Ensemble of Feature Sets and Classification Methods for Stance Detection

Jiaming Xu[a], Suncong Zheng[a], Jing Shi[a], Yiqun Yao[a], and Bo Xu[a,b]

[a]Institute of Automation, Chinese Academy of Sciences (CAS). Beijing, China
[b]Center for Excellence in Brain Science and Intelligence Technology, CAS. China
{jiaming.xu,suncong.zheng,shijing2014,yaoyiqun2014,xubo}@ia.ac.cn

Abstract. Stance detection is the task of automatically determining the author's favorability towards a given target. However, the target may not be explicitly mentioned in the text and even someone may refer some positive opinions to against the target, which make the task more difficult. In this paper, we describe an ensemble framework which integrates various feature sets and classification methods, and does not consist any handcrafted templates or rules to help stance detection. We submit our solution to NLPCC 2016 shared task: Detecting Stance in Chinese Weibo (Task A), which is a supervised task towards five targets. The official results show that our solution of the team "CBrain" achieves one 1st place and one 2nd place on these targets, and the overall ranking is 4th out of 16 teams. Our code is available at https://github.com/jacoxu/2016NLPCC_Stance_Detection.

Keywords: Stance Detection, Ensemble Framework, Text Classification, Chinese Weibo.

1 Introduction

Stance detection is the task of automatically determining from text whether the author of the text is in favor of, against, or neutral towards a given target [18], which has widespread applications in information retrieval, text summarization [9], and textual entailment [14]. This task can be viewed as a subtask of opinion mining and it stands next to the sentiment analysis [12].

Conventional techniques in stance detection generally follow topical text classification methods [11], where a text is regarded as a bag of words (BOW), and then classified by machine learning techniques. However, the given target, for stance detection, may not be explicitly presented in the text and even someone may refer other people's positive opinions to against the target, which has significant difference from the topical text classification task. From another perspective to detect stance, Hasan and Ng [19] employed additional linguistic features to train a stance classifier, and Jindal and Liu [22] learned patterns of opinion expression in the texts. However, social media data are often event-driven temporal information. Extracting the accurate linguistic features and learning practicable patterns form these social media are also challenge tasks which may introduce additional noises to the current task.

Considering that integrating different types of features and classifications may overcome their individual drawbacks and benefit from each other's merits [22], we propose an ensemble framework to solve stance detection task. In particular, we first generated various types of semantic features to describe the stance representations, such as Paragraph Vector (Para2vec) [13], Latent Dirichlet Allocation (LDA) [2], Latent Semantic Analysis (LSA) [5], Laplacian Eigenmaps (LE) [1] and Locality Preserving Indexing (LPI) [8]. Then, these features are ranked and selected to train various classification methods, including Random Forest (RF) [3], Linear Support Vector Machines (SVM-Linear) [10], SVM with RBF Kernel (SVM-RBF) [21] and AdaBoot [7]. Finally, we use ensemble techniques to integrate multiple classification methods to further improve the performance.

Our contributions are three-fold: (1) We explore an ensemble framework by integrating various feature sets and classification methods to solve stance detection task. (2) Our framework can be successfully conducted without designing any handcrafted templates or rules to detect the author's stance. (3) Two feature selection strategies are exploited to help choose the best feature groups, and we further investigate the influence of these features.

2 Problem Description

The Detecting Stance in Chinese Weibo (Task A) at NLPCC 2016 is a supervised task to test stance towards five targets: *#Firecracker* (“春节放鞭炮”), *#Iphone* (“IphoneSE”), *#Terrorism* (“俄罗斯在叙利亚的反恐行动”), *#Child* (“开放二胎”) and *#Motorcycle* (“深圳禁摩限电”). Participants were provided 600 labeled training Weibo texts as well as some unlabeled Weibo texts

Table 1. Statistics of the five target datasets. Note that we drop out 14 bad labeled samples (lack of texts or labels) in *#Motorcycle* and 1 bad unlabeled sample (lack of texts) in *#Child* from the raw datasets.

Dataset	*#Firecracker*	*#Iphone*	*#Terrorism*	*#Child*	*#Motorcycle*
FAVOR	250 (41.7%)	245 (40.8%)	250 (41.7%)	260 (43.3%)	160 (27.3%)
AGAINST	250 (41.7%)	209 (34.8%)	250 (41.7%)	200 (33.3%)	300 (51.2%)
NONE	100 (16.7%)	146 (24.3%)	100 (16.7%)	140 (23.3%)	126 (21.5%)
Labeled	600	600	600	600	586
Unlabeled	600	600	600	599	600
Test	3,000	3,000	3,000	3,000	3,000

and 3,000 test Weibo texts for each target. The statistics of these datasets are summarized in Table 1, where the training Weibo texts are labeled with three stances: FAVOR, AGAINST and NONE. It can be seen that the distribution is not uniform and there is always a preference towards a certain stance. For example, 51.2% of Weibo texts about *#Motorcycle* are labeled as AGAINST.

We further report the vocabulary size of the original datasets in Table 2. It can be seen that the labeled data of each target dataset has a large vocabulary but still has a huge gap to cover the vocabulary of the test data. For example, for *#Firecracker*, the vocabulary sizes of labeled data and +Unlabeled are 5,543 and 8,220 respectively while the vocabulary size is rapidly extended to 19,268 by adding test data. In order to control the computational complexity and filter some nonsense words, it is necessary to limit the vocabulary to contain a smaller size of most meaningful words by using word selection methods, such as Mutual Information (MI), Information Gain (IG) and Chi-square test (CHI) [23].

3 Feature Engineering

The BOW model for text representation is simple and quite efficient in classification task. However, the texts in social media is very short and have a large vocabulary size which make the BOW based text representation, as a high dimensionality of feature space, is very sparse. In order to compress the native feature space, we use Chi-square test (CHI) to automatically remove the non-informative words and form a refined feature space. Nonetheless, BOW based representation has a semantic gap problem which fails to construct the latent

Table 2. Original vocabulary size of the five target datasets after preprocessing. +Test means the vocabulary size of labeled data, unlabeled data and test data together.

Dataset	*#Firecracker*	*#Iphone*	*#Terrorism*	*#Child*	*#Motorcycle*
Labeled	5,543	3,409	4,076	5,334	4,475
+Unlabeled	8,220	5,461	5,992	7,882	6,632
+Test	19,268	13,302	10,318	15,969	8,787

semantic relevance, thus we also use some latent semantic techniques, such as Para2vec, LDA, LSA, LE and LPI, to capture the semantic representations of texts based on the native feature space and the refined feature space respectively. Furthermore, we extract some stance-lexicon features from two public subjectivity lexicons. More details are described in the following sections.

3.1 Text Preprocessing

For the Chinese Weibo texts, we process the raw texts via the following steps: (1) Removing all the hashtag #target by considering that the content in hashtag mostly just highlight the target and rarely contains the author's stance; (2) Removing all URLs and @tags, such as "http://t.cn/RqGPSED" and "@ChinaDaliy"; (3) Transferring the full-width characters into half-width characters and converting letters into lower case; (4) Removing all the special symbols; (5) Segmenting Chinese words using Ansj[1].

3.2 Word Selection

As the comparative study of feature selection methods in [23], the results show that MI had relatively poor performance due to its bias towards favoring rare terms while IG and CHI showed most effective in aggressive term removal without losing categorization accuracy. In this paper, we choose CHI as the feature selection to reduce the vocabulary size. CHI test measures the lack of independence between the i-th word w_i and the j-th class C_j [6].

After obtaining the CHI statistic values of the words toward to the stances, the top 500 words in each stance of each target dataset are selected by ranking their Chi-square values to form the refined vocabulary as shown in Table 3. In our

[1] `https://github.com/NLPchina/ansj_seg`.

Table 3. Refined vocabulary size of the five target datasets, where the top 500 in each stance is selected by CHI. Null/Test means that the number of Weibo texts with empty content after word selection.

Dataset	*#Firecracker*	*#Iphone*	*#Terrorism*	*#Child*	*#Motorcycle*
Vocab size	1,227	1,234	1,216	1,218	1,219
Null/Labeled	1/600	1/600	6/600	0/600	2/586
Null/Unlabeled	1/600	1/600	49/600	2/599	0/600
Null/Test	2/3,000	19/3,000	243/3,000	1/3,000	43/3,000

experiments, we use the refined vocabulary to construct BOW representations of the Weibo texts which are respectively weighted with term frequency (TF) and term frequency-inverse document frequency (TFIDF).

3.3 Latent Semantic Features

Para2vec features. Paragraph Vector (Para2vec) is an unsupervised method to learn distributed representations of word and paragraphs [13]. The key idea is to learn a compact vector by predicting nearby words in a fixed context window. In our experiments, we use the open-source software released by Mesnil et al. [17]. The dimension of the embedding is set to 50, and the model is trained on the original datasets of each target, including labeled, unlabeled and test datasets.

LDA features. Latent Dirichlet Allocation (LDA), first introduced by Blei et al. [2], is a probabilistic generative model that can be used to estimate the multinomial observations by unsupervised learning [20]. The number of topic is set to 50 and the model is estimated on the original datasets of each target.

LSA features. Latent Semantic Analysis (LSA) [5] is the most popular global matrix factorization method, which applies a dimension reducing linear projection, Singular Value Decomposition (SVD), of the corresponding BOW matrix. For our task, we apply LSA on two matrices, one is constructed based on the original vocabulary and another is constructed based on the refined vocabulary, to map the BOW representations into two 50-dimensional subspaces, denoted as LSA features and LSA (CHI) features respectively in this paper.

LE and **LPI** features. Laplacian Eigenmaps (LE) [1] discover the manifold structure of the BOW features by extracting the top eigenvectors of graph Laplacian, that is the similarity matrix of texts. Locality Preserving Indexing (LPI)

Table 4. ACC results obtained via 4-fold cross validation by using RF method on five target training datasets. The three highest scores on each target are shown in bold.

Feature	*#Firecracker*	*#Iphone*	*#Terrorism*	*#Child*	*#Motorcycle*
Para2vec	66.83	48.50	47.33	60.83	61.26
LDA	63.83	42.00	47.83	49.67	57.85
LSA	68.67	51.83	49.33	61.33	66.21
LE	64.83	48.00	50.17	61.00	64.99
LPI	63.83	48.17	51.27	61.17	65.02
TF	71.00	52.50	**57.17**	61.83	64.69
TFIDF	71.50	53.16	**57.33**	61.33	64.34
LSA (CHI)	**72.33**	**58.83**	**61.00**	**68.33**	**74.06**
LE (CHI)	**73.17**	**61.67**	45.33	**68.33**	**68.95**
LPI (CHI)	**73.17**	**61.33**	54.50	**67.33**	**68.94**
Lex. Fea.	38.83	41.33	40.50	42.50	43.34

[8] can be seen as an extended version of LE to deal with the unseen texts by approximating a linear function. Here, we construct two local similarity matrices, using heat kernel measure, based on the original text representation and the refined text representation, which are all weighting with TFIDF. Finally, we get four 50-dimensional feature spaces, denoted as LE features, LE (CHI) features, LPI features and LPI (CHI) features respectively.

3.4 Lexical Features

The lexical features are extracted from two public subjectivity lexicons, one is an evaluation word set generated from HowNet and the other one is an emotion word set released by Li and Sun [15]. Both of these lexicons organize the subjective words into two groups: positive and negative. Take one Weibo text as an example, let P_{ev} be the number of the positive evaluation words in the text, N_{ev} be the number of the negative evaluation words, P_{em} be the number of the positive emotion words, and N_{em} be the number of the negative emotion words. The following four features are then generated:

(1). Positive evaluation feature: $(P_{ev}+1)/(P_{ev}+{N_{ev}}^{\lambda}+2)$;

(2). Negative evaluation feature: $({N_{ev}}^{\lambda}+1)/(P_{ev}+{N_{ev}}^{\lambda}+2)$;

(3). Positive emotion feature: $(P_{em}+1)/(P_{em}+{N_{em}}^{\lambda}+2)$;

(4). Negative emotion feature: $({N_{em}}^{\lambda}+1)/(P_{em}+{N_{em}}^{\lambda}+2)$.

In these features, the exponent λ is used to control the effect of the negative words, and λ is set to 1.3 in our experiments.

4 Model Training

Here, we first rank and select the above features to train various classification methods, including Random Forest (RF) [3], Linear Support Vector Machines (SVM-Linear) [10], SVM with RBF Kernel (SVM-RBF) [21] and AdaBoot [7]. Then, an ensemble framework is applied to further improve the performance.

4.1 Feature Ranking and Selection

In Section 3, we get 11 feature vectors in total. It is crucial to identify important features and remove the redundant features to reduce the train cost and noises [23, 4]. Inspired by [16], we exploit two feature selection strategies, one is top k based selection and the other one is leave-out k based selection. Before conducting the two selection methods, all features are ranked based on the results of each classification method on each target dataset. As an example, Table 4 shows the ACC (Accuracy) scores of RF method on five target training datasets via cross validation. It can be seen that the features based on the refined vocabulary space, such as TF, TFIDF, LSA (CHI), LE (CHI) and LPI (CHI), can achieve a better performance, and lexical feature perform a worst results on the five target datasets. We also observe the similar results of the other classification methods. But due to the limit of space, they are not presented in this paper.

For top k based selection, we simply select the k best features, based on the feature ranking results, as the best feature groups. For example, we apply top 3 selection for RF method on *#Terrorism*, the best feature group is generated as {TF, TFIDF, LSA (CHI)}. For leave-out k based selection, we iteratively evaluate the importance of each feature by leaving it out from all the current feature group and remove the most insignificant feature, until k features are removed. The optimal selection strategy and parameter k are tuned by using each classification on each target training dataset via cross validation.

4.2 Model Ensemble

To further improve the performance, ensemble technique is utilized to integrate the outputs of the various classification methods as $p(C|x) = \sum_{i=1}^{m} w_i \cdot p_i(C|x)$, where $p(C|x)$ is the final probability that the author of a Weibo text x toward to the stance C, $p_i(C|x)$ is the probability predicted by the i-th classification method, m is the number of classification methods, and w is the weight parameter learned by a linear model.

5 A Performance Study

5.1 Importance of The Features

In this section, we evaluate the importance of all the features on each target training dataset via cross validation. Two feature selection strategies are conducted based on the ranked results of the features by using the classification methods. Figure 1 shows the ACC results obtained via 4-fold cross validation by using the classification method on five target training datasets. It can be seen that RF and SVM-Linear show the best performances and SVM-RBF not well deal with integrated features. the optimal feature selection strategies and the best feature groups for each classification are selected and reported in Table 5. Two feature selection strategies perform their respective advantages on different target datasets and different classification methods, and our ensemble framework improves upon the single models on 3 out of 5 target datasets. One interesting result as shown in Table 5 is that the lexical features, showing the worst performances, are not dropped by the leave-out k selection on all target datasets. An explanation maybe that leave-out k selection prefers to drop the redundant features from all feature groups, such as LE (CHI) and LPI (CHI).

5.2 Performance on Test Data

Part of the official ranking results on test datasets are summarized in Table 6. We can see that our solution of the team “CBrain” achieves one 1st place on *#Child* and one 2nd place on *#Motorcycle*, and the overall ranking is 4 out of all 16 teams, where the official metric of our solution is very close to the above two teams within 0.4%. Note that the official result of our solution on *#Terrorism* is lower than the top team about 10%. The reason to explain this problem maybe that, we empirically select top 500 words in each stance of each target dataset via CHI to form the refined vocabulary which leads to lots of null samples, as shown in Table 3, and lots of null samples may hurt the prediction of our system.

6 Conclusion

This paper presents an ensemble framework for stance detection in Chinese Weibo hosted at NLPCC 2016 conference. In the framework, a lots of semantic features are captured from Weibo text and two feature selection strategies are exploited to generate the optimal feature groups. Moreover, we train various classification methods based on the optimal feature groups and integrate the results of these methods to further improve the performance. Our solution can be successfully conducted without designing any handcrafted templates or rules to detect the author's stance which has good scalability for other related tasks.

Acknowledgments We thank the anonymous reviewers for their insightful comments, and this work was supported by the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No. XDB02070005), the National High Technology Research and Development Program of China (863 Program) (Grant No. 2015AA015402) and the National Natural Science Foundation (Grant No. 61602479 and 61403385).

References

1. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation 15(6), 1373–1396 (2003)
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research 3(Jan), 993–1022 (2003)
3. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)
4. Chen, M., Jin, X., Shen, D.: Short text classification improved by learning multi-granularity topics. In: IJCAI. pp. 1776–1781. Citeseer (2011)
5. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. JASIS 41(6), 391 (1990)
6. Dunning, T.: Accurate methods for the statistics of surprise and coincidence. Computational Linguistics 19(1), 61–74 (1993)
7. Freund, Y., Schapire, R.E., et al.: Experiments with a new boosting algorithm. In: ICML. vol. 96, pp. 148–156 (1996)

8. He, X., Cai, D., Liu, H., Ma, W.Y.: Locality preserving indexing for document representation. In: SIGIR. pp. 96–103. ACM (2004)
9. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: KDD. pp. 168–177. ACM (2004)
10. Joachims, T.: Learning to classify text using support vector machines: Methods, theory and algorithms. Kluwer Academic Publishers (2002)
11. Kim, S.M., Hovy, E.H.: Crystal: Analyzing predictive opinions on the web. In: EMNLP-CoNLL. pp. 1056–1064 (2007)
12. Krejzl, P., Steinberger, J.: Uwb at semeval-2016 task 6: Stance detection. Proceedings of SemEval pp. 408–412 (2016)
13. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: ICML. vol. 14, pp. 1188–1196 (2014)
14. Lendvai, P., Augenstein, I., Bontcheva, K., Declerck, T.: Monolingual social media datasets for detecting contradiction and entailment. In: LREC (2016)
15. Li, J., Sun, M.: Experimental study on sentiment classification of chinese review using machine learning techniques. In: NLPKE. pp. 393–400. IEEE (2007)
16. Liu, G., Nguyen, T.T., Zhao, G., Zha, W., Yang, J., Cao, J., Wu, M., Zhao, P., Chen, W.: Repeat buyer prediction for e-commerce. In: KDD. ACM (2016)
17. Mesnil, G., Mikolov, T., Ranzato, M., Bengio, Y.: Ensemble of generative and discriminative techniques for sentiment analysis of movie reviews. arXiv preprint arXiv:1412.5335 (2014)
18. Mohammad, S.M., Kiritchenko, S., Sobhani, P., Zhu, X., Cherry, C.: Semeval-2016 task 6: Detecting stance in tweets. In: SemEval. vol. 16 (2016)
19. Ng, V., Hasan, K.S.: Predicting stance in ideological debate with rich linguistic knowledge. In: COLING. p. 451 (2012)
20. Phan, X.H., Nguyen, L.M., Horiguchi, S.: Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In: WWW (2008)
21. Schölkopf, B., Smola, A.J.: Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press (2002)
22. Xia, R., Zong, C., Li, S.: Ensemble of feature sets and classification algorithms for sentiment classification. Information Sciences 181(6), 1138–1152 (2011)
23. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: ICML. vol. 97, pp. 412–420 (1997)

Table 5. ACC results obtained via 4-fold cross validation on five target datasets.

Dataset	Mothod	Selection	Features	ACC
#Firecracker	RF	Top 3	LE (CHI), LPI (CHI), LSA (CHI)	74.17
	SVM-Linear	Top 2	LPI (CHI), LSA (CHI)	**76.50**
	SVM-RBF	Top 1	LPI (CHI)	76.00
	AdaBoost	Leave-out 3	TFIDF, LE (CHI), LSA	73.17
	Ensemble	–	–	73.83
#Iphone	RF	Leave-out 4	LPI, LSA, Para2vec, TF	63.33
	SVM-Linear	Leave-out 3	TFIDF, Para2vec, LE	61.83
	SVM-RBF	Top 1	LE (CHI)	59.83
	AdaBoost	Leave-out 1	TFIDF	61.67
	Ensemble	–	–	**63.67**
#Terrorism	RF	Top 4	LSA (CHI), TFIDF, TF, LPI (CHI)	62.00
	SVM-Linear	Leave-out 3	TFIDF, LPI (CHI), LE	61.50
	SVM-RBF	Top 1	LSA (CHI)	63.50
	AdaBoost	Leave-out 4	TFIDF, LPI (CHI), LSA, Para2vec	52.67
	Ensemble	–	–	**65.00**
#Child	RF	Top 3	LE (CHI), LSA (CHI), LPI (CHI)	70.17
	SVM-Linear	Top 1	LPI (CHI)	70.67
	SVM-RBF	Top 1	LSA (CHI)	69.50
	AdaBoost	Leave-out 4	TFIDF, Para2vec, LSA, LPI (CHI)	66.33
	Ensemble	–	–	**71.33**
#Motorcycle	RF	Top 1	LSA (CHI)	74.06
	SVM-Linear	Top 1	LSA (CHI)	72.87
	SVM-RBF	Top 1	LSA (CHI)	**74.92**
	AdaBoost	Top 2	LSA (CHI), LPI (CHI)	71.00
	Ensemble	–	–	74.24

Table 6. Part of the official results (Official metric: (F_FAVOR + F_AGAINST)/2).

TeamID	*#Firecracker*	*#Iphone*	*#Terrorism*	*#Child*	*#Motorcycle*	Overall
RUC_MMC	77.30	57.80	**58.14**	80.36	76.52	71.06
TopTeam	74.49	57.64	52.32	76.61	**79.49**	68.94
SDS	**77.84**	**58.52**	53.32	79.48	68.83	68.61
CBrain	76.04	55.28	47.87	**81.35**	78.55	68.56
...	...	...	...	...	...	...
SCHOOL	34.22	42.22	39.03	46.13	36.76	39.95

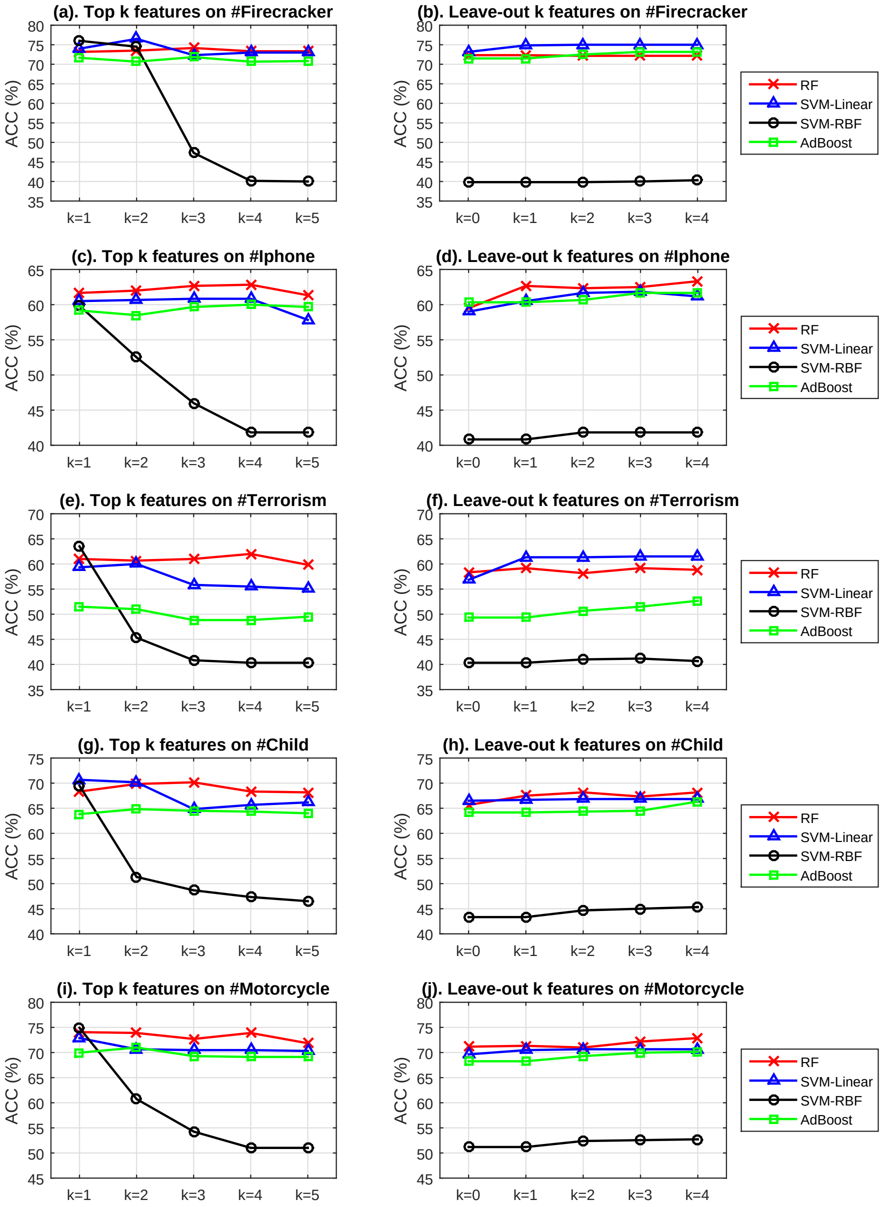

Fig. 1. ACC curves on five target training datasets via 4-fold cross validation by varying the parameter k of the two feature selection methods (top k selection and leave-out k selection). Note that leave-out 0 means that all features are reserved.