Focus Location Extraction from Political News Reports with Bias Correction

Conference Paper · December 2017

DOI: 10.1109/BigData.2017.8258141

CITATIONS
13

READS
952

5 authors, including:

Maryam Bahojb Imani
University of Texas at Dallas
13 PUBLICATIONS **169** CITATIONS
SEE PROFILE

Swarup Chandra
University of Texas at Dallas
30 PUBLICATIONS **497** CITATIONS
SEE PROFILE

Latifur Khan
University of Texas at Dallas
205 PUBLICATIONS **1,670** CITATIONS
SEE PROFILE

Focus Location Extraction from Political News Reports with Bias Correction

Maryam Bahojb Imani, Swarup Chandra, Samuel Ma, Latifur Khan, Bhavani Thuraisingham
Computer Science Department
The University of Texas at Dallas, Richardson, TX, United States
{maryam.bahojbimani, swarup.chandra, samuel.ma, lkhan, bhavani.thuraisingham }@utdallas.edu

***Abstract*—**
Automatic identification of geolocation mentioned in online news articles provide vital information for understanding associated events. While numerous open-source and commercial tools exist for geolocation extraction, they lack in reliable identification of fine-grained location, i.e., they identify location at country-level rather than a fine-grained city or locality level. The problem of location identification has been widely studied. Yet, most techniques depend on external knowledge-base or view the problem only in terms of Named Entity Recognition (NER), only to identify country-level location information. In this paper, we focus on news articles describing an event. A set of locations directly associated with the event are called *focus* locations. However, an event can occur only at a single location. Therefore, we aim to extract this location among focus locations, and call this as *primary focus location*. We propose a mechanism that utilizes the named entities to identify potential sentences containing focus locations, and then employ a supervised classification mechanism over sentence embedding to predict the primary focused geolocation. However, the main issue with such an approach is the unavailability of ground truth (i.e., whether words in a sentence is focus or non-focus) for training a classifier. In practice, labels from only a small number of news articles may be available for training due to high cost of manual labeling. If these articles are not a good representation of news articles in the wild, the classifier may not perform well. Therefore, we utilize an adaptation mechanism to overcome sampling bias in training data. Particularly, we train a classifier by using bias-corrected training data obtained from news articles published by an agency, while testing it on news articles published by a different agency. Our empirical results show superior performance compared to baseline approaches on real-world datasets consisting of news articles.

***Keywords*-Focus Location Extraction; Sentence Embedding; Bias Correction**

I. Introduction

With an ever-increasing number of news reports, there is a need for automatic news analysis. Particularly, articles containing stories about people, events and places form a rich source of information that can be used in applications that search and organize daily news stories for analysis. These applications include determining crime pattern locations, predicting the place of protests and political unrest, and identifying the geolocation of natural disasters. Such applications can largely benefit from identifying precise geolocation information in a timely manner to provide better support for decision making.

by Ola' Audu 399 words 19 May 2014 10:03 All Africa AFNWS English May 19, 2014 (Premium Times/All Africa Global Media via COMTEX)
-- At least 40 villagers were killed and several others injured on Saturday as gunmen believed to be Boko Haram members attacked Dalwa-Masuba Village in Damboa Local Government Area of Borno State, security sources and a witness said. The gunmen who stormed the village in large numbers also burnt down virtually all buildings in the village as well as three pickup vans carrying woods to Damboa. A member of the vigilante in Dalwa-Masuba, who spoke to journalists in Maiduguri on phone, said no security personnel had reached the attacked town at the time he was speaking. "We were on patrol somewhere near Damboa when we heard about the attack from some of the villagers who ran from the village", said the source. "We had to drive to the town on our patrol van; we met the entire village on fire, and about 40 persons dead, there were bodies all over the place; three firewood pickups were also set ablaze." The police and the military in Borno are yet to formally confirm the attack, although security sources in Maiduguri, the state capital, said they had been briefed of the attack. Dalwa-Masuba is a farming community 40km away from Damboa Town and about 80km south-west of Maiduguri. The attack follows similar patterns of attacks on communities in Borno by the Boko Haram. The group has continued its attacks and killed thousands of people despite a state of emergency imposed on Borno, Yobe, and Adamawa in May last year. The atrocities of the group, including its kidnap of over 250 teenage, female students in Chibok, Borno State, on April 14, has drawn international attention and condemnation. At a security summit in the French capital, Paris, Saturday, attended by President Goodluck Jonathan, the leaders of Chad, Niger, Benin, and Cameroun, agreed to share intelligence, and co-ordinate action against the group which is based in northeast Nigeria, but has operated somewhat freely in northwest Cameroun, parts of Chad and Niger. A central intelligence platform will be based in Chad, the summit agreed, and will allow all countries involved, including the world powers, to stage a response as necessary. Representatives of the United States, United Kingdom, France, and the European Union, also attended the Saturday's meeting.

Figure 1: A sample news report with different place names from Atrocity dataset [2]

Typically, the term *Location* is used in a variety of contexts. A location can be precise GPS co-ordinates, or in general a continent. But, the term *Locality* is used to describe a more precise area [1]. In this paper, we focus on identifying the associated locality information mentioned in a news article. Specifically, we aim to precisely identify the locality of an event described in a news article. A news article may contain multiple related localities. These are called *Focus Locations*. However, we aim to identify the place of occurrence of the event. We call this particular locality as *Primary Focus Location*.

For example, consider a news report given in Figure 1. This report[1] was published on May 18, 2014, on the AllAfrica news website [2]. The report describes an atrocity event that occurred in the village of *Dalwa-Masuba*, Nigeria. Moreover, the report also mentions other locations including Damboa, Maiduguri, Borno, Yobe, Adamawa, Chibok, Paris, etc. Here, we say the “Dalwa-Masuba" is the primary focus location since the event took place in that location. However,

[1]http://allafrica.com/stories/201405180007.html

Tool	Extracted Country	Extracted Locations	Focus Country	Focus Location
Cliff-Clavin	NG, FR, TD, NE, BJ,CM,US,IT	Adamaoua, Benin, Borno, Cameroun, Chad, Chibok, Damboa, France, Niger, Maiduguri, Paris, United Kingdom, United States, Yobe	NG	Borno, Damboa Maiduguri
Geoparser	NG, NE, FR, USA AE,CM,TD,BJ	Adamawa,Benin,Borno,Cameroon,Chad,Chibok,European Union,Faransā, Maiduguri, Niger, Nigeria, Paris, United States, Yobe	-	-
Stanford CoreNLP	-	Adamawa, Benin, Boko Haram, Borno, Chad, Chibok, Cameroun, Dalwa-Masuba Damboa, France, Maiduguri, Niger, United Kingdom, United States, Yobe	-	-
Mordecai	NG	Borno, Cameroun-Gbene, Chibok, Dalwa, Damboa, Komadugu, Yobe, Maiduguri	NG	-
Edinburgh	-	Adamawa, Benin, Borno, Cameroun, Chad, Chibok, Damboa, France, Maiduguri, Niger, Nigeria, Paris, United Kingdom, United States, Yobe	-	-

Table I: Output of focus location extraction from existing tools including Cliff-Clavin, Geoparser, Stanford and Mordecai for the given example in Figure 1

other localities associated with the event is Borno and Damboa, which are course-grained. They form the elements of the focus location set.

Even though several geoparsers such as Cliff-Clavin [1], Mordecai [3], and Stanford-CoreNLP [4] have been developed to automatically extract named locations from unstructured text, location extraction from a text is still a challenging task due to the complexity, diversity, and ambiguity of location information. However, those tools cannot extract the *focus location* with good accuracy, and most of them cannot differentiate between different locations in the text, i.e. focus locality versus non-focus locality. Following the example in Figure 1, Table I show the output of these different tools for extracting focus location. Clearly, these tools identify multiple locations mentioned in the news article. Among them, only Cliff-Calvin is able to identify a few focus locations. Yet, it is unable to identify the desired primary focus location.

In the paper, we investigate extracting the primary focus geolocation automatically from unstructured text-based news reports. The main challenge is to distinguish between the different candidate locations to identify the primary focus location. We address this challenge by using a supervised classification model that leverages contextual patterns in occurrences of focus locations. Concretely, we first use a named entity recognition tool to extract candidate locations and identify the sentences in which they occur. We then extract semantic features from these sentences by using word2vec model and sentence embedding approaches [5], [6]. Finally, we train a classifier on labeled training instances and predict the primary focus location on unlabeled test sentences. We denote this approach by PRimary Focus Location Extraction or *Profile*.

A major challenge with the above approach is the requirement of suitable labeled data instances for training. In the real-world, these labeled instances are not readily available, or may be sparingly available. Traditional supervised learning methods assume that the training and test data sets are generated from the same data distribution. In practice, however, this assumption may not be true. In our scenario, true labels of sentences (focus or non-focus) may be only available for a small number of news articles or from news articles associated with a single news agency. In such cases, these labeled articles may not be a good representative of the population. For example, news articles from different agencies typically have dissimilar writing styles, linguistic content, vocabularies, or type of emotions (e. g., acted, elicited, or naturalistic). Such differences negatively affect classifier performance when employed to predict focus locations in news articles in the wild, and prevents scalability of our approach.

We address this challenge by manually labeling only a small amount of sentences. This creates sampling bias between the training and test data sets. We then leverage the approach of sampling bias correction by appropriately weighting each training data instance using density ratio estimates between the test and training data distributions. Particularly, we apply a well-known technique called Kernel Mean Matching (KMM) [7] to estimate these density ratios (or importance weights), and use them to train a classifier for prediction of focus location.

Key contributions of this paper are as follows:

- We address the problem of automatically predicting a primary focus location for event-based news reports by extracting semantic features that aid in capturing patterns of focus location occurrences, rather than using an external database such as gazetteer [2].
- We propose a supervised mechanism called *Profile* that identify the primary focus location in each news article. Particularly, we address the label scarcity problem to train a classifier by using a smaller set of biased training data and leverage a well-known bias-correction mechanism to evaluate test data from the same domain.
- We empirically evaluate Profile over real-world Atrocity and New York Times news articles and compare its performance against well known geoparsers.

Rest of the paper is organized as follows. We review related works and present relevant background of geolocation extraction in Section II. We then present our proposed approach in Section III to address the above challenges. Finally, we empirically evaluate our approach in Section IV and conclude in Section V.

[2]http://www.geonames.org/

Tools	NER Extraction	Location Extraction	Focus Country	Focus Location
Cliff-Clavin	Stanford CoreNLP	✓	✓	✓ (City, State)
Mordecai	MITIE	✓	✓	✗
Geoparser.io	(Not mentioned)	✓	✗	✗
Edinburgh	Based on rules and lexicons	✓	✗	✗

Table II: Capabilities of different tools in focus country and focus location extraction. Here, ✓ indicates presence and ✗ indicates absence of corresponding capability.

II. Background

A. Geolocation Extraction

Different tasks and analyses have been conducted in the field of geolocation extraction from the text. The three main tasks in the area of geolocation extraction are as follows.

- Location named entity extraction [8], [9];
- Location named entity resolution [10];
- Event's location extraction [1], [3].

In this paper, we are focusing on the last task, i.e. the event's location extraction by using geoparser.

Silva et al. [11] presented a framework to identify geographic scopes in Portuguese web pages automatically. They employed different external sources such as WHOIS and DNS registrars, and the Portuguese postal codes database to locate the geographical entities in web pages. Web-a-where [12] is another web page geotagger that uses a gazetteer to identify all location names in web pages, assign a geographic location and confidence level to each page, and derive a focus location for the entire page. NewsStand [13] extracts geographic locations from news articles, but it focuses on the geographic focus of a collection of news reports about the same subject by applying the document clustering algorithm. Mordecai [3] is an open source geoparser that uses MITIE to extract place names from the text. It then identifies the focus country and all place names in the text using a gazetteer. Cliff-Clavin is also an open source geoparser that parses news articles or other documents. It uses Stanford CoreNLP to extract organizations and locations mentioned in the text, and employs context-based geographic disambiguation. It uses a simple method to identify focus places based on frequency of place mentioned at city, state, and country levels [1]. Geoparser.io [14] is a web service that helps to identify place names and handle contextual ambiguity in those place names. The Edinburgh Geoparser [15] is another geoparser designed to analyze unstructured text in order to identify occurrences of locations, and map them to the correct latitude and longitude. We compare the empirical results of our approach with these competing methods.

In Table II, the capability of each geoparsers in focus country and focus location extraction is presented. Accordingly, Mordecai and Cliff-Clavin are the only geoparsers that can extract the focus country. Cliff-Clavin is the only tool that can extract focus location on two different levels, i.e. city and state.

B. Bias Correction (Covariate Shift)

An element belonging to a training set is indicated by a subscript tr, while that of a test set is indicated by a subscript te. Since a set may contain multiple elements, each element is indexed by a superscript integer. For example, $\mathbf{x}^{(i)} \in \mathbf{X}_{tr}$ indicates the i^{th} data instance (array of d covariates) of a training dataset $\mathbf{X}_{tr}$ (a set containing arrays). Also, a hat indicates estimated value. In general, we use a bold letter to indicate an array, and a capital-bold letter to indicate a set of arrays.

In the case of data classification — a binary focus or non-focus classification in this paper — inequality in probability distribution of training and test data sets can be represented in the form of joint probability distribution $p_{tr}(\mathbf{x}, y) \neq p_{te}(\mathbf{x}, y)$, where $\mathbf{x} \in \mathbb{R}^d$ is the d-dimensional covariate of a data instance with class label y. p_{tr} and p_{te} are the training and test probability distribution respectively. According to Ben-David et al. [16], if the two distributions are arbitrarily different, then learning is not possible with bounded error. However, this challenge can be addressed by a method that transfer knowledge (model) from training data to test data using instances or feature representation under several assumptions [17].

One such assumption is the equality in class conditional distribution. Concretely, $p_{tr}(y|\mathbf{x}) = p_{te}(y|\mathbf{x})$. Therefore, the inequality in joint probability distribution is attributed to the covariate distribution, i.e., $p_{tr}(\mathbf{x}) \neq p_{te}(\mathbf{x})$. This is known as *covariate shift*. Overall, a correction to the inequality between $p_{tr}(\mathbf{x})$ and $p_{te}(\mathbf{x})$ is provided by computing an importance weight $\beta(\mathbf{x}) = \frac{p_{te}(\mathbf{x})}{p_{tr}(\mathbf{x})}$ for each instance $\mathbf{x}$. A supervised classifier can then be trained using this weighted training data set whose data distribution is equivalent to the test data distribution. Various studies have focussed on directly estimating the importance weighting function (or density ratio) rather than computing $p_{te}(\mathbf{x})$ and $p_{tr}(\mathbf{x})$ separately. These include Kernel Mean Matching (KMM) [7], Kullback-Leibler Importance Estimation Procedure (KLIEP) [18], and unconstrained Least Square Importance Fitting (uLSIF) [19].

1) Kernel Mean Matching: The main idea in KMM is to decrease the mean distance between weighted training data distribution $\beta(\mathbf{x})p_{tr}(\mathbf{x})$ and the observed test data distribution $p_{te}(\mathbf{x})$ in a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{F}$ with feature map $\phi : \mathcal{D} \rightarrow \mathcal{F}$. Mean distance is determined by computing the *Maximum Mean Discrepancy* (MMD), given by

$$\left\| E_{\mathbf{x}\sim p_{tr}(\mathbf{x})}[\beta(\mathbf{x})\phi(\mathbf{x})] - E_{\mathbf{x}\sim p_{te}(\mathbf{x})}[\phi(\mathbf{x})] \right\| \qquad (1)$$

where $\|\cdot\|$ is the l_2 norm, and $\mathbf{x} \in \mathbf{X} \subseteq \mathcal{D}$ is a data instance in a dataset $\mathbf{X}$. Here, it is assumed that p_{te} is absolutely continuous with respect to p_{tr}, i.e. $p_{te}(\mathbf{x}) = 0$

whenever $p_{tr}(\mathbf{x}) = 0$. Furthermore, the RKHS kernel h is universal in the domain. It has been proven that under these conditions, minimizing MMD in Equation 1 converges to $p_{te}(\mathbf{x}) = \beta(\mathbf{x})p_{tr}(\mathbf{x})$ [20].

In general, minimizing MMD to find desired importance weights is equivalent to minimizing the corresponding quadratic program that estimates the population expectation with an empirical expectation. The empirical approximation of MMD (Equation 1) to get the optimal solution for $\hat{\beta}(\mathbf{x})$ is given by

$$\hat{\boldsymbol{\beta}} \approx \arg\min_{\beta} \left\| \frac{1}{n_{tr}} \sum_{i=1}^{n_{tr}} \beta(\mathbf{x}_{tr}^{(i)}) \phi(\mathbf{x}_{tr}^{(i)}) - \frac{1}{n_{te}} \sum_{j=1}^{n_{te}} \phi(\mathbf{x}_{te}^{(j)}) \right\|^2 \tag{2}$$

where $\hat{\beta}(\mathbf{x}) \in \hat{\boldsymbol{\beta}}$. The equivalent quadratic program is as follows.

$$\hat{\boldsymbol{\beta}} \approx \underset{\beta}{\text{minimize}} \frac{1}{2}\boldsymbol{\beta}^T \mathbf{K} \boldsymbol{\beta} - \boldsymbol{\kappa}^T \boldsymbol{\beta} \tag{3}$$

$$\text{subject to } \beta(\mathbf{x}^{(i)}) \in [0, B], \forall i \in \{1 \dots n_{tr}\}$$

$$\& \left| \sum_{i=1}^{n_{tr}} \beta(\mathbf{x}^{(i)}) - n_{tr} \right| \leq n_{tr}\epsilon$$

where $\mathbf{K}$ and $\boldsymbol{\kappa}$ are matrices of a RKHS kernel $h(\cdot)$ with $K^{(ij)} = h(\mathbf{x}_{tr}^{(i)}, \mathbf{x}_{tr}^{(j)}) \in \mathbf{K}$, and $\kappa^{(i)} = \frac{n_{tr}}{n_{te}} \sum_{j=1}^{n_{te}} h(\mathbf{x}_{tr}^{(i)}, \mathbf{x}_{te}^{(j)}) \in \boldsymbol{\kappa}$. $B > 0$ is an upper bound on the solution search space, and ϵ is the normalization error. In this paper, we utilize the KMM algorithm for bias correction on training data.

III. Approach

A. Focus Location Extraction

Figure 2 shows an overview of Profile for primary focus location extraction. We first pre-process the given news reports to extract candidate focus location, and then utilize a supervised classifier to identify a *primary* focus location among them. In the pre-processing step, we choose a user defined number (denoted by γ) of sentences in each news report, since focus locations are mostly mentioned in the first few sentences. Then, by using Stanford CoreNLP, we identify the location named entities in the training news report among these first few sentences. Next, we select sentences that contain locations, and extract the sentence features. If the sentence includes a focus location, we assign a *Focus* label to it; otherwise, we assign a *Non-Focus* label to it. Finally, we train a binary classifier in a supervised manner using this labeled training data.

On the other hand, in the test phase, after preprocessing step, we employ the model to assign a label to each sentence in each report. The labels consists of either "Focus" or "Non-Focus". We call the labeled sentences as *focus sentences*. Note that each news report may include more than one focus sentence. In order to identify the primary focus location, we use a frequency-based approach to select the focus location among candidate location names in the collection of focus sentences for each report. Particularly, the frequency-based approach is as follows. We form a histogram of each location detected by NER tools. The location having the highest count is selected as the focus location.

In the next two subsections, we will present the features we have extracted from a text-based dataset. Then we will present our learning method used to identify the focus locations in the unstructured text.

1) Word Embedding: Our feature extraction algorithm is based on using pre-trained word embedding model from raw text. We utilized the publicly available word2vec vectors that were trained on 100 billion words from Google. The length of these vectors is 300. Words not present in the set of pre-trained words are initialized as zeros. An interesting property of the word embeddings is that these vectors effectively encode the semantic meanings of the words in the context. In other words, they are able to represent meaningful syntactic and semantic regularities in a very simple way [6].

2) Sentence Embedding: Our basic sentence feature extraction method follows the Sentence Embedding [5]. We employed this approach because it gives more weight to uncommon words in the corpus. In other words, common words become less important in the dataset. An alternative approach to find the sentence vector is computing the mean of words' vectors in the sentence. We will compare the effectiveness of the Sentence Embedding approach with assigning different weight to each word and the alternative approach empirically in Section IV.

Let c_s be a discourse vector, s be a given sentence, S be a set of sentences and α is a scalar. The discourse vector represents"what is being talked about." Assume that $p(w)$ is the unigram probability of a word in a corpus. Given the discourse vector c_s, the probability of a word w in the sentence s is $p(w|c_s)$.

$$p(w|c_s) = \alpha p(w) + (1-\alpha)\frac{exp(< v_w, \tilde{c_s} >)}{Z_{\tilde{c_s}}} \tag{4}$$

where

$$\tilde{c_s} = \beta c_0 + (1-\beta)c_s, c_0 \perp c_s$$

$c_0 \in \mathbb{R}^d$ is a common discourse vector which serves as a correction term for the most frequent discourse that is often related to syntax, and $Z_{\tilde{c_s}}$ is a normalizing constant given as follows.

$$Z = \sum_{w \in \mathcal{V}} exp(< v_w, \tilde{c_s} >)$$

So, the likelihood for the sentence s is:

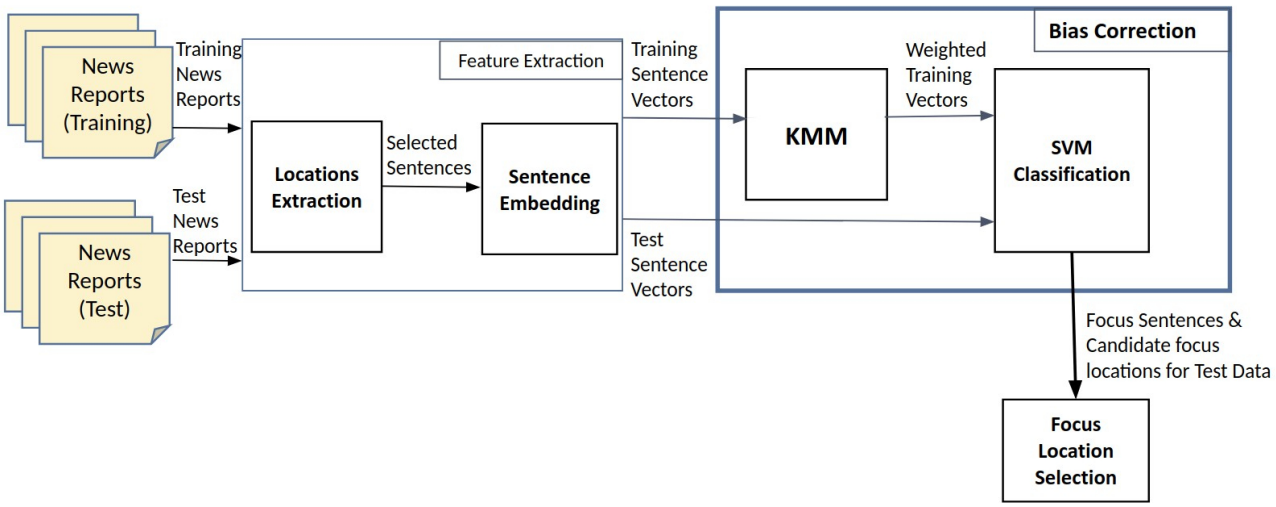


Figure 2: A high level schema of Profile (Primary Focus Location Extraction).

$$p(s|c_s) = \prod_{w \in s} p(w|c_s)$$
$$= \prod_{w \in s} \left(\alpha p(w) + (1-\alpha) \frac{exp(< v_w, \tilde{c_s} >)}{Z} \right) \quad (5)$$

where Z is roughly the same as $Z_{\tilde{c_s}}$.

The maximum likelihood estimator for $f_w(c_s) = log(p[s|c_s])$ is approximately,

$$\arg\max f_w(\tilde{c_s}) \propto \sum_{w \in s} \frac{a}{p(w)+a} v_w \quad (6)$$

where

$$a = \frac{1-\alpha}{\alpha Z}$$

The MLE is approximately a weighted average of the vectors of the words in the sentence. To estimate c_s, we estimate the direction c_0 by computing the first principal component of $\tilde{c_s}$ for a set of sentences. Therefore, in order to omit the effect of a common discourse vector which is often related to the syntax, the final sentence embedding is computed by subtracting the first principle component from $\tilde{c_s}$. More details of this method are described in [5].

The process of feature extraction by using sentence embedding is summarized in Algorithm 1. The inputs of the algorithm are News_Reports, focus_locations, Word_Embedding, and Parameters a and γ. In the first For-loop of the Algorithm (line 1 to 11), we extract set of the locations (loc) by using Stanford CoreNLP as a named entity recognizer (NER) for each news report (line 3), and exclude countries' name from them in line 4. Then, we select the first γ sentences for each news report which contain at least one location name (line 6 to 10). In the next for-loop, we compute the sentence embedding vector ($\mathbf{v_s}$) for each sentence, based on equation 6 (line 12 to 14). For more frequent words w, the weight $\frac{a}{a+p(w)}$ is smaller, so this leads to smaller weights for frequent words. Finally, we compute the first principle component u and decrease it from sentence vector $\mathbf{v_s}$ (line 16 to 18). After extracting the feature vectors, we trained the SVM classification model.

In test process, we apply the same algorithm. However, in the first for-loop, we just use the locations extracted by using Stanford CoreNLP ($loc \leftarrow \mathrm{StanfordNER}(\mathrm{News}_i)$). Then, we classify the feature vectors by using the model. Since there may be more than one Focus sentence per report (i.e., sentence containing potential focus location), we extract the locations from Focus sentences. Next, we use the frequent-based approach to extract the Focus locality. In frequency-based approach, we select the most frequent item in the list. In other words, if we find several sentences from one article with a focus label, the most frequent location name will be a candidate for focus location.

As mentioned earlier, in Section I, we may not have sufficient labeled data to train an unbiased classifier. In such a case, we employ the following approach for bias correction over training data. We first perform pre-processing steps by extracting feature vectors from the given biased training data. We then apply the bias correction method over these feature vectors. Particularly, we compute instance weight for each training data using KMM. This utilizes the given test data instances to estimate density ratios. We then use the weighted training data in RKHS to train a suitable classifier. This classifier is used to predict focus location over test focus sentences.

Algorithm 1: Feature Extraction in Profile using Sentence Embedding

Data:
News_Reports $\text{News}_1, ..., \text{News}_N$, focus_locations $\text{Floc}_1, ..., \text{Floc}_N$, Word_Embedding $\{\mathbf{v_w} : w \in \mathcal{V}\}$, Parameter a and γ
Result: Sentence_Embedding $\mathbf{v_s}$

for ***each*** *News_Report* (News_i) **do**
  /* Extract the Location Named Entities for News_i by using Stanford NER */
  $loc \leftarrow \text{StanfordNER}(News_i) \cup Floc_i$
  $loc \leftarrow \text{loc} \setminus \text{Country\_Names}$
  /* Select the first γ sentences which contain a location in loc. S is a list of these sentences (s). */
  for ***each sentence*** *(s)* **do**
    if $(\#s <= \gamma \; and \; \exists i : \text{loc}_i \in s)$ **then**
      $S \leftarrow S \cup s$
    end
  end
end
for ***each sentence*** s_i *in* S **do**
  $\mathbf{v_{s_i}} \leftarrow \frac{1}{|s_i|} \sum_{w \in S} \frac{a}{a+p(w)} \mathbf{v_w}$
end
/* Compute the first principal component u of v_{s_i} */
for ***each*** *Sentences* s_i *in* S **do**
  $\mathbf{v_{s_i}} \leftarrow \mathbf{v_{s_i}} - \mathbf{u}\mathbf{u}^T\mathbf{v_{s_i}}$;
end

Dataset	# News Reports	# Sentences
Atrocity Event Data	15K	175K
New York Times	1.8M	30M

Table III: Datasets Statistics

IV. Experiments

In this section, we first explain the dataset used to evaluate the proposed method to extract focus locations, and then present the evaluation results while comparing it with the other competing methods.

A. Dataset

The Atrocities Event Data [2] is a collection of recent news reports on atrocities and mass killings in several locations. Human coders have read the reports and extracted metadata about the events reported. The annotated reports includes victims, focus location, and the reports that reported the event For the training and testing dataset, we excluded the reports that contain multiple events. Moreover, we only select reports whose locations were correctly extracted by different NERs such as Stanford and MITIE since the performance of NER is beyond the scope of this paper. The original size of Atrocity dataset is about $15K$ reports, and almost $5K$ of them are annotated.

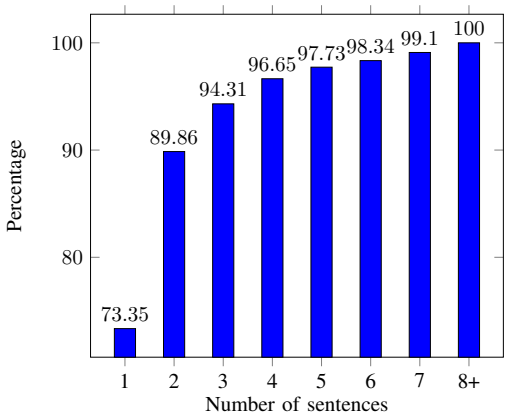


Figure 3: Percentage of documents containing focus location in the initial set of sentences.

Another dataset that we used is the New York Times (NYT)[3] news reports dataset. The New York Times Annotated Corpus includes more than 1.8 million articles composed and published by the New York Times between January 1, 1987 and June 19, 2007 with article metadata. Similar to the Atrocity Event dataset, we only select political news articles that contain special keywords such as kill, die, injure, dead, death, wounded and massacre in their title. Although NYT corpus includes location annotations, all of them are not focus locations. Accordingly, we randomly selected 1000 news reports and manually tagged them. The overall number of news reports and sentences for both corpus are given in Table III.

B. Experiments

The experiments were conducted on an Intel machine having Core-i7 3.40GHz CPU with 64 GB of RAM, running a standard Ubuntu Linux version 16.04 LTS. We also set $a = 0.1$ and $\gamma = 7$ as inputs for Algorithm 1 as default. We choose $\gamma = 7$ (which means first seven sentences of any news reports are selected to input the algorithm) since we observed that focus locations were present in the first 7 sentences of the training set in more that 99% of news reports. This is illustrated in Figure 3. The chart demonstrates that focus location in 73% of articles can be found in the first sentence, and less than 1% of articles contain location information in sentences that occur after the 8^{th} sentence. After applying this filtering scheme, the overall number of resulting news reports and sentences for Atrocity dataset are about $3.6K$ and $40K$, respectively. Similarly, $10K$ annotated sentences in NYT dataset remain.

Profile[4] uses a support vector machine (SVM) with a RBF kernel as a base classifier since it supports weighted training

[3]https://catalog.ldc.upenn.edu/ldc2008t19
[4]https://github.com/Maryam-Imani/Profile

Dataset	Method	Precision	Recall	F1
Atrocity	Profile_w	70.05	76.37	73.07
Event	Profile_w (γ=7)	73.60	81.47	77.34
Data	Profile_s	70.33	78.29	74.10
	Profile_s (γ=7)	**74.51**	**92.66**	**82.90**
New	Profile_w	60.75	51.36	55.66
York	Profile_w (γ=7)	72.45	75.53	73.95
Times	Profile_s	60.22	55.75	57.90
	Profile_s (γ=7)	**76.41**	**77.14**	**76.77**

Table IV: In-agency focus location extraction performance.

data in RKHS. Here, SVM parameter values are $c_{svm} = 1000$ and $\gamma_{svm} = 0.1$.

First, we demonstrate the effectiveness of using sentence embedding and sentence filtering by comparing the performance of Profile when employing word embedding schemes. Here, a sentence embedding is formed from word embedding by taking the mean of word embeddings occurring in the sentence. We denote this as Profile_w. On the contrary, we denote the use of sentence embedding scheme explained in Section III as Profile_s. Later, we compare the performance of Profile with other tools, including Cliff-Clavin. We also use the frequency-based approach to extract focus locations from Mordecai and Stanford-CoreNLP. Since Stanford-CoreNLP was only developed for named entities such as person and location names, it does not distinguish between different levels of location, such as locality and country. Therefore, we modified the Stanford-CoreNLP output and excluded country names from the resulting location names, and then use the frequency-based approach to obtain the top location name as a surrogate for primary focus location. To train the model for each dataset, we randomly picked 60% of articles for training data, and use the remaining 40% of news reports as test dataset. As shown in Figure 2, Profile first performs the pre-processing steps on both training and test datasets to extract sentences containing potential primary focus location. Then, it then extracts their related features from the text.

The above experiments assume that the training and test data occur from the same agency. However, a more practical scenario for focus location identification is to study a setting where a biased training data is available. We generate a training bias by selecting training data only contain articles from one agency, while the test data contains data from another agency. We utilize the KMM method to obtain instance weight of each training data and train a SVM classifier on the weighted training data. We denote this by $\text{Profile}_s^{\text{KMM}}$. For comparison, we also train another SVM, but without the instance weight. We denote this by $\text{Profile}_s^{\text{SVM}}$. We then evaluate these classifiers on the same test dataset.

1) Focus Location Extraction Results: We now present the results of Profile where the training and test data occur from the same agency.

On average, there are more than five different location names per news report. Note that our goal is to determine the primary focus location while the rest are non-focus locations. The experiment results for assigning focus and non-focus location labels based on the sentence embedding approach are presented in Table IV. In this table, we show the effectiveness of this approach rather than using the mean of word embedding for a sentence. Here, $\gamma = 7$ is mentioned for Profile_w and Profile_s to indicate that only the first γ sentences where considered. Otherwise, we consider all sentences. The table shows that Profile_s with sentence filtering outperforms all other embedding methods on both datasets. This shows that the weighted mean of word vectors in Profile_s is better than a simple mean of word vectors in Profile_w. Moreover, using all sentences in an article to extract primary focus location seem to add noise to the data, reducing the classifiers performance.

Method	Accuracy (%)
Profile_s	69.47
Profile_s ($\gamma = 7$)	**71.27**
Cliff-Clavin	63.75
Modified Stanford-CoreNLP	60.83
Modified Mordecai	49.96

Table V: Primary focus location accuracy comparison between different methods (Atrocity Event Data).

Method	Accuracy (%)
Profile_s	54.27
Profile_s ($\gamma = 7$)	**64.21**
Cliff-Clavin	53.65
Modified Stanford-CoreNLP	36.25
Modified Mordecai	22.97

Table VI: Primary focus location accuracy comparison between different methods (NYT).

Next, we compare the classification performance of Profile_s with other existing location estimation approaches, including Cliff-Clavin, Mordecai, and Modified Stanford-CoreNLP. The results are presented in Table V for the Atrocity data set, and Table VI for the NYT data set. In both these tables, Cliff-Clavin has better accuracy than modified Mordecai and Stanford CoreNLP since it is able to extract the focus country and exclude place names which are not in the focus country. Although Mordecai can also extract focus country, it does not identify focus location. While it uses a different named entity recognizer than Stanford CoreNLP, it also discards place names that cannot be found in the gazeteer. Therefore, the modified Mordecai approach has the lowest performance. The accuracy of Cliff-Clavin with 53.65% is marginally equivalent to the accuracy of Profile_s (i.e., 54.27%) when considering all of the news sentences. While Profile_s performance significantly better when using the first 7 sentences. This may be due to focus location usually comes in the first few sentences, and we exclude the rest of the sentences from training and test phase.

All the proposed approaches work better than the existing

methods, since we utilize Word2vec and sentence embedding model which encoded word semantics and relationships between words in a sentence. However, Stanford-CoreNLP and Mordecai are not able to extract focus location at the locality level. In addition, Cliff-Clavin can extract locations at a more course-level based on the dictionary, and it uses the frequency-based approach to identify the focus locations. As a result, $\mathrm{Profile}_s$ outperforms the other methods with 71.27% for Atrocity and 64.21% for NYT.

2) Bias Correction Results: Here, we assume the training and test data are from two different publishers. However, both of them are related to atrocity news. Figure 4 presents the performance of the $\mathrm{Profile}_s^{\mathrm{KMM}}$ model for focus location extraction with Atrocity Event data as the training set and NYT as the test set. Similarly, we also consider NYT as the training set and Atrocity Event Data as the test set. The result is shown in Figure 5 with different sets of randomly selected training data size, following [7].

The main conclusions from these two figures are as follows.

- We consistently achieve the best adaptation performance for a training size of 200 and 500 samples from both experiments on $\mathrm{Profile}_s^{\mathrm{KMM}}$.
- When comparing $\mathrm{Profile}_s^{\mathrm{KMM}}$ and $\mathrm{Profile}_s^{\mathrm{SVM}}$ for 700 and 1000 samples, we see that $\mathrm{Profile}_s^{\mathrm{KMM}}$ performs comparably with the baseline systems, specially for accuracy and precision. However, $\mathrm{Profile}_s^{\mathrm{KMM}}$ method achieves considerably better recall and F1-measure.
- Overall, the $\mathrm{Profile}_s^{\mathrm{KMM}}$ method achieve higher performance than $\mathrm{Profile}_s^{\mathrm{SVM}}$ in both of the tables.
- Finally, selecting less training instances will provide more bias for these two domains. As a result, $\mathrm{Profile}_s^{\mathrm{KMM}}$ significantly outperforms the baseline method when the bias is more.

V. Conclusion and Future Work

We have presented and developed a focus location extraction method executable on unstructured text-based news reports. In this method, we extract all of the possible locations with a named entity recognition tool, and propose a semantic approach to find the focus location. First, we extract the features by using sentence embedding algorithm. In this algorithm, we utilized the word2vec model which encoded the meaning of words and their relationship semantically into a vector. Then we trained a SVM classifier to detect the sentences that contain Focus or non-focus locations. Finally, we used the most we evaluated the proposed method on a Atrocity news event dataset and a subset of New York Times corpus.

Since the training and test domain, specially in our problem, are not always the same, we also applied the domain adaptation technique and conducted experiments on two different domains. The experimental results demonstrate the effectiveness of KMM for focus location extraction when there is bias between different domains.

Our key contributions in this work are extracting the exact focus location at the locality level where an event happened. The proposed approach is independent of geographical dictionary and works based on the semantic relationship among the words in the sentences. The performance of our method exceeds the other approaches considerably. Furthermore, we proposed using a bias correction method to prevent performance loss when the training and test domains are dissimilar.

Our future work includes: (i) To extract multiple focus locations in news reports in which more than one event is reported and evaluating the proposed approach on different datasets; (ii) To find focus location from news reports in other languages; (iii) To use different bias correction algorithms on different domains; (iv) To apply proposed method of focus location extraction and domain adaptation on a stream of event data in a large scale.

Acknowledgment

This material is based upon work supported by National Science Foundation (NSF) award number SBE-SMA-1539302, DMS-1737978, AFOSR award number FA9550-14-1-0173, and IBM faculty award (Research). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF or IBM.

References

[1] C. D'Ignazio, R. Bhargava, E. Zuckerman, and L. Beck, "Cliff-Clavin: Determining geographic focus for news," *NewsKDD: Data Science for News Publishing, at KDD 2014*, 2014.

[2] P. Schrodt and J. Ulfelder, "Political instability task force worldwide atrocities dataset," *Lawrence, KS: Univ. Kansas, updated*, vol. 8, 2009. [Online]. Available: http://eventdata.parusanalytics.com/data.dir/atrocities.html/

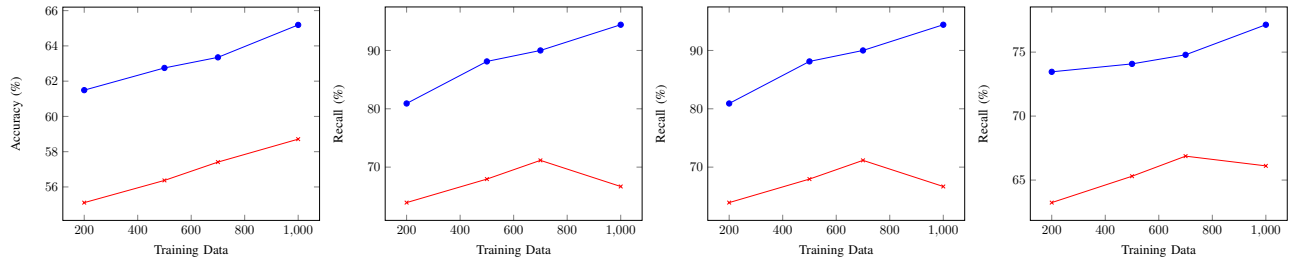
[3] mordecai, "[online]," *URL: Available: https://github.com/openeventdata/mordecai*.

[4] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The Stanford CoreNLP Natural Language Processing Toolkit." in *ACL (System Demonstrations)*, 2014, pp. 55–60.

[5] S. Arora, Y. Liang, and T. Ma, "A simple but tough-to-beat baseline for sentence embeddings," in *International Conference on Learning Representations. To Appear*, 2017.

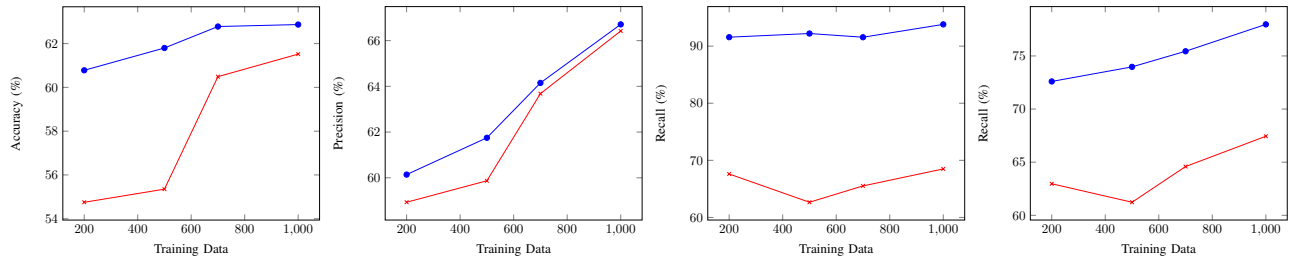
[6] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[7] J. Huang, A. Gretton, K. M. Borgwardt, B. Schölkopf, and A. J. Smola, "Correcting sample selection bias by unlabeled data," in *Advances in neural information processing systems*, 2006, pp. 601–608.



(a) Accuracy (b) Precision (c) Recall (d) F1

Figure 4: Performance of bias-corrected classifier $\text{Profile}_s^{\text{KMM}}$ with Atrocity dataset as training and NYT dataset as test, compared to a biased classifier $\text{Profile}_s^{\text{SVM}}$.



(a) Accuracy (b) Precision (c) Recall (d) F1

Figure 5: Performance of bias-corrected classifier $\text{Profile}_s^{\text{KMM}}$ with NYT dataset as training and Atrocity dataset as test, compared to the biased classifier $\text{Profile}_s^{\text{SVM}}$.

[8] J. R. Finkel, T. Grenager, and C. Manning, "Incorporating non-local information into information extraction systems by gibbs sampling," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2005, pp. 363–370.

[9] A. Ritter, S. Clark, O. Etzioni *et al.*, "Named entity recognition in tweets: an experimental study," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011, pp. 1524–1534.

[10] M. Gritta, M. T. Pilehvar, N. Limsopatham, and N. Collier, "What's missing in geographical parsing?" *Language Resources and Evaluation*, pp. 1–21, 2017.

[11] M. J. Silva, B. Martins, M. Chaves, A. P. Afonso, and N. Cardoso, "Adding geographic scopes to web resources," *Computers, Environment and Urban Systems*, vol. 30, no. 4, pp. 378–399, 2006.

[12] E. Amitay, N. Har'El, R. Sivan, and A. Soffer, "Web-a-where: geotagging web content," in *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2004, pp. 273–280.

[13] B. E. Teitler, M. D. Lieberman, D. Panozzo, J. Sankaranarayanan, H. Samet, and J. Sperling, "Newsstand: A new view on news," in *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*. ACM, 2008, p. 18.

[14] geoparser, "[online]," *URL: Available: https://geoparser.io/*.

[15] B. Alex, K. Byrne, C. Grover, and R. Tobin, "Adapting the Edinburgh geoparser for historical georeferencing," *International Journal of Humanities and Arts Computing*, vol. 9, no. 1, pp. 15–35, 2015.

[16] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine learning*, vol. 79, no. 1-2, pp. 151–175, 2010.

[17] S. J. Pan and Q. Yang, "A survey on transfer learning," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, no. 10, pp. 1345–1359, 2010.

[18] M. Sugiyama, S. Nakajima, H. Kashima, P. V. Buenau, and M. Kawanabe, "Direct importance estimation with model selection and its application to covariate shift adaptation," in *Advances in neural information processing systems*, 2008, pp. 1433–1440.

[19] T. Kanamori, S. Hido, and M. Sugiyama, "A least-squares approach to direct importance estimation," *The Journal of Machine Learning Research*, vol. 10, pp. 1391–1445, 2009.

[20] Y.-l. Yu and C. Szepesvári, "Analysis of kernel mean matching under covariate shift," in *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, 2012, pp. 607–614.