	
Books	DVD	.802	.804	.846	.758	.814	.818	**.844**	.826	.796	.829	.815	.840	.796	.822	.834	(+4.8%)	.823	(+0.1%)
	Electronics	.733	.741	.874	.759	.725	.757	.806	.802	.749	.804	.808	.766	.740	.823	.790	(+6.7%)	**.829**	(+0.7%)
	Kitchen	.772	.779	.908	.789	.788	.789	.804	.828	.778	**.843**	.815	.796	.794	.841	.832	(+4.8%)	.836	(-0.6%)
DVD	Books	.800	.798	.842	.797	.775	.792	.807	.809	.747	.825	.820	**.849**	.788	.825	.828	(+5.1%)	.826	(+0.1%)
	Electronics	.754	.780	.874	.741	.767	.778	.802	.809	.759	.809	.800	.771	.725	.847	.804	(+11.0%)	**.849**	(+0.2%)
	Kitchen	.776	.783	.908	.814	.808	.812	.835	.828	.757	.849	.835	.809	.768	.848	.733	(+8.5%)	**.851**	(+0.4%)
Electronics	Books	.715	.712	.842	.754	.757	.759	.768	.750	.691	.774	.780	.785	.722	.820	.741	(+2.6%)	**.824**	(+0.5%)
	DVD	.742	.739	.846	.762	.772	.773	.777	.765	.718	.781	.778	.796	.738	.800	.773	(+4.8%)	**.802**	(+0.3%)
	Kitchen	.858	.861	.908	.859	.868	.863	**.902**	.879	.837	.881	.900	.870	.848	.876	.889	(+4.8%)	.871	(-0.6%)
Kitchen	Books	.737	.731	.842	.686	.748	.748	.724	.748	.706	.718	.793	.766	.730	.803	.760	(+4.2%)	**.807**	(+0.5%)
	DVD	.750	.746	.846	.769	.766	.785	.803	**.876**	.744	.789	.803	.764	.741	.797	.786	(+6.1%)	.801	(+0.5%)
	Electronics	.840	.851	.874	.868	.851	.856	.872	.861	.831	.856	.820	.864	.818	.855	**.882**	(+7.8%)	.856	(+0.1%)
	Average	.773	.777	.868	.780	.786	.794	.812	.808	.759	.813	.814	.806	.767	.830	.812	(+5.9%)	**.831**	(+0.1%)

Table 6. Cross-lingual adaptation on the Webis-CLS-10 dataset.

			♦				♦	♦		♦		♦			♦		♦	
Target Language	Domain	**ISVM**	**TSVM**	**Upper**	**CL-MT**	**CL-SCL** [43]	**SSMC** [57]	**CL-SLF** [62]	**BM** [58]	**BiDRL** [64]	**Bi-PV** [41]	**CLDFA** [59]	**ISCL**	**IDCI**	**TSCL**		**TDCI**	
German	Books	.516	.610	.863	.808	.833	.819	.799	.825	.841	.795	.840	.758	.849	.841	(+11.0%)	**.857**	(+0.9%)
	DVD	.568	.621	.832	.800	.809	.823	.819	.815	**.841**	.786	.831	.717	.832	.810	(+13.0%)	.834	(+0.2%)
	Music	.568	.633	.845	.790	.829	.813	.796	.830	.847	.825	.790	.774	.852	.834	(+7.7%)	**.860**	(+0.9%)
French	Books	.527	.695	.842	.820	.813	.831	.826	.825	**.844**	.843	.834	.803	.817	.823	(+2.4%)	.831	(+1.7%)
	DVD	.541	.702	.849	.794	.804	.827	.827	.819	.836	.796	.826	.744	.836	.809	(+8.7%)	**.847**	(+1.3%)
	Music	.558	.632	.872	.764	.781	.805	.802	.816	.826	.801	**.833**	.770	.820	.798	(+3.6%)	.829	(+1.1%)
Japanese	Books	.499	.419	.804	.692	.770	.738	.735	.709	.732	.718	.774	.754	**.784**	.672	(-10.9%)	.754	(-3.8%)
	DVD	.503	.535	.808	.722	.764	.776	.771	.746	.768	.754	.805	.677	.801	.789	(+16.5%)	**.816**	(+1.9%)
	Music	.509	.597	.832	.714	.773	.775	.768	.765	.788	.755	.765	.719	.812	.808	(+12.4%)	**.831**	(+2.3%)
	Average	.532	.605	.838	.767	.797	.801	.794	.794	.813	.786	.811	.746	.823	.798	(+7.0%)	**.829**	(+0.8%)

Note that in these cases the inductive variant performed very well (actually outperforming all other competitors in the case of IDCI), which may be an indication that transduction might come at a risk (this is indeed confirmed by the relative performance between the ISVM and TSVM baselines in those cases).

The smallest improvements are achieved in the MDS dataset for TDCI. Probably, the reason is that the contribution of the object set is limited since in this case a 5-fold cross-validation is adopted for evaluation; this means that in each experiment only 400 object documents documents are observed, while the number of object documents observed during training is comparatively higher in other datasets (see Table 3).

TDCI outperforms on average all other competitors in the transductive setting (Table 4) even considering the *comp vs. rec* and *comp vs. talk* anomaly described above. A direct comparison between the performance of TDCI and the baselines in the inductive settings (Tables 5 and 6) is to be taken with a grain of salt (that, is indeed a core claim of this paper) since the baselines are assumed to be inductive (though we argued in Section 4 that some of them are actually transductive). In particular, SSMC, CL-SLF, BiDRL, CLDFA, and TKC access the test data during training and, not surprisingly, most of them rank on top of the inductive transfer learning competitors in terms of performance.

Finally, a Wilcoxon signed-rank test reveals the differences in performance between ISCL vs TSCL and between IDCI vs TDCI to be statistically significant at confidence level 0.05 (with p-values of $2.4E^{-12}$ and $5.2E^{-3}$, respectively), and at a much higher confidence level (p-values of $4.0E^{-13}$ and $3.5E^{-5}$) if we discard the anomalous cases.

6 CONCLUSIONS

Quite obviously, the accuracy of a classifier improves when the learner knows at training time the set of documents the classifier will later be evaluated on. Transductive approaches focus on devising ways of improving the prediction of labels in cases when the specific object sets is available and known in advance. This improvements comes at the cost of sacrificing the generalization ability that inductive approaches show off. Inductive and transductive approaches thus pursue radically different goals, and are thus not interchangeable at will (they are only interchangeable in lab experiments, by wrongly assuming the test set to play the role of an object set). This is a major difference that has largely been overlooked in the transfer learning literature, fostered by a misuse of terminology in the field and leading to unfair comparisons. We have proposed a clarification of terminology, and shown empirical evidences that there was a need for it. To this aim, we have produced a transductive variant of two representative inductive methods, SCL and DCI that we used to deliberately reproduce a wrong experimental practice (imitating past evaluations in the field), in which we compare the performance of TSCL and TDCI to their inductive counterparts in inductive transfer learning problems. The goal of this evaluation is to show evidence that confounding terminology may lead to unfair comparisons, and that the differences in performance can be statistically significant.

ACKNOWLEDGMENTS

The present work has been supported by the ARIADNEplus project, funded by the European Commission (Grant 823914) under the H2020 Programme INFRAIA-2018-1, by the SoBigdata++ project, funded by the European Commission (Grant 871042) under the H2020 Programme INFRAIA-2019-1, and by the AI4Media project, funded by the European Commission (Grant 951911) under the H2020 Programme ICT-48-2020 . The authors' opinions do not necessarily reflect those of the European Commission.

REFERENCES

[1] Andrew Arnold, Ramesh Nallapati, and William W. Cohen. 2007. A comparative study of methods for transductive transfer learning. In *Workshops Proceedings of the 7th IEEE International Conference on Data Mining (ICDM 2007)*. Omaha, US, 77–82. https://doi.org/10.1109/ICDMW.2007.109

[2] Amar P. Azad, Dinesh Garg, Priyanka Agrawal, and Arun Kumar. 2018. Deep domain adaptation under deep label scarcity. *arXiv preprint arXiv:1809.08097* (2018).

[3] Mohammad T. Bahadori, Yan Liu, and Dan Zhang. 2011. Learning with minimum supervision: A general framework for transductive transfer learning. In *Proceedings of the 11th IEEE International Conference on Data Mining (ICDM 2011)*. Vancouver, CA, 61–70. https://doi.org/10.1109/ICDM.2011.92

[4] Yang Bao, Nigel Collier, and Anindya Datta. 2013. A partially supervised cross-collection topic model for cross-domain text classification. In *Proceedings of the 22nd ACM International Conference on Conference on Information and Knowledge Management (CIKM 2013)*. San Francisco, US, 239–248. https://doi.org/10.1145/2505515.2505556

[5] Vahid Behbood, Jie Lu, and Guangquan Zhang. 2011. Long term bank failure prediction using fuzzy refinement-based transductive transfer learning. In *Proceedigns of the 20th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*. Taipei, TW, 2676–2683. https://doi.org/10.1109/FUZZY.2011.6007633

[6] Giacomo Berardi, Andrea Esuli, and Fabrizio Sebastiani. 2014. Optimising human inspection work in automated verbatim coding. *International Journal of Market Research* 56, 4 (2014), 489–512. https://doi.org/10.2501/ijmr-2014-032

[7] John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL 2007)*. Prague, CZ, 440–447.

[8] John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the 4th Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*. Sydney, AU, 120–128. https://doi.org/10.3115/1610075.1610094

[9] Danushka Bollegala, Tingting Mu, and John Yannis Goulermas. 2016. Cross-domain sentiment classification using sentiment-sensitive embeddings. *IEEE Transactions on Knowledge and Data Engineering* 28, 2 (2016), 398–410. https://doi.org/10.1109/tkde.2015.2475761

[10] Danushka Bollegala, David Weir, and John Carroll. 2013. Cross-domain sentiment classification using a sentiment-sensitive thesaurus. *IEEE Transactions on Knowledge and Data Engineering* 25, 8 (2013), 1719–1731. https://doi.org/10.1109/tkde.2012.103

[11] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. 2006. A discussion of semi-supervised learning and transduction. In *Semi-Supervised Learning*, Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien (Eds.). The MIT Press, Cambridge, US, 457–462. https://doi.org/10.7551/mitpress/9780262033589.003.0025

[12] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. 2006. Introduction to semi-supervised learning. In *Semi-Supervised Learning*, Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien (Eds.). The MIT Press, Cambridge, US, 105–117. https://doi.org/10.7551/mitpress/9780262033589.003.0001

[13] Wenyuan Dai, Gui-Rong Xue, Qiang Yang, and Yong Yu. 2007. Co-clustering based classification for out-of-domain documents. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2007)*. San Jose, US, 210–219. https://doi.org/10.1145/1281192.1281218

[14] Wenyuan Dai, Qiang Yang, Gui-Rong Xue, and Yong Yu. 2007. Boosting for transfer learning. In *Proceedings of the 24th International Conference on Machine Learning (ICML 2007)*. Corvallis, US, 193–200. https://doi.org/10.1145/1273496.1273521

[15] Oscar Day and Taghi M. Khoshgoftaar. 2017. A survey on heterogeneous transfer learning. *Journal of Big Data* 4 (2017), Article 17 (1–42). https://doi.org/10.1186/s40537-017-0089-0

[16] Alexander Gammerman, Volodya G. Vovk, and Vladimir Vapnik. 1998. Learning by transduction. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI 1998)*. Madison, US, 148–155.

[17] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *Journal of Machine Learning Research* 17, 1 (2016), 2096–2030.

[18] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th International Conference on Machine Learning (ICML 2011)*. Bellevue, US, 513–520.

[19] Pablo González, Alberto Castaño, Nitesh V. Chawla, and Juan José del Coz. 2017. A review on quantification learning. *Comput. Surveys* 50, 5 (2017), 74:1–74:40. https://doi.org/10.1145/3117807

[20] Quanquan Gu and Jie Zhou. 2009. Learning the shared subspace for multi-task clustering and transductive transfer classification. In *Proceedings of the 9th IEEE International Conference on Data Mining (ICDM 2009)*. Miami, US, 159–168. https://doi.org/10.1109/ICDM.2009.32

[21] Zellig S. Harris. 1954. Distributional structure. *Word* 10, 23 (1954), 146–162. https://doi.org/10.1007/978-94-009-8467-7_1

[22] Xingchang Huang, Yanghui Rao, Haoran Xie, Tak-Lam Wong, and Fu Lee Wang. 2017. Cross-domain sentiment classification via topic-related TrAdaBoost. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI 2017)*. San Francisco, US, 4939–4940.

[23] Radu T. Ionescu and Andrei M. Butnaru. 2018. Transductive learning with string kernels for cross-domain text classification. In *Proceedings of the 25th International Conference on Neural Information Processing (ICONIP 2018)*. Siam Rep, KH, 484–496. https://doi.org/10.1007/978-3-030-04182-3_42

[24] Thorsten Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning (ECML 1998)*. Chemnitz, DE, 137–142. https://doi.org/10.1007/bfb0026683

[25] Thorsten Joachims. 1999. Transductive inference for text classification using support vector machines. In *Proceedings of the 16th International Conference on Machine Learning (ICML 1999)*. Bled, SL, 200–209.

[26] Matthew Lease, Gordon V. Cormack, An Thanh Nguyen, Thomas A. Trikalinos, and Byron C. Wallace. 2016. Systematic review is e-discovery in doctor's clothing. In *Proceedings of the SIGIR 2016 Medical Information Retrieval Workshop (MedIR 2016)*. Pisa, IT.

[27] Lianghao Li, Xiaoming Jin, and Mingsheng Long. 2012. Topic correlation analysis for cross-domain text classification. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI 2012)*. Toronto, CA, 998–1004.

[28] Zheng Li, Ying Wei, Yu Zhang, and Qiang Yang. 2018. Hierarchical attention transfer network for cross-domain sentiment classification. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI 2018)*. New Orleans, US.

[29] Zheng Li, Yu Zhang, Ying Wei, Yuxiang Wu, and Qiang Yang. 2017. End-to-end adversarial memory network for cross-domain sentiment classification. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017)*. Melbourne, AU, 2237–2243. https://doi.org/10.24963/ijcai.2017/311

[30] Xiao Ling, Wenyuan Dai, Gui-Rong Xue, Qiang Yang, and Yong Yu. 2008. Spectral-domain transfer learning. In *Proceedings of the 14th ACM International Conference on Knowledge Discovery and Data Mining (KDD 2008)*. Las Vegas, US, 488–496. https://doi.org/10.1145/1401890.1401951

[31] Zachary C. Lipton and Jacob Steinhardt. 2019. Research for practice: Troubling trends in machine-learning scholarship. *Commun. ACM* 62, 6 (2019), 45–53. https://doi.org/doi/10.1145/3316774

[32] Jie Lu, Vahid Behbood, Peng Hao, Hua Zuo, Shan Xue, and Guangquan Zhang. 2015. Transfer learning using computational intelligence: A survey. *Knowledge-Based Systems* 80 (2015), 14–23.

[33] Jose G. Moreno-Torres, Troy Raeder, Rocío Alaíz-Rodríguez, Nitesh V. Chawla, and Francisco Herrera. 2012. A unifying view on dataset shift in classification. *Pattern Recognition* 45, 1 (2012), 521–530. https://doi.org/10.1016/j.patcog.2011.06.019

[34] Alejandro Moreo, Andrea Esuli, and Fabrizio Sebastiani. 2016. Distributional correspondence indexing for cross-lingual and cross-domain sentiment classification. *Journal of Artificial Intelligence Research* 55 (2016), 131–163. https://doi.org/10.1613/jair.4762

[35] Alejandro Moreo, Andrea Esuli, and Fabrizio Sebastiani. 2018. Revisiting distributional correspondence indexing: A Python reimplementation and new experiments. arXiv:1810.09311 [cs.CL].

[36] Douglas W. Oard, Fabrizio Sebastiani, and Jyothi K. Vinjumur. 2018. Jointly minimizing the expected costs of review for responsiveness and privilege in e-discovery. *ACM Transactions on Information Systems* 37, 1, Article 11 (2018), 11:1–11:35 pages. https://doi.org/10.1145/3268928

[37] Sinno J. Pan, Xiaochuan Ni, Jian-Tao Sun, Qiang Yang, and Zheng Chen. 2010. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th International Conference on the World Wide Web (WWW 2010)*. Raleigh, US, 751–760. https://doi.org/10.1145/1772690.1772767

[38] Sinno J. Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering* 22, 10 (2010), 1345–1359. https://doi.org/10.1109/tkde.2009.191

[39] Weike Pan, Erheng Zhong, and Qiang Yang. 2012. Transfer learning for text mining. In *Mining Text Data*, Charu C. Aggarwal and ChengXiang Zhai (Eds.). Springer, Heidelberg, DE, 223–258. https://doi.org/10.1007/978-1-4614-3223-4_7

[40] Novi Patricia and Barbara Caputo. 2014. Learning to learn, from transfer learning to domain adaptation: A unifying perspective. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2014)*. Columbus, US, 1442–1449. https://doi.org/10.1109/CVPR.2014.187

[41] Hieu Pham, Thang Luong, and Christopher Manning. 2015. Learning distributed representations for multilingual text sequences. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*. Denver, US, 88–94. https://doi.org/10.3115/v1/w15-1512

[42] Peter Prettenhofer and Benno Stein. 2010. Cross-language text classification using structural correspondence learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL 2010)*. Uppsala, SE, 1118–1127.

[43] Peter Prettenhofer and Benno Stein. 2011. Cross-lingual adaptation using structural correspondence learning. *ACM Transactions on Intelligent Systems and Technology* 3, 1 (2011), Article 13. https://doi.org/10.1145/2036264.2036277

[44] Brian Quanz and Jun Huan. 2009. Large margin transductive transfer learning. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM 2009)*. Hong Kong, CN, 1327–1336. https://doi.org/10.1145/1645953.1646121

[45] Joaquin Quiñonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence (Eds.). 2009. *Dataset shift in machine learning*. The MIT Press, Cambridge, US. https://doi.org/10.7551/mitpress/9780262170055.001.0001

[46] Marcus Rohrbach, Sandra Ebert, and Bernt Schiele. 2013. Transfer learning in a transductive setting. In *Proceedings of the 27th Annual Conference on Neural Information Processing Systems (NIPS 2013)*. Lake Tahoe, US, 46–54.

[47] Ozan Sener, Hyun Oh Song, Ashutosh Saxena, and Silvio Savarese. 2016. Learning transferrable representations for unsupervised domain adaptation. In *Proceedings of the 29th Conference on Advances in Neural Information Processing Systems (NIPS 2016)*. Barcelona, ES, 2110–2118.

[48] Hidetoshi Shimodaira. 2000. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference* 90, 2 (2000), 227–244. https://doi.org/10.1016/s0378-3758(00)00115-4

[49] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. 2018. A survey on deep transfer learning. In *International conference on artificial neural networks*. Springer, 270–279.

[50] Dirk Tasche. 2017. Fisher consistency for prior probability shift. *Journal of Machine Learning Research* 18 (2017), 95:1–95:32.

[51] Vladimir Vapnik. 1998. *Statistical learning theory*. Wiley, New York, US.

[52] Ricardo Vilalta, Christophe Giraud-Carrier, Pavel Brazdil, and Carlos Soares. 2011. Inductive transfer. In *Encyclopedia of Machine Learning*, Claude Sammut and Geoffrey I. Webb (Eds.). Springer, Heidelberg, DE, 545–548.

[53] Karl R. Weiss, Taghi M. Khoshgoftaar, and Dingding Wang:. 2016. A survey on transfer learning. *Journal of Big Data* 3 (2016), Article 9 (1–40). https://doi.org/10.1186/s40537-016-0043-6

[54] Gerhard Widmer and Miroslav Kubat. 1996. Learning in the presence of concept drift and hidden contexts. *Machine Learning* 23, 1 (1996), 69–101. https://doi.org/10.1007/bf00116900

[55] Meng-Sung Wu and Jen-Tzung Chien. 2010. A new topic-bridged model for transfer learning. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2010)*. Dallas, US, 5346–5349. https://doi.org/10.1109/icassp.2010.5494947

[56] Min Xiao and Yuhong Guo. 2013. A novel two-step method for cross-language representation learning. In *Proceedings of the 27th Annual Conference on Neural Information Processing Systems (NIPS 2013)*. Lake Tahoe, US, 1259–1267.

[57] Min Xiao and Yuhong Guo. 2014. Semi-supervised matrix completion for cross-lingual text classification. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI 2014)*. Québec City, CA, 1607–1614.

[58] Kui Xu and Xiaojun Wan. 2017. Towards a universal sentiment classifier in multiple languages. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*. Copenhagen, DE, 511–520. https://doi.org/10.18653/v1/d17-1053

[59] Ruochen Xu and Yiming Yang. 2017. Cross-lingual distillation for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*. Vancouver, CA, 1415–1425. https://doi.org/10.18653/v1/p17-1130

[60] Gui-Rong Xue, Wenyuan Dai, Qiang Yang, and Yong Yu. 2008. Topic-bridged PLSA for cross-domain text classification. In *Proceedings of the 31st ACM International Conference on Research and Development in Information Retrieval (SIGIR 2008)*. Singapore, SN, 627–634. https://doi.org/10.1145/1390334.1390441

[61] Xiaoshan Yang, Tianzhu Zhang, and Changsheng Xu. 2015. Cross-domain feature learning in multimedia. *IEEE Transactions on Multimedia* 17, 1 (2015), 64–78. https://doi.org/10.1109/tmm.2014.2375793

[62] Guangyou Zhou, Tingting He, Jun Zhao, and Wensheng Wu. 2015. A subspace learning framework for cross-lingual sentiment classification with partial parallel data. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI 2015)*. Buenos Aires, AR, 1426–1433.

[63] Guangyou Zhou, Yin Zhou, Xiyue Guo, Xinhui Tu, and Tingting He. 2015. Cross-domain sentiment classification via topical correspondence transfer. *Neurocomputing* 159 (2015), 298–305. https://doi.org/10.1016/j.neucom.2014.12.006

[64] Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2016. Cross-lingual sentiment classification with bilingual document representation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*. Berlin, DE, 1403–1412. https://doi.org/10.18653/v1/p16-1133

[65] Fuzhen Zhuang, Ping Luo, Hui Xiong, Qing He, Yuhong Xiong, and Zhongzhi Shi. 2011. Exploiting associations between word clusters and document classes for cross-domain text categorization. *Statistical Analysis and Data Mining* 4, 1 (2011), 100–114. https://doi.org/10.1002/sam.10099