Low-resource Cross-lingual Event Type Detection in Documents via Distant Supervision with Minimal Effort

Aldrian Obaja Muis Naoki Otani Nidhi Vyas Ruochen Xu
Yiming Yang Teruko Mitamura Eduard Hovy
Language Technology Institute
Carnegie Mellon University
Pittsburgh, PA
{amuis,notani,nkvyas,ruochenx,yiming,teruko,ehovy}@andrew.cmu.edu

Abstract

The use of machine learning for NLP generally requires resources for training. Tasks performed in a low-resource language usually rely on labeled data in another, typically resource-rich, language. However, there might not be enough labeled data even in a resource-rich language such as English. In such cases, one approach is to use a hand-crafted approach that utilizes only a small bilingual dictionary with minimal manual verification to create distantly supervised data. Another is to explore typical machine learning techniques, for example adversarial training of bilingual word representations. We find that in event-type detection task—the task to classify [parts of] documents into a fixed set of labels—they give about the same performance. We explore ways in which the two methods can be complementary and also see how to best utilize a limited budget for manual annotation to maximize performance gain.

1 Introduction

For most languages of the world, few or no language processing tools or resources exist (Baumann and Pierrehumbert, 2014). This hinders efforts to apply certain language technologies enjoyed by languages like English, in which much current research is done.

To perform natural language processing tasks in resource-poor languages, one way to overcome data scarcity is to tap on resources from another resource-rich language. Assuming that there are already good resources and tools to solve the same tasks in the more resource-rich language (henceforth, *auxiliary language*), the only remaining challenge is to transfer the learning process into the resource-poor language (henceforth, *target language*) and adapt it to the specifics of that language. One way to do this is to build a shared word representation across the two languages and train an NL engine on this shared representation, perhaps using an adversarial domain adaptation approach to handle the domain (language) shift (Chen et al., 2017). Usually, these approaches assume the availability of large labeled data in the auxiliary language, on the order of hundred thousands to millions.

However, for some more complex or specialized tasks, there might not be enough available training data even in a resource-rich language such as English. A case in point is the event-type classification task over the publicly available datasets, such as ACE 2005[1] and TAC KBP [2] datasets, which usually contain only a few hundred to a few thousand documents. The situation frame (SF) detection task is one example of event-type classification task, where the objective is to extract from each document one or more *situation frames* with their corresponding arguments. A situation frame (SF) is either an *issue* being described in the articles, such as civil unrest or terrorism, or a *need* situation such as the need for water or medical aid. In our task there are 11 situation frame types, each associated with a set of arguments, namely the location, status, relief, and urgency. For example, an article titled "Millions of people are at the risk of starvation due to the food shortage in South Sudan", with content describing the details and the cause of the food shortage, including a mention of difficulty accessing certain regions, can be classified as describing a *food need* and an *infrastructure need*.

[1] https://catalog.ldc.upenn.edu/ldc2006t06

[2] https://tac.nist.gov/2016/KBP/

Proceedings of the 27th International Conference on Computational Linguistics, pages 70–82
Santa Fe, New Mexico, USA, August 20-26, 2018.

As described below, we have tried two approaches: (1) a simple keyword-matching system utilizing only a small bilingual dictionary and minimal manual verification and (2) a sophisticated neural adversarial network that learns bilingual word representations for cross-lingual transfer. We find that the methods have similar performance. We therefore explore ways in which few keyword-based models can create additional, distantly supervised data to improve the performance of a neural cross-lingual event type detection system. Our contributions are: (1) an evaluation of a state-of-the-art method in a different task showing its similar result against a simple baseline, (2) ways to improve performance of such models, and (3) an analysis of the result, with insights to practitioners as to where to focus the available, yet limited, budget for manual annotation work.

This paper is organized as follows: we first describe the related work on cross-lingual NLP tasks in low-resource settings, specifically how the available resources are used. Based on previous work, we then apply our proposed training data augmentation methods and run experiments to show the effectiveness of our methods. We then analyze the results, and follow with a few suggestions on how to best utilize the available annotation effort for maximum gain.

2 Related Work

Keyword-based Models A keyword-based heuristic model is a simple yet effective approach to extract specific information such as events (Keane et al., 2015), because keywords often indicate a strong presence of important information contained in documents (Marujo et al., 2015). Such methods have been used in different tasks like text categorization (Özgür et al., 2005) and information retrieval (Marujo et al., 2013) to extract the required information. Keyword heuristics have also been used to overcome language and domain barriers using bilingual dictionaries (Szarvas, 2008; Tran et al., 2013). However, a weak bilingual dictionary could result in low coverage with this method. Hence, to overcome the limiting bilingual dictionary people employ bootstrapping methods to improve the coverage (Knopp, 2011; Ebrahimi et al., 2016).

Cross-Lingual Text Classification Cross-lingual event type detection is closely related to cross-lingual text classification (CLTC), which aims to classify text in a target language using training data from an auxiliary language (Bel et al., 2003).

To bridge the language gap, early approaches of CLTC relied on a comprehensive bilingual dictionary to translate documents between languages (Bel et al., 2003; Shi et al., 2010; Mihalcea et al., 2007). However, in resource-poor languages, bilingual dictionaries may be small and sparse. Therefore, the performance of direct word translation will be unsatisfactory. Some researchers utilized the bilingual dictionary to translate the models instead (Xu et al., 2016; Littell et al., 2017).

Another line of work focuses on the use of automatic machine translation as an oracle. The various learning algorithms treated the translations as a second view of document and facilitate cross-lingual learning with co-training (Wan, 2009), majority learning (Amini et al., 2009), matrix completion (Xiao and Guo, 2013) and multi-view co-regularization (Guo and Xiao, 2012a).

Instead of word-level or sentence-level translation, various other approaches seek some cross-lingual mapping between document representation (Littman et al., 1998; Vinokourov et al., 2002; Platt et al., 2010; Jagarlamudi et al., 2011; De Smet et al., 2011; Guo and Xiao, 2012b; Zhou et al., 2015; Zhou et al., 2016a; Zhou et al., 2016b; Chen et al., 2017) or label distribution (Xu and Yang, 2017).

Bilingual Word Embedding The most recent method for sharing document representation between languages is bilingual word embedding (Mikolov et al., 2013a; Faruqui and Dyer, 2014; Luong et al., 2015). The goal is to learn a shared embedding space between words in two languages. With the shared embedding, we are able to project all documents into a shared space. The model trained in one language can, therefore, be used in the other language.

3 Models

To see how well recent state-of-the-art methods for CLTC work in our task, we implemented a convolutional neural classifier. We compare this against a simple keyword-based method.

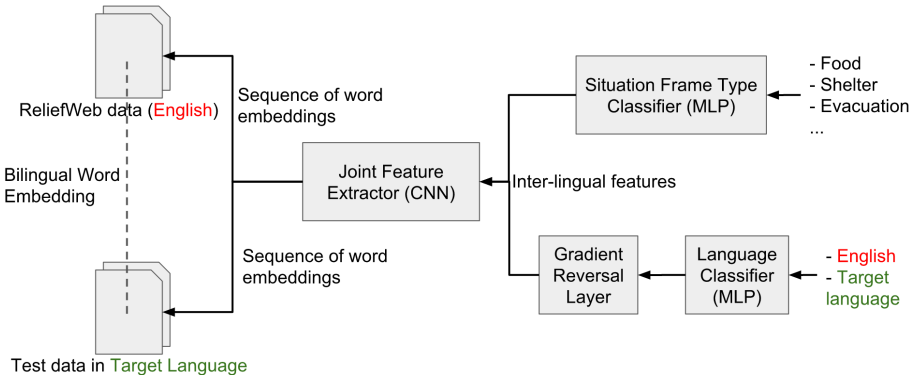


Figure 1: Architecture of the neural classifier with adversarial domain (language) adaptation by Ganin and Lempitsky (2015). Arrows show the flow of gradient.

3.1 Adversarial Convolutional Network

The first step is to train a bilingual word embedding as a shared feature representation space between the two languages. We trained our bilingual word embedding for English and the incident language using the method proposed in XlingualEmb (Duong et al., 2016). This method is a cross-lingual extension from word2vec model (Mikolov et al., 2013b) to bilingual text using two large monolingual corpora and a bilingual dictionary.

Based on the shared representation, we then used a convolutional neural network (CNN) (Kim, 2014) to perform the classification. There are two main advantages of choosing a deep neural classifier over a shallow one. First, CNN outperforms shallow models like SVM or Logistic Regression in various text classification benchmark datasets (Kim, 2014; Lai et al., 2015; Johnson and Zhang, 2015; Xu and Yang, 2017). Second, CNN takes dense word vector representations as input, allowing one to incorporate the state-of-the-art bilingual word embedding methods into the pipeline.

The CNN model takes a sequence of word embeddings as input and applies 1-D convolutional operation on the input to extract semantic features. The features are then passed through a fully-connected layer before reaching the final soft-max layer. The model is trained in English using the ReliefWeb dataset (Littell et al., 2017, Sec 2.3), which is annotated at sentence level with disaster relief needs and emergency situations. Thanks to the bilingual word embedding, which maps the words from the two languages to the same distributional semantic space, the model trained in English can be applied to documents in the target language.

Ideally, if the bilingual word embedding captures the ground-truth mapping between two languages, a classifier learned from English training documents should generalize well on the target language. However, in practice, we can observe obvious domain gaps between documents in different languages when we represent them with bilingual word embeddings. In order to close the domain (language) gap between training and testing, we adapt our learned model in English to the target language with similar adversarial training techniques used in (Xu and Yang, 2017; Chen et al., 2017). In order to alleviate the domain mismatch, we are essentially looking for a feature extractor that only captures the semantics of the event types but not the difference in language usage between English and the target language. In other words, we want the features captured by CNN to be informative for the event type classification and to be language-invariant at the same time. To achieve this goal, we include an auxiliary classifier that takes the features extracted by CNN and predicts the language that the input belongs to. During training, we update our parameter to simultaneously minimize the loss of the event type classification and maximize the loss of language classification through Gradient Reversal Layer (Ganin and Lempitsky, 2015).

3.2 Keyword-based Model

As mentioned previously, a keyword-based model is a simple, quick, yet effective approach to perform text classification without much training data. In our case, we do this in two steps: (1) build a list of keywords for each SF type in English, then (2) translate the keywords into the target language automatically

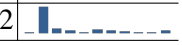
	Instances	Distribution of Situation Frames (%)												Visualization
		terror	violence	regime	food	water	med	infra	shelter	evac	utils	search	none	
Eng-Orig (Ⓞ)	82,096	2.4	1.8	3.9	14.0	33.0	8.2	6.0	9.2	3.7	4.3	8.6	4.9	
Eng-KW (Ⓔ)	1,356,425	17.5	29.2	6.0	11.3	12.6	9.9	2.7	4.0	3.6	1.6	1.5	0.3	
Tigrinya														
Tgt-Boot (Ⓣ)	98	12.2	33.7	10.2	4.1	6.1	20.4	0.0	11.2	1.0	0.0	1.0	0.0	
Tgt-Ann (Ⓐ)	1,012	6.7	14.3	17.6	2.0	0.8	6.6	2.2	3.3	2.4	0.9	2.7	40.6	
Test Data	2,991*	3.2	6.2	0.7	2.5	0.9	3.2	0.4	0.3	0.8	0.3	0.8	80.7	
Oromo														
Tgt-Boot (Ⓣ)	92	3.3	13.0	25.0	21.7	8.7	10.9	1.1	6.5	3.3	1.1	5.4	0.0	
Tgt-Ann (Ⓐ)	652	3.2	4.0	3.5	0.8	0.2	0.6	0.2	0.6	0.0	0.0	0.5	86.5	
Test Data	2,810*	0.6	11.4	2.3	1.4	0.5	2.1	1.7	1.2	0.7	0.5	1.5	76.2	

Table 1: Statistics of the various sources of training data. Eng-Orig and Eng-KW refer to training data in English described in Littell et al. (2017, Sec 2.3) and from our English keyword model's output on ReliefWeb corpus, respectively, while Tgt-Boot and Tgt-Ann refer to training data in the target language obtained from bootstrapped keyword-spotting and from annotation, respectively. The "none" class signifies negative examples in the data. The last column shows a visualization of the SF types, excluding "none". Note: for Test Data, the instances refer to documents, while for the rest, instances refer to sentences.

using a bilingual dictionary. We also asked native speakers of the target language to refine the translation, especially for domain-specific keywords which are not adequately captured by the bilingual dictionary.[3]

Building English keywords is again a two-step process. First, we use the ReliefWeb dataset to generate a list of 100 candidate keywords for each class by taking the top-100 words with the highest TF.IDF scores. Similar to the keyword generation method described by Littell et al. (2017), we manually refined the keyword list by pruning based on world knowledge. For each candidate keyword, we added 30 most similar words using the English word2vec model trained on the Google News corpus[4]. We retained only the words that have cosine-similarity greater than 70%. For each candidate keyword in this extended list, we computed a label affinity score with each SF class label (e.g., *water*, *evacuation*) using cosine-similarity between their word2vec embeddings. Candidate keywords with similarity above a certain threshold th_1 were retained and used as keywords for the corresponding classes[5].

4 Method: Training Data Augmentation

Chen et al. (2017) assumed the auxiliary language contains a large amount of labeled data for the task, about 700k Yelp reviews. For our case, the original training data, which was built semi-automatically by Littell et al. (2017, Sec 2.3), contains only about 80k sentences (Table 1, first row). To improve the performance of the neural model, therefore, we propose to utilize the keyword-based system to automatically augment the original training data. We also explore using additional training data obtained via manual annotation for comparison.

Figure 2 is a summary of the various training data sources we compare in this paper: the original training data (Ⓞ), keyword-spotting in the auxiliary language (Ⓔ), keyword-spotting with bootstrapping in the target language (Ⓣ), and annotated data in the target language (Ⓐ). The additional training data from keyword-spotting in English (Ⓔ) can be directly obtained by using the keyword list in English that we used for the keyword model (Section 3.2) to label a larger ReliefWeb corpus. We describe the other two ways (Ⓣ and Ⓐ) to obtain additional training data in the following sections.

[3]We showed the native speakers translation pairs obtained through the bilingual dictionary, and asked them to verify the translation as acceptable, or to supply a better translation.

[4]https://code.google.com/archive/p/word2vec/

[5]Threshold $th_1 = 0.9$ was determined by a grid search on a held-out English dataset.

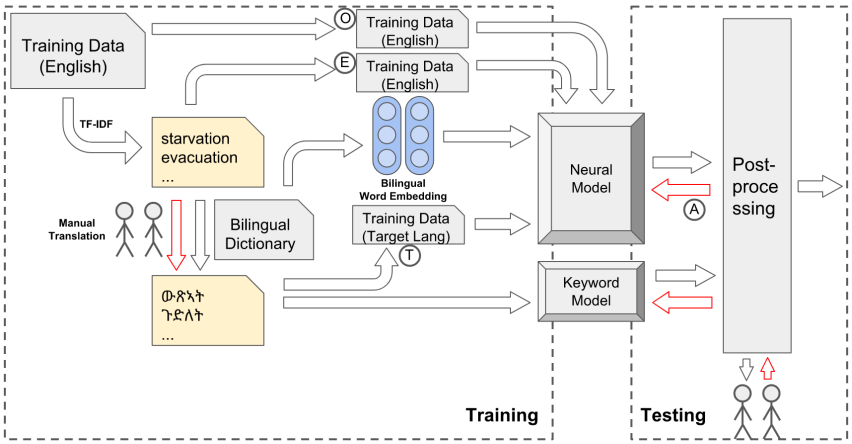


Figure 2: A summary of the various training data sources that we compare in this paper (Ⓞ, Ⓔ, Ⓣ, Ⓐ).

4.1 Bootstrapping Language-specific Keywords

We note that using simple keyword matching can result in low coverage due to missing keywords in the bilingual dictionary or word-variations in the target language. To overcome this, we developed an iterative bootstrapping algorithm that takes into account the newly labeled documents from keyword-spotting and generates additional language-specific keywords in a two-step process (Ⓣ in Figure 2).

Clustering: In the first step, we collected labeled documents from each class, and generated clusters for them ($D = \{D_{c_1},..,D_{c_m}\}$, where D_{c_p} is the cluster of the class c_p). For each cluster D_{c_p} and non-keyword w_i in it, we then computed the label affinity score $S_p(w_i)$, defined as follows:

$$S_p(w_i) = tfidf(w_i) + \frac{\sum_{w_j \in W_p} cos(w_i, w_j)}{|W_p|}$$

where W_p was the set of keywords present in D_{c_p}. We then appended the words which exceed a certain threshold th_2 to the keywords list of class c_p.

The rationale for this step is that the keywords that were missed in the initial keyword list (due to an incomplete bilingual dictionary, the keywords being language-specific, or incident-specific) may appear more frequently in the document cluster, and the second term in $S_p(w_i)$ will capture this.

Labeling: With the updated set of keywords for each class, we relabeled the documents to obtain a new set of labeled documents and again executed the clustering step to get more keywords. We can repeat this two-step process n times until we have the desired coverage or until this process no longer gives useful new keywords. In our experiments, we found that $n = 10$ generally gives good coverage. To generate the training data, we ran this procedure on the test set and took the top-100 most confident predictions.[6]

4.2 Annotation in Target Language

When we have the budget and annotators to do so, we can also annotate documents in the target language with class labels of interest. Given the limited budget and the rarity of documents with SFs (14-18% in our dataset), however, one question remains: how to best pick the documents to be annotated to maximize the gain from the additional training data? Seeing that the number of documents with at least one positive class is much less common compared to the number of documents without any positive class (see Table 1), simply taking a randomly sampled document from the unlabeled documents will likely give a document with no class, which is less useful compared to document with at least one positive class. Thus, we opt for a simpler method of asking annotators to make a binary decision on a

[6] As explained below, we used sentences as our training data, by taking the sentences which contain the keywords found.

	Tigrinya	Oromo
#Documents	2,991 (100%)	2,810 (100%)
– single sentence	2,508 (83.9%)	2,432 (86.6%)
– with 0 SFs	2,565 (85.7%)	2,307 (82.1%)
– with 1 SF	295 (9.9%)	361 (12.9%)
– with 2 SFs	95 (3.2%)	99 (3.5%)
– with 3 SFs	26 (0.9%)	26 (0.9%)
#Sentences	9,412	11,905
#SFs	612	721

Table 2: Data statistics for Situation Frame (SF) type extraction task in Tigrinya and Oromo dataset.

subset of our model's output on a separate dataset, different from the test set. We obtained 653 annotated sentences in Tigrinya this way (and 652 in Oromo). In addition to the native speakers, we also had non-speaker linguists annotate another separate (359) sentences in Tigrinya, assisted by grapheme-to-phoneme conversion, morphological glossing, and machine translation (MT) output.[7] This results in 1,012 sentences annotated in Tigrinya (Ⓐ in Figure 2 and Table 1). Overall, we spent less than 12 man-hours with native speakers of the target language to do the keyword translation and the annotation, with the larger amount of time spent on keyword translation.

5 Experiments

5.1 Dataset

For the purpose of the experiments and analysis, we used the dataset from the LoReHLT 2017[8] shared task, which consists of news articles in two Ethiopian languages: Tigrinya and Oromo.[9] The statistics of the dataset is shown in Table 2.

The available resources that we used for this experiment consist of:

1. Monolingual articles in Tigrinya and Oromo in various genres (news, discussion, social media).
2. Bilingual word dictionary (English-Tigrinya and English-Oromo).
3. A few hours of interaction with volunteers who are native speakers of Tigrinya or Oromo.
4. English documents about disaster recovery from ReliefWeb[10] and CrisisNet[11] annotated semi-automatically with disaster type and theme (Littell et al., 2017, Sec 2.3).[12]

5.2 Setup

We summarize more details about the experiment setup.

Sentence-level prediction: Although the model we used can be applied to produce document-level predictions directly, working at sentence-level provided more training data for the model and made it easier to train. Doing so also enabled some insight on which sentences contain the information about the document-level predictions.

Document-level aggregation: We then aggregate sentence-level predictions to a document-level prediction by assigning to each SF type the maximum confidence score of that type across all sentences in the document. Based on these scores, we calculate the mean confidence score μ_{c_p} of each SF type c_p. We then took the top-k ($k = 3$ in our experiments) SF types as our document-level prediction and filter out the predicted SF types which have confidence scores below μ_{c_p}. In the absence

[7] The MT model was also trained in a low-resource setting, with BLEU score around 12 for Tigrinya.

[8] https://www.nist.gov/itl/iad/mig/lorehlt-evaluations#lorehlt17

[9] For Oromo, the original dataset includes one annotator (out of 4) which annotated most of the documents with a single class. We did not consider this outlier annotator in our experiments.

[10] https://reliefweb.int/

[11] http://http://crisis.net/

[12] Also available at http://dx.doi.org/10.7910/DVN/TGOPRU

	Tigrinya			Oromo		
	P	R	F	P	R	F
KW	48.72	55.63	51.95	14.83	24.40	18.45
KW (bootstrap)	45.90	60.71	52.28	13.35	**44.56**	20.55
NN (Ⓞ)	50.30	58.38	54.04	9.09	18.24	12.13
NN (Ⓞ+Ⓔ)	**56.53**	65.86	60.84	13.65	22.10	16.87
NN (Ⓞ+Ⓐ)	53.32	67.67	59.64	14.82	23.79	18.27
NN (Ⓞ+Ⓔ+Ⓐ)	55.40	65.69	60.11	25.76	28.62	**27.12**
NN (Ⓞ+Ⓣ)	48.01	65.42	56.46	17.45	14.76	16.00
NN (Ⓞ+Ⓔ+Ⓣ+Ⓐ)	55.39	**70.25**	**61.94**	**32.80**	14.07	19.70

Table 3: Performance of the neural model (NN) with various sources of training data. Ⓞ is the original training data, Ⓔ is the additional training data in English from keyword-spotting, Ⓣ is the additional training data in target language from bootstrapping, and Ⓐ is the additional training data in target language from annotation. The result on keyword model (KW) is also shown for comparison.

of labeled data in the target language to be used as development set, this is one method that we can use without much tuning. In later sections we show how different document-level aggregation procedures may affect the performance.

Metric: We followed the metric defined in LoReHLT 2017 guidelines[13], which is *occurrence-weighted scores*, defined as follows:

$$P_{occ} = \frac{\sum \alpha_{tp}}{\sum \alpha_{tp} + \sum \alpha_{fp}}, \qquad R_{occ} = \frac{\sum \alpha_{tp}}{\sum \alpha_{tp} + \sum \alpha_{fn}}, \qquad F_{occ} = \frac{2 \cdot P_{occ} \cdot R_{occ}}{P_{occ} + R_{occ}}$$

where $\sum \alpha_{tp}$ is the number of true positives, weighted with the number of annotators that agree with it. $\sum \alpha_{fp}$ and $\sum \alpha_{fn}$ are similarly defined for false positives and false negatives. False negatives always have weight 1. For brevity, we drop the occ subscript when referring to these scores.

Model hyperparameters: In our neural CNN model, we used filter lengths of $\{3, 4, 5\}$ and 300 filters for each length. We also applied dropout on the extracted feature by CNN at a rate of 0.2. The model was optimized in mini-batches of size 128 by Adam (Kingma and Ba, 2014) optimizer at the learning rate of 0.001. The optimization was terminated after 30 epochs or a convergence criteria was satisfied on the held-out training data.

5.3 Results

Table 3 shows the results in Tigrinya and Oromo with the varying training data described in Section 4.

First, the keyword model (KW) gave results comparable to the neural model (NN Ⓞ), even outperforming it in Oromo. This suggests that in a low-resource setting, a keyword-based model can be used as a way to quickly get a working classifier, without the hassle of training a machine learning classifier or getting a large additional training data.

Next, the additional training data did help to significantly improve the performance of the baseline neural model in both languages. The large additional English data (+Ⓔ) provided a large boost both in Tigrinya (+6.8) and Oromo (+4.7). Interestingly, with only about 900 examples in the target language, the additional annotation in the target language (+Ⓐ) gave about the same improvements in F_1-score in Tigrinya, and even 1.4 points higher in Oromo compared to the large additional training data in English. Recall that the annotation was done on a subset of the neural model's output (trained on Ⓞ). This suggests that when an annotation budget is available, using that to verify the output of a model is a good investment.

It is interesting to note that each source of additional training data improved a different aspect of the model. The additional training data in English (Ⓞ+Ⓔ) seemed to improve precision more, while the

[13] https://goo.gl/FwRCwj

additional training data in target language from annotation seemed to improve recall more (Ⓞ+Ⓐ), and combining both (Ⓞ+Ⓔ+Ⓐ) provided the best of both worlds, especially in Oromo.

When we included the training data in target language from the keyword model with bootstrapping together with all other training data (Ⓞ+Ⓔ+Ⓣ+Ⓐ), it further improved the result in Tigrinya, but not in Oromo, although when it was used alone (Ⓞ+Ⓣ) it still gave some improvements. This could be due to the lower quality of the keyword system in Oromo. Recall that it was created by taking the top-100 most confident predictions of the keyword model. This set of predictions gave 75.9% precision in Tigrinya and 47.1% precision in Oromo. This lower quality of Oromo bootstrapping method can also be seen in the diverging SF Type distribution, as can be seen in Table 1.

The best overall improvement was more pronounced in Oromo (+14.99 points in F_1 for Ⓞ+Ⓔ+Ⓐ) than in Tigrinya (+7.90 points in F_1 for Ⓞ+Ⓔ+Ⓣ+Ⓐ). This could be related to the fact that the baseline score was much lower in Oromo than in Tigrinya to begin with.

For completeness, we also compare the results of the keyword model (KW) in target language without and with bootstrapping in the first two rows of Table 3. As anticipated, the bootstrapping process increased recall significantly, almost doubling the recall in Oromo. Although the precision was slightly reduced, it still resulted in an overall improvement in F_1-score for both languages.

In summary, there are four main observations:

1. The Neural model (NN Ⓞ) gave similar performance to the keyword (KW) model in Tigrinya and lower performance in Oromo, although the keyword model was a much simpler system.
2. With large additional training data in English (+Ⓔ), we obtained large improvements both in Tigrinya (+6.8) and Oromo (+4.7).
3. With only small additional annotations in target language (+Ⓐ) we obtained similar performance to using large English training data in Tigrinya, and even better in Oromo.
4. Getting additional training data in the target language through the keyword model can help if the quality of the keyword model is good enough.

6 Discussion

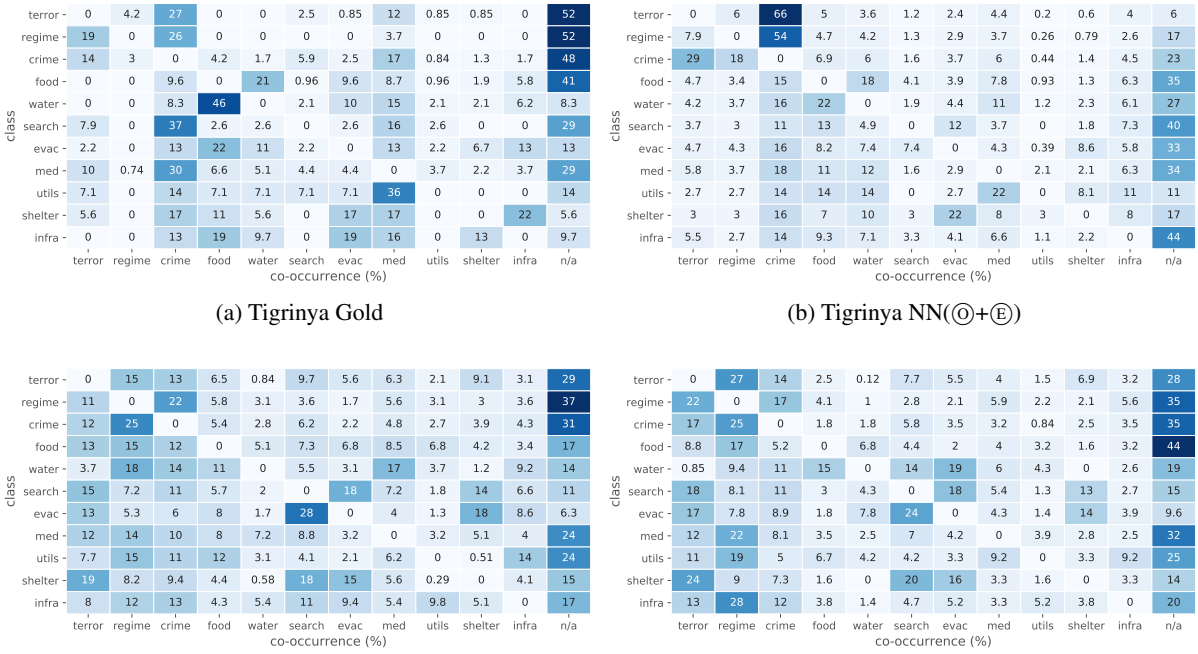
We hypothesize that the focused improvements on precision when using additional training data in English (Ⓔ) could be attributed to the similarity of the SF distribution to the original training data, since both are in English. This causes the model to be more confident in its prediction, at the expense of diverging away from the true distribution of SF types in the target language. In contrast, the annotated dataset in the target language (Ⓐ) has similar distribution to the true distribution, making the model able to rank correct SF types higher. We can see this from the SF type distribution shown in Table 1 by comparing the visualization at the last column.

Another explanation for the higher recall when adding annotated data in the target language is word coverage. The other two additional sources of data rely on keywords, and although bootstrapping helps improve coverage, the annotated data in the target language will cover more subtle correlations between word forms and class labels.

To analyze the differences between the various source of additional training data, we plot the co-occurrence of classes on the Tigrinya dataset in Figure 3. Each row describes the percentage of a particular SF type co-occurring with other SF types in the same document (recall that each document might be labeled with multiple SF types), including none, in which the SF type is the single label for that document. The numbers in a row sums to unity.

As can be seen, predictions of the NN system trained on the additional English data (Figure 3b) and target language data (Figure 3c) have different co-occurrence patterns. The additional English data apparently allowed the NN to find a strong correlation between the *crime violence* class and the *terrorism* and *regime change* classes, which is consistent with our intuition. On the other hand, the NN fine-tuned on the Tigrinya annotations apparently found the *terrorism* and *regime change* classes tend to occur alone.

There is also an interesting phenomenon that arises from the correlation between keywords and class labels. We found that the SF type *terrorism* is associated with the keyword "መንእሰይ" which means



(a) Tigrinya Gold

(b) Tigrinya NN(Ⓞ+Ⓔ)

(c) Tigrinya NN(Ⓞ+Ⓐ)

(d) Tigrinya NN(Ⓞ+Ⓔ+Ⓐ)

Figure 3: Co-occurrence of classes.

	Method	Tigrinya P	Tigrinya R	Tigrinya F	Oromo P	Oromo R	Oromo F
NN (Ⓞ+Ⓔ+Ⓐ)	top-1	68.77	52.45	59.51	36.03	19.93	25.66
NN (Ⓞ+Ⓔ+Ⓐ)	top-2	61.05	62.94	61.98	27.15	24.76	25.90
NN (Ⓞ+Ⓔ+Ⓐ)	top-3	55.40	65.69	60.11	25.76	28.62	27.12
NN (Ⓞ+Ⓔ+Ⓐ)	mean	42.11	71.54	53.01	17.74	36.71	23.92
NN (Ⓞ+Ⓔ+Ⓐ)	Tuned on ref.	80.00	55.03	65.21	36.43	44.08	39.89

Table 4: Impact of various aggregation strategy to the performance.

"youth" or "juvenile", as in the example sentence (English translation, the word recognized as keyword in the original language underlined) "According to the information, the Eritrean girls killed in the incident hide near the tyre of a car and was hot by the Sudanese soldiers." Examining the various examples in the dataset, we found that the violence inherent in terrorism is often depicted with youths as the victims. This could be related to the tendency of news outlets to focus on the suffering experienced by young people to make more emotional appeal.

6.1 Impact of Document-level Aggregation Strategy

In Section 5.2 we showed one heuristic to do document-level aggregation. One might wonder whether one can do better in the classification performance by using another aggregation strategy, such as filtering out classes with confidence scores under certain threshold, or using different k when taking top-k classes. In this section, we explore the impact of different aggregation strategies on the performance under different conditions. Assuming the more realistic case of having no development set to prefer one strategy over another, we can use the top-k strategy like we did in our experiments, or set a fixed threshold on the confidence score based on the average confidence score of each type across all documents in the test set. We also show the result when we set the threshold based on the reference annotation, to show how well the result can be in the case that we have a development set to find the best threshold. The result is shown in Table 4.

The significant gap of performance between the one tuned on the reference annotation and the rest suggests that if additional training data can be obtained in the target language, independently from the model's predictions, we should allocate a portion of them to be used for validation, since there are still large room for improvements just from tuning the thresholds (3-4% in Tigrinya and over 12% in Oromo). Note that in our experiments, since the annotation was done on the output of our neural model, we cannot use them as validation set, as it is biased towards the output of our model. So there is a trade-off between ease of annotation process and the amount of data that can be used as validation set.

7 Conclusion and Future Work

In this paper we tackled the problem of event type detection and classification in low-resource setting. We found that a neural model with adversarial training compared about the same as a simple keyword-based model using a small bilingual dictionary. Given that the problem lies with the limited amount of training data available, we proposed and compared methods to increase the amount of training data: to get significant gain in performance one can either use a very large additional semi-automatically labeled dataset in a resource-rich language, or annotate a small amount of documents in the target resource-poor language. We also showed how investing in a development set for tuning might also be a good strategy when there is a limited budget for annotation, after allocating some of them for keyword translation and additional training data.

One possible direction for future work is to address the mismatch of distribution of the classes between the additional training data and the actual test data as we see in Oromo. One way to mitigate the mismatch would be to make the classifier itself less prone to overfitting. In Section 6.1 we have shown how the document-level aggregation strategy may significantly affect the final result. Thus, exploring ways to effectively select the thresholds might be worthwhile. We could also incorporate the correlation between classes as evident from Figure 3, which has proven useful in multi-label classification (Zhang and Zhang, 2010).

Acknowledgments

We acknowledge NIST for coordinating the SF type evaluation and providing the test data. NIST serves to coordinate the evaluations in order to support research and to help advance the state-of-the-art. NIST evaluations are not viewed as a competition, and such results reported by NIST are not to be construed, or represented, as endorsements of any participants system, or as official findings on the part of NIST or the U.S. Government. We thank Lori Levin for the inputs for an earlier version of this paper. This project was sponsored by the Defense Advanced Research Projects Agency (DARPA) Information Innovation Office (I2O), program: Low Resource Languages for Emergent Incidents (LORELEI), issued by DARPA/I2O under Contract No. HR0011-15-C-0114.

References

Massih Amini, Nicolas Usunier, and Cyril Goutte. 2009. Learning from multiple partially observed views-an application to multilingual text categorization. In *Advances in Neural Information Processing Systems (NIPS)*, pages 28–36, Vancouver, British Columbia, Canada, December.

Peter Baumann and Janet Pierrehumbert. 2014. Using resource-rich languages to improve morphological analysis of under-resourced languages. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, pages 3355–3359, Reykjavik, Iceland, May.

Nuria Bel, Cornelis H A Koster, and Marta Villegas. 2003. Cross-lingual text categorization. *The 7th European Conference on Research and Advanced Technology for Digital Libraries*, 2769:126–139.

Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger. 2017. Adversarial deep averaging networks for cross-lingual sentiment classification. *ArXiv e-prints*.

Wim De Smet, Jie Tang, and Marie-Francine Moens. 2011. Knowledge transfer across multilingual corpora via latent topics. In *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, pages 549–560, Shenzhen, China, May. Springer.

Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2016. Learning crosslingual word embeddings without bilingual corpora. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1285–1295, Austin, Texas, USA, November. Association for Computational Linguistics (ACL).

Javid Ebrahimi, Dejing Dou, and Daniel Lowd. 2016. Weakly supervised tweet stance classification by relational bootstrapping. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1012–1017, Texas, USA, November. Association for Computational Linguistics (ACL).

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 462–471, Gothenburg, Sweden, April. Association for Computational Linguistics (ACL).

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning (ICML)*, pages 1180–1189, Lille, France, July.

Yuhong Guo and Min Xiao. 2012a. Cross language text classification via subspace co-regularized multi-view learning. *arXiv preprint arXiv:1206.6481*.

Yuhong Guo and Min Xiao. 2012b. Transductive representation learning for cross-lingual text classification. In *2012 IEEE 12th International Conference on Data Mining (ICDM)*, pages 888–893. Institute of Electrical and Electronics Engineers (IEEE).

Jagadeesh Jagarlamudi, Raghavendra Udupa, Hal Daumé III, and Abhijit Bhole. 2011. Improving bilingual projections via sparse covariance matrices. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 930–940, Edinburgh, Scotland, UK. Association for Computational Linguistics (ACL).

Rie Johnson and Tong Zhang. 2015. Semi-supervised convolutional neural networks for text categorization via region embedding. In *Advances in Neural Information Processing Systems (NIPS)*, pages 919–927, Montréal, Canada, December.

Nathan Keane, Connie Yee, and Liang Zhou. 2015. Using topic modeling and similarity thresholds to detect events. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 34–42, Denver, Colorado, USA, June. Association for Computational Linguistics (ACL).

Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Stroudsburg, PA, USA. Association for Computational Linguistics (ACL).

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*.

Johannes Knopp. 2011. Extending a multilingual lexical resource by bootstrapping named entity classification using wikipedia's category system. In *Proceedings of the Fifth International Workshop On Cross Lingual Information Access*, pages 35–43. Asian Federation of Natural Language Processing (AFNLP).

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI)*, pages 2267–2273, Austin, Texas, USA, January. Association for the Advancement of Artificial Intelligence (AAAI).

Patrick Littell, Tian Tian, Ruochen Xu, Zaid Sheikh, David Mortensen, Lori Levin, Francis Tyers, Hiroaki Hayashi, Graham Horwood, Steve Sloto, Emily Tagtow, Alan Black, Yiming Yang, Teruko Mitamura, and Eduard Hovy. 2017. The ARIEL-CMU situation frame detection pipeline for LoReHLT16: a model translation approach. *Machine Translation*, pages 1–22, October.

Michael L Littman, Susan T Dumais, and Thomas K Landauer. 1998. Automatic cross-language information retrieval using latent semantic indexing. In *Cross-Language Information Retrieval*, pages 51–62. Springer.

Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159, Berlin, Germany, August. Association of Computational Linguistics (ACL).

Luís Marujo, Miguel Bugalho, João Paulo da Silva Neto, Anatole Gershman, and Jaime Carbonell. 2013. Hourly traffic prediction of news stories. *arXiv preprint arXiv:1306.4608*.

Luis Marujo, Wang Ling, Isabel Trancoso, Chris Dyer, Alan W Black, Anatole Gershman, David Martins de Matos, João Neto, and Jaime Carbonell. 2015. Automatic keyword extraction on twitter. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 637–643, Beijing, China, July. Association for Computational Linguistics (ACL).

Rada Mihalcea, Carmen Banea, and Janyce Wiebe. 2007. Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL)*, pages 976–983, Czech Republic, June. Association of Computational Linguistics (ACL).

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3111–3119, Lake Tahoe, Nevada, United States, December.

Arzucan Özgür, Levent Özgür, and Tunga Güngör. 2005. Text categorization with class-based and corpus-based keyword selection. In *Proceedings of the 20th International Symposium on Computer and Information Sciences (ISCIS)*, pages 606–615, Istanbul, Turke, October. Springer.

John C Platt, Kristina Toutanova, and Wen-tau Yih. 2010. Translingual document representations from discriminative projections. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 251–261, Cambridge, Massachusetts, USA, October. Association for Computational Linguistics (ACL).

Lei Shi, Rada Mihalcea, and Mingjun Tian. 2010. Cross language text classification by model translation and semi-supervised learning. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1057–1067, Cambridge, Massachusetts, USA, October. Association for Computational Linguistics (ACL).

György Szarvas. 2008. Hedge classification in biomedical texts with a weakly supervised selection of keywords. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, pages 281–289, Columbus, Ohio, June. Association for Computational Linguistics (ACL).

Dang Tran, Cuong Chu, Son Pham, and Minh Nguyen. 2013. Learning based approaches for vietnamese question classification using keywords extraction from the web. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing (IJCNLP)*, pages 740–746, Nagoya, Japan, October. Asian Federation of Natural Language Processing (AFNLP).

Alexei Vinokourov, Nello Cristianini, and John S Shawe-Taylor. 2002. Inferring a semantic representation of text via cross-language correlation analysis. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1473–1480, Vancouver, British Columbia, Canada, December.

Xiaojun Wan. 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 (ACL-IJCNLP)*, pages 235–243, Singapore, August. Association for Computational Linguistics (ACL).

Min Xiao and Yuhong Guo. 2013. A novel two-step method for cross language representation learning. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1–9, Lake Tahoe, Nevada, United States, December.

Ruochen Xu and Yiming Yang. 2017. Cross-lingual distillation for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1415–1425, Vancouver, British Columbia, Canada, July. Association for Computational Linguistics (ACL).

Ruochen Xu, Yiming Yang, Hanxiao Liu, and Andrew Hsi. 2016. Cross-lingual text classification via model translation with limited dictionaries. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM)*, pages 95–104, Indianapolis, Indiana, USA, October. Association for Computing Machinery (ACM).

Min-Ling Zhang and Kun Zhang. 2010. Multi-label Learning by Exploiting Label Dependency. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 999–1008, Washington DC, USA, July. Association for Computing Machinery (ACM).

Huiwei Zhou, Long Chen, Fulin Shi, and Degen Huang. 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 (ACL-IJCNLP)*, pages 430–440, Beijing, China, July. Association for Computational Linguistics (ACL).

Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2016a. Attention-based LSTM network for cross-lingual sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 247–256, Austin, Texas, USA., November. Association for Computational Linguistics (ACL).

Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2016b. Cross-lingual sentiment classification with bilingual document representation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1403–1412, Berlin, Germany, August. Association for Computational Linguistics (ACL).