# CROSS-LINGUAL SENTIMENT QUANTIFICATION

A PREPRINT

**Andrea Esuli, Alejandro Moreo, Fabrizio Sebastiani***
Istituto di Scienza e Tecnologie dell'Informazione
Consiglio Nazionale delle Ricerche
56124 Pisa, Italy
firstname.lastname@isti.cnr.it

April 18, 2019

**ABSTRACT**

We discuss *Cross-Lingual Text Quantification* (CLTQ), the task of performing text quantification (i.e., estimating the relative frequency $p_c(D)$ of all classes $c \in \mathcal{C}$ in a set $D$ of unlabelled documents) when training documents are available for a source language $\mathcal{S}$ but not for the target language $\mathcal{T}$ for which quantification needs to be performed. CLTQ has never been discussed before in the literature; we establish baseline results for the binary case by combining state-of-the-art quantification methods with methods capable of generating cross-lingual vectorial representations of the source and target documents involved. We present experimental results obtained on publicly available datasets for cross-lingual sentiment classification; the results show that the presented methods can perform CLTQ with a surprising level of accuracy.

## 1 Introduction

In *Cross-Lingual Text Classification* (CLTC) documents may be expressed in either a *source* language $\mathcal{S}$ or a *target* language $\mathcal{T}$, and training documents are available only for $\mathcal{S}$ but not for $\mathcal{T}$; CLTC thus consists of leveraging the training documents in the source language in order to train a classifier for the target language, also using the fact that the classification scheme $\mathcal{C}$ is the same for both $\mathcal{S}$ and $\mathcal{T}$. CLTC has been widely investigated in the literature (see e.g., Prettenhofer and Stein (2011); Moreo et al. (2016)). A companion task which instead has never been tackled, and which is the object of this paper, is *Cross-Lingual Text Quantification*, the task of performing "quantification" across a source language $\mathcal{S}$ and a target language $\mathcal{T}$. *Quantification* is a supervised learning task that consists of predicting, given a set of classes $\mathcal{C}$ and a set $D$ (a *sample*) of unlabelled items drawn from some domain $\mathcal{D}$, the *prevalence* (i.e., relative frequency) $p_c(D)$ of each class $c \in \mathcal{C}$ in $D$. Put it another way, given an unknown distribution $p_{\mathcal{C}}(D)$ of the members of $D$ across $\mathcal{C}$ (the *true distribution*), quantification consists in generating a *predicted distribution* $\hat{p}_{\mathcal{C}}(D)$ that approximates $p_{\mathcal{C}}(D)$ as accurately as possible González et al. (2017). Quantification is important for many application fields characterised by an interest in aggregate (rather than individual) data, such as the social sciences, market research, political science, and epidemiology.

In principle, quantification can be trivially solved via classification, i.e., by training a classifier $h$ using training data labelled according to $\mathcal{C}$, classifying the unlabelled data in $D$ via $h$, and counting, for each $c \in \mathcal{C}$, how many items in $D$ have been attributed to $c$ (the "classify and count" method). However, research has conclusively shown (see e.g., Bella et al. (2010); Esuli et al. (2018); Forman (2008)) that this approach leads to suboptimal quantification accuracy.

In this paper we establish baseline results for (binary) CLTQ by combining a number of quantification methods with state-of-the-art cross-lingual projection methods, i.e., methods capable of generating language-agnostic vectorial representations of the source and target documents involved. For performing this latter task we explore *Structural Correspondence Learning* (SCL – Prettenhofer and Stein (2011)) and *Distributional Correspondence Indexing* (DCI

*The order in which the authors are listed is purely alphabetical; each author has given an equally important contribution to this work.

– Moreo et al. (2016)), since (i) SCL is arguably the most representative cross-lingual projection method in the literature, and a mandatory baseline in lab experiments of related research, while DCI is a cross-lingual projection method that has recently demonstrated state-of-the-art performance in CLTC Moreo et al. (2018), (ii) both methods provide a general procedure for projecting source and target documents onto a common vector space, and (iii) the code implementing both methods is publicly available and easily modifiable.

The rest of the paper is organized as follows. Section 2 describes the CLTQ methods we use; Section 3 tests the presented methods on standard datasets for cross-lingual sentiment classification; Section 4 wraps up.

## 2 Method

Different quantification methods have been proposed that exploit the classification outcomes that a previously trained classifier delivers on unlabelled data. We explore different CLTQ methods that result from the combination of a cross-lingual projection method (Section 2.1), a "classify and count" policy (Section 2.2), and an estimate correction method (Section 2.3). In this paper we only address the binary case, where the classes are indicated as $\mathcal{C} = \{\oplus, \ominus\}$.

### 2.1 Cross-Lingual Document Representations

SCL and DCI rely on the concept of *pivot term* (or simply *pivot*) Blitzer et al. (2007) in order to bridge the gap between the different feature spaces which the different languages generate. In cross-lingual adaptation, pivots are defined as highly predictive pairs of translation-equivalent terms which behave in a similar way in their respective languages. Typical examples of pivots for sentiment-related applications are adjectives with domain-independent meaning such as "excellent" or "poor", and partially domain-dependent terms such as "fancy" (as found in the kitchen appliance domain and in the clothing domain) or "masterpiece" (as found in the book domain, movie domain, and music domain), with their respective translations in other languages.

A common strategy to select the pivots consists of taking the top elements from a list of terms ranked according to their mutual information to the label representing the domain (as computed from source-language training data), and filtering out those candidates whose translation equivalent shows a substantial prevalence drift in the target language. A word translation oracle, with a fixed budget of allowed calls, is assumed available. Once pivots are selected, different methods can be defined in order to produce cross-lingual vectorial representations. Both SCL and DCI first represent documents as vectors $\mathbf{x}$ in a (weighted) bag-of-words model of dimension $|V|$ (with $V$ being the vocabulary), and then apply a linear projection (parameterized by a matrix $\theta \in \mathbb{R}^{|V|L}$) of type $\mathbf{x}^\top\theta$, thus mapping $|V|$-dimensional vectors into $L$-dimensional vectors in a cross-lingual latent space. To achieve this, the unlabelled collections from the source and target domains are inspected. Once defined, the matrix can be subsequently used to project source documents (to train a classifier) and target documents (to classify them).

SCL builds the projection matrix by resolving an auxiliary prediction problem for each pair of translation-equivalent pivot terms. Each problem consists of predicting the presence of a pivot term based on the observation of the other terms. By solving the auxiliary problems (via linear classification), structural correspondences among terms and pivots are captured and collected as a matrix of correlations. This matrix is later decomposed using truncated SVD to generate the final projection matrix $\theta$. DCI relies instead on the distributional hypothesis to directly model correspondences between terms and pivots. Each row of the projection matrix DCI computes represents a term profile, where each dimension quantifies the degree of correspondence (as measured by a *distributional correspondence function*) of the term to a pivot.

### 2.2 Classifying and Counting

An obvious way to solve quantification is by aggregating the scores assigned by a classifier to the unlabelled documents. In connection to each of SCL and DCI we experiment with two different aggregation methods, one that uses a "hard" classifier (i.e., a classifier $h_\oplus : \mathcal{D} \rightarrow \{0, 1\}$ that outputs binary decisions, 0 for $\ominus$ and 1 for $\oplus$) and one that uses a "soft" classifier (i.e., a classifier $s_\oplus : \mathcal{D} \rightarrow [0, 1]$ that outputs posterior probabilities $\Pr(\oplus|\mathbf{x})$, representing the probability that the classifier attributes to the fact that $\mathbf{x}$ belongs to the $\oplus$ class).

The (trivial) *classify and count* (CC) quantifier then comes down to computing

$$\hat{p}^{CC}_\oplus(D) = \frac{\sum_{\mathbf{x}\in D} h_\oplus(\mathbf{x})}{|D|} \tag{1}$$

while the *probabilistic classify and count* quantifier (PCC – Bella et al. (2010)) is defined by

$$\hat{p}_{\oplus}^{CC}(D) = \frac{\sum_{\mathbf{x}\in D} s_{\oplus}(\mathbf{x})}{|D|} \tag{2}$$

### 2.3 Adjusting the Results of Classify and Count

A popular quantification method consists of applying an *adjustment* to the prevalence $\hat{p}_{\oplus}(D)$ estimated via "classify and count". It is easy to check that, in the binary case, the true prevalence $p_{\oplus}(D)$ and the estimated prevalence $\hat{p}_{\oplus}(D)$ are such that

$$p_{\oplus}(D) = \frac{\hat{p}_{\oplus}^{CC}(D) - fpr}{tpr - fpr} \tag{3}$$

where $tpr$ and $fpr$ stand for the *true positive rate* and *false positive rate* of the classifier $h_{\oplus}$ used to obtain $\hat{p}_{\oplus}^{CC}$. The values of $tpr$ and $fpr$ are unknown, but can be estimated via $k$-fold cross-validation on the training data. In the binary case this comes down to using the results $h_{\oplus}(\mathbf{x})$ obtained in the $k$-fold cross-validation (i.e., $\mathbf{x}$ ranges on the training documents) in equations

$$\hat{tpr} = \frac{\sum_{\mathbf{x}\in\oplus} h_{\oplus}(\mathbf{x})}{|\{\mathbf{x}\in\oplus\}|} \qquad \hat{fpr} = \frac{\sum_{\mathbf{x}\in\ominus} h_{\oplus}(\mathbf{x})}{|\{\mathbf{x}\in\ominus\}|} \tag{4}$$

We obtain $p_{\oplus}^{ACC}(D)$ estimates, which define the *adjusted classify and count* method (ACC – Forman (2008)), by replacing $tpr$ and $fpr$ in Equation 3 with the estimates of Equation 4.

If the soft classifier $s_{\oplus}(\mathbf{x})$ is used in place of $h_{\oplus}(\mathbf{x})$, analogues of $\hat{tpr}$ and $\hat{fpr}$ from Equation 4 can be defined as

$$\hat{tpr} = \frac{\sum_{\mathbf{x}\in\oplus} s_{\oplus}(\mathbf{x})}{|\{\mathbf{x}\in\oplus\}|} \qquad \hat{fpr} = \frac{\sum_{\mathbf{x}\in\ominus} s_{\oplus}(\mathbf{x})}{|\{\mathbf{x}\in\ominus\}|} \tag{5}$$

We obtain $p_{\oplus}^{PACC}(D)$ estimates, which define the *probabilistic adjusted classify and count* method (PACC – Bella et al. (2010)), by replacing $tpr$ and $fpr$ in Equation 3 with the estimates of Equation 5.

ACC and PACC define two simple linear adjustments to the aggregated scores of general-purpose classifiers. We also investigate the use of a more recently proposed adjustment method called QuaNet Esuli et al. (2018). QuaNet models a neural *non-linear* adjustment by taking as input all estimated prevalences ($\hat{p}_{\oplus}^{CC}$, $\hat{p}_{\oplus}^{ACC}$, $\hat{p}_{\oplus}^{PCC}$, $\hat{p}_{\oplus}^{PACC}$), several statistics (the $\hat{tpr}$ and $\hat{fpr}$ estimates from Equations 4 and 5), the posterior probabilities $\Pr(\oplus|\mathbf{x})$ for each document $\mathbf{x}$, and the document vectors themselves. QuaNet relies on a recurrent neural network to produce "quantification embeddings", which are then used to generate the final prevalence estimates.

## 3 Experiments

In this section we report on the experiments we have run on sentiment classification data in order to empirically evaluate the effectiveness of our CLTQ approaches. The code to replicate all these experiments is available from GitHub.[2]

We use the Webis-CLS-10 dataset Prettenhofer and Stein (2011) as the benchmark for our experiments.[3] Webis-CLS-10 is a dataset originally proposed for CLTC experiments, and consisting of Amazon product reviews written in four languages (English, German, French, Japanese) and concerning three product domains (Books, DVDs, Music). There are 2,000 training documents, 2,000 test documents, and a number of unlabelled documents ranging from 9,000 to 50,000 for each combination of language and domain. The examples of $\oplus$ and $\ominus$ (which indicate positive and negative sentiment, resp.) are balanced in all sets (training, test, unlabelled). Following a consolidated practice in CLTC, we always use English as the source language.

We use the NUT package[4] for SCL and the PYDCI package[5] Moreo et al. (2018) for DCI in order to generate the vectorial representations of all training and test documents. As the hard classifiers, we stick to the ones used by the original proponents of SCL and DCI, i.e., a linear classifier trained via Elastic Net Zou and Hastie (2005) (implemented via the

[2] `https://github.com/AlexMoreo/cl-quant.git`

[3] We use the preprocessed version available at `http://www.uni-weimar.de/medien/webis/corpora/corpus-webis-cls-10/cls-acl10-p`

[4] `https://github.com/pprett/nut`

[5] `https://github.com/AlexMoreo/pydci`

BOLT package[6]) for SCL, and a linear classifier trained via SVMs (implemented via the SCIKIT-LEARN package[7]) for DCI. As the soft classifier we instead use one trained via logistic regression (in its SCIKIT-LEARN implementation) for both SCL and DCI, since such classifiers are known to return "well-calibrated" posterior probabilities.[8]

We set all the hyper-parameters in SCL (number $m$ of pivots, minimum support frequency $\phi$ for pivot candidates, dimensionality $k$ of the cross-lingual representation, and the Elastic Net coefficient $\alpha$) to the values found optimal by the authors of Prettenhofer and Stein (2011) when optimizing for the German book review task ($m = 450$, $\phi = 30$, $k = 100$, $\alpha = 0.85$). Along with Moreo et al. (2018), in DCI we set the number of pivots and minimum support to $m = 450$ and $\phi = 30$ (the dimensionality is defined as $k = 450$ since in DCI each pivot defines a dimension[9]); as the distributional correspondence function we use cosine since it is the one which delivered the best performance in Moreo et al. (2018). For each setup we independently optimize the parameter $C$ (which controls the regularization strength in the SVM and in the logistic regressor) via grid search in the log space defined by $C \in \{10^i\}_{i=-5}^{5}$, and via 5-fold cross-validation. The classifiers with the optimized hyper-parameters are then used in a 10-fold cross-validation run on the training data to produce the $\hat{tpr}$ and $\hat{fpr}$ estimates.

For the neural correction of QuaNet we use the publicly available implementation released by its authors.[10] We optimize the hyper-parameters of QuaNet using the German book review task (as done in Prettenhofer and Stein (2011)); we end up using 64 hidden units in the recurrent cell of a two-layer stacked bidirectional LSTM, 1024 and 512 hidden units in the next-to-last feed-forward layers, and a drop probability of 0. We set the rest of the parameters as in Esuli et al. (2018).

As the measures of quantification error we use *Absolute Error* (*AE*), *Relative Absolute Error* (*RAE*), and the *Kullback-Leibler Divergence* (*KLD*), defined as:

$$AE(p, \hat{p}, D) = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} |\hat{p}_c(D) - p_c(D)| \tag{6}$$

$$RAE(p, \hat{p}, D) = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \frac{|\hat{p}_c(D) - p_c(D)|}{p_c(D)} \tag{7}$$

$$KLD(p, \hat{p}, D) = \sum_{c \in \mathcal{C}} p_c(D) \log \frac{p_c(D)}{\hat{p}_c(D)} \tag{8}$$

since they are (see (Sebastiani, 2018, p. 23)) the most frequently used measures for evaluating quantification error.

The evaluation of a quantifier cannot be carried out on the basis on one single set of test documents. The reason is that, while in text classification experiments a test set consisting of $n$ documents enables the evaluation of $n$ different decision outcomes, in quantification the same test set would only allow to validate one single prevalence prediction. In order to allow statistically significant comparisons, Forman (2008) proposed to run quantification experiments on a set of test samples, randomly sampled from the original set of test documents at different prevalence levels. Along with Forman (2008), as the range of prevalences for the $\oplus$ class we use $\{0.01, 0.05, 0.10, \ldots, 0.90, 0.95, 0.99\}$. As in Esuli et al. (2018), we generate 100 random samples for each of the 21 prevalence levels, and report quantification error as the average across $21 \times 100 = 2100$ test samples. All samples consist of 200 documents. For each target language (German, French, Japanese) and product domain (Books, DVD, Music) the samples are the same across the different methods, which will enable us to evaluate the statistical significance of the differences in performance; to this aim, we rely on the non-parametric Wilcoxon signed-rank test on paired samples.

For each combination of target language and product domain, Table 1 reports quantification error (for each CLTQ method and for each evaluation measure) as an average across the 2100 test samples; we recall that English is always used as the source language, so that, e.g., the "German Books" experiment is about training on English book reviews and testing on German book reviews. Since QuaNet depends on a stochastic optimization, Table 1 reports the average and standard deviation across 10 runs.

[6] `https://github.com/pprett/bolt`

[7] `https://scikit-learn.org/`

[8] "Well calibrated probabilities" is usually considered a synonym of "good-quality probabilities". Posterior probabilities $\Pr(c|\mathbf{x})$ are said to be *well calibrated* when, given a sample $D$ drawn from some population, $\lim_{|D|\rightarrow\infty} \frac{|\{\mathbf{x} \in c | \Pr(c|\mathbf{x})=\alpha\}|}{|\{\mathbf{x} \in D | \Pr(c|\mathbf{x})=\alpha\}|} = \alpha$. Intuitively, this property implies that, as the size of the sample $D$ goes to infinity, e.g., 90% of the documents $\mathbf{x} \in D$ that are assigned a well calibrated posterior probability $\Pr(c|\mathbf{x}) = 0.9$ belong to class $c$.

[9] In preliminary experiments we had used the same value $k = 450$ both for DCI and SCL, on grounds of "fairness". The results for SCL were slightly worse with respect to using $k = 100$; for SCL we thus decided to stick to the $k = 100$ value originally used in Prettenhofer and Stein (2011).

[10] `https://github.com/HLT-ISTI/QuaNet`

Table 1: CLTQ results for Webis-CLS-10. **Boldface** indicates the best result. Superscripts † and †† denote the method (if any) whose score is not statistically significantly different from the best one at $\alpha = 0.05$ (†) or at $\alpha = 0.005$ (††).

| | Target | | SCL | | | | | DCI | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Language | Domain | CC | ACC | PCC | PACC | QuaNet | CC | ACC | PCC | PACC | QuaNet |
| *AE* | German | Books | 0.092 | 0.040 | 0.237 | 0.375 | 0.203 (±0.006) | 0.090 | 0.037 | 0.119 | **0.027** | 0.030 (±0.002) |
| | German | DVDs | 0.104 | 0.045 | 0.221 | 0.331 | 0.178 (±0.009) | 0.086 | 0.030 | 0.147 | **0.028** | 0.030 (±0.003)†† |
| | German | Music | 0.097 | 0.037†† | 0.151 | 0.101 | 0.072 (±0.007) | 0.078 | 0.037†† | 0.109 | 0.039†† | **0.030** (±0.002) |
| | French | Books | 0.098 | 0.037 | 0.202 | 0.288 | 0.151 (±0.007) | 0.098 | 0.038 | 0.122 | **0.025** | 0.036 (±0.003) |
| | French | DVDs | 0.110 | 0.056 | 0.174 | 0.113 | 0.072 (±0.002) | 0.091 | 0.037 | 0.117 | **0.027** | 0.045 (±0.005) |
| | French | Music | 0.119 | 0.060 | 0.178 | 0.090 | 0.072 (±0.001) | 0.074 | 0.030 | 0.160 | **0.024** | 0.047 (±0.010) |
| | Japanese | Books | 0.127 | 0.072 | 0.194 | 0.124 | 0.095 (±0.002) | 0.117 | **0.060** | 0.174 | 0.064 | 0.073 (±0.003) |
| | Japanese | DVDs | 0.131 | 0.079 | 0.329 | 0.485 | 0.270 (±0.005) | 0.104 | 0.045 | 0.128 | **0.037** | 0.058 (±0.006) |
| | Japanese | Music | 0.118 | 0.059 | 0.242 | 0.377 | 0.228 (±0.007) | 0.092 | 0.029 | 0.161 | **0.027** | 0.044 (±0.009) |
| | Average | | 0.111 | 0.054 | 0.214 | 0.254 | 0.149 | 0.092 | 0.038 | 0.138 | **0.033** | 0.044 |
| *RAE* | German | Books | 0.888 | 0.164 | 0.878 | 0.807 | 0.513 (±0.015) | 1.135 | 0.246 | 1.411 | **0.136** | 0.248 (±0.034) |
| | German | DVDs | 1.086 | 0.267 | 1.047 | 0.733 | 0.428 (±0.031) | 1.070 | 0.223 | 1.709 | **0.144** | 0.234 (±0.020)†† |
| | German | Music | 1.056 | 0.194† | 1.364 | 0.268 | 0.216 (±0.011) | 0.947 | 0.194†† | 1.310 | **0.153** | 0.245 (±0.022)†† |
| | French | Books | 1.021 | 0.313 | 1.041 | 0.666 | 0.383 (±0.025) | 1.227 | 0.407 | 1.426 | **0.159** | 0.330 (±0.026) |
| | French | DVDs | 1.307 | 0.682 | 1.642 | 0.475 | 0.543 (±0.019) | 0.938 | 0.176 | 1.284 | **0.144** | 0.223 (±0.016) |
| | French | Music | 1.310 | 0.496 | 2.099 | 1.181 | 0.817 (±0.026) | 0.834 | **0.138** | 1.803 | 0.208 | 0.276 (±0.039)† |
| | Japanese | Books | 1.423 | 0.781 | 2.287 | 1.572 | 1.122 (±0.026) | 1.196 | **0.450** | 1.935 | 0.639 | 0.570 (±0.032) |
| | Japanese | DVDs | 1.392 | 0.785 | 0.833 | 0.947 | 0.557 (±0.012) | 1.097 | 0.292 | 1.380 | **0.213** | 0.350 (±0.021) |
| | Japanese | Music | 1.232 | 0.304 | 0.910 | 0.806 | 0.527 (±0.016) | 0.973 | **0.175** | 1.800 | 0.198† | 0.293 (±0.034) |
| | Average | | 1.191 | 0.443 | 1.345 | 0.828 | 0.567 | 1.046 | 0.256 | 1.562 | **0.222** | 0.308 |
| *KLD* | German | Books | 0.041 | 0.016 | 0.194 | 1.778 | 0.274 (±0.043) | 0.040 | 0.032 | 0.062 | 0.028 | **0.007** (±0.001) |
| | German | DVDs | 0.050 | 0.013 | 0.172 | 0.987 | 0.139 (±0.034) | 0.038 | 0.019 | 0.086 | 0.028 | **0.007** (±0.001) |
| | German | Music | 0.045 | 0.017†† | 0.090 | 0.062 | 0.027 (±0.005) | 0.032 | 0.046 | 0.054 | 0.072 | **0.008** (±0.001) |
| | French | Books | 0.046 | 0.010†† | 0.146 | 0.748 | 0.115 (±0.024) | 0.046 | 0.014 | 0.064 | 0.014 | **0.010** (±0.001) |
| | French | DVDs | 0.055 | 0.019 | 0.111 | 0.055 | 0.029 (±0.001) | 0.040 | 0.012 | 0.060 | **0.008** | 0.012 (±0.002) |
| | French | Music | 0.062 | 0.021 | 0.114 | 0.040 | 0.028 (±0.000) | 0.030 | 0.040 | 0.097 | **0.007** | 0.014 (±0.004) |
| | Japanese | Books | 0.068 | 0.028 | 0.132 | 0.065 | 0.043 (±0.001) | 0.060 | **0.020** | 0.110 | 0.024 | 0.029 (±0.002) |
| | Japanese | DVDs | 0.071 | 0.033 | 0.376 | 5.133 | 0.250 (±0.013) | 0.051 | 0.014 | 0.069 | **0.011** | 0.020 (±0.003) |
| | Japanese | Music | 0.061 | 0.022 | 0.202 | 1.629 | 0.234 (±0.024) | 0.042 | 0.011 | 0.098 | **0.009** | 0.013 (±0.004) |
| | Average | | 0.055 | 0.020 | 0.171 | 1.166 | 0.127 | 0.042 | 0.023 | 0.078 | 0.022 | **0.013** |

Overall, the results indicate that the combination DCI+PACC is the best performer in terms of *AE* and *RAE*, while DCI+QuaNet seems to behave slightly better in terms of *KLD*. Given the recent theoretical study on the properties of evaluation measures for quantification Sebastiani (2018), that indicates that *AE* and *RAE* are to be preferred to *KLD*, this leads us to prefer DCI+PACC. In both SCL and DCI the "hard" classifier works comparatively better than the "soft" logistic regressor. As expected, ACC (the "adjusted" version of CC) performs substantially better than CC in all cases. What comes as a surprise, though, is the fact that the remarkable benefit PACC brings about in DCI with respect to its unadjusted variant PCC, is not consistently mirrored in the case of SCL (where the effect of adjusting is instead harmful, and especially so in terms of *KLD*). The neural adjustment of QuaNet, when applied to DCI vectors, performs somehow similarly to the best performer in several cases, and actually delivers the lowest average *KLD* error. That QuaNet does not perform as well with SCL can be explained by two facts (which are not independent of each other), i.e., the importance of the estimated posterior probabilities within QuaNet, and the suboptimal ability (as shown by the PCC and PACC results) in delivering accurate posterior probabilities for SCL vectors that the logistic regressor has shown.

## 4 Conclusions

The experiments we have performed show that structural correspondence learning (SCL) and distributional correspondence indexing (DCI), two previously proposed methods for cross-lingual text classification, can effectively be used in cross-lingual text quantification, a task that had never been tackled before in the literature. The tested methods yield quantification predictions that are fairly close to the true prevalence; in terms of absolute error (arguably the most easy-to-interpret error criterion), and on average, DCI+PACC differs from the true prevalence by a margin of 3.3%, while this difference is 5.4% for SCL+ACC. These results are encouraging, especially if we consider the fact that the quantifier is trained on a language different from the one on which quantification is performed, and that a range of prevalences different from the one present in the training set are tested.

The combination of transfer learning (of which cross-lingual transfer is an instance) with quantification is an interesting task in general, that should prompt a body of dedicated research. We believe end-to-end approaches for cross-lingual quantification, not necessarily relying on classification as an intermediate step, would be worth exploring. Likewise, a natural extension of this work would be to explore applications of transfer learning to quantification different from the cross-lingual one, such as cross-domain quantification.

## References

Antonio Bella, Cèsar Ferri, José Hernández-Orallo, and María José Ramírez-Quintana. 2010. Quantification via Probability Estimators. In *Proceedings of the 11th IEEE International Conference on Data Mining (ICDM 2010)*. Sydney, AU, 737–742. https://doi.org/10.1109/icdm.2010.75

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.

Andrea Esuli, Alejandro Moreo, and Fabrizio Sebastiani. 2018. A Recurrent Neural Network for Sentiment Quantification. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM 2018)*. Torino, IT, 1775–1778. https://doi.org/10.1145/3269206.3269287

George Forman. 2008. Quantifying counts and costs via classification. *Data Mining and Knowledge Discovery* 17, 2 (2008), 164–206. https://doi.org/10.1007/s10618-008-0097-y

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

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

Alejandro Moreo, Andrea Esuli, and Fabrizio Sebastiani. 2018. Revisiting Distributional Correspondence Indexing: A Python Reimplementation and New Experiments. arXiv:1810.09311 [cs.CL].

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

Fabrizio Sebastiani. 2018. Evaluation Measures for Quantification: An Axiomatic Approach. (2018). arXiv:1809.01991.

Hui Zou and Trevor Hastie. 2005. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B* 67, 2 (2005), 301–320. https://doi.org/10.1111/j.1467-9868.2005.00503.x