Identifying Domains and Concepts in Short Texts via Partial Taxonomy and Unlabeled Data

Yihong Zhang[1], Claudia Szabo[2], Quan Z. Sheng[3], Wei Emma Zhang[2], and Yongrui Qin[4]

[1] School of Computer Science and Engineering, Nanyang Technological University, Singapore
[2] School of Computer Science, The University of Adelaide, Australia
[3] Department of Computing, Macquarie University, Australia
[4] School of Computing and Engineering, University of Huddersfield, United Kingdom
yihong.zhang@ntu.edu.sg, claudia.szabo@adelaide.edu.au, michael.sheng@mq.edu.au, wei.zhang01@adelaide.edu.au, y.qin2@hud.ac.uk

Abstract. Accurate and real-time identification of domains and concepts discussed in microblogging texts is crucial for many important applications such as earthquake monitoring, influenza surveillance and disaster management. Existing techniques such as machine learning and keyword generation are application specific and require significant amount of training in order to achieve high accuracy. In this paper, we propose to use a multiple domain taxonomy (MDT) to capture general user knowledge. We formally define the problems of domain classification and concept tagging. Using the MDT, we devise domain-independent pure frequency count methods that do not require any training data nor annotations and that are not sensitive to misspellings or shortened word forms. Our extensive experimental analysis on real Twitter data shows that both methods have significantly better identification accuracy with low runtime than existing methods for large datasets.

Keywords: text classification, concept extraction, unsupervised method, Twitter

1 Introduction

Popular microblogging services such as Twitter can generate as many as 600 million short texts in a day[5]. Similar to real world human conversations, these short texts, henceforth called *tweets*, cover all types of topics, including politics, sports, weather, product promotion, and interesting personal discoveries. Exploiting such *mixed-domain* data for the information needs of a *narrow domain* can prove extremely useful for identifying crucial information. Existing work in this field usually requires selecting small portions of data from a large, mixed-domain data body. For example, as a service such as Twitter allows public access to all its data[6], certain portions of this data have been collected for applications in narrow domains, including earthquake monitoring [15], influenza

[5] http://www.tweetstats.com/
[6] https://dev.twitter.com/streaming/firehose

surveillance [2], election result prediction [18, 19], ideal point estimation [1], and rumor detection [5, 9].

In domain applications such as the above, the required data represents an extremely small portion of the collected datastream. For example, in a work attempting to capture disaster and crime events from tweets, the authors find that only 0.05% data in all collected data is related to the application [7]. Thus the first step in existing approaches is to filter the required data from mixed-domain, unclassified data. In [15], earthquake-related tweets are classified. In [19] tweets related to two candidates are collected. In [9], only tweets related to the bombing incident are selected. The techniques for such filtering range from machine learning based approaches [15, 13], to keyword generation [11] and clustering [20]. However, most of the filtering solutions are designed specifically for the corresponding application, and are not suitable for other domains or applications. In this paper, we focus on providing an information extraction solution that can be tailored to different specific applications based on existing domain knowledge.

Our approach relies on the insight that a narrow domain information consumer has some initial but not complete knowledge of the data, including knowledge about the key elements or topics within the domain, which is quite often the case when a domain expert in an organization wants to build an information system based on text data. This knowledge often can be translated into a taxonomy. For example, in a previous work [21], short text messages containing the keyword "shooting" are collected for detecting shooting crimes, where distinctions must be made about the meaning of "shooting", such as in "shooting photo", "shooting gun", or "shooting basketball". Users may note that "photo" is a "imaging product" and "gun" is a "weapon". They may also note the domain background, that "gun" is used in a "crime", while "ball" is used in a "game". We can construct a Multiple Domain Taxonomy (MDT) that contains these two kinds of relationships, namely, is_a, and in_a, to represent user knowledge. We show that using the concepts and relationships defined in a partially constructed MDT, we can effectively provide functions such as message domain classification and key concept recognition.

Given the MDT, we use a pure frequency approach on unlabeled data to identify the domain and concepts in the short text, as we describe in detail in the Section 3. There are several advantages using this approach. First, a pure frequency approach that does not involve grammar-based NLP (Natural Language Processing) techniques is language independent, and suitable for processing informal microblog messages. Unlike formal texts, microblog messages are filled with common misspellings and word shortening that cannot be found in a dictionary, but can be captured by frequency-based analysis with large data. Second, it is an unsupervised approach that does not require annotating data. As Twitter allows free access to one percent of its data traffic, one can easily collect millions of tweets in a day, very few of which, however, can be manually annotated. Our approach takes advantage of the large number of unlabeled data and effectively improves the identification accuracy. Finally, our approach does not require an external knowledge source. Since existing knowledge sources such as Wikipedia[7] only provide information for more common concepts, the use of

[7] https://en.wikipedia.org/

external knowledge sources generally limits the applicability of the method. Instead, our approach considers the unlabeled data as the context of the key terms and provides similar accuracy improvement effect. To summarize, we make the following contributions:

- We formally define the problem of domain classification and concept tagging given an existing taxonomy called MDT. We propose MDT as a new type of taxonomy based on the reality of narrow domain information consumption from mixed-domain data.
- We propose an unsupervised, pure-frequency approach for solving identification of domain and concepts in short texts. Our approach does not require annotation of training data and captures common misspellings and word shortenings, thus is suitable for processing informal social media messages. Our approach is also a general solution that is applicable in any narrow domain, and except for the partial MDT that requires some initial knowledge of data to construct, our approach does not need any external input.
- We test our approach extensively using real Twitter data. Our results show that the proposed domain classification method achieves much higher accuracy than existing classification methods, with up to 52% precision increase; our concept tagging method similarly achieved relatively high accuracy.

2 Related Work

Given the emerging popularity of social media, short text classification has been widely studied. Sriram et al. [16] propose a classification method to identify pre-defined message categories, such as news, opinions, and deals. Targeting such categories, their method is a supervised learning approach based on text features such as opinion words, time-event phrases, and the use of dollar sign. Li et al. [7] propose a classification method to find the Crime and Disaster Events (CDE). They use a supervised classifier that incorporates features that include hashtag, URL, and CDE-specific features such as time mention. They found that including CDE features provides about 80% accuracy versus 60% accuracy without them. These works are proposing classification solutions with presumed target domains are often considering specific domain characteristics. However, as we will show in our experiments, such solutions are usually not applicable with a different target domain. Olteanu et al. [11] propose a method for filtering relevant information based on keywords, and claim that the method can be applied to any domain. Their method generates discriminative keywords based on labeled data, and the discriminative strength is measured using PMI and frequency. However, their experiments show poor performance, with the proposed method providing almost no accuracy improvement over simple keyword filtering.

Some research exploits the message categories inherently associated with the messages. Ritter et al. [14] propose a method to automatically generate message types in addition to message classification. Based on the event messages and related phrases, the event type of each message is determined based on the distribution of name entity and time. The event messages and related phrases, however, are initially classified under a broad “event” label, which is first extracted using a supervised method based on signal words such as “announcement” and “new”, and thus may not be applicable depending on the application domain.

Lucia et al. [8] propose an unsupervised message classification method based on expanding lexical meanings using external knowledge sources. They automatically generate message categories based on existing type definitions provided by knowledge sources such as YAGO[8]. Most works that automatically generate message categories, however, tend to result only in general categories such as sports, politics, and religion, and are insufficient for more specific classification needs in a particular domain.

Name and entity recognition (NER) has been widely studied in computational linguistics, and well known solutions have been developed, such as StanfordNER[9] and OpenNLP[10]. Traditional NER solutions, however, focus only on pre-defined term categories, such as person, organization, and location [12, 3]. Recently, some solutions are proposed to tag names and entities without predefined categories. Tuan et al. [17] propose a method to find the taxonomy relations between unlabeled terms in data. In addition to string inclusion method and lexical-based rules, their method calculates subsumptions of contexts between terms, which rely on existing tools to extract *(Object, Verb, Subject)* triples. Their method achieves high precision recognizing taxonomy relations in formal texts such as journal papers and government reports. However, it is unlikely their method can be applied to informal texts, since there is no existing tool to effectively extract structures from such texts. Topics extracted from topic models can also be regarded as concepts for a short text. Li et al. [6] propose a topic model, GPU-DMM, for extracting topics from short texts. The method enriches the topic model with learned word embeddings. Semantically related words under the same topic are promoted during the sampling process by using a GPU model. However, this method highly depends on the word embeddings, which requires long time to learn and may not provide the specified categories. The work by Han et al. [4] has a similar aim to our work. They propose a frequency-based approach to link name mentions in texts to a concept in a knowledge graph, based on local compatibility and evidence propagation over the graph. Their method, however, relies on a pre-defined knowledge graph that has articles associated with each entity and thus is difficult to tailor to a specific classification task in a user-defined domain. Our proposed method, on the other hand, can work on user-defined domains and only requires a handful of domain concepts.

3 Domain Classification and Concept Tagging

We define the Multiple Domain Taxonomy (MDT) as a taxonomy with two types of relationships[11], namely, *domain association* and *taxonomy association*, denoted as *in_a* and *is_a*. *Domain associations* define the domain to which a concept belongs. *Taxonomy associations* define taxonomical hierarchies between concepts. One such MDT is shown in Figure 1. In this example, the domains are `crime` and `imaging activity`, which could both present in a text dataset

[8] http://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/

[9] http://nlp.stanford.edu/software/CRF-NER.shtml

[10] http://opennlp.apache.org/

[11] We refer to the MDT as a taxonomy due to the simple nature of the relationships defined.

regarding a *shooting*. The concepts of suspect and victim are defined as "*in a*" crime, and camera "*is a*" tool "*in a*" imaging activity.

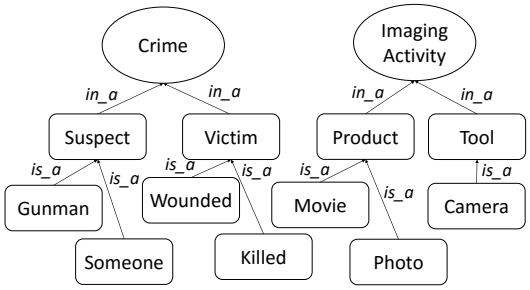

Fig. 1. An Example Multiple Domain Taxonomy

We define a multiple domain taxonomy as $MDT = \{D, V, I, S\}$, where D is the set of domains, V is the taxonomy vocabulary, and each $c \in V$ is a concept. $I = V \mapsto D$ is the mapping of in_a relationship between concepts and domains, and $S = V \mapsto V$ is the mapping of is_a relationship that describes the hierarchy of concepts. Here we consider if $\{c_1 \mapsto d\} \in I$, and $\{c_2 \mapsto c_1\} \in S$, then $\{c_2 \mapsto d\} \in I$, in other words, if a parent concept belongs to a domain, all its children concepts also belong to the same domain. In this way we do not need to explicitly define in_a relationship for lower level concepts.

3.1 Problem Statement

We show that using a partial MDT constructed with some initial knowledge of that data, we can solve the problem of message domain classification and concept tagging. The problem of domain classification looks at determining the domain for a message given a number of known domains. The problem of concept tagging looks at tagging unknown terms in a message with a concept label. An example of a tagged message would look like: "`I took a photo[IMAGING:PRODUCT] of my girlfriend[IMAGING:TARGET] with my new camera[IMAGING:TOOL]`". We note that the text transformation is straightforward once we identify the compatible taxonomy concept for the term. We formally define the two problems as the following:

Problem 1 (Domain Classification). Given a number of possible domains $D = \{d_1, ..., d_l\}$, and the message m consisting of terms $\{t_1, ..., t_k\}$, find the domain association of m, such that $\{m \mapsto d\}$ for some $d \in D$.

Problem 2 (Concept Tagging). Given a number of concepts V, and a number of terms in a message m, $T_m = \{t_1, ..., t_k\}$, find a taxonomy association for each t such that $\{t \mapsto c\}$, for some $c \in V$.

For solving the problems, we assume a $MDT = \{D, V, I, S\}$ has been constructed, such that D contains all known domains, and V contains an incomplete list of concepts that are mapped to D with I.

3.2 Message Domain Classification

To classify the domain of a message, we compare the semantic relatedness between message terms and the concepts in each domain. After aggregating the relatedness for all terms in each domain, we can determine which domain is more semantically related to the message.

To calculate the semantic relatedness between a term and a concept, we use a method proposed by Milne et al. [10], which utilizes the presence of the term and the concept in the unlabeled data, and calculates the semantic relatedness score (SRS) as following:

$$SRS(t,c) = 1 - \frac{log(max(|T|,|C|)) - log(|T \bigcap C|)}{log(|W|) - log(min(|T|,|C|))} \tag{1}$$

where t and c are the term and the concept, T and C are the sets of all messages that contain t and c, respectively, and W is the entire dataset.

We use the highest SRS obtained when matching the term with different domain concepts as the domain score for the term. After retrieving the SRS for each term in a domain, we calculate a message score for the domain (DS):

$$DS(m,d) = \sum_{i=1}^{k} max(SRS(t_k, c_j), \forall\{c_j \mapsto d\} \in I) \tag{2}$$

where the message m consists of terms $\{t_1, ..., t_k\}$.

We calculate a domain score for each domain. Then the predicted domain for m is the domain that provides the highest domain score, $\arg\max_i DS(m, d_i)$.

3.3 Concept Tagging

We approach the concept tagging problem by finding the *compatible* concept in the taxonomy for a term. If a term is compatible with a concept, then it can inherit its taxonomy associations. For example, if we find "film" is compatible with "movie", and "movie" is defined as a product in the taxonomy, then we can consider "film" is also a product. To calculate the concept compatibility, we take into account the message contexts, which is formed from the words surrounding the term and the concept. We argue that if a term is in the same domain as the concept, and the context they appear in are similar, then it is very likely they are compatible.

The context of a term is usually represented as a number of words neighboring the keyword. Traditionally, the position of context words is ignored, and the context words are considered interchangeable. However, we found that the position of context words contains crucial information and should not be overlooked. For example, suppose we have two message, "He took a new photo of the house", and "the house of cards took a new view on US politics". In this example, if we ignore the position, the two terms "photo" and "cards" have the same context, but they are certainly semantically incompatible. Based on this insight, in our solution we take into account the position of context words.

To calculate the context similarity between a term and a concept, we first set a context width parameter q, which defines how many neighboring words

will be considered as the context. From a number of unlabeled messages that contains the term, we extract a set of words at each position between $p-q$ and $p+q$, where p is the position of the term in the message. A total of $2q$ sets will be extracted, denoted as $Q_t^1, ..., Q_t^{2q}$. Similarly we extract the context word sets for the compared concept, $Q_c^1, ..., Q_c^{2q}$. The context similarity is thus calculated based on the similarity of context words in the same position:

$$contextSimilarity(t, c) = \frac{\sum_{i=1}^{2q} sim(Q_t^i, Q_c^i)}{2q} \tag{3}$$

where $sim(Q_1, Q_2)$ is a similarity function that compares two cluster of words.

From existing work, we choose a similarity function proposed by Unankard et al. [20], which is based on term frequency and cosine similarity:

$$sim(Q_1, Q_2) = \frac{\sum_i tf(Q_1, t_i) \times tf(Q_2, t_i)}{\sqrt{\sum_i tf(Q_1, t_i)^2} \times \sqrt{\sum_i tf(Q_2, t_i)^2}} \tag{4}$$

where $t_i \in T$ is all the terms in $Q_1 \bigcup Q_2$, and $tf(Q, t)$ is the term frequency of term t in set Q.

To tag a term t in a message m, first we determine the domain for m using the method described above. Then we obtain all concepts that belong to the domain, $c_d \in V$ that satisfies $\{c_d \mapsto d\} \in I$. We then calculate the context similarity between the term and each concept, and find the concept that produces the highest context similarity, c_{max}. Finally we consider t and c_{max} compatible, and assign $\{t \mapsto c_p\}$ for any $\{c_{max} \mapsto c_p\}$, in other words, allowing t inherit the taxonomy association that c_{max} has.

We need to note that the identified concepts can be added into the MDT, based on the identified domain and compatible concepts, and thus the MDT can be iteratively improved. As more data being processed, and more concepts added to the ontology, we expect a better recognition performance of our system with the improved MDT. In this work, however, we focus on the first iteration of this process. We will explore iterative MDT improvement with identified concepts in future works.

3.4 Improving Computational Efficiency

It is computationally expensive to collect context and calculate similarity for every concept-term pair in large unlabeled datasets. For example, for 100,000 unlabeled messages and $q = 3$, a total of 600,000 words will be compared for each pair. To improve efficiency, we compute some term frequency information at the start of the system and store it in a memory heap to quickly estimate the significance of context similarity between a term and a concept, thus eliminating most contextual comparisons between insignificant pairs.

We call our runtime reduction technique *reverse contextualizing* (RC). First we compute the significance between a concept c and a context word w in position i. We collect the context of c in position i as Q_c^i, the significance of a context word w is calculated as:

$$sig(c, w, i) = \frac{tf(Q_c^i, w)}{|Q_c^i|} \tag{5}$$

This score shows the percentage of a context word in all words appearing in the concept's context at the given position. Then for each context w, we also collect its contexts, with the reversed position of $2q-i$, as Q_w^{2q-i}. For each term t of this reversed context set of w, the significance is calculated as:

$$sig(t, w, i) = \frac{tf(Q_w^{2q-i}, t)}{|Q_w^{2q-i}|} \tag{6}$$

Finally we compute a significance score between concept c and term t as:

$$sig(c, t) = 100 \times \sum_{i=1}^{2q} \sum_{w \in |Q_c^i|} sig(c, w, i) + sig(t, w, i) \tag{7}$$

As an example, suppose $q = 3$ and $i = q+1$. Then for concept *police*, we calculate significance of contextual word *shooting* in the position next to the concept as $sig(\text{“}police\text{”}, \text{“}shooting\text{”}, q+1)$, based on the frequency of the phrase “police shooting”. Then for word *shooting*, we calculate the significance of its context word *kids* in the position previous to the word as $sig(\text{“}kids\text{”}, \text{“}shooting\text{”}, q+1)$, based on the frequency of phrase “kids shooting”. Finally based on two calculation results we obtain significance score between *police* and *kids*.

We compute this score for each pair of concept and term appearing in the same position in the data with respect to context words, and store it in memory. We also set a significance threshold τ. When tagging a term in a message, we first retrieve the significance score between the term and the concept, $sig(c, t)$, and only when $sig(c, t) > \tau$ we proceed to calculate the actual context similarity.

4 Experimental Analysis

We have presented our domain and concept identifying method as an effective unsupervised method. We expect our method to achieve better accuracy than current supervised and unsupervised methods, while keep low computational cost. We conduct experiments on real Twitter data to validate our approach. First we test the accuracy of our domain classification method. Then we test the accuracy of our concept tagging method. Finally we study the runtime of our approach, and provide insights into the impact of different training data size and pre-computation on computational costs.

4.1 Datasets

Our experiments are conducted on two sets of real Twitter data. The first dataset, called the *shooting* dataset, is collected using the Twitter Filter API[12] during September and October, 2014. The dataset has about 2 million tweets containing the keyword *shooting*. After removing retweets, we obtain a set of 284,343 tweets. We examine the data and discover that the tweets are mainly related to four domains, namely, *crime*, *imaging*, *game*, and *metaphor*. After deciding the domains, we label a number of tweets according to their domains. The labeled data contains 1,083 tweets.

[12] https://dev.twitter.com/streaming/reference/post/statuses/filter

The second dataset is called the *crisis* dataset and is a publicly available dataset[13] introduced by Olteanu et al. [11]. It contains sets of tweets related to 26 natural disasters and other crisis events and labeled and unlabeled tweets. There are two types of labels, based on whether the tweet is related and informative, and based on the source of the tweet, respectively. We use only related tweets. Combining tweets for all 26 events, we obtain 201,078 unlabeled tweets, and 3,646 labeled tweets. The labeled tweets contain five categories, namely, *eyewitness*, *business*, *government*, *media*, and *ngo*.

For each dataset we manually construct an MDT shown in Table 1. Both MDTs have a flat structure, with the first level as domains, and the second and third levels as concepts. Between domains and concepts, *in_a* relationships are defined. Between second and third levels of concepts, *is_a* relationships are defined. We have not spent more than two hours per MDT. For the crisis dataset, the five domains are taken from the five categories of labeled data.

Table 1. MDT used in the experiments

`Shooting dataset`		
crime	actor	police, officer, cops, somebody, someone, gunman
	victim	wounded, killed
	weapon	gun, handgun
	location	office, street, house, crib, backyard, block
imaging	product	movie, film, photo, video, commercial, ad
	maker	cameraman, director, assistant, production, crew
	target	wedding, party, girlfriend
	location	studio, set, indoor, outdoor
	tool	camera, script, iphone, canon
game	type	games, range, ball, hoops, dice, ranch, duck, clay, match
	result	won, wins, lost, losses, leads, point, foul
	participant	player, team, shooter, guard, opponent
metaphor	object	star, pain, slugs
	target	foot, moon, face, wall, myself
	environment	sky, space, ecstasy, fantasy
`Crisis dataset`		
eyewitness	observation	windy, raining, baha, ulan, habagat
	reaction	my, friend, everyone, scary, hope, think
	location	house, backyard, outside
business	person	customers, ceo, employees
	unit	company, stores, plant, site, railway, google
	operation	sales, schedule, license
govenment	sector	public, federal, fdny, cpa, rfs, fbi, ntsb, mta, gov
	service	warning, hotlines, forms, school
	person	governor, premier, police, commissioner
media	type	blog, news, article, journal, press, tv, video, paper
	agent	bbc, reuters, cnn, fox, yahoo, times
	report	says, reports, kills, victims, accused, missing, hits, reported, coverage, source, update, story
ngo	organization	communities, centre, redcross, members
	activity	donating, fundraising, volunteering, charities
	support	donations, goods, money, aide

4.2 Results for Domain Classification

In the first set of experiments, we test the domain classification accuracy for our approach. We first focus on the first domain for the two datasets, namely, *crime*

[13] http://crisislex.org/

in the shooting dataset, and *eyewitness* in the crisis dataset. We focus on these two domains because *crime* and *eyewitness* are more desirable information, and have been the topic in several studies [7, 22].

We compare our approach with three baselines. The first is *accept all* which considers all messages as positive. The *accept all* method would always achieve the highest recall of 1.0. The second baseline, proposed by Sriram et al. [16], is a supervised method based on eight features and the Naive Bayes model. The eight features include author name, use of slang, time phrase, opinionated words, and word emphasis, presences of currency signs, percentage signs, mention sign at the beginning and the middle of the message. The evaluation is based on the five-fold cross validation. The Sriram classifier is shown to be effective in classifying tweets into categories such as news, opinions, deals and events, but has not been tested in other applications. The third baseline (PA) is from our previous work [22]. It is an unsupervised approach that incorporates lexical analysis and user profiling. This method is shown to be effective for filtering personal observations from tweet messages.

The classification accuracy of the first domain in two datasets achieved by three baselines and our MDT-based approach is shown in Table 2. As can be seen from the results, our approach achieves extremely high precision comparing to the baselines. For classifying *crime* domain, it achieves 0.92 precision, which is a 52% increase from the baseline methods, as well as 0.78 f-value, a 27% increase from the baseline methods. For classifying *eyewitness*, it also achieves a high precision of 0.73, a 9% increase from the baseline method, and 6% increase in f-value. The PA method is designed to distinguish observation messages according to their source, and thus it achieves a low accuracy classifying *crime* as it includes messages from various sources; for *eyewitness* ,it achieves the highest accuracy among baseline methods. Our MDT-based method, nevertheless, surpasses the PA method both in precision and recall for classifying *eyewitness*.

Table 2. Classification accuracy of the first domain

	Accept All	Sriram	PA	MDT
`Shooting dataset`				
precision	0.30	0.40	0.31	**0.92**
recall	**1**	0.71	0.49	0.68
f-value	0.46	0.51	0.38	**0.78**
`Crisis Dataset`				
precision	0.14	0.32	0.64	**0.73**
recall	**1**	0.52	0.50	0.54
f-value	0.24	0.40	0.56	**0.62**

We also look at other domains. Table 3 shows the classification accuracy across four domains for the shooting dataset. As can be seen from the result, classification on other domain also achieves high accuracy as the crime domain, indicated by similar f-values. However, the *crime* domain do provide the highest precision, mainly due to that it is a narrower domain that can be better identified with a simple taxonomy.

Table 3. Classification accuracy for the shooting dataset

	crime	imaging	game	metaphor
precision	0.92	0.82	0.67	0.67
recall	0.68	0.68	0.78	0.91
f-value	0.78	0.75	0.72	0.77

4.3 Results for Concept Tagging

In the second set of experiments, we test the accuracy of our concept tagging approach. We conduct two experiments. In the first experiment, which we call *take-out-one* experiment, the leaf level concepts in the MDT are taken out one-by-one and put back to the MDT using our approach. For example, for the shooting MDT, we first take out the *police* concept, and then use the proposed tagging method to match it with the MDT, now without the *police* concept. This process is run for every concept in the MDT. For the shooting MDT, 71 concepts are tested. For the crisis MDT, 80 concepts are tested. The proportion of correctly tagged concept with respect to different training data sizes is shown in Table 4. From the results we can see that the take-out-one experiment reaches a very high precision. With only 15,000 training data, we have over 0.95 precision for the shooting MDT, and over 0.92 precision for the crisis MDT. According to this result, we can confidently tag a concept with a MDT even with a small number of training data, if it is known that the concept must be compatible with the MDT.

Table 4. Precision in take-out-one experiment

training size	5k	10k	15k	20k	25k	30k
shooting	0.915	0.943	0.955	0.955	0.955	0.985
crisis	0.850	0.875	0.925	0.925	0.962	0.987

In the next experiment, we run concept tagging on the raw data. We take 10,000 tweets from the shooting dataset and 3,000 tweets from the crisis dataset, and employ our concept tagging method. We use 30k training data, which should provide optimal effectiveness based on to the previous experiment. The detected taxonomy and the context similarity score are recorded for each tagged term, and thus a large number of tagged terms are generated. Table 5 shows the terms with the highest context similarity score for the second-level concepts.

We can identify some errors in the above tagging, such as identifying *baby* instead of *baby milk product commercial* as the shooting target, and *Queen* in *Queen Elizabeth High School* as a person. Such errors are caused by the limitation of not considering multi-word terms, which we will explore in the future. It is worth noting that word shortenings such as *ppl* are captured correctly.

To evaluate the overall accuracy, we manually check all the tagged terms with a context similarity score above 0.3. There are 296 terms and 554 terms that satisfy this requirement in the shooting and crisis test data, respectively. The tagging accuracy of these terms with respect to different context similarity

Table 5. Top terms for second-level concept

Tagged tweet	Term	Concept	Score
Shit crazy ppl[CRIME:ACTOR] shooting omg :(	ppl	actor	0.496
On set wishes to Tyneea C and Romarni C shooting for baby[IMAGING:PRODUCT] milk product commercial - enjoy girlz !	baby	product	0.381
I enjoy shooting pool[GAME:TYPE]	pool	type	0.431
Like a shooting star, I will go the distance. I will search the world[METAPHOR:TARGET]. I will face its harm and I don't care how far.	world	target	0.455
The windows are shaking at home[EYEWITNESS:LOCATION], the wind is crazy!! And it's getting worse - #GoldCoast #bigwet	home	location	0.487
Bid now on this one of a kind SIGNED canvas print of our @RedRocksOnline poster. ALL proceeds[BUSINESS:OPERATION] go to #Coflood relief:	proceeds	operation	0.512
BBC News - In pictures[MEDIA:TYPE]: Brazil nightclub fire	pictures	type	0.653
AB relief Cards are available today - if you are in Sunnyside, please go to Queen[GOVERNMENT:PERSON] Elizabeth High School! #yycflood	Queen	person	0.355
Raise funds[NGO:SUPPORT] for #Boston or West #Texas tonight if #party planning - (Between 6pm and 11pm, ET, Tuesday April 30th)	funds	support	0.694

score range is shown in Table 6. As a comparison, we also show the expected accuracy if we randomly choose a second-level concept for tagging.

Table 6. Tagging accuracy in different context similarity score range

context similarity	> 0.3	> 0.35	> 0.4	> 0.45	random
shooting	0.44	0.46	0.48	0.57	0.014
crisis	0.55	0.567	0.60	0.64	0.012

We can obtain around 50% tagging accuracy for terms that generate a context similarity score > 0.3. The low accuracy is possibly due to many terms that do not have a taxonomy relationship with the MDT but still have similar context with the concepts, such as time and location words. This problem can be overcome by, for example, adding the time-related concepts to the MDT. Nevertheless, comparing to randomly assigning tags, our approach achieves much higher tagging accuracy.

4.4 Runtime Analysis

We test the effectiveness of our runtime reduction technique (RC). We measure the runtime of concept tagging for the 1,083 shooting tweets, with different training data sizes and two τ values. The results are shown in Figure 2. All

experiments are run on a desktop computer with a 3.7GHz eight-core Intel Xeon CPU, 15.6 GB memory, and Ubuntu 16.04.

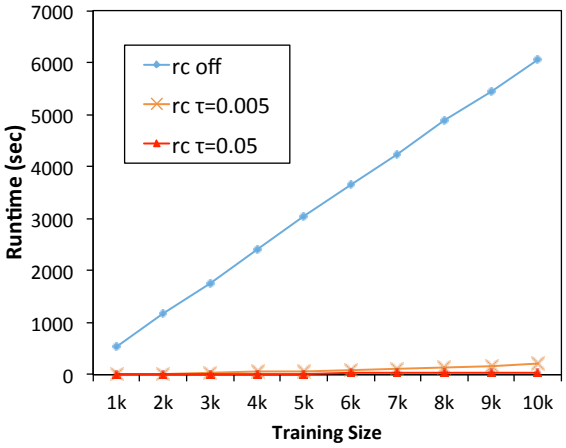

Fig. 2. Runtime with different RC options

As we can see from the figure, our RC technique effectively removes most of the computation in the otherwise computation-heavy concept tagging task. Using 5k training data, the runtime without RC is 3,045 seconds, while with RC the runtime is 72 second for $\tau = 0.005$ and 23 seconds for $\tau = 0.05$. Using 10k training data, the runtime without RC is 6,065 seconds, while with RC the runtime is 202 seconds for $\tau = 0.005$ and 49 seconds for $\tau = 0.05$. In both cases, the runtime is reduced to a few hundredth of the original runtime. Looking at the absolute values, using 10k training data, with which we have seen satisfactory tagging accuracy, the average tagging time for a single tweet is 5.98 seconds without RC, but only 0.04 seconds with RC ($\tau = 0.05$). With the improved tagging speed, our concept tagging method becomes suitable even for realtime tweet processing.

5 Discussion

One of the hurdles of deploying our approach is the construction of the MDT. It is nearly impossible to extract narrow domain information from a large, mixed-domain data without any manual input. Comparing to training data annotation in supervised approaches, though, we consider that constructing a MDT requires much less effort, and translates knowledge in an efficient manner. We can also see that the extraction accuracy varies depending on the quality of the MDT. In our experiments, the extraction accuracy for the *shooting* data is higher than the *crisis* dataset, most likely because we have more experience with the first dataset than with the latter, and thus constructed a more representative MDT for the first dataset. Adding identified concepts to the MDT can improve the system performance, but manually checking is required given the errors in concept recognition we discussed in the previous section. Based on our experiences,

adding wrong or ambiguous concept will not improve identification accuracy, but rather decrease it.

Currently our method only considers single-word concepts, but in reality many concepts are expressed in multiple words, and we will run into error if we cannot recognize them, for example, "video camera". This can be done by generating all possible bi-grams and multi-grams from data, as existing works have suggested [8].

6 Conclusion

Social media produces significantly large volume of data covering a wide range of topics, and there is an increasing need of extracting information for narrow domain applications from large, mixed-domain datasets. However, currently most applications develop classification and extraction solutions tailored to a narrow domain, and are usually unsuitable for use in other applications and domains. Developing individual solutions is expensive including efforts to develop algorithms and annotate training data for supervised solutions. We therefore focus on a general solution that can be easily tailored to narrow domain needs and does not require training data annotation and other manual involvement.

In this paper, we propose Multiple Domain Taxonomy (MDT), a representation of mixed-domain data. We show that using a partially constructed MDT, we can effectively classify and extract key concepts from short text messages. The MDT can be constructed with some initial knowledge of the data, and can be quickly tailored to narrow domain needs. Our approach is frequency-based and unsupervised. It is robust to common misspellings and word shortenings, and does not require training data annotation. The effectiveness of our approach is verified extensively using real datasets, and comparing to baseline methods such as the Sriram classifier and the PA method, our approach increased the accuracy by up to 52%. In the future, we plan to further improve the concept tagging accuracy, as well as investigating the case of multi-word concepts.

References

1. P. Barberá. Birds of the same feather tweet together: Bayesian ideal point estimation using Twitter data. *Political Analysis*, 23(1):76–91, 2015.
2. M. Dredze, M. J. Paul, S. Bergsma, and H. Tran. Carmen: A Twitter geolocation system with applications to public health. In *AAAI Workshop on Expanding the Boundaries of Health Informatics Using AI*, pages 20–24, 2013.
3. T. Finin, W. Murnane, A. Karandikar, N. Keller, J. Martineau, and M. Dredze. Annotating named entities in twitter data with crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 80–88. Association for Computational Linguistics, 2010.
4. X. Han, L. Sun, and J. Zhao. Collective entity linking in web text: a graph-based method. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 765–774. ACM, 2011.
5. S. Kwon, M. Cha, K. Jung, W. Chen, and Y. Wang. Prominent features of rumor propagation in online social media. In *Proceedings of 13th International Conference on Data Mining*, pages 1103–1108, 2013.

6. C. Li, H. Wang, Z. Zhang, A. Sun, and Z. Ma. Topic modeling for short texts with auxiliary word embeddings. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 165–174. ACM, 2016.
7. R. Li, K. H. Lei, R. Khadiwala, and K.-C. Chang. TEDAS: A Twitter-based event detection and analysis system. In *Proceedings of 28th International Conference on Data Engineering*, pages 1273–1276, 2012.
8. W. Lucia and E. Ferrari. Egocentric: Ego networks for knowledge-based short text classification. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*, pages 1079–1088. ACM, 2014.
9. J. Maddock, K. Starbird, H. Al-Hassani, D. E. Sandoval, M. Orand, and R. M. Mason. Characterizing online rumoring behavior using multi-dimensional signatures. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work & Social Computing*, pages 228–241, 2015.
10. D. Milne and I. H. Witten. Learning to link with Wikipedia. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, pages 509–518. ACM, 2008.
11. A. Olteanu, C. Castillo, F. Diaz, and S. Vieweg. CrisisLex: A lexicon for collecting and filtering microblogged communications in crises. In *In Proceedings of the 8th International AAAI Conference on Weblogs and Social Media*, pages 376–385, 2014.
12. T. Poibeau and L. Kosseim. Proper name extraction from non-journalistic texts. *Language and Computers*, 37(1):144–157, 2001.
13. A.-M. Popescu and M. Pennacchiotti. Detecting controversial events from Twitter. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 1873–1876, 2010.
14. A. Ritter, O. Etzioni, S. Clark, et al. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1104–1112. ACM, 2012.
15. T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes Twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th International World Wide Web Conference*, pages 851–860, 2010.
16. B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas. Short text classification in Twitter to improve information filtering. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 841–842, 2010.
17. L. A. Tuan, J.-j. Kim, and N. S. Kiong. Taxonomy construction using syntactic contextual evidence. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 810–819, 2014.
18. A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe. Predicting elections with Twitter: What 140 characters reveal about political sentiment. In *Proceedings of the Fourth International Conference on Weblogs and Social Media*, pages 178–185, 2010.
19. S. Unankard, X. Li, M. Sharaf, J. Zhong, and X. Li. Predicting elections from social networks based on sub-event detection and sentiment analysis. In *Proceedings of 15th International Conference on Web Information Systems Engineering, Part II*, pages 1–16. Springer, 2014.
20. S. Unankard, X. Li, and M. A. Sharaf. Emerging event detection in social networks with location sensitivity. *World Wide Web*, 18(5):1393–1417, September 2015.
21. Y. Zhang, C. Szabo, and Q. Z. Sheng. Sense and focus: Towards effective location inference and event detection on twitter. In *Proceedings of the 16th International Conference on Web Information Systems Engineering Part I*, pages 463–477, 2015.
22. Y. Zhang, C. Szabo, and Q. Z. Sheng. Improved object and event monitoring on twitter through lexical analysis and user profiling. In *Proceedings of the International Conference on Web Information System Engineering*, 2016.