J. Cent. South Univ. (2014) 21: 1376−1384
DOI: 10.1007/s11771-014-2075-4

Springer

Construction of unsupervised sentiment classifier on idioms resources

XIE Song-xian(谢松县), WANG Ting(王挺)

School of Computer Science, National University of Defense Technology, Changsha 410073, China

Abstract: Sentiment analysis is the computational study of how opinions, attitudes, emotions, and perspectives are expressed in language, and has been the important task of natural language processing. Sentiment analysis is highly valuable for both research and practical applications. The focuses were put on the difficulties in the construction of sentiment classifiers which normally need tremendous labeled domain training data, and a novel unsupervised framework was proposed to make use of the Chinese idiom resources to develop a general sentiment classifier. Furthermore, the domain adaption of general sentiment classifier was improved by taking the general classifier as the base of a self-training procedure to get a domain self-training sentiment classifier. To validate the effect of the unsupervised framework, several experiments were carried out on publicly available Chinese online reviews dataset. The experiments show that the proposed framework is effective and achieves encouraging results. Specifically, the general classifier outperforms two baselines (a Naïve 50% baseline and a cross-domain classifier), and the bootstrapping self-training classifier approximates the upper bound domain-specific classifier with the lowest accuracy of 81.5%, but the performance is more stable and the framework needs no labeled training dataset.

Key words: sentiment analysis; sentiment classification; bootstrapping; idioms; general classifier; domain-specific classifier

1 Introduction

The amount of user-generated content (UGC) on the Internet has risen exponentially over the last decade with the emergence and advance of Web 2.0 technology, and such content is now always at our fingertips. UGC, in particular, becomes an ever-growing source of opinions and sentiments which are spread worldwide through blogs, wikis, chats and diverse social networks such as Twitter and Facebook [1]. The distillation of subjective knowledge from such abundant sources is an important part of applications in fields such as commerce, tourism, politics and health. As a live example of daily life, more and more review sites continue to grow in popularity as more and more people begin to refer the reviews of fellow users regarding services and products before they make their dealing decision. However, with the explosion of such information, users are often forced through large quantities and sometimes low quality reviews in order to find the useful information they really need. This has led to increasing research interests in the areas of opinion mining and sentiment analysis.

Sentiment analysis is the computational study of how opinions, attitudes, emotions, and perspectives are expressed in language (especially in written text), so as to provide tools and techniques for extracting this kind of evaluative information from large datasets [2−3]. With the growing need of identifying opinions and sentiments automatically from text data on the web, sentiment analysis has received considerable attention recently, and been applied to business intelligence, public opinion analysis, election prediction, etc.

Sentiment classification, which deals with determining sentiment orientation of target text, is one of sentiment analysis tasks [4]. The task can be viewed as a specific text categorization problem. Given an instance of opinionated text, the goal is to classify it as positive or negative, or neutral in multi-class classification sometimes. In fact, sentiment classification is a more challenging task than text classification. Firstly, methods and techniques developed in traditional text classification usually do not work well on this task, since they tend to take frequent-occurring words, which are also called keywords, as good indicators of the class a document belongs to. However, for sentiment classification of an opinionated document, words indicating sentiment are usually ambiguous and maybe infrequent. Secondly and most importantly, sentiment expressions critically depend on domains and contexts, and opinions are often hidden in a large amount of domain dataset, so there are often no universal sentiment resources available for sentiment classification. However, human-labeled resources for each domain are costly and difficult, for

Foundation item: Projects(61170156, 60933005) supported by the National Natural Science Foundation of China
Received date: 2012−12−02; **Accepted date:** 2013−03−25
Corresponding author: XIE Song-xian, PhD; Tel: +86−731−84574627; E-mail: xsongx@nudt.edu.cn

manual annotation is very expensive and time-consuming.

Recently, many researchers have cast their eyes on all kinds of machine learning techniques. With more and more work on document-level sentiment classification using machine learning methods, various classifiers and feature sets have been explored [4], which can be categorized into supervised, semi-supervised and unsupervised approaches.

Supervised approaches were firstly applied to sentiment classification by PANG et al [4] by comparing multiple supervised machine learning algorithms for the task of movie reviews sentiment classification. Afterwards various classifiers and features selection has manifested in many other researches [5−7]. Performance of supervised approaches is reasonably satisfying because of the requirement that test data should be similar to manually annotated training data in the same domain, and the sound performance is often the upper bound for other approaches to compete with. But collecting annotated data in the new domain and retraining the classifier are unavoidable for adapting a supervised sentiment classifier to another domain. The dependency on domain annotated training dataset is one major shortcoming of all supervised approaches.

Semi-supervised approaches try to improve the performance of classification by fitting labeled and unlabeled datasets together with various methods such as EM on Naïve Bayes, co-training, transductive SVMs, and co-regularization [8−10]. But just as supervised methods, labeled training datasets are needed for semi-supervised approaches, which are mainly annotated by hand because the reliability of training data is the main consideration for learning methods. Although in recent years many researchers have tried to solve the shortage problem of training dataset by adaptive techniques (such as transfer learning) to realize cross-domain [11−12] or cross-language [13] sentiment classification, most are inefficient in that adaptive learning is needed for the changing of target domain. Furthermore, the low accuracy is another problem of adaptive methods because of disambiguation of sentiment word in different domain or languages.

From the very beginning, many unsupervised approaches have been brought forth to tackle the problem of annotated training data shortage and domain dependence, which are mostly domain-independent rules based, and try to get some highly confident examples produced by the rule based classifier as the training data for bootstrapping learning of next stage [14−16]. These rules rely on universal sentiment expressions in the language recognized based on linguistic expertise knowledge. Therefore, there are two problems of these methods: firstly, linguistic expertise knowledge is needed for the manually produced or automatically learned rules; secondly, discovering all sentiment expressions in the text by a few rules is impossible because human intuition may not be always correct and comprehensive for the complication and variety of language [4], so limited or biased examples are returned as the low coverage of the rules, which will influence the performance of pipeline classifier of the unsupervised procedure.

The problem of domain dependence of sentiment classification was focused on in this work. An unsupervised framework based on general resources independent of domains was put forward. In the framework, without the need of laborious labeling training data, a general classifier was trained on the off-the-shelf resources which are highly opinionated but do not depend on any context and domain. The proposed general classifier could output highly confident instances as training data for the next stage bootstrapping domain classifier.

2 Formulation of sentiment classification

Sentiment classification task aims to automatically classify document as predefined sentiment polarity classes of binary (negative or positive) or multi-class (negative, positive or neutral), and binary classification has been studied for simplicity. Formally, given document corpus $D=\{d_1, \cdots, d_n\}$, and predefined sentiment category set $C=\{1, -1|\text{positive}=1, \text{negative}=-1\}$, the task of sentiment classification is to predict each d_i in D with a label c_i expressed in C should belong to. To be along with text categorization, each document can be represented as a vector of bag-of-words features $\boldsymbol{x}=\mathbf{R}^n$, where n is the size of a pre-specified vocabulary V. The weight of each entry in this vector usually is specified as binary, with weight equal to 1 for terms present in the vector and 0 for absent. Given a training dataset $X=\{x_1, \cdots, x_m\}$, we can build a binary classifier:

$$f: X \to Y, Y = \{-1, 1\} \tag{1}$$

and employ it to predict label for an unseen instance $\boldsymbol{x}$ by computing $f(\boldsymbol{x})$, with each instance being represented as a vector $\boldsymbol{x}=(w_1, \cdots, w_v)$, in which w_i is the weight of the i-th feature.

3 Hypothesis of feature space division

Often there is an implicit hypothesis underlying previous sentiment classification researches, which considers all features appearing in a document vector representing the document's sentiment polarity equally with different binary weights, for example, in the following English book review which obviously expresses positive sentiment:

Example 1: The book is recommended and sent to me by one of my good friends, and he told me to put it beside my pillow so as to read it anytime available. I read it through without any letup with the same idea, and I feel the translation very accurate without any mark of translation, the language very lively and vividly and the story easily understood to make the reader personally on the scene, and there is much use for reference in the book. I am going to read it again. Recommend!

When computationally classifying sentiment polarity of the review, all words extracted as features are considered potentially indicators of positive evaluation for the book equally. However with careful considerations, it could be found that words "recommend" and "accurate" are positive indicators of reviews almost across all domains, while "lively and vividly", "personally on the scene" and "use for reference" are more frequently used in the domain of book reviews to express positive evaluation. With this intuition, an assumption has been proposed as:

Assumption 1: In the feature space of sentiment classification, bag-of-words features can be divided into two different parts:

1) Domain-independent part, i.e. general sentiment features, which are indicators of sentiment polarity across all domains and independent of any context of any domain;

2) Domain-dependent part, i.e. specific sentiment features, sentiment polarity of which depend on specific context of each domain.

Formally, feature vector representing a document of the sentiment classification task can be expressed as $\boldsymbol{x}=(w_1, \cdots, w_l, w_{l+1}, \cdots, w_v)$, and based on Assumption 1, feature vector $\boldsymbol{x}$ could be divided into two parts:

$$\boldsymbol{x}=\begin{cases}\boldsymbol{x}_{\mathrm{g}}: \text{General sentiment features}\\ \boldsymbol{x}_{\mathrm{s}}: \text{Specific sentiment features}\end{cases} \tag{2}$$

where $\boldsymbol{x}_{\mathrm{g}}=(w_1, \cdots, w_l)$ denotes the weights of general part of features, and $\boldsymbol{x}_{\mathrm{s}}=(w_{l+1}, \cdots, w_v)$ denotes the weights of specific part of features.

Questions might arise about Assumption 1:

Firstly, what's the meaning of the division of space, and how to identify each part of feature space?

Imagine such a scenario, when reading a review about a professional book that one could still distinguish which polarity (recommend or not), the reviewer prefer even if he knows nothing about the domain knowledge what the book describes, as long as "good", "accurate", "recommend" appear in the review. Intuitively, this kind of phenomenon may be explained by the general part of the text which is used to express the holistic sentiment polarity of the author, and the polarity of general sentiment words are prone to be recognized by anyone independent of domain knowledge. Comparably, in sentiment classification, we put forward that the sentiment polarity of a document could still be recognized with only the text segment of general part of feature space $\boldsymbol{x}_{\mathrm{g}}$. That is to say, theoretically, if general sentiment knowledge could be modeled, what sentiment polarity a review prefers for could be still classified based on such general sentiment models.

The second question is how to establish such kind of general sentiment model?

Many researchers have tried to establish all kinds of sentimental ontology lexicons to represent general knowledge about human's sentiment, such as SentiWordNet [17] and General Inquiry [18] in English, Hownet [19] and NTUSD (Chinese Network Sentiment Dictionary) in Chinese [20], etc. However, they all failed in modeling the universal sentiment knowledge in that many entries of these lexical resources have multiple senses with different sense representing different sentiment polarity, and the exact sense unavoidably depends on the context of each domain. Actually, such knowledge exists in many cases, in which one word or combination of a few words could identify its own exact sentiment polarity independent of domains, such as idioms and proverbs. So, the way that models the universal sentiment knowledge could be transformed as training a classifier on such resources with sentimental polarity independent of context and domains.

Then, another question is how to find such kind of instances?

In fact, this question has motivated our research at the very beginning. There are many linguistic resources highly valuable for sentiment classification, of which idiom resources attract interests of this research. Idioms are common phenomena of many languages beside Chinese, such as "castles in the air", "a bed of thorns", "bring down the house" in English. The form of idioms is succinct and the meaning is penetrating, which is a quintessence part of language. Generally speaking, the structure of idioms is fixed and can't be changed randomly; idioms have semantic intactness, which is not generally the simple summation of the literal meaning of each component; idioms chiefly use metaphor, exaggerator and comparison in the rhetoric to express its real meaning; most importantly, the sentimental orientation of idioms is independent and unchangeable under any context. There are many off-the-shelf lexical idiom resources in all kinds of languages with entries taking the example form as follows.

Example 2: Castles in the air: a derogatory term, indicate the illusive things or impractical fanciness metaphorically.

In this example, the entry is composed of three parts: the idiom "castles in the air", the semantic orientation "a derogatory term" representing negative sentiment polarity and a short paraphrase with three general

negative words (“illusive”, “impractical” and “fanciness”). The entry provides us with an illustrational universally-labeled sentimental example with general sentiment features and a negative label. Most importantly, the sentiment polarity of such instance is independent of any domain just as the idiom it explains. Based on such observation, another assumption has been proposed as follows.

Assumption 2: The sentiment polarity of the idiom paraphrase is independent of domains as the idiom it describes.

With Assumption 2 admitted, the labeled training dataset could be constructed on idiom resources with the paraphrases of idioms as train instances, and the semantic orientation values as sentimental labels. With such kind of training dataset at hand, a domain-independent classifier could be trained to model the universal sentiment knowledge, with the paraphrases being represented as vectors of general sentiment features.

4 Framework of unsupervised sentiment classification

Although theoretically a general classifier could be trained on Chinese idiom resources to model the domain-independent sentiment knowledge, the coverage and efficiency of such model are limited by the quality and quantity of idiom resources. Besides, it is obvious that the paraphrase of idiom is usually short, so the training instance for the general classifier must be very sparse, which would degrade the effect of such classifier.

For above reasons, a consistent self-training bootstrapping framework of machine learning has been chosen to upgrade the performance of the general classifier. The framework is illustrated in Fig. 1. As could be seen from the framework, a general domain-independent classifier is trained on the dataset extracted from idioms resources, which makes use of general sentiment part of features, and the general classifier is applied to each domain dataset to be classified so that the domain data are labeled initially. The labeled dataset are fed into the iterated convergent procedure of bootstrapping learning to train a domain-specific classifier which makes use of the domain specific part of features. At the same time, the output of such unsupervised machine learning framework is result dataset of the sentiment classification. Above all, the whole framework could be segmented as two pipelining training and classifying phrases.

4.1 General classifying algorithm

The same methods as PANG et al [4] adopted, since they had applied Naïve Bayes, maximum entropy and support vector machine techniques to identify the effectiveness of machine learning on sentiment classification of movie reviews, and they got satisfying result (accuracy of 82.9%).

4.1.1 Naïve Bayesian classifier

Naïve Bayesian method is one of the most popular techniques for text classification. Given a set of training documents D, with each document being considered as an ordered list of words, entry $w_{d_i,k}$ is used to denote the word in position k of document d_i, where each word is from the vocabulary $V=\langle w_1, \cdots, w_{|v|}\rangle$. The vocabulary is

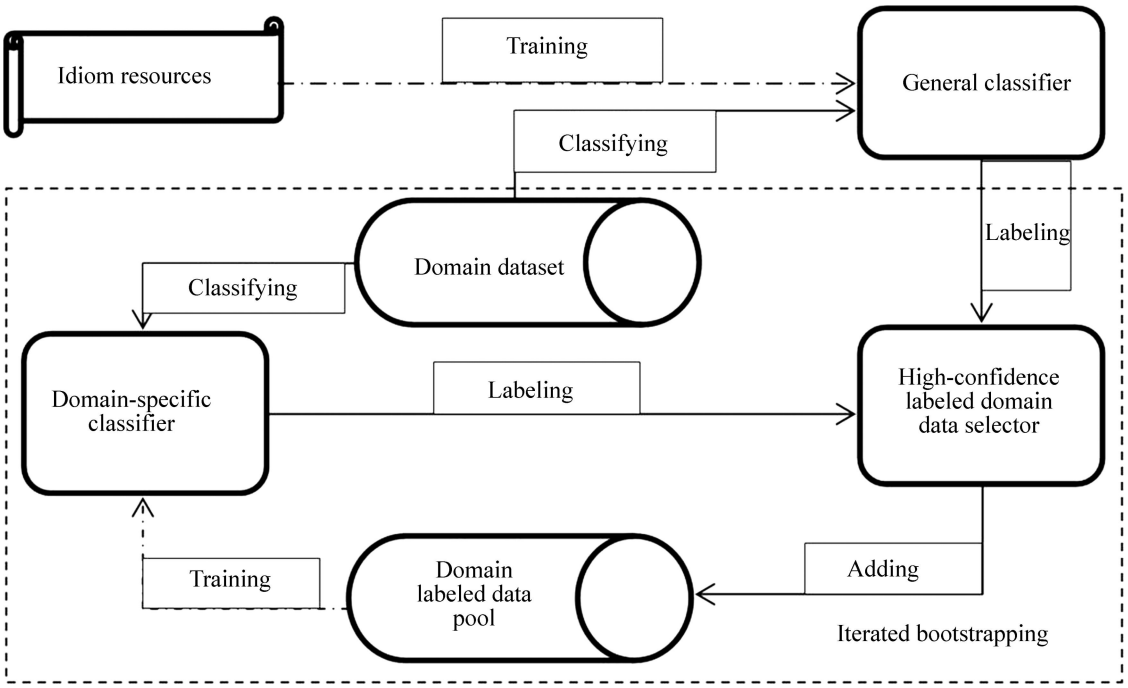

Fig. 1 Unsupervised sentiment classification Framework

the set of all words considered as features for classification. A set of pre-defined classes, $C=\{c_1, c_2\}$ is used to label a document. To determine which class a document belongs to, it is needed to compute the posterior probability, $P(c_j|d_i)$, where c_j is a class label and d_i is a document. Based on the Bayesian probability and the multinomial model, the following equation is got:

$$P(c_j)=\frac{\sum_{i=1}^{|D|}P(c_j\mid d_j)}{|D|} \tag{3}$$

Based on the hypothesis that the probabilities of document words are independent given the class:

$$P(c_j\mid d_i)=\frac{P(c_j)\prod_{k=1}^{d_i}P(w_{d_i,k}\mid c_j)}{\sum_{r=1}^{|C|}P(c_r)\prod_{k=1}^{|d_i|}P(w_{d_k,k|c_r})} \tag{4}$$

For the Naïve Bayes classifier, the class label with the highest probability is assigned as the class label of the document.

4.1.2 Maximum entropy classifier

Maximum entropy classifier has been widely applied in many natural language processing tasks [21]. The classifier assigns the class label with the higher conditional probability given the document as follows:

$$P(c_j\mid d_i,\boldsymbol{\theta})=\frac{1}{z}\exp(\boldsymbol{\theta}, f(d_i,c_j)) \tag{5}$$

where $\boldsymbol{\theta}$ is a vector of feature weights and $f(d_i, c_j)$ is a feature function that maps pairs (d_i, c_j) to a nonnegative real-valued feature vector. Each feature has an associated parameter θ_i, called its weight, and z is the corresponding normalization factor.

With a set of labeled documents D, maximum likelihood parameter estimation (training) procedure for such a model is trying to solve such a optimization problem as

$$\boldsymbol{\theta}^*=\arg\max_{\boldsymbol{\theta}}\prod_{i=1}^{|D|}P(c_j\mid d_i,\boldsymbol{\theta}) \tag{6}$$

4.1.3 Support vector machines classifier

Support vector machines classifier, abbreviated as SVM, is a kind of discriminative method of machine learning techniques. Based on the structural risk minimization principle of the computational learning theory, SVM tries to find a decision surface to separate the training data points into two classes and makes decisions based on the support vectors that are selected as the only effective discriminative points from the training dataset.

Multiple variants of SVM have been developed for different tasks [22−23]. In this work, linear SVM has been adopted due to its popularity and sound performance in sentiment classification task. The optimization of SVM (dual form) is to minimize

$$\boldsymbol{\alpha}^*=\arg\min\left\{-\sum_{i=1}^{n}\alpha_i+\sum_{i=1}^{n}\alpha_i\alpha_j x_i x_j\left\langle \boldsymbol{x}_i,\boldsymbol{x}_j\right\rangle\right\} \tag{7}$$

Subject to $\sum_{i=1}^{n}\alpha_i y_i=0;\ 0\le\alpha_i\le C$

4.2 Domain-specific classifying algorithm

As the general sentiment features are only one part of all features in the whole feature space, the other part of domain-dependent features should be considered in order to capture the subtle clues embedded in the specific sentiment expressions in each domain, so that test data of each domain could be classified more accurately than just using the general part of features. Usually, unsupervised approaches improve their performance by combining with supervised learning in target domain, and such ways were adopted to verify the rationality of the feature space division hypothesis. Specifically, initial label and the confidence of classification were identified for each instance of each domain after applying general classifier to the multi-domain dataset, so that a domain-specific supervised classifier could be trained on the labeled instances with high confidence under the unsupervised self-training framework. The main idea of this kind of bootstrapping method was to apply a domain- independent classifier to label each domain dataset, and the instances labeled by the general classifier with high confidence served as training data for a supervised machine learning classifier. Generally speaking, the resulting domain-specific supervised classifier should be more effective in each domain than the general classifier as long as the general classifier is dependable enough. Otherwise, the performance would be inferior because too many false-labeled instances passing into iterative self-training process would bring down the domain-specific classifier.

Alogrithm 1: Self-training algorithm

Input: General classifier C_g;

Domain test dataset U

Self-training procedure:

1) Classify each domcument d_i in U with classifier C_g;

2) Initiate domain training dataset S:

(1) Sort d_i according to classifying confidence,

(2) Initiate S with high-confidence documents,

(3) Remove s documents from U;

3) Iterative self-training procedure:

(1) Train domain classifier C_d on dataset S,

(2) Classify each document d_i in U with classifier

C_d,

(3) If convergent, end the procedure,

(4) Extend S, sort d_i according to classifying confidence, extend S with s high-confidence documents, remove s documents from U;

4) Get A by classifying U with the self-training classifier C_s

Output: Automatically labeled domain dataset A; self-training classifier C_s

5 Experiments

In this section, systematical evaluation of the proposed approaches designed for document-level sentiment classification is introduced.

5.1 Experiment description

5.1.1 Dataset

Idioms are common phenomena of Chinese, most of which embody the sentimental polarity of the users, but sentiment classification of Chinese hasn't make full use of such kind of rich language resources. So, in sentiment classification, the Chinese idioms become the target resources of constructing the training dataset to model the universal sentiment. Because of limited entries of most existent Chinese idioms dictionaries, and to make the case worse, some of entries do not contain sentiment polarity labels, and the abundant idiom resources on the web labeled with collective intelligence become the main resources for constructing training dataset. After crawling from the online idiom dictionary of "China Education Network" [24], an idiom dataset of 24395 entries were achieved, with each entry labeled with positive, negative and neutral class. Since the research has focused on binary classification, the neutral entries were removed from the dataset. As a result, a final dataset of 8160 instances labeled with positive and negative classes were used to train the general classifier. To evaluate the performance of general classifier and train the self-training classifier, three publicly available Chinese reviews corpus of three domains (book, hotel and notebook PC) [25] were adopted, each of which consisted of 4000 reviews (2000 positives and 2000 negatives).

5.1.2 Packages and classifiers

Text written in Chinese is not well formatted in that words in a sentence are not separated by space as English. All the text in Chinese must be segmented before bag-of-words features being extracted. In the experiment, Chinese text of train and test dataset was segmented with an open source Chinese segmentation package named "Mmseg" [26]. As for classifiers, Naïve Bayes classifier, maximum entropy classifier of Natural Language ToolKits (NLTK) [27] package and support vector machine classifier of Libsvm [28] package were used for classification. All the parameters and settings were optimized by cross-validation.

5.1.3 Feature selection

Feature selection is often used to pick out discriminating features from training dataset for efficient classification. Among various methods, I statistic measure which calculates the association value between features and categories was chosen [29], defined as

$$I(t,c_i)=\frac{N\times(AD-BE)^2}{(A+E)\times(B+D)\times(A+B)\times(E+D)} \tag{8}$$

where A is the number of times that feature t and category c_i co-occur; B is the number of times that t occurs without that c_i; E is the number of times that c_i occurs without t; D is the number of times that neither c_i nor t occurs; N is the total number of documents. Features with high I values were selected and the last 5% low-value features were removed.

5.1.4 Baselines and upper bound

1) Baselines

Two baselines were used to compare with the proposed method: the first one was Naïve 50% baseline since the test corpuses were all balanced with respect to the sentiment classes, and the other one was the cross-domain classifier with the same algorithms and settings as the general classifier. The latter baseline was used to demonstrate the superiority of the proposed general classifier to cross-domain classifier in that not only could it be applied to any domain without any labeled training dataset but be more robust and dependable than using labeled dataset of other domains.

2) Upper bound

As mentioned in introduction section, supervised machine learning methods in each domain are often set up as upper bound whose performance can be used to be challenged by semi-supervised and unsupervised methods. In the experiments, an upper bound was also set up by training supervised classifiers with the same algorithms and settings as general classifier except for the dataset settings. For each classifier, each domain dataset was split five-folded with one fifth for testing and others for training, and the performance was measured by averaging the results of five iterative computations on split dataset.

5.1.5 Performance measurement

There are various complicated measurements to evaluate the performance of computational algorithm, the simplest accuracy index of which was chosen to evaluate the performance of sentiment classifiers, because the comparison between measurements was not the

important points of our research. Accuracy was defined as

$$a = \frac{N_c}{N} \tag{9}$$

where N_c denotes the number of correct predictions, and N denotes the number of examples in test dataset.

5.2 Experiment results

5.2.1 General classifier versus cross-domain classifier

Because of the imbalance of the idiom dataset with 5611 negative instances and only 2549 positive instances, over-sampling techniques were adopted which specifically aimed to balance the class populations through replicating the minority class samples [30]. After over-sampling the positive data, a balanced training dataset of 11222 instances was achieved. For the cross-domain classifier, three supervised classifiers were trained separately on full dataset of each domain and tested on the other two domains. The results are shown in Table 1. The following results can be observed. Firstly, the accuracies of general classifier tested on three domains all surpass the Naïve baseline (50%), with the least gap (11.640%) of maximum entropy classifier tested on the hotel corpus and the largest gap (30.7%) of SVM classifier tested on the book corpus. The result proves that the general classifier is superior to random selection and may be better choice when there is no labeled dataset available for supervised or semi-supervised machine learning sentiment classification.

Secondly, as for three cross-domain classifiers, the performance of each classifier varies diversely, from the worst hotel SVM classifier tested on book corpus (17.5% below 50%) to best notebook Naïve Bayes classifier tested on hotel corpus (26.344% above 50%). The result shows inequality of cross-domain classifiers. Besides, labeled dataset is needed for training cross-domain classifier despite of dataset out of domain.

Finally, for comparison between two kinds of classifier, the general classifier outperforms the cross-domain classifier with 7 highest accuracies versus 2 highest accuracies marked boldly in Table 1. It is obvious enough that the average accuracy of general classifier is much higher than that of the cross-domain classifier, and more stable. In addition to the above observation, no laborious labeled data are needed for training the general classifier.

Above all, it is more robust and dependable for general classifier than cross-domain classifier.

5.2.2 Domain-specific versus self-training classifier

Three supervised domain-specific classifiers were trained in five-folded in cross-validation mode and the average accuracies were reported as the upper bound for unsupervised self-training classifier. And as illustrated in Table 1, the general SVM sentiment classifier showed the best performance of all three general classifiers. So, according to Algorithm 1, the SVM classifier was used to activate the bootstrapping procedure by outputting high-confidence instances to a self-training classifier. Table 2 illustrates the result.

From Table 2, it is provable of the result that the SVM classifier performs best of three domain-specific classifiers in accordance with conclusion of other researchers [4]. The improvement of performance from the general classifier to the self-training classifier is undoubtedly outstanding after the bootstrapping training procedure in each domain, with 5.5% improvement in book domain, 15.146% improvement in hotel domain and 17.65% improvement in notebook domain. But only in hotel domain does self-training classifier surpass all three domain-specific classifiers. In book and hotel domains, the self-training classifier also surpasses domain-specific Naïve Bayes and Maximum Entropy classifiers. In a whole, the performance of the self-training classifier based on general classifier approximates to the upper bound domain-specific classifier, which demonstrates the efficiency of the proposed framework.

Table 1 Results for general vs cross-domain classifier

Domain	Classifier	Book classifier	Hotel classifier	Notebook classifier	General classifier
Book	Naïve Bayes	—	48.425	59.350	**69.995**
	Maximum entropy	—	47.525	57.850	**63.675**
	Support vector machine	—	32.500	61.225	**80.700**
Hotel	Naïve Bayes	50.312	—	**76.344**	66.425
	Maximum entropy	50.837	—	**76.341**	61.640
	Support vector machine	43.685	—	59.665	**71.775**
Notebook	Naïve Bayes	50.100	62.675	—	**65.100**
	Maximum entropy	50.400	63.050	—	**70.500**
	Support vector machine	50.075	63.250	—	**63.850**

Table 2 Results for in-domain and self-training classifier

Domain	Naïve Bayes	Maximum entropy	Support vector machine	Self-training
Book	55.087	57.487	93.375	86.200
Hotel	72.407	77.910	87.479	87.521
Notebook	90.337	90.212	90.700	81.500

6 Discussion and future work

In the proposed unsupervised framework, a self-training classifier which takes the high-confidence predictions of the general classifier as input is trained in each domain to exploit domain-dependent part of features, and the performance is much like the upper bound domain-specific classifier. Obviously, the output quality of general classifier has a critical impact on the performance of the self-training classifier. Usually, the more accurate the prediction of the general classifier is, the better the resulting performance of self-training classifier is.

Therefore, further performance improvement of general classifiers is one of target future works, and there are some ways to do so. First of all, larger domain-independent language resources are needed to improve the coverage of features and reduce sparseness. Table 3 illustrates feature coverage among three domains and coverage between idioms dataset and each domain. It can be observed that the coverage of idiom features is relatively much less compared with three domains, so improvement of feature coverage would improve performance of general classifier. Sparseness is another factor that influences performance. According to the statistic, the paraphrase of each idiom is very short (19 characters of mean length) and the abandonment of words caused by segmenting error also causes sparseness.

Table 3 Statistic of feature coverage

Domain	Book	Hotel	Notebook	Idiom
Book (20219)	—	0.413 (8365)	0.243 (4933)	0.358 (7251)
Hotel (16220)	0.515 (8365)	—	0.307 (4994)	0.329 (5345)
Notebook (7595)	0.649 (4933)	0.657 (4994)	—	0.408 (3100)

From results of Table 3 and Table 1, it can be analyzed that, although some coverage among three domains is much larger than the idiom features, the performance of the cross-domain classification is worse than the general classifier. The reason lies in that the different distribution of common part of features in different domains will make cross-domain classifier unsuitable in another domain, while the distribution of common part between idiom features and other three domains features is independent of domains according to two assumptions discussed above.

In addition, only unigrams have been used to model the sentiment polarity and extracted to train the polarity classifiers, while many researches in machine learning sentiment classification suggest that more advanced linguistic modeling is likely to improve the performance of sentiment classification because complicated human sentiment could not be fully modeled by simple linguistic features. So, the next future work will focus on mining more useful linguistic features to improve performance of general classifier.

7 Conclusions

1) With the need of sentiment classification in various domains, domain dependency has become the bottleneck of machine learning techniques for the shortage of labeled domain dataset. The problem has been solved by a novel unsupervised learning framework on the off-the-shelf domain-independent resources.

2) A novel perspective of sentimental features is put forward with the assumption that feature space is divided into the general part and the specific part. Elicited by human's identification of sentiment polarity, a general classifier is proposed to make use of the general part features, which suggests that the general features play an important role in sentiment classification. The general classifiers can be obtained by training on linguistic resources independent of domains such as Chinese idioms resources.

3) The performance of general classifier can be improved by adopting bootstrapping procedure in the target domain to make use of the domain-specific part features. Thus, a self-training classifier is got by taking high-confidence predictions of the general classifier as initial input to an iterative training procedure. The performance of self-training classifier can approximate the upper bound supervised learning domain-specific classifier.

References

[1] MILLER M, SATHI C, WIESENTHAL D, LESKOVEC J, POTTS C. Sentiment flow through hyperlink networks [C]// Proceedings of the 5th International AAAI Conference on Weblogs and Social Media. Barcelona, Spain: Association for the Advancement of Artificial Intelligence, 2011: 550−553.

[2] PANG B, LEE L. Opinion mining and sentiment analysis [J]. Foundations and Trends in Information Retrieval, 2008, 2(1/2): 1−135.

[3] LIU B. Sentiment analysis and opinion mining [J]. Synthesis Lectures on Human Language Technologies, 2012, 5(1): 1−167.

[4] PANG B, LEE L, VAITHYANATHAN S. Thumbs up: Sentiment classification using machine learning techniques [C]// Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: Association for Computational Linguistics, 2002: 79−86.

[5] BOIY E, MOENS M. A machine learning approach to sentiment analysis in multilingual Web texts [J]. Information Retrieval, 2009, 12(5): 526−558.

[6] JIANG L, YU M, ZHOU M, LIU X, ZHAO T. Target-dependent Twitter sentiment classification [C]// Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Oregon, Portland: Association for Computational Linguistics, 2011: 151−160.

[7] MAAS A, DALY R, PHAM P, HUANG D, NG A, POTTS C. Learning word vectors for sentiment analysis [C]// Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011: 142−150.

[8] YESSENALINA A, YUE Y, CARDIE C. Multi-level structured models for document-level sentiment classification [C]// Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010: 1046−1056.

[9] YU N, KUBLER S. Semi-supervised learning for opinion detection [C]// Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on. Toronto, Canada: IEEE, 2010: 249−252.

[10] SINDHWANI V, MELVILLE P. Document-word co-regularization for semi-supervised sentiment analysis [C]// Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on. Pisa, Italy: IEEE, 2008: 1025−1030.

[11] TAN S, WU G, TANG H, CHENG X. A novel scheme for domain-transfer problem in the context of sentiment analysis [C]// Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management. New York, NY, USA: ACM, 2007: 979−982.

[12] PAN S J, NI X, SUN J T, YANG Q, CHEN Z. Cross-domain sentiment classification via spectral feature alignment [C]// Proceedings of the 19th International Conference on World Wide Web. New York, NY, USA: ACM, 2010: 751−760.

[13] LU B, TAN C, CARDIE C, TSOU K Y B. Joint bilingual sentiment classification with unlabeled parallel corpora [C]// Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011: 320−330.

[14] BRODY S, ELHADAD N. An unsupervised aspect-sentiment model for online reviews [C]// Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010: 804−812.

[15] TAN S, WANG Y, CHENG X. Combining learn-based and lexicon-based techniques for sentiment detection without using labeled examples [C]// Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, NY, USA: ACM, 2008: 743−744.

[16] LIU J, SENEFF S. Review sentiment scoring via a parse-and-paraphrase paradigm [C]// Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1. Stroudsburg, PA, USA: Association for Computational Linguistics, 2009: 161−169.

[17] BACCIANELLA S, ESULI A, SEBASTIANI F. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining [C]// Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10), Valletta, Malta. European Language Resources Association (ELRA). 2010: 417−422.

[18] STONE P J, DUNPHY D C, SMITH M S, OGILVIE D M. The general inquirer: A computer approach to content analysis [J]. Journal of Regional Science, 1968, 8(1): 113−116.

[19] DONG Z, DONG Q. HowNet and the computation of meaning [M]. Beijing: World Scientific, 2006: 53−56.

[20] KU L W, LIANG Y T, CHEN H H. Opinion extraction, summarization and tracking in news and blog corpora [C]// Proceedings of AAAI-2006 Spring Symposium on Computational Approaches to Analyzing Weblogs. Stanford, US: American Association for Artificial Intelligence, 2006: 568−575.

[21] LOU D, YAO T. Semantic polarity analysis and opinion mining on Chinese review sentences [J]. Journal of Computer Applications, 2006, 11: 30−45.

[22] SUN Y, WERNER V, ZHANG X. A robust feature extraction approach based on an auditory model for classification of speech and expressiveness [J]. Journal of Central South University, 2012, 19: 504−510.

[23] NIU D, WANG J, LIU J. Knowledge mining collaborative DESVM correction method in short-term load forecasting [J]. Journal of Central South University of Technology, 2011, 18: 1211−1216.

[24] TeacherCn. China Education Network [EB/OL]. [06/06/2012]. http://chengyu. teachercn.com (in Chinese).

[25] TAN Song-bo. ChnSentiCorp [EB/OL]. [06/06/2012]. http://www. searchforum.org.cn/tansongbo/corpus/(in Chinese).

[26] ZHANG Chi-yuan. Pymmseg-cpp [EB/OL]. [06/06/2012]. https://github.com/ pluskid/pymmseg-cpp.

[27] NLTK. Natural Language ToolKits [EB/OL]. [06/06/2012]. http://nltk.org.

[28] CHANG C C, LIN C J. LIBSVM: A library for support vector machines [J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2011, 2(3): 1−27.

[29] GALAVOTTI L, SEBASTIANI F, SIMI M. Experiments on the use of feature selection and negative evidence in automated text categorization [J]. Research and Advanced Technology for Digital Libraries, 2000, 1(3): 59−68.

[30] BLUM A, CHAWLA S. Learning from labeled and unlabeled data using graph mincuts [C]// Proceedings of the Eighteenth International Conference on Machine Learning. Morgan Kaufmann, San Francisco, CA, 2001: 19−26.

(Edited by YANG Bing)