Embedding Projection for Targeted Cross-Lingual Sentiment: Model Comparisons and a Real-World Study

Jeremy Barnes jeremycb@ifi.uio.no
Language Technology Group
University of Oslo
Gaustadalléen 23 B, N-0373
Oslo, Norway

Roman Klinger klinger@ims.uni-stuttgart.de
Institut für Maschinelle Sprachverarbeitung
University of Stuttgart
Pfaffenwaldring 5b, 70569
Stuttgart, Germany

Abstract

Sentiment analysis benefits from large, hand-annotated resources in order to train and test machine learning models, which are often data hungry. While some languages, *e. g.*, English, have a vast array of these resources, most under-resourced languages do not, especially for fine-grained sentiment tasks, such as aspect-level or targeted sentiment analysis. To improve this situation, we propose a *cross-lingual* approach to sentiment analysis that is applicable to under-resourced languages and takes into account target-level information. This model incorporates sentiment information into bilingual distributional representations, by jointly optimizing them for semantics and sentiment, showing state-of-the-art performance at sentence-level when combined with machine translation. The adaptation to targeted sentiment analysis on multiple domains shows that our model outperforms other projection-based bilingual embedding methods on binary targeted sentiment tasks. Our analysis on ten languages demonstrates that the amount of unlabeled monolingual data has surprisingly little effect on the sentiment results. As expected, the choice of a annotated source language for projection to a target leads to better results for source-target language pairs which are similar. Therefore, our results suggest that more efforts should be spent on the creation of resources for less similar languages to those which are resource-rich already. Finally, a domain mismatch leads to a decreased performance. This suggests resources in any language should ideally cover varieties of domains.

1. Introduction

1.1 Targeted Sentiment Classification

Opinions are everywhere in our lives. Every time we open a book, read the newspaper, or look at social media, we scan for opinions or form them ourselves. We are cued to the opinions of others, and often use this information to update our own opinions (Asch, 1955; Das et al., 2014). This is true on the Internet as much as it is in our face-to-face relationships. In fact, with its wealth of opinionated material available online, it has become feasible and interesting to harness this data in order to automatically identify opinions,

which had previously been far more expensive and tedious when the only access to data was offline.

Sentiment analysis, sometimes referred to as *opinion mining*, seeks to create data-driven methods to classify the polarity of a text. The information obtained from sentiment classifiers can then be used for tracking user opinions in different domains (Pang et al., 2002; Socher et al., 2013; Nakov et al., 2013), predicting the outcome of political elections (Wang et al., 2012; Bakliwal et al., 2013), detecting hate speech online (Nahar et al., 2012; Hartung et al., 2017), as well as predicting changes in the stock market (Pagolu et al., 2016).

Sentiment analysis can be modeled as a classification task, especially at sentence- and document-level, or as a sequence-labeling task at target-level. Targeted sentiment analysis aims at predicting the polarity expressed towards a particular entity or sub-aspect of that entity. This is a more realistic view of sentiment, as polarities are directed towards targets, not spread uniformly across sentences or documents. Take the following example, where we mark the sentiment target with green, positive sentiment expressions with blue, and negative sentiment expressions with red.:

> The café near my house has great coffee but I never go there because the service is terrible.

In this sentence, it is not stated what the sentiment towards the target "café" is, while the sentiment of the target "coffee" is positive and that of "service" is negative. In order to correctly classify the sentiment of each target, it is necessary to (1) detect the targets, (2) detect polarity expressions, and (3) resolve the relations between these.

In order to model these relationships and test the accuracy of the learned models, hand-annotated resources are typically used for training machine learning algorithms. Resource-rich languages, *e. g.*, English, have high-quality annotated data for both classification and sequence-labeling tasks, as well as for a variety of domains. However, under-resourced languages either completely lack annotated data or have only a few resources for specific domains or sentiment tasks. For instance, for aspect-level sentiment analysis, English has datasets available in the news domain (Wiebe et al., 2005), product review domain (Hu & Liu, 2004; Ding et al., 2008; Pontiki et al., 2014, 2015), education domain (Welch & Mihalcea, 2016), medical domain (Gräßer et al., 2018), urban neighborhood domain (Saeidi et al., 2016), and financial (Maia et al., 2018) domain. Spanish, on the other hand, has only three datasets (Agerri et al., 2013; Pontiki et al., 2016), while Basque and Catalan only have one each for a single domain (Barnes et al., 2018). The cost of annotating data can often be prohibitive as training native-speakers to annotate fine-grained sentiment is a long process. This motivates the need to develop sentiment analysis methods capable of leveraging data annotated in other languages.

1.2 Cross-Lingual Approaches to Sentiment Analysis

Previous work on *cross-lingual sentiment analysis* (CLSA) offers a way to perform sentiment analysis in an under-resourced language that does not have any annotated data available. Most methods relied on the availability of large amounts of parallel data to transfer sentiment information across languages. *Machine translation* (MT), for example, has been the most common approach to cross-lingual sentiment analysis (Banea et al.,

2013; Almeida et al., 2015; Zhang & Wallace, 2017). Machine translation, however, can be biased towards domains (Wu et al., 2008; Bertoldi & Federico, 2009; Koehn & Knowles, 2017), does not always preserve sentiment (Mohammad et al., 2016), and requires millions of parallel sentences (Gavrila & Vertan, 2011; Vaswani et al., 2017), which places a limit on which languages can benefit from these approaches. The following example illustrates that MT does not preserve sentiment (hotel review in Basque, automatically translated via `translate.google.com`):

> Hotel[1] txukuna da, nahiko berria. Harreran zeuden langileen arreta[2] ez zen onena izan. Tren geltoki bat[3] du 5 minutura eta kotxez[4] berehala iristen da baina oinez[5] urruti samar dago.
>
> The hotel[1] is tidy, quite new. The care of the workers at reception[2] was not the best. It's 5 minutes away from a train station[3] and it's quick to reach the car[4], but it's a short distance away.

While the first two sentences are mostly well translated for the purposes of sentiment analysis, in the third, there are a number of reformulations and deletions that lead to a loss of information. It should read "It has a train station five minutes away and by car you can reach it quickly, but by foot it's quite a distance." We can see that one of the targets has been deleted and the sentiment has flipped from negative to positive. Such common problems degrade the results of cross-lingual sentiment systems that use MT, especially at target-level.

Although high quality machine translation systems exist between many languages and have been shown to enable cross-lingual sentiment analysis, for the vast majority of language pairs in the world there is not enough parallel data to create these high quality MT systems. This lack of parallel data coupled with the computational expense of MT means that approaches to cross-lingual sentiment analysis that do not require MT should be preferred. Additionally, most cross-lingual sentiment approaches using MT have concentrated on sentence- and document-level, and have not explored targeted or aspect-level sentiment tasks.

1.3 Bilingual Distributional Models and the Contributions of this Paper

Recently, several *bilingual distributional semantics models* (bilingual embeddings) have been proposed and provide a useful framework for cross-lingual research without requiring machine translation. They are effective at generating features for bilingual dictionary induction (Mikolov et al., 2013; Artetxe et al., 2016; Lample et al., 2018a), cross-lingual text classification (Prettenhofer & Stein, 2011; Chandar et al., 2014), or cross-lingual dependency parsing (Søgaard et al., 2015), among others. In this framework, words are represented as n-dimensional vectors which are created on large monolingual corpora in order to (1) maximize the similarity of words that appear in similar contexts and use some bilingual regularization in order to (2) maximize the similarity of translation pairs. In this work, we concentrate on a subset of these bilingual embedding methods that perform a post-hoc mapping to a bilingual space, which we refer to as *embedding projection methods*. One of the main advantages of these methods is that they make better use of small amounts of parallel data

than MT systems, even enabling unsupervised machine translation (Artetxe et al., 2018b; Lample et al., 2018b).

With this paper, we provide the first extensive evaluation of cross-lingual embeddings for targeted sentiment tasks. We formulate the task of targeted sentiment analysis as classification, given the targets from an oracle[1]. The question we attempt to address is *how to infer the polarity of a sentiment target in a language that does not have any annotated sentiment data or parallel corpora with a resource-rich language.* In the following Catalan sentence, for example, how can we determine that the sentiment of "servei" is negative, while that of "menjar" is positive if we do not have annotated data in Catalan or parallel data for English-Catalan?

El servei al restaurant va ser péssim. Al menys el menjar era bo.

Specifically, we propose an approach which requires (1) minimal bilingual data and instead makes use of (2) high-quality monolingual word embeddings in the source and target language. We take an intermediate step by first testing this approach on sentence-level classification. After confirming that our approach performs well at sentence-level, we propose a targeted model with the same data requirements. The main contributions are that we

- compare projection-based cross-lingual methods to MT,
- extend previous cross-lingual approaches to enable targeted cross-lingual sentiment analysis with minimal parallel data requirements,
- compare different model architectures for cross-lingual targeted sentiment analysis,
- perform a detailed error analysis, and detailing the advantages and disadvantages of each method,
- and, finally, deploy the methods in a realistic case-study to analyze their suitability beyond applications on (naturally) limited language pairs.

In addition, we make our code and data publicly available at `https://github.com/jbarnesspain/targeted_blse` to support future research. The rest of the article is organized as follows: In Section 2, we detail related work and motivate the need for a different approach. In Section 3, we describe both the sentence-level and targeted projection approaches that we propose. In Section 4, we detail the resources and experimental setup for both sentence and targeted classification. In Section 5, we describe the results of the two experiments, as well as perform a detailed error analysis. In Section 6, we perform a case study whose purpose is to give a more qualitative view of the models. Finally, we discuss the implications of the results in Section 7.

2. Previous Work

Sentiment analysis has become an enormously popular task with a focus on classification approaches on individual languages, but there has not been as much work on cross-lingual

1. This is a common assumption when studying target-level sentiment analysis (Dong et al., 2014; Zhang et al., 2016).

approaches. In this section, we detail the most relevant work on cross-lingual sentiment analysis and lay the basis for the bilingual embedding approach we propose later.

2.1 Machine Translation Based Methods

Early work in cross-lingual sentiment analysis found that machine translation (MT) had reached a point of maturity that enabled the transfer of sentiment across languages. Researchers translated sentiment lexicons (Mihalcea et al., 2007; Meng et al., 2012) or annotated corpora and used word alignments to project sentiment annotation and create target-language annotated corpora (Banea et al., 2008; Duh et al., 2011; Demirtas & Pechenizkiy, 2013; Balahur & Turchi, 2014).

Several approaches included a multi-view representation of the data (Banea et al., 2010; Xiao & Guo, 2012) or co-training (Wan, 2009; Demirtas & Pechenizkiy, 2013) to improve over a naive implementation of machine translation, where only the translated version of the data is considered. There are also approaches which only require parallel data (Meng et al., 2012; Zhou et al., 2016; Rasooli et al., 2017), instead of machine translation.

All of these approaches, however, require large amounts of parallel data or an existing high quality translation tool, which are not always available. To tackle this issue, Barnes et al. (2016) explore cross-lingual approaches for aspect-based sentiment analysis, comparing machine translation methods and those that instead rely on bilingual vector representations. They conclude that MT approaches outperform current bilingual representation methods.

Chen et al. (2016) propose an adversarial deep averaging network, which trains a joint feature extractor for two languages. They minimize the difference between these features across languages by learning to fool a language discriminator. This requires no parallel data, but does require large amounts of unlabeled data and has not been tested on fine-grained sentiment analysis.

2.2 Bilingual Embedding Methods

Recently proposed bilingual embedding methods (Hermann & Blunsom, 2014; Chandar et al., 2014; Gouws et al., 2015) offer a natural way to bridge the language gap. These particular approaches to bilingual embeddings, however, also require large parallel corpora in order to build the bilingual space, which gives no advantage over machine translation. Another approach to creating bilingual word embeddings, which we refer to as *Projection-based Bilingual Embeddings*, has the advantage of requiring relatively little parallel training data while taking advantage of larger amounts of monolingual data. In the following, we describe the most relevant approaches.

Bilingual Word Embedding Mappings (VecMap): Mikolov et al. (2013) find that vector spaces in different languages have similar arrangements. Therefore, they propose a linear projection which consists of learning a rotation and scaling matrix. Artetxe et al. (2016, 2017) improve upon this approach by requiring the projection to be orthogonal, thereby preserving the monolingual quality of the original word vectors.

Given source embeddings S, target embeddings T, and a bilingual lexicon L, Artetxe et al. (2016) learn a projection matrix W by minimizing the square of Euclidean distances

$$\arg\min_{W} \sum_{i} ||S'W - T'||_F^2 \,, \tag{1}$$

where $S' \in S$ and $T' \in T$ are the word embedding matrices for the tokens in the bilingual lexicon L. This is solved using the Moore-Penrose pseudoinverse $S'^{+} = (S'^{T}S')^{-1}S'^{T}$ as $W = S'^{+}T'$, which can be computed using SVD. We refer to this approach as VecMap.

Multilingual Unsupervised and Supervised Embeddings (Muse) Lample et al. (2018a) propose a similar refined orthogonal projection method to Artetxe et al. (2017), but include an adversarial discriminator, which seeks to discriminate samples from the projected space WS, and the target T, while the projection matrix W attempts to prevent this making the projection from the source space WS as similar to the target space T as possible.

They further refine their projection matrix by reducing the hubness problem (Dinu et al., 2015), which is commonly found in high-dimensional spaces. For each projected embedding Wx, they define the k nearest neighbors in the target space, $\mathcal{N}_T$, suggesting $k = 10$. They consider the mean cosine similarity $r_T(Wx)$ between a projected embedding Wx and its k nearest neighbors

$$r_T(Wx) = \frac{1}{k} \sum_{y \in \mathcal{N}_T(Wx)} \cos(Wx, y) \tag{2}$$

as well as the mean cosine of a target word y to its neighborhood, which they denote by r_S.

In order to decrease similarity between mapped vectors lying in dense areas, they introduce a cross-domain similarity local scaling term (CSLS)

$$\text{CSLS}(Wx, y) = 2\cos(Wx, y) - r_T(Wx) - r_S(y) \,, \tag{3}$$

which they find improves accuracy, while not requiring any parameter tuning.

Barista Gouws and Søgaard (2015) propose a method to create a pseudo-bilingual corpus with a small task-specific bilingual lexicon, which can then be used to train bilingual embeddings (Barista). This approach requires a monolingual corpus in both the source and target languages and a set of translation pairs. The source and target corpora are concatenated and then every word is randomly kept or replaced by its translation with a probability of 0.5. Any kind of word embedding algorithm can be trained with this pseudo-bilingual corpus to create bilingual word embeddings.

2.3 Sentiment Embeddings

Maas et al. (2011) first explored the idea of incorporating sentiment information into semantic word vectors. They proposed a topic modeling approach similar to latent Dirichlet allocation in order to collect the semantic information in their word vectors. To incorporate the sentiment information, they included a second objective whereby they maximize the probability of the sentiment label for each word in a labeled document.

Tang et al. (2014) exploit distantly annotated tweets to create Twitter sentiment embeddings. To incorporate distributional information about tokens, they use a hinge loss and

maximize the likelihood of a true n-gram over a corrupted n-gram. They include a second objective where they classify the polarity of the tweet given the true n-gram. While these techniques have proven useful, they are not easily transferred to a cross-lingual setting.

Zhou et al. (2015) create bilingual sentiment embeddings by translating all source data to the target language and vice versa. This requires the existence of a machine translation system, which is a prohibitive assumption for many under-resourced languages, especially if it must be open and freely accessible. This motivates approaches which can use smaller amounts of parallel data to achieve similar results.

2.4 Targeted Sentiment Analysis

The methods discussed so far focus on classifying textual phrases like documents or sentences. Next to these approaches, others have concentrated on classifying aspects (Hu & Liu, 2004; Liu, 2012; Pontiki et al., 2014) or targets (Zhang et al., 2015, 2016; Tang et al., 2016) to assign them with polarity values.

A common technique when adapting neural architectures to targeted sentiment analysis is to break the text into left context, target, and right context (Zhang et al., 2015, 2016), alternatively keeping the target as the final/beginning token in the respective contexts (Tang et al., 2016). The model then extracts a feature vector from each context and target, using some neural architecture, and concatenates the outputs for classification.

More recent approaches attempt to augment a neural network with memory to model these interactions (Chen et al., 2017; Xue & Li, 2018; Wang et al., 2018; Liu et al., 2018). Wang et al. (2017) explore methods to improve classification of multiple aspects in tweets, while Akhtar et al. (2018) attempt to use cross-lingual and multilingual data to improve aspect-based sentiment analysis in under-resourced languages.

As mentioned before, MT has traditionally been the main approach for transferring information across language barriers (Klinger & Cimiano, 2015, *i. a.*, for cross-lingual target-level sentiment analysis). But this is particularly problematic for targeted sentiment analysis, as changes in word order or loss of words created during translation can directly affect the performance of a classifier (Lambert, 2015).

3. Projecting Sentiment Across Languages

In this section, we propose a novel approach to incorporate sentiment information into bilingual embeddings, which we first test on *sentence-level* cross-lingual sentiment classification[2]. We then propose an extension in order to adapt this approach to *targeted* cross-lingual sentiment classification. Our model, *Bilingual Sentiment Embeddings* (Blse), are embeddings that are jointly optimized to represent both (a) semantic information in the source and target languages, which are bound to each other through a small bilingual dictionary, and (b) sentiment information, which is annotated on the source language only. We only need three resources: (1) a comparably small bilingual lexicon, (2) an annotated sentiment corpus in the resource-rich language, and (3) monolingual word embeddings for the two involved languages.

2. This first contribution in this paper is an extended version of the work presented as Barnes et al. (2018a).

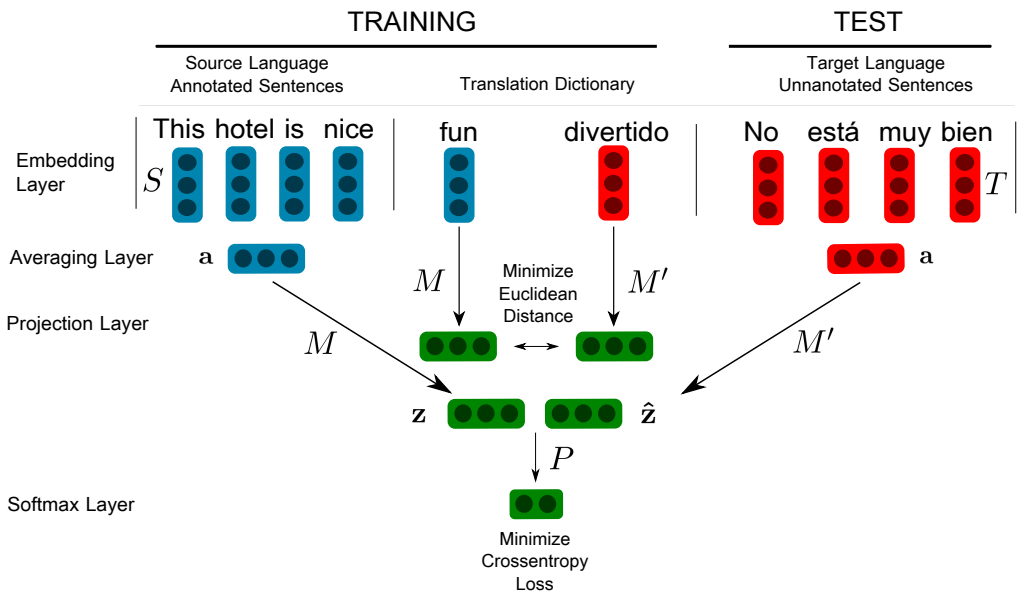

Figure 1: Bilingual Sentiment Embedding Model (Blse)

3.1 Sentence-level Model

In this section, we detail the projection objective, the sentiment objective, and finally the full objective for sentence-level cross-lingual sentiment classification. A sketch of the full sentence-level model is depicted in Figure 1.

3.1.1 Cross-lingual Projection

We assume that we have two precomputed vector spaces $S = \mathbb{R}^{v \times d}$ and $T = \mathbb{R}^{v' \times d'}$ for our source and target languages, where v (v') is the length of the source vocabulary (target vocabulary) and d (d') is the dimensionality of the embeddings. We also assume that we have a bilingual lexicon L of length n which consists of word-to-word translation pairs $L = \{(s_1, t_1), (s_2, t_2), \ldots, (s_n, t_n)\}$ which map from source to target.

In order to create a mapping from both original vector spaces S and T to shared sentiment-informed bilingual spaces $\mathbf{z}$ and $\hat{\mathbf{z}}$, we employ two linear projection matrices, M and M'. During training, for each translation pair in L, we first look up their associated vectors, project them through their associated projection matrix and finally minimize the mean squared error of the two projected vectors. This is similar to the approach taken by (Mikolov et al., 2013) , but includes an additional target projection matrix.

The intuition for including this second matrix is that a single projection matrix does not support the transfer of sentiment information from the source language to the target language. Without M', any signal coming from the sentiment classifier (see Section 3.1.2) would have no affect on the target embedding space T, and optimizing M to predict sentiment and projection would only be detrimental to classification of the target language.

We analyze this further in Section 5.1.4. Note that in this configuration, we do not need to update the original vector spaces, which would be problematic with such small training data.

The projection quality is ensured by minimizing the mean squared error[34]

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{z_i} - \hat{\mathbf{z}}_\mathbf{i})^2\,, \tag{4}$$

where $\mathbf{z_i} = S_{s_i} \cdot M$ is the dot product of the embedding for source word s_i and the source projection matrix and $\hat{\mathbf{z}}_\mathbf{i} = T_{t_i} \cdot M'$ is the same for the target word t_i.

3.1.2 Sentiment Classification

We add a second training objective to optimize the projected source vectors to predict the sentiment of source phrases. This inevitably changes the projection characteristics of the matrix M, and consequently M' and encourages M' to learn to predict sentiment without any training examples in the target language.

In order to train M to predict sentiment, we require a source-language corpus $C_{\text{source}} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_i, y_i)\}$ where each sentence x_i is associated with a label y_i.

For classification, we use a two-layer feed-forward averaging network, loosely following (Iyyer et al., 2015) [5]. For a sentence x_i we take the word embeddings from the source embedding S and average them to $\mathbf{a}_i \in \mathbb{R}^d$. We then project this vector to the joint bilingual space $\mathbf{z}_i = \mathbf{a}_i \cdot M$. Finally, we pass $\mathbf{z}_i$ through a softmax layer P to obtain the prediction $\hat{y}_i = \text{softmax}(\mathbf{z}_i \cdot P)$.

To train our model to predict sentiment, we minimize the cross-entropy error of the predictions

$$H = -\sum_{i=1}^{n} y_i \log \hat{y_i} - (1 - y_i) \log(1 - \hat{y_i})\,. \tag{5}$$

3.1.3 Joint Learning

In order to jointly train both the projection component and the sentiment component, we combine the two loss functions to optimize the parameter matrices M, M', and P by

$$J = \sum_{(x,y)\in C_{\text{source}}} \sum_{(s,t)\in L} \alpha H(x, y) + (1 - \alpha) \cdot \text{MSE}(s, t)\,, \tag{6}$$

where α is a hyperparameter that weights sentiment loss vs. projection loss.

3.1.4 Target-language Classification

For inference, we classify sentences from a target-language corpus C_{target}. As in the training procedure, for each sentence, we take the word embeddings from the target embeddings T

3. We omit parameters in equations for better readability.
4. We also experimented with cosine distance, but found that it performed worse than Euclidean distance.
5. Our model employs a linear transformation after the averaging layer instead of including a non-linearity function. We choose this architecture because the weights M and M' are also used to learn a linear cross-lingual projection.

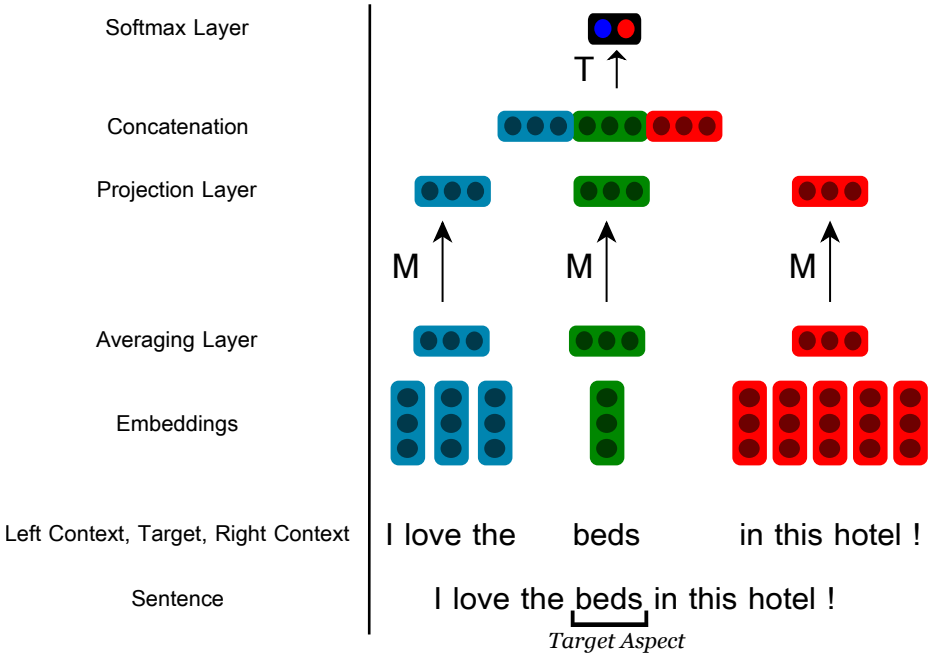

Figure 2: The Split adaptation of our Blse model to targeted sentiment analysis. At test time, we replace the matrix M with the matrix $M^{'}$.

and average them to $\mathbf{a}_i \in \mathbb{R}^d$. We then project this vector to the joint bilingual space $\hat{\mathbf{z}}_i = \mathbf{a}_i \cdot M'$. Finally, we pass $\hat{\mathbf{z}}_i$ through a softmax layer P to obtain the prediction $\hat{y}_i = \text{softmax}(\hat{\mathbf{z}}_i \cdot P)$.

3.2 Targeted Model

In our targeted model, we assume that the list of sentiment targets as they occur in the text is given. These can be extracted previously either by using domain knowledge (Liu et al., 2005), by using a named entity recognizer (Zhang et al., 2015) or by using a number of aspect extraction techniques (Zhou et al., 2012). Given these targets, the task is reduced to classification. However, what remains is how to represent the target, to learn to subselect the information from the context which is relevant, how to represent this contextual information, and how to combine these representations in a meaningful way that enables us to classify the target reliably.

Our approach to adapt the Blse model to targeted sentiment analysis, which we call Split (depicted in Figure 2), is similar to the method proposed by Zhang et al. (2016) for gated recurrent networks. For a sentence with a target a, we split the sentence at a in order to get a left and right context, $\text{con}_\ell(a)$ and $\text{con}_r(a)$ respectively.

Unlike the approach from Zhang et al. (2016), we do not use recurrent neural networks to create a feature vector, as Atrio et al. (2019) showed that, in cross-lingual setups, they overfit too much to word order and source-language specific information to perform well on our

		EN	ES	CA	EU
Binary	+	1258	1216	718	956
	−	473	256	467	173
	Total	1731	1472	1185	1129
4-class	++	379	370	256	384
	+	879	846	462	572
	−	399	218	409	153
	−−	74	38	58	20
	Total	1731	1472	1185	1129

Table 1: Statistics for the OpeNER English (EN) and Spanish (ES) as well as the Multi-Booked Catalan (CA) and Basque (EU) datasets.

tasks. Therefore, we instead average each left context $\text{con}_\ell(a_i)$, right context $\text{con}_r(a_i)$, and target a_i separately. Although averaging is a simplified approach to create a compositional representation of a phrase, it has been shown to work well for sentiment (Iyyer et al., 2015; Barnes et al., 2017). After creating a single averaged vector for the left context, right context, and target, we concatenate them and use these as input for the softmax classification layer $T \in \mathbb{R}^{d \times 3}$, where d is the dimensionality of the input vectors. The model is trained on the source language sentiment data using M to project, and then tested by replacing M with $M^{'}$, similar to the sentence-level model.

4. Experiments

In this section, we describe the resources and datasets, as well as the experimental setups used in both the sentence-level (Experiment 1 in Subsection 4.2) and targeted (Experiment 2 in Subsection 4.3) experiments.

4.1 Datasets and Resources

The number of datasets and resources for under-resourced languages are limited. Therefore, we choose a mixture of resource-rich and under-resourced languages for our experiments. We treat the resource-rich languages as if they were under-resourced by using similar amounts of parallel data.

4.1.1 Sentence-level Datasets

To evaluate our proposed model at sentence-level, we conduct experiments using four benchmark datasets and three bilingual combinations. We use the OpeNER English and Spanish datasets (Agerri et al., 2013) and the MultiBooked Catalan and Basque datasets (Barnes et al., 2018). All datasets contain hotel reviews which are annotated for targeted sentiment analysis. The labels include *Strong Negative* (−−), *Negative* (−), *Positive* (+), and *Strong*

	Spanish	Catalan	Basque
Sentences	23 M	9.6 M	0.7 M
Tokens	610 M	183 M	25 M
Embeddings	0.83 M	0.4 M	0.14 M

Table 2: Statistics for the Wikipedia corpora and monolingual vector spaces.

Positive (++). We map the aspect-level annotations to sentence level by taking the most common label and remove instances of mixed polarity. We also create a binary setup by combining the strong and weak classes. This gives us a total of six experiments. The details of the sentence-level datasets are summarized in Table 1. For each of the experiments, we take 70 percent of the data for training, 20 percent for testing and the remaining 10 percent are used as development data for tuning meta-parameters.

4.1.2 Targeted Datasets

We use the following corpora to set up the experiments in which we train on a source language corpus C_S and test on a target language corpus C_T. Statistics for all of the corpora are shown in Table 3. We include a binary classification setup, where neutral has been removed and strong positive and strong negative have been mapped to positive and negative, as well as a multiclass setup, where the original labels are used.

OpeNER Corpora: The OpeNER corpora (Agerri et al., 2013) are composed of hotel reviews, annotated for aspect-based sentiment. Each aspect is annotated with a sentiment label (Strong Positive, Positive, Negative, Strong Negative). We perform experiments with the English and Spanish versions.

MultiBooked Corpora: The MultiBooked corpora (Barnes et al., 2018) are also hotel reviews annotated in the same way as the OpeNER corpora, but in Basque and Catalan. These corpora allow us to observe how well each approach performs on low-resource languages.

SemEval 2016 Task 5: We take the English and Spanish restaurant review corpora made available by the organizers of the SemEval event (Pontiki et al., 2016). These corpora are annotated for three levels of sentiment (positive, neutral, negative).

USAGE Corpora: The USAGE corpora (Klinger & Cimiano, 2014) are Amazon reviews taken from a number of different items, and are available in English and German. Each aspect is annotated for three levels of sentiment (positive, neutral, negative). As the corpus has two sets of annotations available, we take the annotations from annotator 1 as the gold standard.

4.1.3 Resources

Monolingual Word Embeddings For Blse, VecMap, Muse, and MT, we require monolingual vector spaces for each of our languages. For English, we use the publicly

		Binary		Multiclass				
		+	−	++	+	0	−	−−
OpeNER	EN	1658	661	472	1132		556	105
	ES	2404	446	813	1591		387	59
MultiBooked	CA	1453	883	645	808		741	142
	EU	1461	314	686	775		273	41
SemEval	EN	2268	953		2268	145	953	
	ES	2675	948		2675	168	948	
USAGE	EN	2985	1456		2985	34	1456	
	DE	3115	870		3115	99	870	

Table 3: Number of aspect-polarity tuples for the targeted datasets.

available GoogleNews vectors[6]. For Spanish, Catalan, and Basque, we train skip-gram embeddings using the Word2Vec toolkit[6] with 300 dimensions, subsampling of 10^{-4}, window of 5, negative sampling of 15 based on a 2016 Wikipedia corpus[7] (sentence-split, tokenized with IXA pipes (Agerri et al., 2014) and lowercased). The statistics of the Wikipedia corpora are given in Table 2.

Bilingual Lexicon For Blse, VecMap, Muse, and Barista, we also require a bilingual lexicon. We use the sentiment lexicon from Hu and Liu (2004) (to which we refer in the following as Hu and Liu) and its translation into each target language. We translate the lexicon using Google Translate and exclude multi-word expressions.[8] This leaves a dictionary of 5700 translations in Spanish, 5271 in Catalan, and 4577 in Basque. We set aside ten percent of the translation pairs as a development set in order to check that the distances between translation pairs not seen during training are also minimized during training.

4.2 Setting for Experiment 1: Sentence-level Classification

We compare Blse (Sections 3.1.1–3.1.3) to VecMap, Muse, and Barista (Section 2) as baselines, which have similar data requirements and to machine translation (MT) and monolingual (Mono) upper bounds which request more resources. For all models (Mono, MT, VecMap, Muse, Barista), we take the average of the word embeddings in the source-language training examples and train a linear SVM[9]. We report this instead of using the same feed-forward network as in Blse as it is the stronger upper bound. We choose the parameter c on the target language development set and evaluate on the target language test set.

6. https://code.google.com/archive/p/word2vec/
7. http://attardi.github.io/wikiextractor/
8. Note that we only do that for convenience. Using a machine translation service to generate this list could easily be replaced by a manual translation, as the lexicon is comparably small.
9. LinearSVC implementation from scikit-learn.

Upper Bound Mono. We set an empirical upper bound by training and testing a linear SVM on the target language data. Specifically, we train the model on the averaged embeddings from target language training data, tuning the c parameter on the development data. We test on the target language test data.

Upper Bound MT. To test the effectiveness of machine translation, we translate all of the sentiment corpora from the target language to English using the Google Translate API[10]. Note that this approach is not considered a baseline, as we assume not to have access to high-quality machine translation for low-resource languages of interest.

Baseline Unsup We compare with the unsupervised statistical machine translation approach proposed by Artetxe et al. (2018a). This approach uses a self-supervised method to create bilingual phrase embeddings which then populates a phrase table. Monolingual n-gram language models and an unsupervised variant of MERT are used to create a MT model which is improved through iterative backtranslation. We use the Wikipedia corpora from Section 4.1.3 to create the unsupervised SMT system between English and the target languages and run the training proceedure with default parameters. Finally, we translate all test examples in the target languages to English.

Baseline VecMap. We compare with the approach proposed by Artetxe et al. (2016) which has shown promise on other tasks, *e. g.*, word similarity. In order to learn the projection matrix W, we need translation pairs. We use the same word-to-word bilingual lexicon mentioned in Section 3.1.1. We then map the source vector space S to the bilingual space $\hat{S} = SW$ and use these embeddings.

Baseline Muse. This baseline is similar to VecMap but incorporates and adversarial objective as well as a localized scaling objective, which further improve the orthogonal refinement so that the two language spaces are even more similar.

Baseline Barista. The approach proposed by Gouws and Søgaard (2015) is another appropriate baseline, as it fulfills the same data requirements as the projection methods. The bilingual lexicon used to create the pseudo-bilingual corpus is the same word-to-word bilingual lexicon mentioned in Section 3.1.1. We follow the authors' setup to create the pseudo-bilingual corpus. We create bilingual embeddings by training skip-gram embeddings using the Word2Vec toolkit on the pseudo-bilingual corpus using the same parameters from Section 4.1.3.

Our method: BLSE. Our model, Blse, is implemented in Pytorch (Paszke et al., 2016) and the word embeddings are initialized with the pretrained word embeddings S and T mentioned in Section 4.1.3. We use the word-to-word bilingual lexicon from Section 4.1.3, tune the hyperparameters α, training epochs, and batch size on the target development set and use the best hyperparameters achieved on the development set for testing. ADAM (Kingma & Ba, 2014) is used in order to minimize the average loss of the training batches.

Ensembles. In order to evaluate to what extent each projection model adds complementary information to the machine translation approach, we create an ensemble of MT and each projection method (Blse, VecMap, Muse, Barista). A random forest classifier is trained on the predictions from MT and each of these approaches.

10. `https://translate.google.com`

4.3 Setting for Experiment 2: Targeted Classification

For the targeted classification experiment, we compare the same models mentioned above, but adapted to the setting using the Split method from Section 3.2.

A simple majority baseline sets the lower bound, while the MT-based model serves as an upper bound. We assume our models to perform between these two, as we do not have access to the millions of parallel sentences required to perform high-quality MT and particularly aim at proposing a method which is less resource-hungry.

Simplified Models: Target only and Context only We hypothesize that cross-lingual approaches are particularly error-prone when evaluative phrases and words are wrongly predicted. In such settings, it might be beneficial for a model to put emphasis on the target word itself and learn a prior distribution of sentiment for each target independent of the context. For example, if you assume that all mentions of Steven Segal are negative in movie reviews, it is possible to achieve good results (Bird et al., 2009). On the other hand, it may be that there are not enough examples of target-context pairs, and that it is better to ignore the target and concentrate only on the contexts.

To analyze this, we compare our model to two simplified versions. In addition, this approach enables us to gain insight in the source of relevant information. The first is Target-only, which means that we use the model in the same way as before but ignore the context completely. This serves as a tool to understand how much model performance originates from the target itself.

In the same spirit, we use a Context-only model, which ignores the target by constraining the parameters of all target phrase embeddings to be the same. This approach might be beneficial over our initial model if the prior distribution between targets was similar and the context actually carries the relevant information.

Baseline: Sentence Assumption As the baseline for each projection method, we assume all targets in each sentence respectively to be of the same polarity (Sent). This is generally an erroneous assumption, but can give good results if all of the targets in a sentence have the same polarity. In addition, this baseline provides us with the information about whether the models are able to handle information from different positions in the text.

5. Results

5.1 Experiment 1: Sentence-level Classification

In Table 4, we report the results of all four methods. Our method outperforms the other projection methods (the baselines VecMap, Muse, and Barista) on four of the six experiments substantially. It performs only slightly worse than the more resource-costly upper bounds (MT and Mono). This is especially noticeable for the binary classification task, where Blse performs nearly as well as machine translation and significantly better than the other methods. Unsup also performs similarly to Blse on the binary tasks, while giving stronger performance on the 4-class setup. We perform approximate randomization tests (Yeh, 2000) with 10,000 runs and highlight the results that are statistically significant (*p < 0.01) in Table 4.

		Upper Bounds			Baselines				Ensemble			
		Mono	MT	**BLSE**	Unsup	VecMap	Muse	Barista	VecMap	Muse	Barista	Blse
Binary	ES	73.5	79.0	**74.6*	76.8	67.1	73.4	61.2	62.6	58.7	56.0	**80.3**
	CA	79.2	77.2	**72.9*	79.4	60.7	71.1	60.1	63.3	64.3	62.5	**85.0**
	EU	69.8	69.4	**69.3*	65.5	45.6	59.8	54.4	66.4	68.4	49.8	**73.5**
4-class	ES	45.5	48.8	*41.2*	49.1	34.9	37.1	39.5	43.8	49.3	47.1	**50.3**
	CA	49.9	52.7	35.9	47.7	23.0	39.0	36.2	47.6	52.0	53.0	**53.9**
	EU	47.1	43.6	30.0	39.3	21.3	25.8	33.8	49.9	46.4	47.8	**50.5**

Table 4: Macro F_1 of four models trained on English and tested on Spanish (ES), Catalan (CA), and Basque (EU). The **bold** numbers show the best results for each metric per column and the *highlighted* numbers show where Blse is better than the other projection methods, VecMap, Muse, and Barista (* $p < 0.01$).

In more detail, we see that MT generally performs better than the projection methods (79–69 F_1 on binary, 52–44 on 4-class). Blse (75–69 on binary, 41–30 on 4-class) has the best performance of the projection methods and is comparable with MT on the binary setup, with no significant difference on binary Basque. VecMap (67–46 on binary, 35–21 on 4-class) and Barista (61–55 on binary, 40–34 on 4-class) are significantly worse than Blse on all experiments except Catalan and Basque 4-class. Muse (67–62 on binary, 45–34 on 4-class) performs better than VecMap and Barista. On the binary experiment, VecMap outperforms Barista on Spanish (67.1 vs. 61.2) and Catalan (60.7 vs. 60.1) but suffers more than the other methods on the four-class experiments, with a maximum F_1 of 34.9. Barista is relatively stable across languages. Unsup performs well across experiments (76–65 on binary, 49–39 on 4-class), even performing better than MT on both Catalan tasks and Spanish 4-class.

The Ensemble of MT and Blse performs the best, which shows that Blse adds complementary information to MT. Finally, we note that all systems perform worse on Basque. This is presumably due to the increased morphological complexity of Basque, as well as its lack of similarity to the source language English (Section 6.4.2).

5.1.1 Model and Error Analysis

We analyze three aspects of our model in further detail: 1) where most mistakes originate, 2) the effect of the bilingual lexicon, and 3) the effect and necessity of the target-language projection matrix M'.

5.1.2 Phenomena

In order to analyze where each model struggles, we categorize the mistakes and annotate all of the test phrases with one of the following error classes: vocabulary (voc), adverbial

Model		voc	mod	neg	know	other	*total*
MT	bi	49	26	19	14	5	**113**
	4	147	94	19	21	12	**293**
Unsup	bi	65	31	21	17	7	**141**
	4	170	120	27	26	15	**358**
Muse	bi	75	38	17	18	8	**156**
	4	195	137	27	22	28	**409**
VecMap	bi	80	44	27	14	7	**172**
	4	182	141	19	24	19	**385**
Barista	bi	89	41	27	20	7	**184**
	4	191	109	24	31	15	**370**
Blse	bi	67	45	21	15	8	**156**
	4	146	125	29	22	19	**341**

Table 5: Error analysis for different phenomena for the binary (bi) and multi-class (4) setups. See text for explanation of error classes.

modifiers (mod), negation (neg), external knowledge (know) or other. Table 5 shows the results.

Vocabulary: The most common way to express sentiment in hotel reviews is through the use of polar adjectives (as in "the room was great") or the mention of certain nouns that are desirable ("it had a pool"). Although this phenomenon has the largest total number of mistakes (an average of 72 per model on binary and 172 on 4-class), it is mainly due to its prevalence. MT performed the best on the test examples which according to the annotation require a correct understanding of the vocabulary (81 F_1 on binary /54 F_1 on 4-class), with Blse (79/48) slightly worse. Muse (76/23), VecMap (70/35), and Barista (67/41) perform worse. This suggests that Blse is better than Muse, VecMap and Barista at transferring sentiment of the most important sentiment bearing words.

Negation: Negation is a well-studied phenomenon in sentiment analysis (Pang et al., 2002; Wiegand et al., 2010; Zhu et al., 2014; Reitan et al., 2015) . Therefore, we are interested in how these four models perform on phrases that include the negation of a key element, for example "In general, this hotel isn't bad". We would like our models to recognize that the combination of two negative elements "isn't" and "bad" lead to a *Positive* label.

Given the simple classification strategy, all models perform relatively well on phrases with negation (all reach nearly 60 F_1 in the binary setting). However, while Blse performs the best on negation in the binary setting (82.9 F_1), it has more problems with negation in the 4-class setting (36.9 F_1).

Adverbial Modifiers: Phrases that are modified by an adverb, *e. g.*, the food was *incredibly* good, are important for the four-class setup, as they often differentiate between the base and *Strong* labels. In the binary case, all models reach more than 55 F_1. In the

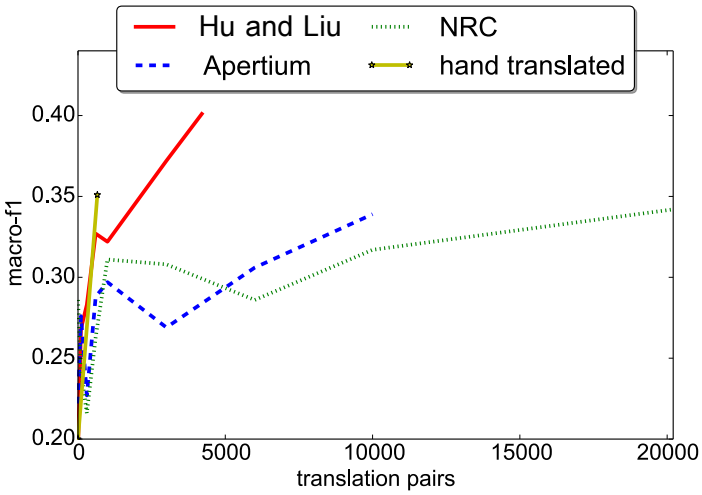

Figure 3: Macro F_1 for translation pairs in the Spanish 4-class setup. Training with the expanded hand translated lexicon and machine-translated Hu and Liu lexicon gives a macro F_1 that grows constantly with the number of translation pairs. Despite having several times more training data, the Apertium and NRC translation dictionaries do not perform as well.

4-class setup, Blse only achieves 27.2 F_1 compared to 46.6 or 31.3 of MT and Barista, respectively. Therefore, presumably, our model does currently not capture the semantics of the target adverbs well. This is likely due to the fact that it assigns too much sentiment to functional words (see Figure 6). Muse performs poorly on modified examples (20.3 F_1).

External Knowledge Required: These errors are difficult for any of the models to get correct. Many of these include numbers which imply positive or negative sentiment (350 meters from the beach is *Positive* while 3 kilometers from the beach is *Negative*). Blse performs the best (63.5 F_1) while MT performs comparably well (62.5). Barista performs the worst (43.6).

Binary vs. 4-class: All of the models suffer when moving from the binary to 4-class setting; an average of 26.8 in macro F_1 for MT, 31.4 for VecMap, 22.2 for Barista, 34.1 for Muse, and 36.6 for Blse. The vector projection methods (VecMap, Muse, and Blse) suffer the most, suggesting that they are currently more apt for the binary setting.

5.1.3 Effect of Bilingual Lexicon

We analyze how the number of translation pairs affects our model. We train on the 4-class Spanish setup using the best hyper-parameters from the previous experiment.

Research into projection techniques for bilingual word embeddings (Mikolov et al., 2013; Lazaridou et al., 2015; Artetxe et al., 2016) often uses a lexicon of the most frequent 8–10 thousand words in English and their translations as training data. We test this approach by taking the 10,000 word-to-word translations from the Apertium English-to-Spanish dic-

tionary[11]. We also use the Google Translate API to translate the NRC hashtag sentiment lexicon (Mohammad et al., 2013) and keep the 22,984 word-to-word translations. We perform the same experiment as above and vary the amount of training data from 0, 100, 300, 600, 1000, 3000, 6000, 10,000 up to 20,000 training pairs. Finally, we compile a small hand translated dictionary of 200 pairs, which we then expand using target language morphological information, finally giving us 657 translation pairs[12]. The macro F_1 score for the Hu and Liu dictionary climbs constantly with the increasing translation pairs. Both the Apertium and NRC dictionaries perform worse than the translated lexicon by Hu and Liu, while the expanded hand translated dictionary is competitive, as shown in Figure 3.

While for some tasks, *e. g.*, bilingual lexicon induction, using the most frequent words as translation pairs is an effective approach, for sentiment analysis, this does not seem to help. Using a translated sentiment lexicon, even if it is small, gives better results.

5.1.4 Analysis of M'

The main motivation for using two projection matrices M and M' is to allow the original embeddings to remain stable, while the projection matrices have the flexibility to align translations and separate these into distinct sentiment subspaces. To justify this design decision empirically, we perform an experiment to evaluate the actual need for the target language projection matrix M': We create a simplified version of our model without M', using M to project from the source to target and then P to classify sentiment.

The results of this model are shown in Figure 4. The modified model does learn to predict in the source language, but not in the target language. This confirms that M' is necessary to transfer sentiment in our model.

5.1.5 No projection

Additionally, we provide an analysis of a similar model to ours, but which uses $M = \mathbb{R}^{d,o}$ and $M' = \mathbb{R}^{d',o}$, where d (d') is the dimensionality of the original embeddings and o is the label size, to directly model crosslingual sentiment, such that the final objective function is

$$J = \sum_{(x,y)\in C_{\text{source}}} \sum_{(s,t)\in L} \alpha \cdot H(x,y) + (1-\alpha) \cdot ||M \cdot s - M' \cdot t|| \tag{7}$$

thereby simplifying the model and removing the P parameter. Table 6 shows that Blse outperforms this simplified model on all tasks.

5.1.6 Qualitative Analyses of Joint Bilingual Sentiment Space

In order to understand how well our model transfers sentiment information to the target language, we perform two qualitative analyses. First, we collect two sets of 100 positive sentiment words and one set of 100 negative sentiment words. An effective cross-lingual sentiment classifier using embeddings should learn that two positive words should be closer in the shared bilingual space than a positive word and a negative word. We test if Blse

11. `http://www.meta-share.org`
12. The translation took approximately one hour. We can extrapolate that manually translating a sentiment lexicon the size of the Hu and Liu lexicon would take no more than 5 hours.

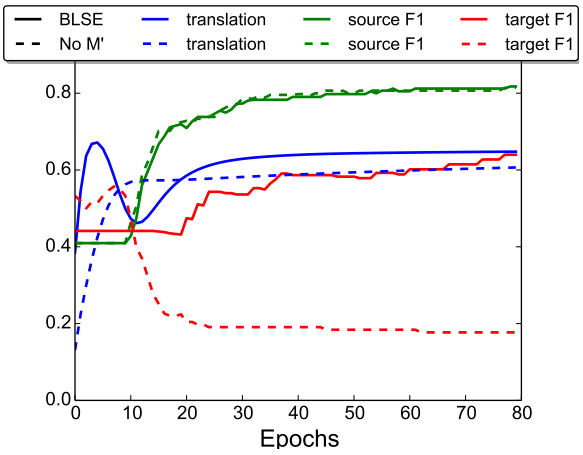

Figure 4: Blse model (solid lines) compared to a variant without target language projection matrix M' (dashed lines). "Translation" lines show the average cosine similarity between translation pairs. The remaining lines show F_1 scores for the source and target language with both variants of Blse. The modified model cannot learn to predict sentiment in the target language (red lines). This illustrates the need for the second projection matrix M'.

		Blse	no proj.
binary	ES	74.6	52.0
	CA	72.9	48.3
	EU	69.3	49.1
4-class	ES	41.2	21.3
	CA	35.9	18.3
	EU	30.0	17.0

Table 6: An empirical comparison of Blse and a simplified model which directly projects the embeddings to the sentiment classes. Blse outperforms the simplified model on all tasks.

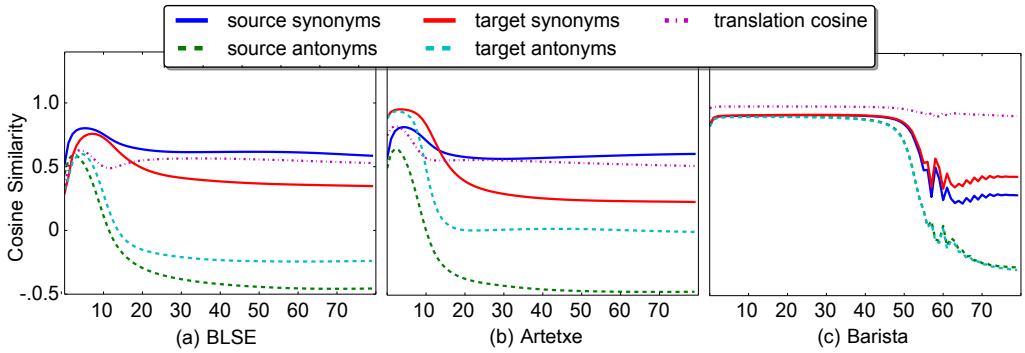

Figure 5: Average cosine similarity between a subsample of translation pairs of same polarity ("sentiment synonyms") and of opposing polarity ("sentiment antonyms") in both target and source languages in each model. The x-axis shows training epochs. We see that Blse is able to learn that sentiment synonyms should be close to one another in vector space and sentiment antonyms should not.

is able to do this by training our model and after every epoch observing the mean cosine similarity between the sentiment synonyms and sentiment antonyms after projecting to the joint space.

We compare Blse with VecMap and Barista by replacing the Linear SVM classifiers with the same multi-layer classifier used in Blse and observing the distances in the hidden layer. Figure 5 shows this similarity in both source and target language, along with the mean cosine similarity between a held-out set of translation pairs and the macro F_1 scores on the development set for both source and target languages for Blse, Barista, and VecMap. From this plot, it is clear that Blse is able to learn that sentiment synonyms should be close to one another in vector space and antonyms should have a negative cosine similarity. While the other models also learn this to some degree, jointly optimizing both sentiment and projection gives better results.

Secondly, we would like to know how well the projected vectors compare to the original space. Our hypothesis is that some relatedness and similarity information is lost during projection. Therefore, we visualize six categories of words in t-SNE, which projects high dimensional representations to lower dimensional spaces while preserving the relationships as best as possible (Van der Maaten & Hinton, 2008): positive sentiment words, negative sentiment words, functional words, verbs, animals, and transport.

The t-SNE plots in Figure 6 show that the positive and negative sentiment words are rather clearly separated after projection in Blse. This indicates that we are able to incorporate sentiment information into our target language without any labeled data in the target language. However, the downside of this is that functional words and transportation words are highly correlated with positive sentiment.

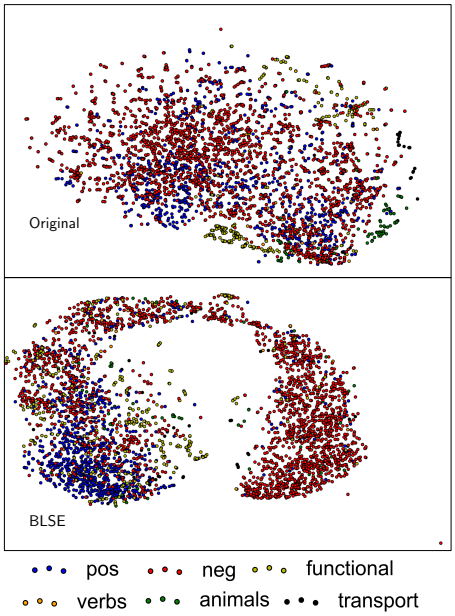

Figure 6: t-SNE-based visualization of the Spanish vector space before and after projection with Blse. There is a clear separation of positive and negative words after projection, despite the fact that we have used no labeled data in Spanish.

5.1.7 Analysis of α parameter

Finally, in order to analyze the sensitivity of the alpha parameter, we train Blse models for 30 epochs each with α between 0 and 1. Figure 7 shows the average cosine similarity for the translation pairs, as well as macro F_1 for both source and target language development data.

Values near 0 lead to poor translation and consecuently poor target language transfer. There is a rather large "sweet spot" where all measures perform best and finally, the translation is optimized to the detriment of sentiment prediction in both source and target languages with values near 1.

5.1.8 Discussion

The experiments in this section have proven that it is possible to perform cross-lingual sentiment analysis without machine translation, and that jointly learning to project and predict sentiment is advantageous. This supports the growing trend of jointly training for multiple objectives (Tang et al., 2014; Klinger & Cimiano, 2015; Ferreira et al., 2016).

This approach has also been exploited within the framework of multi-task learning, where a model learns to perform multiple similar tasks in order to improve on a final task (Collobert et al., 2011). The main difference between the joint method proposed here and multi-task learning is that vector space projection and sentiment classification are not similar enough tasks to help each other. In fact, these two objectives compete against

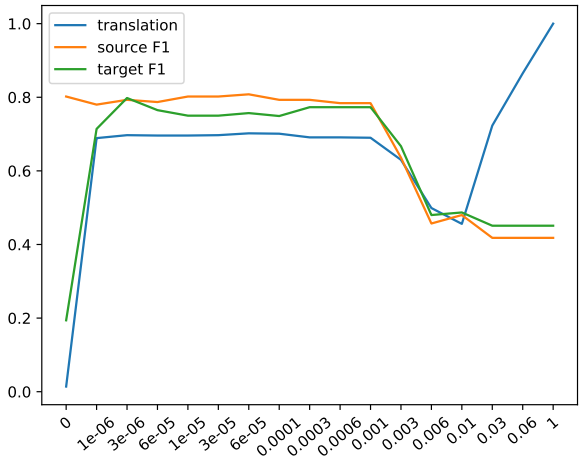

Figure 7: An analysis of the α parameter of Blse showing cosine similarity of translation pairs and macro F_1 for source and target development data. The optimal values range from 1×10^{-6} to 1×10^{-3}.

one another, as a perfect projection would not contain enough information for sentiment classification, and vice versa.

5.2 Experiment 2: Targeted Classification

Table 7 shows the macro F_1 scores for all cross-lingual approaches (Blse, VecMap, Muse, Barista, MT, Unsup) and all targeted approaches (Sent, Split, Context-only, and Target-only). The final column is the average over all corpora. The final row in each setup shows the macro F_1 for a classifier that always chooses the majority class.

Blse outperforms other projection methods on the binary setup, 63.0 macro averaged F_1 across corpora versus 59.0, 57.9, and 51.4 for VecMap, Muse, and Barista, respectively. On the multiclass setup, however, Muse (32.2 F_1) is the best, followed by VecMap (31.0), Barista (28.1) and Blse (23.7). Unsup performs well across all experiments, achieving the best results on OpeNER ES (73.2 on binary and 42.7 on multiclass) and SemEval binary (77.1). VecMap is never the best nor the worst approach. In general, Barista performs poorly on the binary setup, but slightly better on the multiclass, although the overall performance is still weak. These results are similar to those observed in Experiment 1 for sentence classification.

The Split approach to ABSA improves over the Sent baseline on 33 of 50 experiments, especially on binary (21/25), while on multiclass it is less helpful (13/25). Both Sent and Split normally outperform Context-only or Target-only approaches. This confirms the intuition that it is important to take both context and target information for classification. Additionally, the Context-only approach always performs better than

			EN-ES OpeNER	EN-CA MultiBooked	EN-EU MultiBooked	EN-ES SemEval	EN-DE USAGE	Average
Binary	Sent	Blse	64.4	47.3	45.5	61.1	63.8	56.4
		VecMap	52.2	41.8	39.1	42.3	31.2	51.3
		Muse	47.6	40.1	45.8	45.3	47.5	45.3
		Barista	47.3	39.1	45.8	42.3	33.4	41.6
		MT	70.8	**81.5**	**76.2**	70.9	58.8	71.6
		Unsup	**73.7**	69.8	70.1	66.1	-	-
	Split	Blse	*66.8*	*69.8*	*66.3*	*62.2*	50.0	*63.0*
		VecMap	65.8	64.4	65.1	60.0	39.9	59.0
		Muse	58.3	64.3	50.2	59.8	*57.0*	57.9
		Barista	61.9	59.0	56.1	44.5	35.3	51.4
		MT	67.3	77.8	74.8	73.2	69.4	72.5
		Unsup	71.6	73.5	64.0	**77.1**	-	-
	Context-only	Blse	47.3	39.1	45.8	42.3	55.9	46.1
		VecMap	47.3	39.1	45.8	42.3	45.8	44.1
		Muse	55.5	67.5	52.1	61.6	45.4	56.4
		Barista	47.3	60.2	51.9	42.3	45.5	49.4
		MT	66.5	78.1	72.4	74.2	**73.1**	**72.9**
		Unsup	69.9	72.3	63.6	75.5	-	-
	Target-only	Blse	53.1	43.7	42.7	42.3	41.5	44.7
		VecMap	54.4	51.1	35.4	45.5	45.2	46.3
		Muse	56.2	55.4	52.3	46.0	47.5	51.5
		Barista	48.9	53.0	48.5	42.3	44.8	47.5
		MT	46.7	40.1	45.8	47.5	56.0	47.2
		Unsup	52.4	51.0	52.1	43.7	-	-
	Maj.		47.3	39.1	45.8	42.3	43.0	43.5
Multiclass	Sent	Blse	25.2	23.3	16.6	36.0	**40.5**	28.3
		VecMap	28.1	19.9	26.3	28.2	28.3	26.2
		Muse	22.4	23.2	23.5	27.4	24.1	24.1
		Barista	29.3	35.8	27.0	27.4	29.9	29.9
		MT	41.4	**46.5**	**44.3**	33.1	28.9	**38.8**
		Unsup	**42.7**	37.7	37.6	32.5	-	-
	Split	Blse	18.5	14.3	15.7	**40.6**	29.5	23.7
		VecMap	29.2	30.9	28.0	38.9	27.9	31.0
		Muse	*32.9*	*33.5*	*27.3*	27.4	39.7	*32.2*
		Barista	27.9	35.1	27.3	27.4	33.4	28.1
		MT	24.7	29.2	27.0	33.8	33.2	29.6
		Unsup	28.9	26.9	23.9	31.7	-	-
	Context-only	Blse	18.5	12.6	15.7	27.4	38.4	22.5
		VecMap	18.5	12.6	15.7	27.4	28.3	20.5
		Muse	22.7	39.0	27.4	27.4	30.0	29.3
		Barista	32.9	31.6	27.2	27.4	32.1	30.2
		MT	27.5	31.4	27.2	30.6	34.4	30.2
		Unsup	29.4	27.7	23.9	32.8	-	-
	Target-only	Blse	19.1	17.3	16.7	27.4	25.3	21.2
		VecMap	25.8	23.1	19.0	32.1	25.3	25.1
		Muse	23.2	21.6	17.1	29.5	31.1	24.5
		Barista	21.8	21.5	16.8	27.4	33.9	24.3
		MT	26.9	23.3	23.9	30.5	33.6	27.6
		Unsup	22.9	18.7	21.2	19.7	-	-
	Maj.		18.5	12.6	15.7	27.4	28.3	20.5

Table 7: Macro F_1 results for all corpora and techniques. We denote the best performing projection-based method per column with a *blue box* and the best overall method per column with a **green box**.

	correct	incorrect
Blse	2.1	2.5
VecMap	2.5	2.1
Muse	2.1	2.2
Barista	1.7	2.2
MT	2.1	2.2
Unsup	2.1	2.2

Table 8: Average length of tokens of correctly and incorrectly classified targets on the OpeNER Spanish binary corpus.

Target-only, which indicates that context is more important than the prior probability of an target being positive or negative.

Unlike the projection methods, MT using only the Sent representation performs well on the OpeNER and MultiBooked datasets, while suffering more on the SemEval and USAGE datasets. This is explained by the percentage of sentences that contain contrasting polarities in each dataset: between 8 and 12% for the OpeNER and Multibooked datasets, compared to 29% for SemEval or 50% for USAGE. In sentences with multiple contrasting polarities, the Sent baseline performs poorly.

Finally, the general level of performance of projection-based targeted cross-lingual sentiment classification systems shows that they still lag 10+ percentage points behind MT on binary (compare MT (72.9 F_1) with Blse (63.0)), and 6+ percentage points on multiclass (MT (38.8) versus Muse (32.2)). The gap between MT and projection-based approaches is therefore larger on targeted sentiment analysis than at sentence-level.

5.2.1 Error Analysis

We perform a manual analysis of the targets misclassified by all systems on the OpeNER Spanish binary corpus (see Table 8), and found that the average length of misclassified targets is slightly higher than that of correctly classified targets, except for with VecMap. This indicates that averaging may have a detrimental effect as the size of the targets increases.

With the MT upperbounds, there is a non-negligible amount of noise introduced by targets which have been incorrectly translated (0.05% OpeNER ES, 6% MultiBooked EU, 2% CA, 2.5% SemEval, 1% USAGE). We hypothesize that this is why MT with Context-only performs better than MT with Split. This motivates further research with projection-based methods, as they do not suffer from translation errors.

The confusion matrices of the models on the SemEval task, shown in Figure 8, show that on the multilabel task, models are not able to learn the neutral class. This derives from the large class imbalance found in the data (see Table 3). Similarly, models do not learn the Strong Negative class on the OpeNER and MultiBooked datasets.

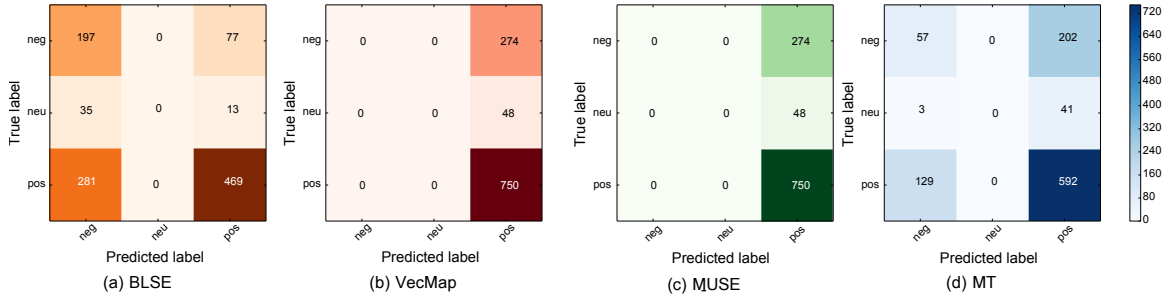

Figure 8: Confusion matrices for all Split models on the SemEval task.

6. Case Study: Real World Deployment

6.1 Motivation

The performance of machine learning models on different target languages depends on the amount of data available, the quality of the data, and characteristics of the target language, *e. g.*, morphological complexity. In the following, we analyze these aspects. There has been previous work that has observed target-language specific differences in multilingual dependency parsing (Agić et al., 2016), machine translation (Johnson et al., 2017), and language modeling (Cotterell et al., 2018; Gerz et al., 2018). We are not aware of any work in cross-lingual sentiment analysis that explores the relationship between target language and performance in such depth and aim at improving this situation in the following.

Additionally, the effect of domain differences when performing cross-lingual tasks has not been studied in depth. Hangya et al. (2018) propose domain adaptation methods for cross-lingual sentiment classification and bilingual dictionary induction. They show that creating domain-specific cross-lingual embeddings improves the classification for English-Spanish. However, the source-language training data used to train the sentiment classifier is taken from the same domain as the target-language test data. Therefore, it is not clear what the effect of using source-language training data from different domains would be. We analyzed the model presented in Section 3.1 in a domain adaptation setup, including the impact of domain differences (Barnes et al., 2018b). The main result was that our model performs particularly well on more distant domains, while other approaches (Chen et al., 2012; Ziser & Reichart, 2017) performed better when the source and target domains were not too dissimilar.

In the following, we transfer this analysis to the target-based projection model in a real-world case study which mimics a user searching for the sentiment on touristic attractions. In order to analyze how well these methods generalize to new languages and domains, we deploy the targeted Blse, Muse, VecMap and MT models on tweets in ten Western European languages with training data from three different domains. Additionally, we include experiments with the Unsup models for a subset of the languages. English is the source language in all experiments, and we test on each of the ten target languages and attempt to answer the following research questions:

- How much does the amount of monolingual data available to create the original embeddings effect the final results?
- How do features of the target language, *i. e.* similarity to source language or morphological complexity, affect the performance?
- How do domain mismatches between source-language training and target-language test data affect the performance?

Section 6.4 addresses our findings regarding these questions and demonstrates that 1) the amount of monolingual data does not correlate with classification results, 2) language similarity between the source and target languages based on word and character n-gram distributions predicts the performance of Blse on new datasets, and 3) domain mismatch has more of an effect on the multiclass setup than binary.

6.2 Experimental Setup

6.2.1 Setup

We collect tweets directed at a number of tourist attractions in European cities using the Twitter API in 10 European languages, including several under-resourced languages (English, Basque, Catalan, Galician, French, Italian, Dutch, German, Danish, Swedish, and Norwegian). We detail the data collection and annotation procedures in Section 6.2.2. For classification, we compare MT the best performing projection-based methods (Blse, Muse, VecMap) using the Split method, detailed further in Section 6.2.4. As we need monolingual embeddings for all projection-based approaches, we create skipgram embeddings from Wikipedia dumps, detailed in Section 6.2.3.

6.2.2 Data Collection

As an experimental setting to measure the effectiveness of targeted cross-lingual sentiment models on a large number of languages, we collect and annotate small datasets from Twitter for each of the target languages, as well as a larger dataset to train the models in English. While it would be possible to only concentrate our efforts on languages with existing datasets in order to enable evaluation, this could give a distorted view of how well these models generalize. In order to reduce the possible ambiguity of the tourist attractions, we do not include those that have two or more obvious senses, *e. g.*, Barcelona could refer either to the city or the football team.

In order to obtain a varied sample of tweets with subjective opinions, we download tweets that contain mentions of these tourist attractions as well as one of several emoticons or keywords[13]. This distant supervision technique has been used to create sentiment lexicons (Mohammad et al., 2016), semi-supervised training data (Felbo et al., 2017), and features for a classifier (Turney & Littman, 2003). We then remove any tweets that are less than 7 words long or which contain more than 3 hashtags or mentions. This increases the probability that a tweet text contains sufficient information for our use case setting.

13. The emoticons and keywords we used were ":)", ":(", "good", "bad", and the translations of these last two words into each target language.

Canonical name	Search terms	City
The Sagrada Familia Church	sagrada familia	Barcelona
Güell Park	parc güell	Barcelona
La Boqueria Market	la boqueria	Barcelona
Tibidabo Theme Park	tibidabo	Barcelona
Santiago de Compostela	santiago de compostela	Santiago de Compostela
The Guggenheim Museum Bilbao	guggenheim bilbao	Bilbao
Txindoki Mountain	txindoki	Basque Country
Anboto Mountain	anboto	Basque Country
The Eiffel Tower	tour eiffel; eiffel tower; torre eiffel; eiffel dorrea; eiffelturm; eiffeltårnet; eiffeltornet; eiffeltoren	Paris
The Louvre museum	louvre	Paris
The Champs-Élysées	champs-élysées	Paris
Big Ben tower	big ben	London
The London Eye	london eye	London
Buckingham Palace	buckingham palace; palacio buckingham; palau buckingham; jauregia buckingham	London
Akershus Castle Oslo	akerhus slot; akerhus fortress; fortaleza akershus; fortezza akershus; fortalesa akershus; akershus gotorlekua; festung akershus; akershus fästning	Oslo
The Oslo Viking Ship Museum	vikingskipshuset oslo; oslo viking ship museum; museo de barcos vikingos oslo; museu de vaixells víkings Oslo; oslo itsasontzi bikingoen museoa; Musée des navires vikings Oslo	Oslo
The Gamla Stan Stockholm	gamla stan stockholm; gamla stan estocolmo; gamla stan estocolm	Stockholm

Table 9: Touristic targets used as tweet search criteria.

We manually annotate all tweets for its polarity toward the target to insure the quality of the data[14]. Note that we only annotate the sentiment towards the predefined list of targets, which leads to a single annotated target per tweet. Any tweets that have unclear polarity towards the target are assigned a neutral label. This produces the three class setup that is commonly used in the SemEval tasks (Nakov et al., 2013, 2016). Annotators were master's and doctoral students between 27 and 35 years old. All had either native or C1 level fluency in the languages of interest. Finally, for a subset of tweets in English, Catalan, and Basque two annotators classify each tweet. Table 11 shows three example tweets from English.

14. Data is available at
`https://github.com/jbarnesspain/targeted_blse/tree/master/case_study/datasets`

	EN	EU	CA	GL	IT	FR	NL	DE	NO	SV	DA
+	388	40	88	27	63	51	30	72	47	40	34
0	645	93	165	57	103	140	48	125	80	93	77
−	251	9	47	15	56	66	8	48	10	20	11
IAA	0.62	0.60	0.61	—	—	—	—	—	—	—	—

Table 10: Statistics of Tweet corpora collected for the deployment study, as well as inter-annotator agreement for English, Basque, and Catalan calculated with Cohen's κ.

Text	Label
I'm so jealous. I want to visit the Sagrada Familia!	Positive
just visited yoko ono's works at the bilbao guggenheim museum	Neutral
low points: I visited the Buckingham Palace	Negative

Table 11: Three example tweets in English. The underlined phrases are the targets.

Table 10 depicts the number of annotated targets for all languages, as well as inter-annotator agreement using Cohen's κ. The neutral class is the largest in all languages, followed by positive, and negative. These distributions are similar to those found in other Twitter crawled datasets (Nakov et al., 2013, 2016). We calculate pairwise agreement on a subset of languages using Cohen's κ. The scores reflect a good level of agreement (0.62, 0.60, and 0.61 for English, Basque, and Catalan, respectively).

6.2.3 Embeddings

We collect Wikipedia dumps for ten languages; namely, Basque, Catalan, Galician, French, Italian, Dutch, German, Danish, Swedish, and Norwegian. We then preprocess them using the Wikiextractor script[15], and sentence and word tokenize them with either IXA pipes (Agerri et al., 2014) (Basque, Galician, Italian, Dutch, and French), Freeling (Padró et al., 2010) (Catalan), or NLTK (Loper & Bird, 2002) (Norwegian, Swedish, Danish).

For each language we create Skip-gram embeddings with the word2vec toolkit following the pipeline and parameters described in Section 4.1.3. This process gives us 300 dimensional vectors trained on similar data for all languages. We assume that any large differences in the embedding spaces derive from the size of the data and the characteristics of the language itself. Following the same criteria laid out in Section 4.1.3, we create projection dictionaries by translating the Hu and Liu dictionary (Hu & Liu, 2004) to each of the target languages and keeping only translations that are single word to single word. The statistics of all Wikipedia corpora, embeddings, and projection dictionaries are shown in Table 12.

15. http://attardi.github.io/wikiextractor/

Type	Measurement	EU	CA	GL	IT	FR	NL	DE	NO	SV	DA
Wiki	sents. (M)	3.1	9.6	2.5	23.7	39.1	19.4	53.7	6.8	35.9	3.6
	tokens (M)	47.9	143.7	51.0	519.6	771.8	327.3	902,1	110.5	457.3	64.4
Emb	vocab. (k)	246.0	400.9	178.6	729.4	967.7	877.9	2,102.7	443.3	1,346.7	294.6
	dimension	300	300	300	300	300	300	300	300	300	300
Dict	pairs	4,616	5,271	6,297	5,683	5,383	5,700	6,391	5,177	5,344	5,007

Table 12: Statistics of Wikipedia corpora, embeddings, and projection dictionaries (M denotes million, k denotes thousand).

6.2.4 Experiments

Since we predetermine the sentiment target for each tweet, we can perform targeted experiments without further annotation. We use the Split models described in Section 3.2. Our model is the targeted Blse models described in Section 3.2. Additionally, we compare to the targeted Muse, VecMap, and MT models, as well as an Ensemble classifier that uses the predictions from Blse and MT before taking the largest predicted class for classification (see Section 4.2 for details). Finally, we set a majority baseline by assigning the most common label (neutral) to all predictions. All models are trained for 300 epochs with a learning rate of 0.001 and α of 0.3.

We train the five models on the English data compiled during this study, as well as on the USAGE, and SemEval English data (the details can be found in Table 3) and test the models on the target-language test set.

6.3 Results

Table 13 shows the macro F_1 scores for all cross-lingual targeted sentiment approaches (Blse, Muse, VecMap, MT) trained on English data and tested on the target-language using the Split method proposed in 3.2. The final column is the average over all languages. Given the results from the earlier experiments, we hypothesize that MT should outperform Muse, VecMap and Blse for most of the languages.

On the binary setup, Blse outperforms all other cross-lingual methods including MT and Unsup, with 56.0 macro averaged F_1 across languages versus 48.7, 49.4, and 48.9 for Muse, VecMap, and MT respectively (54.1 across Basque and Catalan versus 46.0 for Unsup). Blse performs particularly well on Catalan (54.5), Italian (63.4), Swedish (65.3), and Danish (68.3). VecMap performs poorly on Galician (33.3), Italian (38.2), and Danish (43.4), but outperforms all other methods on Basque (56.4), Dutch (55.2) and Norwegian (59.0). MT performs worse than Blse and VecMap, although it does perform best for Galician (56.5). Unlike experiments in Section 3.1, the ensemble approach does not perform better than the individual classifiers and Muse leads to the classifier with the lowest performance overall. Unsup performs better than MT on both Basque and Catalan.

On the multiclass setup, however, MT (36.6 F_1) is the best, followed by VecMap (34.1), Blse (32.6), and Muse (26.1). Compared to the experiments on hotel reviews, the average

	Training Data	Model	EU	CA	GL	IT	FR	NL	DE	NO	SV	DA	avg.
Binary		maj. class	46.0	39.5	38.8	34.6	36.1	44.1	37.5	43.4	45.2	40.0	40.5
	Twitter	Blse	53.7	54.5	52.0	**63.4**	49.2	44.1	53.4	56.4	**65.3**	**68.3**	**56.0**
		Muse	53.7	50.0	55.9	49.5	40.6	18.3	51.5	47.9	52.0	67.4	48.7
		VecMap	56.4	48.1	33.3	38.2	48.6	55.2	51.0	59.0	60.5	43.4	49.4
		MT	41.3	41.4	**56.5**	39.7	54.5	43.3	55.1	52.2	49.8	55.6	48.9
		Unsup	44.7	47.3	-	-	-	-	-	-	-	-	-
		Ensemble	40.5	42.5	41.8	44.2	54.5	44.1	53.0	53.9	52.2	46.7	47.4
	USAGE	Blse	36.4	44.5	46.8	59.4	50.4	52.2	44.6	57.7	65.2	44.3	50.1
		Muse	13.0	31.5	34.9	63.3	43.0	17.4	35.2	25.7	56.9	31.2	35.2
		VecMap	32.9	45.9	35.2	49.8	42.3	49.0	47.3	**59.2**	33.3	44.3	43.9
		MT	49.1	54.3	53.5	58.1	49.8	21.1	55.5	41.4	49.0	45.1	47.7
		Unsup	56.8	48.33	-	-	-	-	-	-	-	-	-
		Ensemble	48.2	55.5	42.7	57.1	50.4	28.9	53.3	48.4	44.5	48.0	47.7
	SemEval	Blse	31.6	55.3	37.9	47.9	56.4	70.3	58.3	43.4	44.5	47.9	49.3
		Muse	18.2	**69.8**	56.6	53.6	63.8	**87.5**	**59.9**	36.7	50.0	57.2	55.3
		VecMap	**59.8**	59.0	45.6	55.3	**60.0**	55.9	39.7	43.4	48.2	40.0	50.7
		MT	57.0	58.7	40.5	58.2	49.0	61.6	57.6	40.3	53.8	50.8	52.8
		Unsup	46.0	50.0	-	-	-	-	-	-	-	-	-
		Ensemble	46.0	47.2	36.9	44.4	37.3	62.8	54.9	41.1	59.3	42.7	47.3
		Average	43.8	49.6	44.3	51.0	49.1	47.2	50.5	46.9	51.9	45.8	

	Training Data	Model	EU	CA	GL	IT	FR	NL	DE	NO	SV	DA	avg.
Multi-class		maj. class	26.6	23.7	24.3	21.1	23.5	23.9	22.5	26.1	24.6	25.2	24.1
	Twitter	Blse	32.6	**35.9**	30.1	26.7	28.0	28.7	36.9	41.4	40.9	24.3	32.6
		Muse	28.3	24.4	31.2	22.2	29.4	23.9	22.5	26.1	26.7	26.1	26.1
		VecMap	26.5	30.2	**39.6**	26.7	**37.2**	34.6	**39.8**	31.7	33.4	**41.0**	34.1
		MT	37.3	34.1	33.9	35.6	35.6	**35.9**	32.5	**43.2**	38.6	39.6	**36.6**
		Unsup	40.1	28.5	-	-	-	-	-	-	-	-	-
		Ensemble	**41.5**	30.5	36.5	26.9	36.3	31.9	30.9	37.9	**42.8**	36.3	35.1
	USAGE	Blse	11.9	15.2	21.4	31.4	22.5	20.1	18.9	23.2	22.0	14.6	20.1
		Muse	3.3	21.1	15.6	28.9	18.9	5.9	18.6	14.3	24.2	16.8	16.8
		VecMap	14.6	17.4	13.7	**49.3**	17.0	20.1	20.0	12.3	13.7	24.2	20.2
		MT	19.8	22.5	23.9	26.2	18.3	10.2	24.8	19.4	16.1	13.2	19.4
		Unsup	18.9	21.5	-	-	-	-	-	-	-	-	-
		Ensemble	16.4	20.9	18.6	27.4	19.9	11.7	24.4	22.6	23.4	15.5	20.1
	SemEval	Blse	13.6	24.9	13.8	20.4	24.9	26.9	24.6	18.5	18.7	19.2	20.6
		Muse	9.5	28.9	21.1	25.6	25.2	21.2	25.2	17.9	17.8	20.5	21.3
		VecMap	14.7	25.6	13.8	31.6	22.7	17.2	16.7	22.3	40.8	14.3	22.0
		MT	15.2	24.0	19.1	26.2	20.0	25.1	26.8	20.6	19.2	15.7	21.2
		Unsup	15.1	17.9	-	-	-	-	-	-	-	-	-
		Ensemble	14.9	15.5	13.8	15.9	16.3	19.8	20.3	17.1	15.5	21.0	17.0
		Average	21.1	24.4	23.2	27.6	24.7	22.3	25.3	24.7	26.2	23.0	

Table 13: Macro F_1 of targeted cross-lingual models on Twitter data in 10 target languages. Twitter refers to models that have been trained on the English data mentioned in Table 10, while USAGE and SemEval are trained on the English data from the datasets mentioned in Section 4.1.2.

differences between models is small (2.5 percentage points between MT and VecMap, and 1.5 between VecMap and Blse). Unsup performs better than MT on Basque (40.1), but worse on Catalan (28.5). Again, all methods outperform the majority baseline.

On both the binary and multiclass setups, the best overall results are obtained by testing and training on data from the same domain (56.0 F_1 for Blse and 36.6 F_1 for MT). Training MT, Muse, and VecMap on the SemEval data performs better than training on USAGE, however.

An initial error analysis shows that all models suffer greatly on the negative class. This seems to suggest that negative polarity towards a target is more difficult to determine within these frameworks. A significant amount of the tweets that have negative polarity towards a target also express positive or neutral sentiment towards other targets. The averaging approach to create the context vectors does not currently allow any of the models to exclude this information, leading to poor performance on these instances.

Finally, compared to the experiments performed on hotel and product reviews in Section 4, the noisy data from Twitter is more difficult to classify. Despite the rather strong majority baseline (an average of 40.5 Macro F_1 on binary), no model achieves more than an average of 56 Macro F_1 on the binary task. A marked difference is that Blse and VecMap outperform MT on the binary setup. Unlike the previous experiment, Muse performs the worst on the multiclass setup. The other projection methods obtain multiclass results similar to the previous experiment (32.6–34.1 F_1 here compared to 23.7–31.0 F_1 previously).

6.4 Discussion

In this section, we present an error analysis. Specifically, Table 14 shows examples where Blse correctly predicts the polarity of a tweet that MT and Unsup incorrectly predict, and vice versa, as well as examples where all models are incorrect.

In general, in examples where Blse outperforms MT and Unsup, the translation-based approaches often mistranslate important sentiment words, which leads to prediction errors. In the first Basque tweet, for example, "#txindoki igo gabe ere inguruaz goza daiteke... zuek joan tontorrera eta utzi arraroei gure kasa...", Unsup incorrectly translates the most important sentiment word in the tweet "goza" (*enjoy*) to "overlook" and subsequently incorrectly predicts that the polarity towards txindoki is negative.

Tweets that contain many out-of-vocabulary words or non-standard spelling (due to dialectal differences, informal writing, etc.), such as the third tweet in Table 14, "kanpora jun barik ehko asko: anboto, txindoki", are challenging for all models. In this example "jun" is a non-standard spelling of "joan" (*go*), "barik" is a Bizcayan Basque variant of "gabe" (*without*) , and "ehko" is an abbreviation of "Euskal Herriko" (*Basque Country's*). These lead to poor translations for MT and Unsup, but pose a similar out-of-vocabulary problem for Blse.

In order to give a more qualitative view of the targeted model, Figure 9 shows t-sne projections of the bilingual vector space before and after training on the Basque binary task, following the same proceedure mentioned in Section 5.1.6. As in the sentence-level experiment, there is a separation of the positive and negative sentiment words, although it is less clear for targeted sentiment. This is not surprising, as a targeted model must learn

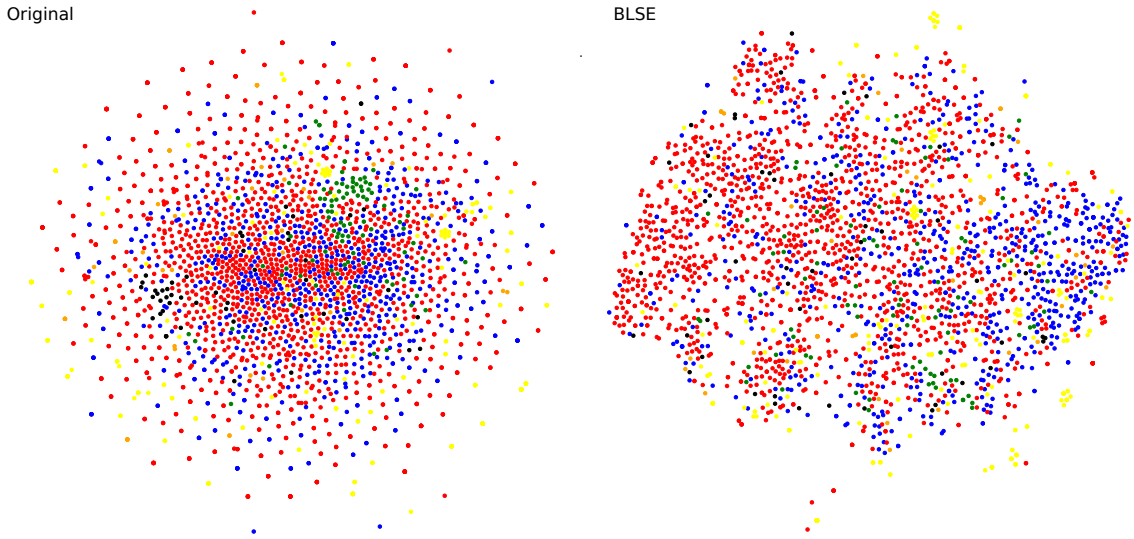

Figure 9: t-SNE-based visualization of the Basque vector space before and after projection with the targeted Blse. The positive and negative sentiment words are separated, although it is less clearly defined at target-level.

not only the prior polarity of words, but how they interact with targets, leading to a more context-dependent representation of sentiment words.

Finally, we further analyze the effects of three variables that are present in cross-lingual sentiment analysis: a) availability of monolingual unlabeled data, b) similarity of source and target languages, and c) domain shift between the source language training data and the target language test data.

6.4.1 Availability of Monolingual Unlabeled Data

We pose the question of what the relationship is between the amount of available monolingual data to create the embedding spaces and the classification results of the models. If the original word embedding spaces are not of high quality, this could make it difficult for the projection-based models to create useful features. In order to test this, we perform ablation experiments by training target-language embeddings on varying amounts of data (1×10^4 to 5×10^9 tokens) and testing the models replacing the full target-language embeddings with these. We plot the performance of the models as a function of available monolingual data in Figure 10.

Figure 10 shows that nearly all models, with the exception of Norwegian, perform poorly with very limited monolingual training data (1×10^4) and improve, although erratically, with more training data. Interestingly, the models require little data to achieve results comparable to using the all tokens to train the embeddings. A statistical analysis of the amount

Model	Tweet	Label
Blse	#txindoki igo gabe ere inguruaz goza daiteke... zuek joan tontorrera eta utzi arraroei gure kasa...	pos
MT	#txindoki it can also be enjoyed by the surrounding area... you go to the summit and leave it strange to us...	neg
Unsup	Without falling also inguruaz overlook... you are rose summit and left arraroei on our own...	neg
Ref.	Even without climbing Txindoki, you can enjoy the surroundings... go to the top and leave the weirdos to us...	pos
Blse	ta gaur eiffel dorrea ikusiko degu, zelako gogoak aaaaaaaiiiiis!	neg
MT	ta today we see eiffel tower, what kind of aaaaaaaiiiiis!	pos
Unsup	i'm torn eifell tower So we see the infamous enjoyment aaaaaaaiiiiis	neg
Ref.	and today we'll visit the Eiffel Tower. So excited, ahhhhh!	pos
Blse	kanpora jun barik ehko asko : anboto, txindoki ...	neu
MT	many out of the jungle: anboto	neu
Unsup	away jun at once congress on many : Shasta, Pine mountain ...	neu
Ref.	Without leaving home, there's a lot of Basque Country: Anboto, Txindoki ...	pos
Blse	é pisar a coruña e boom! venme unha onda de bandeiras españolas, velliñxs falando castelán e señoras bordes e relambidas na cara.	pos
MT	is stepping on a coruña boom! I came across a wave of Spanish flags, talking Spanish and ladies and speaking on the face. PIC	neg
Ref.	As soon as I get to a coruña, boom! A wave of Spanish flags, old people speaking Spanish, and rude, pretentious ladies hits me in the face.	neg
Blse	@musee louvre @parisotc purtroppo non ho visto il louvre.... che file chilometriche !!! (	neg
MT	@musee louve @parisotc unfortunately I did not see the louvre.... that file kilometers! (	pos
Ref.	@musee louve @parisotc unfortunately I did not see the louvre.... the line was kilometers long! (	neg
Blse	io voglio solamente andare al louvre :-(	neg
MT	I just want to go to the louvre :-(	pos
Ref.	I just want to go to the louvre :-(	pos

Table 14: Examples where Blse is better and worse than MT and Unsup. We show the original tweet in Blse, the automatic translation in MT and Unsup, and reference translations (Ref.). The label column shows the prediction of each model and the reference gold label (either pos or neg). Additionally, we underline relevant incorrect translations of words.

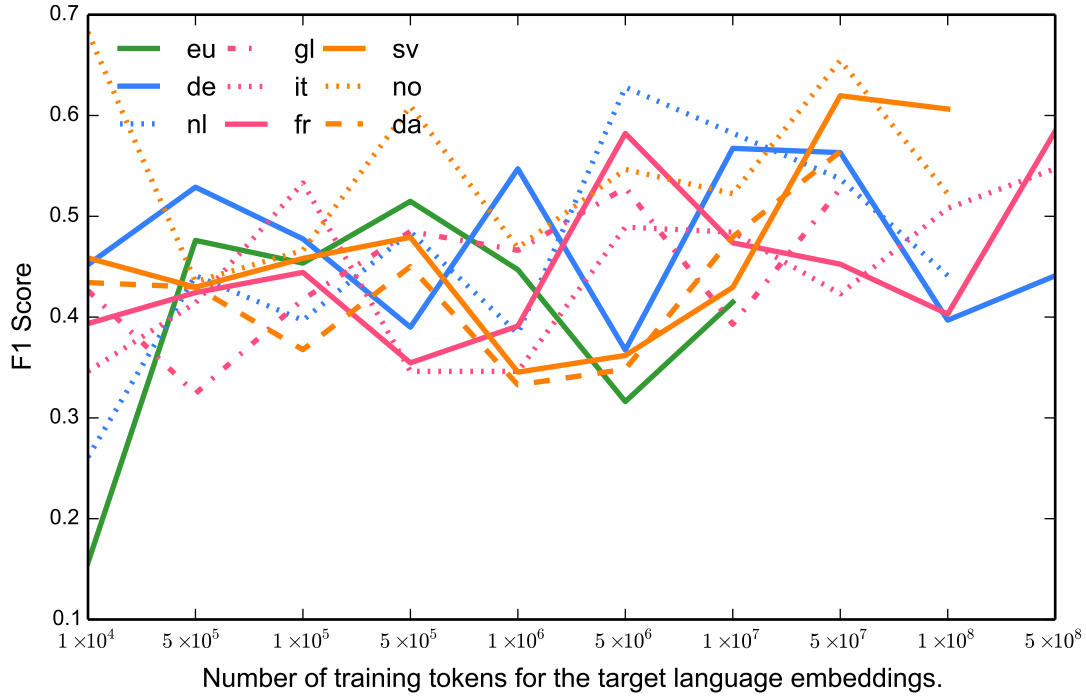

Figure 10: Performance of Blse (Macro F_1) on the binary sentiment task with training and test on Twitter as a function of amount of monolingual data available to train the monolingual embeddings in each language.

of unlabeled data available and the performance of Blse, Muse, VecMap (Pearson's r = −0.14, −0.27, 0.08, respectively) reveals no statistically significant correlation between them. This seems to indicate that all models are not sensitive to the amount of monolingual training data available in the target language.

6.4.2 Language Similarity

One hypothesis to different results across languages is that the similarity of the source and target language has an effect on the final classification of the models. In order to analyze this, we need a measure that models pairwise language similarity. Given that the features we use for classification are derived from distributional representations, we model similarity as a function of 1) universal POS-tag n-grams which represent the contexts used during training, and 2) character n-grams, which represent differences in morphology. POS-tag n-grams have previously been used to classify genre (Fang & Cao, 2010), improve statistical

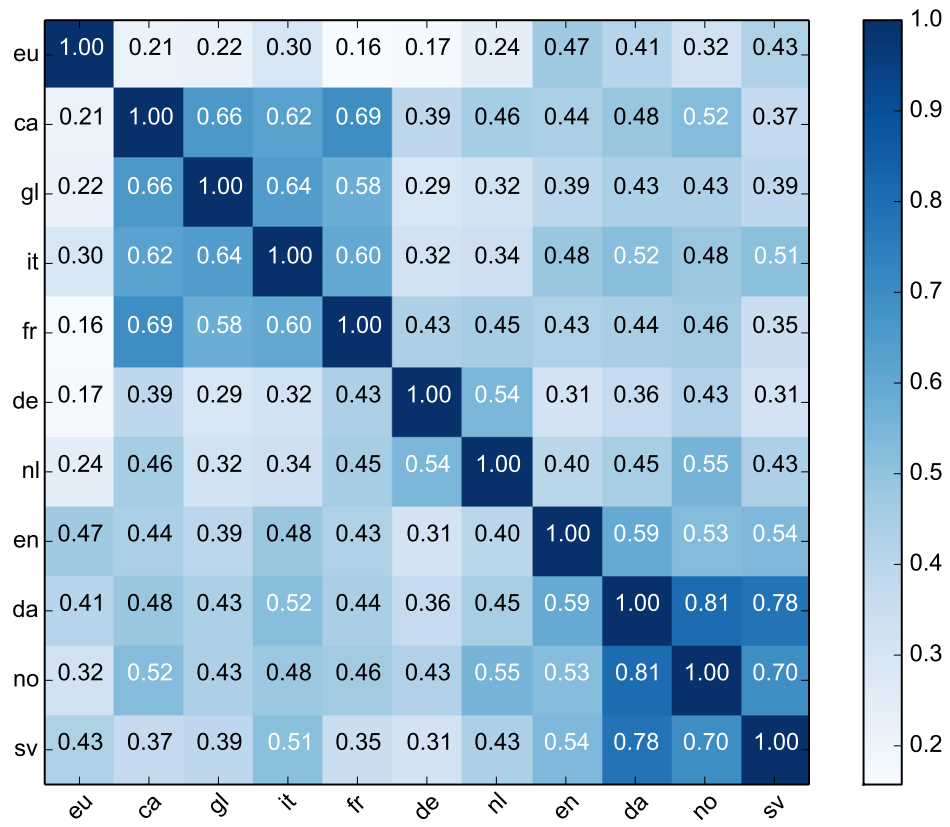

Figure 11: Cosine similarity of 3-gram POS-tag and 3-gram character frequency.

machine translation (Lioma & Ounis, 2005), and the combination of POS-tag and character n-grams have proven useful features for identifying the native language of second language writers in English (Kulmizev et al., 2017). This indicates that these are useful features for characterizing a language. In this section we calculate the pairwise similarity between all languages and then check whether this correlates with performance.

After POS-tagging the test sentences obtained from Twitter using the universal part of speech tags (Petrov et al., 2012), we calculate the normalized frequency distribution P_l for the POS-tag trigrams and C_l for character trigrams for each language l in L = {Danish, Swedish, Norwegian, Italian, Basque, Catalan, French, Dutch, Galician, German, English}. We then compute the pairwise cosine similarity between $\cos(A, B) = \frac{A \cdot B}{||A||\,||B||}$ where A is the concatenation of P_{l_i} and C_{l_i} for language l_i and B is the concatenation of P_{l_j} and C_{l_j} for language l_j.

The pairwise similarities in Figure 11 confirm to expected similarities, and language families are clearly grouped (Romance, Germanic, Scandinavian, with Basque as an outlier that has no more than 0.47 similarity with any language). This confirms the use of our

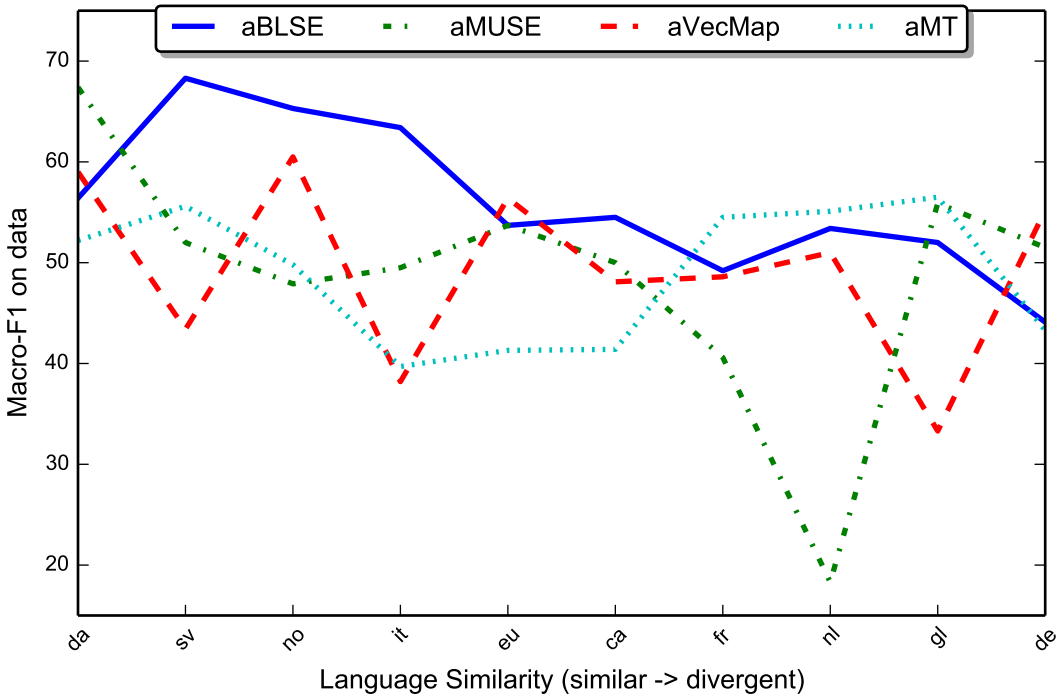

Figure 12: Performance (Macro F_1) on the binary task as a function of cosine similarity between POS-tag and character trigram distributions in the source language (EN) and the target languages.

similarity metric for our purposes. We plot model performance as a function of language similarity in Figure 12. To measure the correlation between language similarity and performance, we calculate Pearson's r and find that for Blse there is a strong correlation between language similarity and performance, $r = 0.76$ and significance $p < 0.01$. Muse, VecMap and MT do not show these correlations ($r = 0.41$, 0.24, 0.14, respectively). For MT this may be due to robust machine translation available in less similar languages according to our metric, *e. g.*, German-English. For Muse and VecMap, however, it is less clear why it does not follow the same trend as Blse.

6.4.3 Domain Similarity

In this section, we determine the effect of source-language domain on the cross-lingual sentiment classification task. Specifically, we use English language training data from three different domains (Twitter, restaurant reviews, and product reviews) to train the cross-lingual

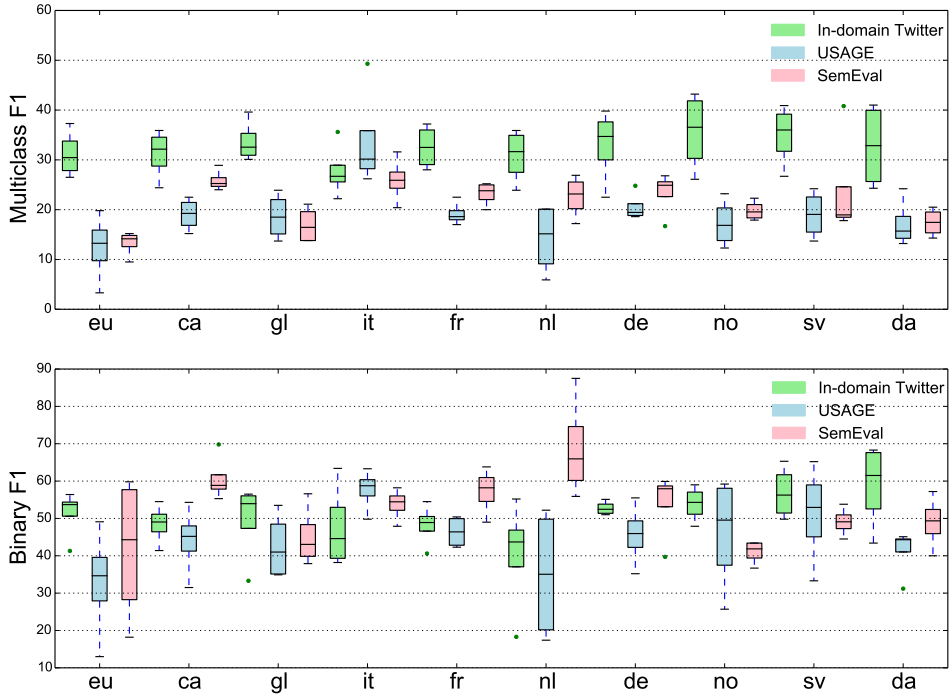

Figure 13: Performance of all models (Macro F_1) on the binary and multiclass task when trained on different source language data. For each target language, we show a boxplot for all models trained on *In-domain Twitter data* (light green), *USAGE product reviews* (light blue), and *SemEval restaurant reviews* (pink). In the multiclass setup, we can see the in-domain data gives better results than the out-of-domain training data. This trend is not found in the binary setup, suggesting that binary classification is more robust to domain changes than multiclass classification.

	Twitter	SemEval	USAGE
Twitter	1.000	0.749	0.749
SemEval	0.749	1.000	0.819
USAGE	0.749	0.819	1.000

Table 15: Domain similarity of English training data measured as Jennson-Shannon divergence between the most common 10,000 unigrams.

classifiers, and then test on the target-language Twitter data. In monolingual sentiment analysis, one would expect to see a drop when moving to more distant domains.

	Blse	Muse	VecMap	MT	Ensemble
Binary	0.32	0.09	0.11	−0.07	−0.01
Multiclass	*0.75	*0.50	*0.60	*0.88	*0.88

Table 16: Pearson's r and p values for correlations between domain and performance of each model. On the binary setup, there is no statistically significant effect of domain, while on the multiclass setup, all results are statistically significant ($p > 0.01$, with Pearson's r).

In order to analyze the effect of domain similarity further, we test the similarity of the domains of the source-language training data using Jensen-Shannon Divergence, which is a smoothed, symmetric version of the Kullback-Leibler Divergence, $D_{KL}(A||B) = \sum_i^N a_i \log \frac{a_i}{b_i}$. Kullback-Leibler Divergence measures the difference between the probability distributions A and B, but is undefined for any event $a_i \in A$ with zero probability, which is common in term distributions. Jensen-Shannon Divergence is then

$$D_{JS}(A, B) = \frac{1}{2}\Big[D_{KL}(A||B) + D_{KL}(B||A)\Big].$$

Our similarity features are probability distributions over terms $t \in \mathbb{R}^{|V|}$, where t_i is the probability of the i-th word in the vocabulary V. For each domain, we create frequency distributions of the most frequent 10,000 unigrams that all domains have in common and measure the divergence with D_{JS}.

The results shown in Table 15 indicate that both the SemEval and USAGE datasets are relatively distinct from the Twitter data described in Section 6.2.2, while they are more similar to each other. Additionally, we plot the results of all models with respect to the training domain in Figure 13.

We calculate Pearson's r on the correlation between domain and model performance, shown in Table 16. On the binary setup, the results show a negligible correlation for Blse (0.32), with no significant correlation for Muse, VecMap or MT. This suggests that the models are relatively robust to domain noise, or rather that there is so much other noise found in the approaches that domain is less relevant. On the multiclass setup, however, there is a significant effect for all models. This indicates that the multiclass models presented here are less robust than the binary models.

Both the SemEval and USAGE corpora differ equally from the Twitter data given the metric defined here. The fact that models trained on SemEval tend to perform better than those trained on USAGE, therefore, seems to be due to the differences in label distribution, rather than to differences in domain. These label distributions are radically different in the multiclass setup, as the English Twitter data has a 30/50/20 distribution over Positive, Neutral, and Negative labels (67/1/32 and 68/4/28 for USAGE and SemEval, respectively). Both undersampling and oversampling help, but the performance is still worse than training on in-domain data.

6.4.4 Conclusion

The case study which we presented in this section showed results of deploying the models from Section 3 to real world Twitter data, which we collect and annotate for targeted sentiment analysis. The analysis of different phenomena revealed that for binary targeted sentiment analysis, Blse performs better than machine translation on noisy data from social media, although it is sensitive to differences between source and target languages. Finally, there is little correlation between performance on the cross-lingual sentiment task and the amount of unlabeled monolingual data used to create the original embeddings spaces which goes against our expectations.

Unlike the experiments in Section 3.1, the ensemble classifier employed here was not able to improve the results. We assume that the small size of the datasets in this experiment does not enable the classifier to learn which features are useful in certain contexts.

One common problem that appears when performing targeted sentiment analysis on noisy data from Twitter is that many of the targets of interest are ambiguous, which leads to false positives. Even with relatively unambiguous targets like "Big Ben", there are a number of entities that can be referenced; Ben Rothlisberger (an American football player), an English language school in Barcelona, and many others. In order to deploy a full sentiment analysis system on Twitter data, it will be necessary to disambiguate these mentions before classifying the tweets, either as a preprocessing step or jointly.

In sentiment analysis, it is not yet common to test a model on multiple languages, despite the fact that current state-of-the-art models are often theoretically language-agnostic. This section shows that good performance in one language does not guarantee that a model transfers well to other languages, even given similar resources. We hope that future work in sentiment analysis will make better use of the available test datasets.

7. Conclusion

With this article, we have presented a novel projection-based approach to targeted cross-lingual sentiment analysis. The central unit of the proposed method is Blse which enables the transfer of annotations from a source language to a non-annotated target language. The only input it relies on are word embeddings (which can be trained without manual labeling by self-annotation) and a comparably small translation dictionary which connects the semantics of the source and the target language.

In the binary classification setting (automatic labeling of sentences or documents), Blse constitutes a novel state of the art on several language and domain pairs. For a more fine-grained classification to four sentiment labels, Barista and Muse perform slightly better. The predictions in all settings are complementary to the strong upper bound of employing machine translations: in an ensemble, even this resource-intense approach is inferior.

The transfer from classification to target-level analysis revealed additional challenges. The performance is lower, particularly for the 4-class setting. Our analyses show that mapping of sentence predictions to the aspects mentioned in each sentence with a machine translation model is a very challenging empirical upper bound – the difference in performance compared to projection-based methods is greater here than for the sentence-classification setting. However, we showed that in resource-scarce environments, Blse constitutes the current state of the art for binary target-level sentiment analysis when incorporated in a

deep learning architecture which is informed about the aspect. Muse performs better in the same architecture for the 4-class setting.

Our analysis further showed that the neural network needs to be informed about both the aspect and the context – limiting the information to a selection of these sentence parts strongly underperforms the combined setting. That also demonstrates that the model does not rely on prior distributions of aspect mentions.

The final experiment in the paper is a real-world deployment of the target-level sentiment analysis system in multilingual setting with 10 languages, where the assumption is that the only supervision is available in English (which is not part of the target languages). We learned here that it is important to have access to in-domain data (even for cross-lingual projection), especially in the multiclass setting. Binary classification however, which might often be sufficient for real-world applications, is more robust to domain changes. Further, machine translation is less sensitive to language dissimilarities, unlike projection-based methods. The amount of available unlabeled data to create embeddings plays a role in the final performance of the system, although only to a minor extent.

The current performance of the projection-based techniques still lags behind state-of-the-art MT approaches on most tasks, indicating that there is still much work to be done. While general bilingual embedding techniques do not seem to incorporate enough sentiment information, they are able to retain the semantics of their word vectors to a large degree even after projection. We hypothesize that the ability to retain the original semantics of the monolingual spaces leads to Muse performing better than MT on multiclass targeted sentiment analysis. The joint approach introduced in this work suffers from the degradation of the original semantics space, while optimizing the sentiment information. Moving from a similarity-based loss to a ranking loss, where the model must predict a ranked list of most similar translations could improve the model, but would require further resource development cross-lingually, as a simple bilingual dictionary would not provide enough information.

One problem that arises when using bilingual embeddings instead of machine translation is that differences in word order are no longer handled (Atrio et al., 2019). Machine translation models, on the other hand, always include a reordering element. Nonetheless, there is often a mismatch between the real source language word order and the translated word order. In this work, we avoided the problem by using a bag-of-embeddings representation, but Barnes et al. (2017) found that the bag-of-embeddings approach does not perform as well as approaches that take word order into account, *e. g.*, Lstms or Cnns. We leave the incorporation of these classifiers into our framework for future work.

Unsupervised machine translation (Artetxe et al., 2018b; Lample et al., 2018b; Artetxe et al., 2018a) shows great promise for sentence-level classification. Like MT, however, it performs worse on noisy data, such as tweets. Therefore, users who want to apply targeted cross-lingual approaches to noisy data should consider currently consider using embedding projection methods, such as Blse. Future work on adapting unsupervised machine translation to noisy text may provide another solution for low-resource NLP.

Acknowledgments

The authors thank Patrik Lambert, Toni Badia, Amaia Oliden, Itziar Etxeberria, Jessie Kief, Iris Hübscher, and Arne Øhm for helping with the annotation of the resources used in this research. This work has been partially supported by the DFG Collaborative Research Centre SFB 732 and a SGR-DTCL Predoctoral Scholarship.

References

Agerri, R., Bermudez, J., & Rigau, G. (2014). Ixa pipeline: Efficient and ready to use multilingual nlp tools. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)* (1.0 edition)., pp. 3823–3828. European Language Resources Association (ELRA).

Agerri, R., Cuadros, M., Gaines, S., & Rigau, G. (2013). OpeNER: Open polarity enhanced named entity recognition.. In *Sociedad Española para el Procesamiento del Lenguaje Natural*, Vol. 51, pp. 215–218.

Agić, Ž., Johannsen, A., Plank, B., Martínez Alonso, H., Schluter, N., & Søgaard, A. (2016). Multilingual projection for parsing truly low-resource languages. *Transactions of the Association for Computational Linguistics*, *4*, 301–312.

Akhtar, M. S., Sawant, P., Sen, S., Ekbal, A., & Bhattacharyya, P. (2018). Solving data sparsity for aspect based sentiment analysis using cross-linguality and multi-linguality. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 572–582. Association for Computational Linguistics.

Almeida, M. S. C., Pinto, C., Figueira, H., Mendes, P., & Martins, A. F. T. (2015). Aligning opinions: Cross-lingual opinion mining with dependencies. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 408–418.

Artetxe, M., Labaka, G., & Agirre, E. (2016). Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2289–2294, Austin, Texas. Association for Computational Linguistics.

Artetxe, M., Labaka, G., & Agirre, E. (2017). Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 451–462, Vancouver, Canada. Association for Computational Linguistics.

Artetxe, M., Labaka, G., & Agirre, E. (2018a). Unsupervised statistical machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium. Association for Computational Linguistics.

Artetxe, M., Labaka, G., Agirre, E., & Cho, K. (2018b). Unsupervised neural machine translation. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*.

Asch, S. E. (1955). Opinions and social pressure. *Scientific American*, *193*(5), 31–35.

Atrio, À. R., Badia, T., & Barnes, J. (2019). On the effect of word order in cross-lingual sentiment analysis. *Procesamiento del Lenguaje Natural*, *63*(0), To Appear.

Bakliwal, A., Foster, J., van der Puil, J., O'Brien, R., Tounsi, L., & Hughes, M. (2013). Sentiment analysis of political tweets: Towards an accurate classifier. In *Proceedings of the Workshop on Language Analysis in Social Media*, pp. 49–58, Atlanta, Georgia. Association for Computational Linguistics.

Balahur, A., & Turchi, M. (2014). Comparative experiments usingsupervised learning and machine translation for multilingual sentiment analysis. *Computer Speech & Language*, *28*(1), 56–75.

Banea, C., Mihalcea, R., & Wiebe, J. (2010). Multilingual subjectivity: Are more languages better?. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pp. 28–36.

Banea, C., Mihalcea, R., & Wiebe, J. (2013). Porting multilingual subjectivity resources across languages. *IEEE Transactions on Affective Computing*, *99*(PrePrints).

Banea, C., Mihalcea, R., Wiebe, J., & Hassan, S. (2008). Multilingual subjectivity analysis using machine translation. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pp. 127–135, Honolulu, Hawaii. Association for Computational Linguistics.

Barnes, J., Klinger, R., & Schulte im Walde, S. (2017). Assessing state-of-the-art sentiment classifiers on state-of-the-art sentiment datasets. In *Proceedings of 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA 2017)*.

Barnes, J., Klinger, R., & Schulte im Walde, S. (2018a). Bilingual sentiment embeddings: Joint projection of sentiment across languages. In *Proceedings of the 56th Annual Meeting of the Association for Comutational Linguistics (Volume 1: Long Papers)*, pp. 2483–2493.

Barnes, J., Klinger, R., & Schulte im Walde, S. (2018b). Projecting embeddings for domain adaptation: Joint modeling of sentiment analysis in diverse domains. In *Proceedings of COLING 2018, the 27th International Conference on Computational Linguistics: Technical Papers*, Santa Fe, USA. The COLING 2018 Organizing Committee.

Barnes, J., Lambert, P., & Badia, T. (2016). Exploring distributional representations and machine translation for aspect-based cross-lingual sentiment classification.. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 1613–1623. The COLING 2016 Organizing Committee.

Barnes, J., Lambert, P., & Badia, T. (2018). Multibooked: A corpus of basque and catalan hotel reviews annotated for aspect-level sentiment classification. In *Proceedings of 11th Language Resources and Evaluation Conference (LREC'18)*.

Bertoldi, N., & Federico, M. (2009). Domain adaptation for statistical machine translation with monolingual resources. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pp. 182–189, Athens, Greece. Association for Computational Linguistics.

Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media.

Chandar, S., Lauly, S., Larochelle, H., Khapra, M., Ravindran, B., Raykar, V. C., & Saha, A. (2014). An autoencoder approach to learning bilingual word representations. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., & Weinberger, K. Q. (Eds.), *Advances in Neural Information Processing Systems 27*, pp. 1853–1861. Curran Associates, Inc.

Chen, M., Xu, Z., Weinberger, K. Q., & Sha, F. (2012). Marginalized denoising autoencoders for domain adaptation. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, pp. 1627–1634, USA. Omnipress.

Chen, P., Sun, Z., Bing, L., & Yang, W. (2017). Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 452–461. Association for Computational Linguistics.

Chen, X., Athiwaratkun, B., Sun, Y., Weinberger, K. Q., & Cardie, C. (2016). Adversarial deep averaging networks for cross-lingual sentiment classification. *CoRR*, *abs/1606.01614*.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, *12*, 2493–2537.

Cotterell, R., Mielke, S. J., Eisner, J., & Roark, B. (2018). Are all languages equally hard to language-model?. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 536–541. Association for Computational Linguistics.

Das, A., Gollapudi, S., & Munagala, K. (2014). Modeling opinion dynamics in social networks. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, WSDM '14, pp. 403–412, New York, NY, USA. ACM.

Demirtas, E., & Pechenizkiy, M. (2013). Cross-lingual polarity detection with machine translation. In *Proceedings of the International Workshop on Issues of Sentiment Discovery and Opinion Mining - WISDOM '13*, pp. 9:1–9:8, Chigaco, Illinois, USA.

Ding, X., Liu, B., & Yu, P. S. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, pp. 231–240, New York, NY, USA. ACM.

Dinu, G., Lazaridou, A., & Baroni, M. (2015). Improving zero-shot learning by mitigating the hubness problem. In *ICLR 2015*, pp. 1–10.

Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., & Xu, K. (2014). Adaptive recursive neural network for target-dependent twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 49–54, Baltimore, Maryland. Association for Computational Linguistics.

Duh, K., Fujino, A., & Nagata, M. (2011). Is machine translation ripe for cross-lingual sentiment classification?. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers*, *2*, 429–433.

Fang, A. C., & Cao, J. (2010). Enhanced genre classification through linguistically fine-grained pos tags. In *Proceedings of the 24th Pacific Asia Conference on Language, Information and Computation*.

Felbo, B., Mislove, A., Søgaard, A., Rahwan, I., & Lehmann, S. (2017). Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 1615–1625, Copenhagen, Denmark. Association for Computational Linguistics.

Ferreira, D. C., Martins, A. F. T., & Almeida, M. S. C. (2016). Jointly learning to embed and predict with multiple languages. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2019–2028, Berlin, Germany. Association for Computational Linguistics.

Gavrila, M., & Vertan, C. (2011). Training data in statistical machine translation - the more, the better?. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pp. 551–556, Hissar, Bulgaria. RANLP 2011 Organising Committee.

Gerz, D., Vulić, I., Ponti, E. M., Reichart, R., & Korhonen, A. (2018). On the relation between linguistic typology and (limitations of) multilingual language modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. To Appear.

Gouws, S., Bengio, Y., & Corrado, G. (2015). BilBOWA: Fast bilingual distributed representations without word alignments. In *Proceedings of The 32nd International Conference on Machine Learning*, pp. 748–756.

Gouws, S., & Søgaard, A. (2015). Simple task-specific bilingual word embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1386–1390, Denver, Colorado. Association for Computational Linguistics.

Gräßer, F., Kallumadi, S., Malberg, H., & Zaunseder, S. (2018). Aspect-based sentiment analysis of drug reviews applying cross-domain and cross-data learning. In *Proceedings of the 2018 International Conference on Digital Health*, DH '18, pp. 121–125, New York, NY, USA. ACM.

Hangya, V., Braune, F., Fraser, A., & Schütze, H. (2018). Two methods for domain adaptation of bilingual tasks: Delightfully simple and broadly applicable. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 810–820. Association for Computational Linguistics.

Hartung, M., Klinger, R., Schmidtke, F., & Vogel, L. (2017). Ranking right-wing extremist social media profiles by similarity to democratic and extremist groups. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and*

Social Media Analysis, pp. 24–33, Copenhagen, Denmark. Association for Computational Linguistics.

Hermann, K. M., & Blunsom, P. (2014). Multilingual models for compositional distributed semantics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 58–68, Baltimore, Maryland. Association for Computational Linguistics.

Hu, M., & Liu, B. (2004). Mining opinion features in customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004)*, pp. 168–177.

Iyyer, M., Manjunatha, V., Boyd-Graber, J., & Daumé III, H. (2015). Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1681–1691, Beijing, China. Association for Computational Linguistics.

Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viégas, F., Wattenberg, M., Corrado, G., Hughes, M., & Dean, J. (2017). Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, *5*, 339–351.

Kingma, D., & Ba, J. (2014). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

Klinger, R., & Cimiano, P. (2014). The USAGE review corpus for fine grained multi lingual opinion analysis. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pp. 2211–2218.

Klinger, R., & Cimiano, P. (2015). Instance selection improves cross-lingual model training for fine-grained sentiment analysis. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pp. 153–163, Beijing, China. Association for Computational Linguistics.

Koehn, P., & Knowles, R. (2017). Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pp. 28–39. Association for Computational Linguistics.

Kulmizev, A., Blankers, B., Bjerva, J., Nissim, M., van Noord, G., Plank, B., & Wieling, M. (2017). The power of character n-grams in native language identification. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 382–389, Copenhagen, Denmark. Association for Computational Linguistics.

Lambert, P. (2015). Aspect-level cross-lingual sentiment classification with constrained smt. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pp. 781–787, Beijing, China. Association for Computational Linguistics.

Lample, G., Conneau, A., Ranzato, M., Denoyer, L., & Jégou, H. (2018a). Word translation without parallel data. In *International Conference on Learning Representations*.

Lample, G., Denoyer, L., & Ranzato, M. (2018b). Unsupervised machine translation using monolingual corpora only. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*.

Lazaridou, A., Dinu, G., & Baroni, M. (2015). Hubness and pollution: Delving into cross-space mapping for zero-shot learning. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 270–280, Beijing, China. Association for Computational Linguistics.

Lioma, C., & Ounis, I. (2005). Deploying part-of-speech patterns to enhance statistical phrase-based machine translation resources. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pp. 163–166, Ann Arbor, Michigan. Association for Computational Linguistics.

Liu, B. (2012). *Sentiment analysis and opinion mining*. Morgan and Claypool.

Liu, B., Hu, M., & Cheng, J. (2005). Opinion Observer: Analyzing and comparing opinions on the web. In *Proceedings of the 14th international World Wide Web conference (WWW-2005)*, Chiba Japan.

Liu, F., Cohn, T., & Baldwin, T. (2018). Recurrent entity networks with delayed memory update for targeted aspect-based sentiment analysis. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pp. 278–283. Association for Computational Linguistics.

Loper, E., & Bird, S. (2002). NLTK: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, pp. 63–70, Philadelphia, Pennsylvania, USA.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Maia, M., Handschuh, S., Freitas, A., Davis, B., McDermott, R., Zarrouk, M., & Balahur, A. (2018). Www'18 open challenge: Financial opinion mining and question answering. In *Companion Proceedings of the The Web Conference 2018*, WWW '18, pp. 1941–1942, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

Meng, X., Wei, F., Liu, X., Zhou, M., Xu, G., & Wang, H. (2012). Cross-lingual mixture model for sentiment classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 572–581, Jeju Island, Korea. Association for Computational Linguistics.

Mihalcea, R., Banea, C., & Wiebe, J. (2007). Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp. 976–983.

Mikolov, T., Le, Q. V., & Sutskever, I. (2013). Exploiting similarities among languages for machine translation. *CoRR*, *1*(1), 1–10.

Mohammad, S. M., Kiritchenko, S., & Zhu, X. (2013). Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. In *Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013)*.

Mohammad, S. M., Salameh, M., & Kiritchenko, S. (2016). How translation alters sentiment. *Journal of Artificial Intelligence Research*, *55*, 95–130.

Nahar, V., Unankard, S., Li, X., & Pang, C. (2012). Sentiment analysis for effective detection of cyber bullying. In *Proceedings of the 14th Asia-Pacific International Conference on Web Technologies and Applications*, APWeb'12, pp. 767–774, Berlin, Heidelberg. Springer-Verlag.

Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., & Stoyanov, V. (2016). Semeval-2016 task 4: Sentiment analysis in twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pp. 1–18, San Diego, California. Association for Computational Linguistics.

Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., & Wilson, T. (2013). Semeval-2013 task 2: Sentiment analysis in twitter. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pp. 312–320, Atlanta, Georgia, USA. Association for Computational Linguistics.

Padró, L., Collado, M., Reese, S., Lloberes, M., & Castellón, I. (2010). Freeling 2.1: Five years of open-source language processing tools. In *Proceedings of 7th Language Resources and Evaluation Conference (LREC'10)*, La Valletta, Malta.

Pagolu, V. S., Reddy, K. N., Panda, G., & Majhi, B. (2016). Sentiment analysis of twitter data for predicting stock market movements. In *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pp. 1345–1350.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical methods in natural language processing-Volume 10*, pp. 79–86. Association for Computational Linguistics.

Paszke, A., Gross, S., Chintala, S., & Chanan, G. (2016). Pytorch deeplearning framework. http://pytorch.org. Accessed: 2017-08-10.

Petrov, S., Das, D., & McDonald, R. (2012). A universal part-of-speech tagset. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., AL-Smadi, M., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O., Hoste, V., Apidianaki, M., Tannier, X., Loukachevitch, N., Kotelnikov, E., Bel, N., Jiménez-Zafra, S. M., & Eryiğit, G. (2016). Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pp. 19–30.

Pontiki, M., Galanis, D., Papageorgiou, H., Manandhar, S., & Androutsopoulos, I. (2015). Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pp. 486–495.

Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., & Manandhar, S. (2014). Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pp. 27–35.

Prettenhofer, P., & Stein, B. (2011). Cross-lingual adaptation using structural correspondence learning. *ACM Transactions on Intelligent Systems and Technology*, *3*(1), 1–22.

Rasooli, M. S., Farra, N., Radeva, A., Yu, T., & McKeown, K. (2017). Cross-lingual sentiment transfer with limited resources. *Machine Translation*, *32*(1-2), 143–165.

Reitan, J., Faret, J., Gambäck, B., & Bungum, L. (2015). Negation scope detection for twitter sentiment analysis. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pp. 99–108.

Saeidi, M., Bouchard, G., Liakata, M., & Riedel, S. (2016). Sentihood: Targeted aspect based sentiment analysis dataset for urban neighbourhoods. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 1546–1556. The COLING 2016 Organizing Committee.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642. Association for Computational Linguistics.

Søgaard, A., Agic, Z., Martínez Alonso, H., Plank, B., Bohnet, B., & Johannsen, A. (2015). Inverted indexing for cross-lingual nlp. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint conference of the Asian Federation of Natural Language Processing (ACL-IJCNLP 2015)*.

Tang, D., Wei, F., Qin, B., Yang, N., Liu, T., & Zhou, M. (2016). Sentiment embeddings with applications to sentiment analysis. *IEEE Trans. on Knowl. and Data Eng.*, *28*(2), 496–509.

Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2014). Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1555–1565.

Turney, P. D., & Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems*, *21*(4), 315–346.

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, *9*, 2579–2605.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., & Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., & Garnett, R. (Eds.), *Advances in Neural Information Processing Systems 30*, pp. 5998–6008. Curran Associates, Inc.

Wan, X. (2009). Co-training for cross-lingual sentiment classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp. 235–243.

Wang, B., Liakata, M., Zubiaga, A., & Procter, R. (2017). Tdparse: Multi-target-specific sentiment recognition on twitter. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pp. 483–493. Association for Computational Linguistics.

Wang, H., Can, D., Kazemzadeh, A., Bar, F., & Narayanan, S. (2012). A system for real-time twitter sentiment analysis of 2012 u.s. presidential election cycle. In *Proceedings of the ACL 2012 System Demonstrations*, pp. 115–120, Jeju Island, Korea. Association for Computational Linguistics.

Wang, S., Mazumder, S., Liu, B., Zhou, M., & Chang, Y. (2018). Target-sensitive memory networks for aspect sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 957–967. Association for Computational Linguistics.

Welch, C., & Mihalcea, R. (2016). Targeted sentiment to understand student comments. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 2471–2481. The COLING 2016 Organizing Committee.

Wiebe, J., Wilson, T., & Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, *39*(2-3), 165–210.

Wiegand, M., Balahur, A., Roth, B., Klakow, D., & Montoyo, A. (2010). A survey on the role of negation in sentiment analysis. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, pp. 60–68.

Wu, H., Wang, H., & Zong, C. (2008). Domain adaptation for statistical machine translation with domain dictionary and monolingual corpora. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pp. 993–1000. Coling 2008 Organizing Committee.

Xiao, M., & Guo, Y. (2012). Multi-view adaboost for multilingual subjectivity analysis. In *Proceedings of COLING 2012*, pp. 2851–2866.

Xue, W., & Li, T. (2018). Aspect based sentiment analysis with gated convolutional networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2514–2523. Association for Computational Linguistics.

Yeh, A. (2000). More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th Conference on Computational linguistics (COLING)*, pp. 947–953.

Zhang, M., Zhang, Y., & Vo, D. T. (2015). Neural networks for open domain targeted sentiment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 612–621, Lisbon, Portugal. Association for Computational Linguistics.

Zhang, M., Zhang, Y., & Vo, D.-T. (2016). Gated neural networks for targeted sentiment analysis. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pp. 3087–3093. AAAI Press.

Zhang, Y., & Wallace, B. (2017). A sensitivity analysis of (and practitioners guide to) convolutional neural networks for sentence classification. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 253–263, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Zhou, G., Zhu, Z., He, T., & Hu, X. T. (2016). Cross-lingual sentiment classification with stacked autoencoders. *Knowledge and Information Systems*, *47*(1), 27–44.

Zhou, H., Chen, L., Shi, F., & Huang, D. (2015). Learning bilingual sentiment word embeddings for cross-language sentiment classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 430–440.

Zhou, X., Wan, X., & Xiao, J. (2012). Cross-language opinion target extraction in review texts. In *IEEE 12th International Conference on Data Mining*, pp. 1200 – 1205, Brussels, Belgium.

Zhu, X., Guo, H., Mohammad, S., & Kiritchenko, S. (2014). An empirical study on the effect of negation words on sentiment. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 304–313.

Ziser, Y., & Reichart, R. (2017). Neural structural correspondence learning for domain adaptation. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pp. 400–410, Vancouver, Canada. Association for Computational Linguistics.