Keyword expansion for understanding crisis events in Indonesian tweets

JinYeong Bak [1] **Imaduddin Amin** [2] **Jong Gun Lee** [2] **Alice Oh** [1]

Abstract

Posts of emergency crises such as forest fires or terror attacks pervade social media. In identifying and analyzing the messages related to such events, keywords play an important role. However, it is difficult to come up with the relevant keywords for many of the resource-limited languages because they lack linguistic knowledge bases such as WordNet. In this paper, we suggest a statistical iterative and interactive learning for keyword expansion (SILK). SILK is a framework that produces a set of candidate keywords based on text modeling techniques, and human experts filter out any irrelevant keywords.

1. Introduction

During an emergency crisis such as an earthquake or a forest fire, Twitter and other social media posts can reveal important details and updates about the situation. However, it can be difficult to automatically identify and understand the event-related posts, and for that reason, keywords play an important role (Imran et al., 2015; Atefeh & Khreich, 2015). For example, researchers use a pre-defined set of keywords to detect earthquake-related tweets (Sakaki et al., 2010), and others use keywords to visualize the summary of the posts (Mathioudakis & Koudas, 2010).

In resource-rich languages such as English, one can automatically build a keyword set using WordNet (Miller, 1995) or Probase (Wu et al., 2012). However, resource-limited languages such as Indonesian suffer from a lack of linguistic knowledge bases, and thus building a lexicon of keywords must rely on human experts.

In this paper, we propose and evaluate a framework for assisting human experts such that they can construct the keyword lexicon with relatively little effort. We call this framework *SILK*, short for statistical iterative learning of keywords. To build a keyword lexicon for a crisis event, SILK first gathers a set of seed keywords from the human expert and uses those to identify a set of tweets. SILK expands the keyword set by using either a probabilistic topic model or a simple tf-idf metric to identify the candidate keywords. Then, for the last step in an iteration, SILK those candidate words to the human experts who simply accept or reject each keyword based on their relevance to the event. After this last step, new keywords would be added to the initial seed keywords, and the steps would be repeated several times until the keyword set is large enough.

To test the effectiveness of SILK, we conduct an experiment with Indonesian tweets about the *haze* crisis, an air pollution that affects tens of millions of people and happens every year. We test SILK with various methods for automatically identifying keywords, and SILK with a non-parametric Bayesian topic model performs best for classification of relevant tweets.

2. Related Work

Various ways to detect events in Twitter are suggested (Atefeh & Khreich, 2015; Imran et al., 2015), using SVM with words and meta information features (Sakaki et al., 2010), bursty words during time (Abdelhaq et al., 2013), and pre-defined keywords (Marcus et al., 2011). Query expansion by pseudo-relevance feedback is also used to identify relevant tweets in Twitter (Lin et al., 2012). However, these methods are focusing on identifying tweets, not to understand the keywords itself.

Building crisis lexicons are done by crowd-sourcing and pseudo-relevance feedback (Olteanu et al., 2014), but it requires annotated tweets as training data. However, SILK do not need the training data, it requires tweets that have keywords and to tag small candidate keywords from topics.

Topic models are one way to extract the topics from corpus. LDA (Blei et al., 2003) is naive Bayesian topic model. We adopt HDSP (Kim & Oh, 2014) which adds observed label information into model. It can measure the distance between labels and topics, so we can extract the closeness of topics over keywords.

[1]KAIST, Daejeon, South Korea [2]Pulse Lab Jakarta, UN Global Pulse, United Nations, Jakarta, Indonesia. Correspondence to: JinYeong Bak <jy.bak@kaist.ac.kr>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

3. Methodology

This section describes our SILK framework.

Algorithm 1 An overview of SILK

1: **loop**
2: Get tweets by keywords
3: Run topic model
4: Extract representative words from topics
5: Make keyword candidates
6: Experts accept/reject keywords
7: Combine keywords
8: **end loop**

Algorithm 1 is an overall description of SILK. We can run several iterations to find new keywords based on the keywords from the output of the previous iteration. Each step of the algorithm is explained in detail below.

Get tweets by keywords In SILK, human experts provide a small set of relevant seed keywords, and the first step of an iteration of SILK is extracting the messages that contain those keywords. We define keyword as a combination of two words. For example, 'kabut & asap' (smog in Indonesian) is a good combination of keywords for the haze crisis, whereas each word 'kabut' or 'asap' alone is not a good keyword. So in identifying relevant tweets, we require the two words to appear in a tweet regardless of ordering.

Run topic model In steps 3, 4, and 5, we identify a set of candidate keywords to be presented to the human experts. This part of the algorithm can be done with several different algorithms, such as simply using the *tfidf* metric, or running a probabilistic topic model. Here we describe using the HDSP topic model (Kim & Oh, 2014) which puts the topics and labels (or other types of meta-information) into the same latent space and thus enables direct comparisons of the topic words and the meta-information. We use this feature of the HDSP to treat the input keywords as labels.

Extract representative words from topics In the next step, we find the representative words for the topics. We first start with the input keywords as the centroids and cluster the topics around them by the similarity between topics and the keyword centroids. Then, we extract the representative words for each topic cluster by computing the word mutual information (Manning et al., 2008) with the topic clusters. We then select the top 10% words with high mutual information among the clusters.

Make keyword candidates We then make new keyword candidates based on the representative words. To combine two words, we look at the average of the two words' joint probability given the a topic. We compute all combinations of the two words, and filter out the words with the probability lower than 0.1 of the highest probability.

Filter out keywords by human experts In the last step, we ask the human experts to accept or reject each keyword in the candidate set. In other words, they tag each of the candidate keywords based on the relevance to the specific event for which the keyword lexicon is being constructed. The accepted keywords are then combined with the input keywords, and fed as input keywords into the next iteration.

4. Experiments

This section describes the experiments and results of SILK as well as the baselines for classification of an event in Indonesian Twitter and efficiency of human tagging.

4.1. Setup

SILK is developed for languages with few linguistic resources. So, we choose Indonesian tweets in Twitter from March to June 2014 as data to show the performance of the methodology. We collect the tweets with geo-tagged location information, and we choose the tweets about haze, which is air pollution caused mainly by forest fires. It affects tens of millions of people and happens every year.

To perform and evaluate classification of the tweets on whether they are relevant to the haze crisis, we make a ground truth dataset. We sample tweets randomly from Sumatra Indonesia where the haze occurs every year, and ask for annotation by three native Indonesian speakers who are familiar with the haze and with Twitter. We select tweets that are dominantly annotated as yes or no. After annotating the tweets, we randomly choose 1104 positive and 1104 negative tweets for relevance to the haze.

To start SILK, we get four seed keywords from the human experts. 'kabut & asap' (haze smog), 'titik & panas' (hotspots), 'sumber & api' (sources of ignition) and 'kebakaran & hutan' (forest fire).

We compare SILK with the following methods for identifying haze tweets and tagging efficiency.

SEED: A simple baseline to classify as positive just the tweets that contain any of the four seed keywords. **SVM**: Support vector machine with radial basis kernel and tf-idf scored unigram word feature. It is used to detect events in Twitter (Sakaki et al., 2010). **PRF**: Pseudo Relevance Feedback based on Relevance Based Language Models (Lavrenko & Croft, 2001) **DPRF**: Dynamic Pseudo Relevance Feedback (Lin et al., 2012) which consider bursty words on certain period. **TFIDF**: SILK with bigrams that have high tf-idf scores. It has been used to extract keywords from a text corpus (Liu et al., 2008). **DPRF+**: SILK with DPRF. Experts filter out candidate keywords on each iteration. **LDA**: SILK with LDA which is a basic topic

	Iter 1	Iter 2	Iter 3	All	Rate
TFIDF	24 (55)	15 (90)	17 (82)	56 (227)	0.25
DPRF+	14 (43)	13 (24)	9 (22)	36 (89)	0.40
LDA	46 (61)	52 (98)	63 (67)	161 (226)	0.71
LFTM	25 (28)	30 (35)	45 (48)	100 (111)	0.90
SILK	51 (58)	79 (88)	44 (46)	174 (192)	**0.91**

Table 1. Efficiency of human tagging to keywords candidates over three iterations. Numbers in a cell means the number of accepted keywords and suggested keywords for each iteration of the methods. SILK shows best efficiency and produces many relevant keywords than others.

	Accuracy		Precision		Recall	
	Prev	Iter 1	*Prev*	Iter 1	*Prev*	Iter 1
SEED	0.949	-	0.972	-	0.745	-
DPRF	0.885	0.84	0.667	0.556	0.804	0.825
DPRF+	0.951	0.955	0.973	0.973	0.755	0.777
LFTM	0.962	0.978	**0.975**	**0.978**	0.819	0.907
SILKrs	-	0.974	-	0.977	-	0.887
SILK	**0.976**	**0.982**	0.966	**0.978**	**0.904**	**0.928**

Table 2. Haze related tweets classification performance using annotated data in September and October 2014. *Prev* means identifying keywords from previous time by each method. SILKrs is re-performing SILK with SEED keywords. SILK outperforms all other methods compared.

model (Blei et al., 2003). **LFTM**: SILK with LFTM which is a model that combines LDA and word2vec to generate words in a document (Nguyen et al., 2015). **SILK**: Our final methodology which uses HDSP (Kim & Oh, 2014).

We use skewed linear distribution for DPRF. We set 0.5 for the parameter between query and relevant model of DPRF, and other parameters are same as (Lin et al., 2012).

LFTM requires pre-trained word2vec (Mikolov & Dean, 2013) from an independent corpus. (Nguyen et al., 2015) uses pre-trained vectors from the English Google News corpus, but it would not work on our Indonesian data. Instead, we crawled 400K Indonesian news articles from 2013 to 2015 in Kompas[1], and make 200-dimensional vectors by the gensim[2].

We set 300 topics for LDA and LFTM. We assume that one tweet has a few topics, so we set the hyper-parameters of the topic proportion in a tweet and the word distribution over the topics for LDA, LFTM and HDSP to 0.01.

To get the representative words from LDA and LFTM results, we compute the distance of the word distribution over the topics by KL divergence and run hierarchical clustering (Manning et al., 2008).

SVM needs training data, so we perform ten-fold cross validation with ground-truth. We check the performance of other kernels, but SVM with radial basis performs the best.

4.2. Results

Identifying Crisis Tweets We look at the identifying haze related tweet performance of methods via annotated data. Figure 1 shows the results. SILK performs better than the other methods for accuracy as well as Recall, preserving the precision. PRF and DPRF perform well at the first iteration, but its performance is decreased since it also produces irrelevant keywords. LDA, LFTM and SILK perform better than TFIDF, and its performance increase over iterations. But, only SILK shows better performance than SVM.

[1] http://www.kompas.com/

[2] https://radimrehurek.com/gensim

Efficiency of Human Annotation SILK is interactive learning which takes human feedback to tag keyword candidates. So reducing tagging efforts is also important issue. We check the proportion of relevant new keywords over keyword candidates.

As table 1 shows, SILK produces many keyword candidates and 91% of the candidates are accepted the human filtering. TFIDF and LDA suggest similar size of candidates, but almost 75% candidates are failed in TFIDF. DPRF+ suggests small candidates and less than half keywords are accepted. It is interesting to note that LFTM produces fewer keyword candidates than others, but most of them are accepted. It gives evidence to us to use external word2vec to improve the performance.

Applying to the Future SILK is iterative method which can apply to future data. So, we use the keywords from March to June 2014 to identify haze related tweets in September and October 2014 when the haze appear again.

We also sample tweets randomly from the same location, and ask for annotation by the same annotators. After annotating the tweets, we randomly choose 197 positive and 308 negative tweets.

Table 2 shows the results. SILK performs better than the other methods for accuracy and recall. We run the one more iteration with new data. Then, SILK with previous keywords are better than with seed keywords. Keywords from previous data are useful to identify the events in the future, and more steps increase the performance.

5. Conclusion and Future Work

In this paper, we have presented SILK which is the iterative methodology for expanding keywords in Twitter with experts feedback. We annotated tweets to make a ground-truth dataset for identifying haze event. With the data, we showed that SILK performs better than other methods in classification accuracy and recall. SILK also reduce the experts efforts rather than others.

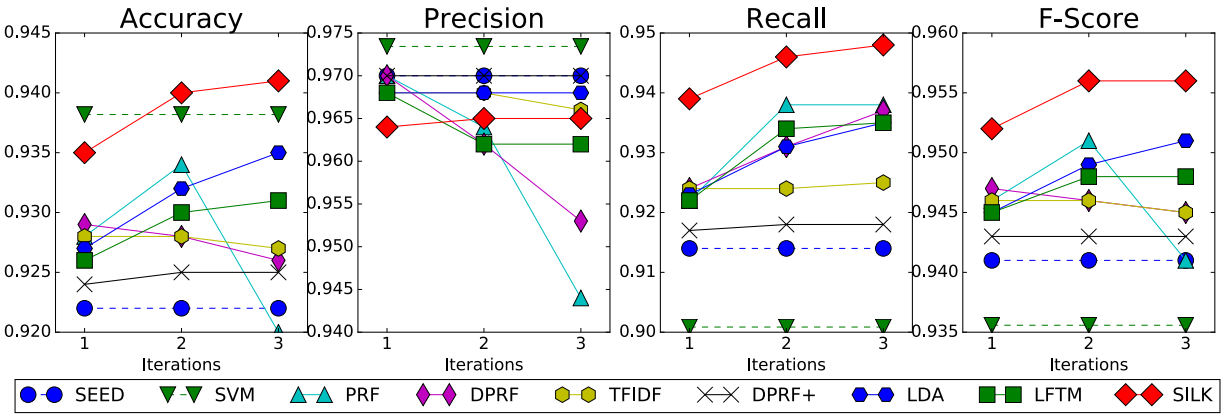

Figure 1. Haze related tweets classification accuracies, precisions, recall and F-measures using annotated data from March to June 2014. SILK outperforms all other methods compared.

This is ongoing work, and we are looking to improve methods for expanding keywords and summarizing the events. We will construct hierarchy of keywords for the event from data and human feedback. We will also apply SILK to various crisis in Twitter such as earthquake and typhoon. Mainly, Indonesian people suffer from natural disasters, so our work hope to be helpful to listen the voice of people.

References

Abdelhaq, Hamed, Sengstock, Christian, and Gertz, Michael. Eventweet: Online localized event detection from twitter. *VLDB*, 6(12):1326–1329, 2013.

Atefeh, Farzindar and Khreich, Wael. A survey of techniques for event detection in twitter. *Computational Intelligence*, 31(1):132–164, 2015.

Blei, David M, Ng, Andrew Y, and Jordan, Michael I. Latent dirichlet allocation. *JMLR*, 3(Jan):993–1022, 2003.

Imran, Muhammad, Castillo, Carlos, Diaz, Fernando, and Vieweg, Sarah. Processing social media messages in mass emergency: A survey. *CSUR*, 47(4):67, 2015.

Kim, Dongwoo and Oh, Alice H. Hierarchical dirichlet scaling process. In *ICML*, pp. 973–981, 2014.

Lavrenko, Victor and Croft, W Bruce. Relevance based language models. In *Proceedings of the SIGIR*, pp. 120–127. ACM, 2001.

Lin, Chen, Lin, Chun, Li, Jingxuan, Wang, Dingding, Chen, Yang, and Li, Tao. Generating event storylines from microblogs. In *CIKM*, pp. 175–184. ACM, 2012.

Liu, Fei, Liu, Feifan, and Liu, Yang. Automatic keyword extraction for the meeting corpus using supervised approach and bigram expansion. In *SLT 2008*, pp. 181–184. IEEE, 2008.

Manning, Christopher D., Raghavan, Prabhakar, and Schütze, Hinrich. *Introduction to Information Retrieval*. Cambridge University Press, NY, USA, 2008.

Marcus, Adam, Bernstein, Michael S., Badar, Osama, Karger, David R., Madden, Samuel, and Miller, Robert C. Twitinfo: Aggregating and visualizing microblogs for event exploration. In *SIGCHI*, pp. 227–236. ACM, 2011.

Mathioudakis, Michael and Koudas, Nick. Twittermonitor: trend detection over the twitter stream. In *Proceedings of the SIGMOD*, pp. 1155–1158. ACM, 2010.

Mikolov, T and Dean, J. Distributed representations of words and phrases and their compositionality. *NIPS*, 2013.

Miller, George A. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

Nguyen, Dat Quoc, Billingsley, Richard, Du, Lan, and Johnson, Mark. Improving topic models with latent feature word representations. *TACL*, 3:299–313, 2015.

Olteanu, Alexandra, Castillo, Carlos, Diaz, Fernando, and Vieweg, Sarah. Crisislex: A lexicon for collecting and filtering microblogged communications in crises. In *ICWSM*, 2014.

Sakaki, Takeshi, Okazaki, Makoto, and Matsuo, Yutaka. Earthquake shakes twitter users: Real-time event detection by social sensors. In *WWW*, pp. 851–860. ACM, 2010.

Wu, Wentao, Li, Hongsong, Wang, Haixun, and Zhu, Kenny Q. Probase: A probabilistic taxonomy for text understanding. In *Proceedings of the SIGMOD*, pp. 481–492. ACM, 2012.