A Pattern Matching Based Model for Implicit Opinion Question Identification

Hadi Amiri[a,b], **Zheng-Jun Zha**[b], **Tat-Seng Chua**[a,b]
[a]NUS Graduate School for Integrative Sciences and Engineering
[b]Department of Computer Science, National University of Singapore
{hadi,zhazj,chuats}@comp.nus.edu.sg

Abstract

This paper presents the results of developing subjectivity classifiers for *Implicit Opinion Question* (IOQ) identification. IOQs are defined as opinion questions with no opinion words. An IOQ example is "*will the U.S. government pay more attention to the Pacific Rim*?" Our analysis on community questions of Yahoo! Answers shows that a large proportion of opinion questions are IOQs. It is thus important to develop techniques to identify such questions. In this research, we first propose an effective framework based on mutual information and sequential pattern mining to construct an opinion lexicon that not only contains opinion words but also patterns. The discovered words and patterns are then combined with a machine learning technique to identify opinion questions. The experimental results on two datasets demonstrate the effectiveness of our approach.

Introduction

In this research we focus on opinion questions, the questions that look for subjective views about the topics of the questions. For example "*what is the **best** Korean BBQ restaurant in the Bay area*?" is an opinion question. Question answering (QA) systems benefit from identifying opinion questions in the sense that for opinion questions, a QA system should return a set of relevant answers with diverse opinions while for factual (objective) questions, e.g. "*which cat has the largest variety of prey*?", the QA system should only provide a unique answer (Stoyanov et al. 2006).

Previous research on opinion sentence or question identification typically resort to opinion lexicons and show that questions/sentences that contain strong opinion words are most likely opinion questions/sentences (Ku, Liang, and Chen 2007; Wiebe and Riloff 2005; Stoyanov et al. 2006; Wilson, Wiebe, and Hoffmann 2009). Similarly, we observed that, in the cQA data samples from *Travel* and *Family & Relationships* (F&R) domains, respectively 90% and 95% of the questions that contain at least one strong opinion word are opinion questions.

Despite the success of previous research, they usually suffer from the following two problems:

Q1: what can one do for 50th wedding anniversary?
Q2: what can you do to help environment?
Q3: my do is scared of me, what can I do?
Q4: what can US do to to grow faster economically?
Q5: what can the government do to help converse energy?

Table 1: IOQs that shares the pattern <{*what*}{*can*}{*do*}>

Limited Lexicon: Current opinion lexicons miss slang and so-called urban opinion words (e.g., *must-see* and *nerdy*) which are usually used in informal language and have high usage on the Internet. This problem poses a challenge to identifying opinion questions.

Implicit Opinion Question Identification: There exist opinion questions that do not contain any opinion words. We call such questions "*Implicit Opinion Questions*" (IOQs). Our observation shows that the number of IOQs is not negligible in the cQA data. For example, in our data sample from F&R and Travel domains, 78% and 30% of the questions are IOQs respectively. As such, it is necessary to develop techniques to identify such opinion questions.

To address the above problems, we propose to discover new opinion words from cQA questions to enrich the current opinion lexicon. We mine the new opinion words based on their contextual similarity to existing opinion words. Beyond opinion words, we discover opinion patterns to supplement the existing opinion lexicons. Two types of opinion patterns are defined and mined: *explicit* and *implicit* opinion patterns. The explicit opinion patterns are sequential patterns containing an opinion word, while implicit opinion patterns are the patterns that do not contain any opinion word but act as subjectivity indicators (e.g. <{*what*}{*to*}{*do*}{*in*}>, <{*do*}{*you*}{*think*}{*to*}>, <{*how*}{*ask*}{*out*}> are samples of such patterns in IOQa). Take the questions in Table 1 as examples, the repeated pattern <{*what*}{*can*}{*do*}> acts as a strong subjectivity indicator and thus is an implicit opinion pattern.

The main contributions of this paper are as follows:

- We propose an effective method to construct a cQA opinion lexicon that contains opinion words and patterns, and
- We show that the discovered words or patterns are effective in opinion question and IOQ identification.

The experimental results on two cQA datasets demonstrate the effectiveness of the opinion words and patterns. On average, the opinion words and patterns achieve an accuracy of 75.5% for opinion question identification while implicit opinion patterns achieve an accuracy of 79.0% for IOQ identification.

The rest of this paper is organized as follows. Section 2 and 3 elaborate our method for mining new opinion words and patterns, respectively. Section 4 explains our method for opinion question and IOQ identification. Section 5 reports the experimental setting and results. Section 6 reviews the related works and, finally, Section 7 concludes the paper.

Mining New Opinion Words

Current opinion lexicons, such as the subjectivity (Wilson, Wiebe, and Hoffmann 2005) and SentiWordNet (Esuli and Sebastiani 2006), do not contain slang and so-called urban opinion words. These words are usually used in user generated contents and are useful for subjectivity prediction.

In this section, we propose a new corpus-based approach to automatically mine new opinion words. Our assumption is that the words that can be substituted with an existing opinion word without changing the subjectivity orientation of the questions are opinion word. Two examples of new opinion words are "*itchy*" and "*nerdy*" that are *interchangeable* with the existing opinion word "*ugly*". To measure the interchangeability of words, a new statistical measurement is proposed in the next Section.

We consider the words that are labeled as strong in subjectivity lexicon or have objectivity score of zero in SentiWordNet as existing opinion words. We only consider *verbs*, *adjectives* and *adverbs* as they are the most dominant opinion-bearing word types. In addition, we incorporate the words "*should*" and "*opinion*" as they often appear in opinion questions and act as strong subjectivity indicators.

Interchangeability

We propose to find the interchangeable words that are distributionally similar with existing opinion words and consider them as candidate new opinion words. We define the interchangeability between two words as follows:

Definition 1: Two words are interchangeable, if they have:

- low co-occurrence (see Equation 2),
- high overlap in their left and right neighboring words.

Due to the intuitive definition of interchangeability, the co-occurrence between two interchangeable words is expected to be low. For example, since "*suggest*" and "*recommend*" are interchangeable, we usually use one of them in a sentence to give a suggestion. Furthermore, we here separately deal with the left and right neighboring words to discard the effect of the words that occur on the opposite sides of the target words in measuring their interchangeability.

Given two words w and v, we first define the side-oriented PMI between them as follows:

$$\begin{aligned} PMI^l(w,v) &= \\ &\log\left(\frac{Count(w \textit{ occur on left side of } v)M}{Count(w)Count(v)}\right) \\ PMI^r(w,v) &= \\ &\log\left(\frac{Count(w \textit{ occur on right side of } v)M}{Count(w)Count(v)}\right) \end{aligned} \tag{1}$$

where $Count(x)$ is the number of sentences that contain x and M is the total number of sentences in our dataset.

In addition, given the word w, we refer to its left (right) significant neighboring words (SNWs) as the words that (a) occur on the left (right) side of w, and (b) have positive PMI^l (PMI^r) values with respect to w. For each word, we only consider its top z left (right) SNWs that have the highest PMI^l (PMI^r) values with respect to the word.

Let v be a *seed* word and w be a given word. We define S^l_{vw} and S^r_{vw} as the common left and right SNWs of v and w respectively and compute the *context similarity* between the two words as follows:

$$\begin{aligned} Sim(v,w) = \tfrac{1}{z} \sum_{O\in\{l,r\}} \sum_{u \in S^O_{vw}} \\ [(PMI^O(v,u))^\zeta + (PMI^O(u,w))^\zeta] \end{aligned} \tag{2}$$

where O indicates left or right, u is a common (left or right) SNW of both v and w, and ζ is a constant. Equation 2 computes the similarity between two words by aggregating the PMI values of their common left and right SNWs. It assigns high similarity scores to the words that either (a) frequently co-occur, or (b) rarely co-occur but have high semantic association, such as "*recommend*" and "*suggest*". According to Definition 1, we are only interested in the latter case, so, we discard the words that frequently co-occur. For this purpose, we use side-oriented PMI as the measure of co-occurrence and compute the interchangeability score between two words as follows:

$$Int(v,w) = \frac{Sim(v,w)}{c + \sum\limits_{O\in\{l,r\}} PMI^O(v,w) + PMI^O(v,w)} \tag{3}$$

where c is a small constant. We first normalize the values into [0, 1] range. For each existing opinion word v, we construct the following interchangeability list in which the words are sorted in descending order of their interchangeability scores with respect to v:

$$Int(v) = \{w_1, ..., w_N\} \tag{4}$$

Each w_i in the list is considered as a new opinion word. We set the opinion weight of each w_i as:

$$weight(w_i) = Int(v', w_i) \tag{5}$$

where $v' = \arg\max_v Int(v, w_i)$ is the existing opinion word that has the highest interchangeability score with w_i.

Mining Opinion Patterns

As aforementioned, opinion patterns are important for (implicit) opinion question identification. Opinion patterns could be created manually or automatically. In contrast to hand-crafted patterns that are hard to design and have coverage problem, automated mining of sequential patterns (Han et al. 2001; Yan and Han 2002) has been shown to be effective in different text mining (Matsumoto, Takamura, and Okumura 2005; Jindal and Liu 2006a; 2006b) and cQA question analysis tasks (Wang and Chua 2010; Cong et al. 2008).

Problem Setup

We define a sequence $S =< s_1, ..., s_k >$ as an ordered list of items, s_i. A question can be represented as a sequence containing an ordered list of terms where each term is treated an item. Here is an example:

- **Question**: *what is the best BBQ restaurant in Seattle?*
- **Sequence**: <{*what*} {*is*} {*the*} {*best*} {*bbq*} {*restaurant*} {*in*} {*seattle*}>

With such representation, we first employ the PrefixSpan algorithm (Han et al. 2001) to mine sequential patterns from cQA questions. This algorithm has been reported to be efficient in discovering sequential patterns. It uses frequent items to recursively divide sequence databases into smaller databases and grows sub-sequences in each database.

In next sections, we introduce our opinion pattern extraction approaches tailored to different types of questions, respectively. In particular, we divide the cQA questions into following three groups:

- **G1**: includes all the questions that contain at least one existing opinion word,
- **G2**: includes questions that contain no existing opinion word but at least one new opinion word, and
- **G3**: includes the remaining questions, i.e., all the questions that do not contain any opinion words.

Mining Opinion Patterns from G1

The questions in G1 contain at least one existing opinion word. We can reliably consider them as opinion questions. We regard the patterns of G1 that contain an existing opinion word as explicit opinion patterns and set their opinion weight to 1. For the other patterns, let p be a pattern of G1 that does not contain any existing opinion word. We consider p as an implicit opinion pattern iff it frequently co-occurs with positive (negative) opinion words in questions, but rarely or never co-occur with negative (positive) opinion words. To detect such patterns, we compute an initial polarity score for each p as follows:

$$PlScore(p) = \sum_{x \in POS} PMI(p, x) - \sum_{y \in NEG} PMI(p, y) \tag{6}$$

where POS and NEG are the sets of positive and negative existing opinion words respectively. We normalize the polarity scores and consider the patterns with sufficiently high $|PlScore(.)|$ as implicit opinion patterns. We weight such patterns as follows:

$$weight(p) = \frac{N_{G1}^{P} + \sum_q N_{G1}^{q}}{T_{G1}} \tag{7}$$

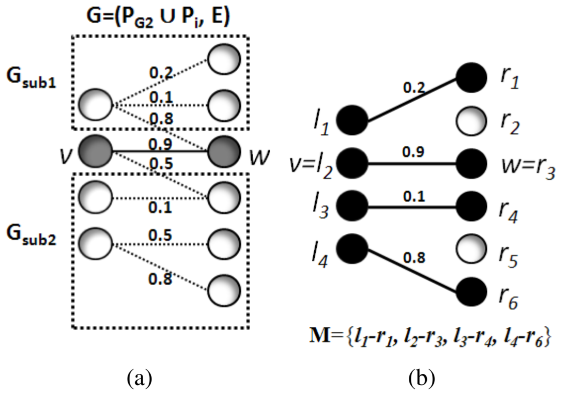


Figure 1: A sample bipartite graph and its corresponding match M.

where p is the selected implicit opinion pattern, q is any super-sequence of p that is different with p in exactly one existing opinion word. For instance, $p =<${*what*}{*to*}{*do*}$>$, and $q =<${*what*}{*fun*}{*to*}{*do*}$>$, with "*fun*" as its existing opinion word. $N_{G_1}^{p}$ and $N_{G_1}^{q}$ are the number of times that p and q occur in G1 respectively, and T_{G1} is the total number of questions in G1. We normalize these weights to [0, 1] range.

Equation 7 favors patterns that have high frequency in G1 or have high co-occurrence with opinion words. The term $\sum_q N_{G1}^{q}$ estimates how easily the pattern p can be converted to an explicit opinion pattern. For example, the pattern "$p =<${*what*}{*to*}{*do*}$>$" has high potential to occur with opinion words like "*fun*", "*cool*" and "*favorite*" and therefore obtains a high opinion weight.

Mining Opinion Patterns from G2

G2 includes the questions that do not contain any existing opinion word but at least one new opinion word. There exist some potential noises in the patterns of G2 due to the fact that not all the extracted new opinion words are indeed opinion-bearing words. Therefore, we don't treat all the patterns in G2 as opinion patterns.

In order to identify the opinion patterns in G2, we resort to the fact that each new opinion word w is interchangeable with an existing opinion word v. In particular, a pattern containing w is labeled as opinion pattern, iff it has high correspondence (similarity) with an opinion pattern containing v in G1. Otherwise, it is not an opinion pattern, and the word w is less likely an opinion word and should be removed from the opinion lexicon.

An effective way to compute the similarity between two patterns is to model them in a bipartite graph and find a matching in the graph that has the maximum weight. This approach has been shown to be effective in computing the similarity between pairs of sentences in machine translation (Chan and Ng 2008; Taskar, Lacoste-Julien, and Klein 2005).

Bipartite Graph Construction Assume that P_{G2} is a pattern from G2 that contains the new opinion word w. We can find a group of opinion patterns in G1, indicated by $\{P_1, ..., P_k\}$, that contain the existing opinion word v where w and v are interchangeable. To compute the similarity score between P_{G2} and each P_i, we model them in a bipartite graph as follows:

Let graph $G = (P_G \cup P_i, E)$ be a bipartite graph with $P_{G2} = \{l_1, ..., w, ..., l_n\}$ and $P_i = \{r_1, ..., v, ..., r_m\}$ as its partite sets and E as the weighted edges that connect the related vertexes of two partite sets (see Figure 1a). Each vertex l_i (r_j) corresponds to a term in P_{G2} (P_i). We weight the edge connecting each two vertexes (l_i, r_j) as follows:

$$w_{ij} = \begin{cases} 1 & \text{if } l_i \text{ and } r_j \text{ represent the same term} \\ Sim(l_i, r_j) & \text{otherwise} \end{cases} \tag{8}$$

where $Sim(l_i, r_j)$ is obtained by Equation (2).

Finding Maximum Weight Match We define a match M in G as a subset of the edges (E) in which no two edges share a common vertex. A match M that satisfies the following constraints best represents the maximum similarity between the two patterns:

- M has the maximum matching weight,
- M contains no crossing edge, and
- M includes the edge connecting the two nodes v and w.

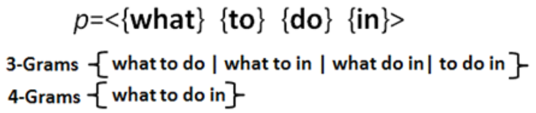
The match that has the maximum weigh indicates the maximum similarity between the two patterns. The crossing edges are also prohibited because we want to preserve the sequential order of the terms in the pattern matching process. Furthermore, v and w should correspond to each other in the final match. Figure 1b shows a match that satisfies the above constraints for the bipartite graph in Figure 1a.

Since the desired match should include the edge connecting the two nodes v and w, we divide G into two sub-graphs, namely G_1 and G_2 (see Figure 1a). We then adopt the algorithm proposed in (Kuhn 1955; Munkres 1957) to find the maximum weight bipartite match from each sub-graph. This algorithm requires that the bipartite graph to be fully connected and have equal number of nodes on both sides. We simply add dummy nodes and edges with weight 0 to the sub-graphs to satisfy this requirement. This algorithm may produce crossing edges in the generated match. In case of any crossing, we reset the weight of the edge that has the maximum crosses and the smallest weight in the sub-graph to zero and find the maximum bipartite match again. The above process is repeated until there is no crossing in the generated match. The final match of G is the combination of the maximum weight bipartite match of G_1 and G_2, and the edge that connect the words v and w.

Let M be a match in G that satisfies the above constraints. We assign an opinion weight to P_{G2} with regard to the match M as follows:

$$weight(P_{G2}, M) = \frac{2 \sum_{e \in M} W(e)}{n_G} \tag{9}$$

where $W(e)$ is the weight of the edge e in M, and n_G is the total number of nodes (including dummy nodes) in G. Note that $\frac{n_G}{2}$ is the greatest possible bipartite matching weight in G. The final opinion weight of P_{G2} is its maximum opinion weight with respect to all its matches with $\{P_1, ..., P_k\}$.



p=<{what} {to} {do} {in}>

3-Grams {what to do | what to in | what do in| to do in}

4-Grams {what to do in}

Figure 2: N-Gram features for a sample pattern

If the opinion weight of P_{G2} is greater than or equal a threshold, $\epsilon \in [0, 1]$, we label P_{G2} as an opinion pattern. The parameter ϵ controls the effect of precision and recall. Smaller ϵ values favor recall while greater ϵ values favor precision. We experimentally detect the optimal value of ϵ using our validation (held-out) data.

Mining Opinion and Factual Patterns from G3

G3 includes questions that do not contain any opinion words. These questions are implicit opinion questions or factual questions. Consequently, the frequent patterns in G3 are either implicit opinion or non-pinion patterns.

Here we employ the one-class SVM to discriminate the implicit opinion patterns from the non-opinion ones. The SVM classifier is trained based on the implicit opinion patterns extracted from G1, which are regarded as positive samples. We construct the feature set based on the n-gram substrings of the training patterns. For each training pattern, we extract all of its n-grams to construct the feature set where $n \geq 3$ is the length of the pattern. For example, the n-grams extracted from the pattern p =<{*what*} {*to*} {*do*} {*in*}> are shown in Figure 3.

Suppose the size of the resultant feature set is m. Given a pattern p, its m-dimensional feature vector is constructed as follows:

$$f_i^p = \begin{cases} \frac{|fea_i|}{\sum_{l=3}^{L}(l)} & \text{if } p \text{ contains } fea_i,\ i = 1...m \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where $|fea_i|$ is the length of i-th feature, and L is the length of p. The above Equation assigns higher weights to longer features since they are more precise than the shorter ones.

We use the learned one-class SVM to indentify the implicit opinion patterns in G3, and regard the remaining patterns of G3 as non-opinion patterns. We weight these patterns using Equation (11) and then normalize their weights to the [0, 1] range:

$$W(p) = \frac{L^p}{L * N_{G3}^p} T_{G3} \tag{11}$$

where L^p is the length of p, L is the maximum pattern length in G3, N_{G3}^p is the number of times that p occur in G3, and T_{G3} is the total number of questions in G3.

If an implicit opinion pattern in G3 is already mined from G1, its final weight is the average of its weights in G1 and G3. In addition, we give priority to G1 over G3, i.e., if a pattern is implicit opinion pattern in G1 and categorized as a non-opinion pattern in G3, then we consider it as an implicit opinion pattern.

Subjectivity Prediction

To predict the subjectivity of questions, we represent them by their opinion words, explicit and implicit opinion patterns, and non-opinion patterns. In particular, if a given question contains a particular feature, we set its weight to the feature's opinion weight; otherwise it is set to zero. Let the size of the resultant feature set to be h, given a question Q, its h-dimensional feature vector is constructed as follows:

$$f_i^Q = \begin{cases} weight(fea_i) & \text{if } Q \text{ contains } fea_i, \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

where $weight(fea_i)$ is the opinion weight of the ith feature obtained in Section 3. Wi this setting, we can employ any classifier to discriminate opinion from factual questions.

Experiments

Data and Experimental Setup

We used the Yahoo! Answers cQA archives from (Surdeanu, Ciaramita, and Zaragoza 2008) as the development dataset. This dataset includes more than 4M questions in different domains. We set the value of z in Equation (2) to the average question length in this dataset (i.e. $z = 6$) and ζ to 3 as suggested by (Islam and Inkpen 2008). For PrefixSpan algorithm, we limited the minimum and maximum pattern length to 3 and 6 respectively and set the minimum support for each pattern to 0.5% of the questions in each domain. Furthermore, we used LibSVM with linear kernel to conduct the one-class SVM learning. The linear kernel with a trade-off parameter of 0.8 is used as it is shown superior in our experiments.

Our approach is able to mine opinion entities including misspelled seeds (like *excellance* and *regretably*), new opinion words (like *fab* and *doomy*), and implicit opinion patterns (like <{*do*}{*you*}{*think*}{*to*}>, <{*how*}{*ask*}{*out*}> and <{*how*}{*to*}{*convince*}> in the F&R domain, and <{*where*}{*go*}{*vacation*}>, <{*what*}{*to*}{*do*}{*in*}> and <{*what*}{*must*}{*visit*}> in the Travel domain).

We used two datasets for evaluation. The first dataset (**D1**) contains 1000 questions as described in (Li, Liu, and Agichtein 2008). The questions are from *Art*, *Education*, *Health*, *Science*, and *Sport* categories. D1 comprises opinion and factual questions and is used as the benchmark for the opinion question identification task.

In order to evaluate our approach on IOQ identification, we developed a second dataset (**D2**) that only contains IOQ and factual questions from the two popular Yahoo! Answer categories: *Travel* and *Family & Relationship*. Two annotators labeled the data with an overall inter-annotator agreement of 79.0%. The *F&R* and *Travel* categories respectively contain 246 and 388 questions with half on IOQs and half on factual questions.

Li, Liu, and Agichtein (2008) showed that a Nave Bayes (NB) classifier (with term unigram (TU) as features and term frequency as feature weights) produces the best performance and outperform SVM for cQA opinion question identification on D1. We consider this configuration as the baseline in our experiments.

Table 2: Opinion question identification F1 results on D1

	TU	PKop	TopN	Pat	EopFact	IopFact	OpFact
Edu	68.1	48.2	64.2	67.6	69.7*	71.0*	**71.7***
Art	**74.5**	63.0	63.4	74.4	73.5	73.9	73.9
Health	63.6	49.4	51.8	64.9*	70.7*	**71.9***	**71.9***
Sport	75.2	69.1	69.5	82.0*	83.5*	82.5*	**83.7***
Science	63.1	49.3	60.4	65.3*	69.1*	**76.1***	**76.1***
Avg F1	68.9	55.8	61.9	70.8	73.3	75.1	**75.5**

Table 3: Values of N for TopN column and ϵ for EopFact column of Table 2

	N	ϵ
Edu	4	0.1
Art	3	0.4
Health	4	0.1
Sport	6	0.1
Science	4	0.2

All the experiments were performed through 10-fold cross validation and the two-tailed paired t-test ($p < 0.01$) is used for significance testing. In addition, for each domain, the parameter ϵ was set by cross validation.

Opinion Question Identification on D1

Table 2 shows the macro F1-score results of different configurations on D1. The *TU* column shows the baseline. The *PKop* column shows the results when we only use the existing (previously-known) opinion words as the features of the NB classifier. The *TopN* column shows the corresponding results by adding new opinion words to existing ones. *TopN* means that the new opinion words are the top *N* words extracted from the interchangeability lists as defined in Equation (4). For each domain, the value of N is computed based on a held-out validation subset (15% of data). Table 3 shows the values of $N \in \{1, ..., 10\}$ that produced the best classification result. The *Pat* column shows the results when we only use the patterns (explicit and implicit opinion and factual patterns) as the features of the NB classifier. The *EopFact* column shows the result when we combine opinion words with explicit opinion and factual patterns while the *IopFact* column indicates the corresponding results when we combine opinion words with implicit opinion and factual patterns. The *OpFact* column shows the results when we combine all the patterns with the opinion words. Table 3 shows the optimal ϵ values that result in the highest classification performance for the *EopFact* column in Table 2. We use the same setting for ϵ for *OpFact* results as well.

The results of *TopN* show that adding the new opinion words to existing ones improves the accuracy of *PKop* by about 6% on average. The results also show that the patterns are highly useful for opinion question identification. In all the domains, except *Art* and *Education*, *Pat* produces significantly better results than *TU*.

Also, as expected, adding opinion words to patterns, i.e. *EopFact*, *IopFact*, and *OpFact*, results in significant improvement over *TU* and *Pat* in all domains except *Art* in which *TU* produces slightly higher accuracy than others.

Table 4: IOQ identification F1 results on D2

	TU-NB	TU-SVM	IoF-NB	IoF-SVM
Travel	74.7	76.8	**81.4***	79.1
F&R	76.8	76.4	76.9	**77.9***
Avg F1	75.8	76.6	**79.0**	78.5

In addition, *EopFact*, *IopFact*, and *OpFact* significantly outperform *Pat* in *Education*, *Health* and *Science* domains; in the other two domains, *Sport* and *Art*, there is no significant difference between the above configurations and *Pat*. This indicates that, in the *Sport* and *Art* domains, the patterns alone convey all the information exist in the opinion words and are sufficient for opinion question identification in these two domains.

In addition, comparing the performance of *IopFact* and *EopFact* indicates that, the extracted implicit opinion patterns highly affect the performance of the opinion question identification. *EopFact* produces significant higher performance than *IopFact* in *Education*, *Health* and *Science* domains and improves the accuracy of *EopFact* by about 2% on average.

We should note that there is no significant difference between *IopFact* and *OpFact*. This indicates that the opinion words that are used as features in the *IopFact* bear most of the information conveyed by the explicit opinion patterns (that are absence in *IopFact*) and lead to close performance between *IopFact* and *OpFact*.

The average performance over all the domains shows that, patterns in conjunction with opinion words produce the best performance on the benchmark dataset and outperform *TU* in four domains. *TU* acts as a hard baseline in the *Art* domain and produces the best classification result in this domain.

IOQ Identification on D2

In this section, we evaluate the performance of our method for IOQ identification on D2. As we discussed before, we can only utilize the implicit opinion and factual patterns to identify IOQs. Table 4 shows the F1 results for *TU* and patterns using NB and SVM classifiers. *IoF-NB* (*IoF-SVM*) means the NB (SVM) classifier using implicit opinion pattern and factual patterns as features.

As Table 4 shows, *IoF-NB* and *IoF-SVM* produce the performance of 81.4% and 77.9% for the Travel and *F&R* domains respectively.

There is no significant difference between *TU-NB* and *TU-SVM* in both domains. However, the implicit opinion patterns (for both classifiers and domains) result in significant improvement over *TU*. The difference between *IoF-NB* and *TU-NB* in the *F&R* domain is interestingly significant. In the *Travel* domain, *IoF-NB* significantly outperforms *IoF-SVM*. In the *F&R* domain, *IoF-SVM* produces an accuracy of 77.9% and significantly outperforms others. The average accuracy over the two domains shows that, the implicit opinion and factual patterns produce the best performance on D2 and outperform *TU* for the IOQ identification task. On average, the NB classifier produces higher performance than SVM.

Related Work

Subjectivity analysis is a well studied field of research with wide variety of applications (Wiebe et al. 2004; Pang and Lee 2008; Liu 2010). Research in subjectivity analysis has been done at different level of granularity and from different linguistic and computational perspectives (Yu and Hatzivassiloglou 2003; Ng, Dasgupta, and Arifin 2006; Wiebe and Riloff 2005; Wilson, Wiebe, and Hoffmann 2009). Previous researches have shown that the opinion words are important features for opinion information identification.

In order to identify subjective sentences, Yu and Hatzivassiloglou (2003) used various features, including the number of opinion words, number of POS tags, and polarity, etc and reported a high accuracy of 91%. Wiebe and Riloff (2005) showed that a rule-based subjective classifier that categorizes each sentence with at least two strong opinion words as subjective can achieve 90.4% precision but a low recall of 34.2%. Some of the features used in their work are the count of weak/strong opinion words in current, previous, and next sentences, appearance of pronouns or modals and etc. The rule-based classifier idea is also employed by other researchers for subjectivity analysis (Stoyanov, Cardie, and Wiebe 2005; Riloff, Wiebe, and Phillips 2005; Wilson, Wiebe, and Hoffmann 2009).

Opinion words have been also used for opinion question identification. Ku et al. (2007) showed that the ”total number of opinion words in question” and the ”question type” (i.e. type of question in factual QA systems, e.g. Yes/No, Location, etc.) are the most effective features in opinion question identification. They reported a high accuracy of 92.50% over 1289 opinion questions. Despite the success of these works, they usually suffer from the limited lexicon problem, and are not able to identify the Implicit Opinion Questions (IOQ), since IOQs do not contain any opinion word.

Different from the above works, Li, Liu, and Agichtein; Li et al. (2008) utilized term unigram (TU) weighted by term frequency as feature of an NB classifier. They showed that this feature is a strong baseline for opinion question identification in cQA data and outperforms all the other features like characters, POS, N-grams, text of answers, etc.

Conclusion And Future Work

In this paper, we explored implicit opinion questions (IOQs). To identify IOQs, we first proposed to discover new opinion words and patterns to enrich the current opinion lexicons. We then utilized the resulting lexicon as classification features to identify opinion questions and IOQs. We found that mining new opinion words and patterns greatly improves the performance of subjectivity prediction in community questions. In the future, we continue our research to utilize the proposed method on wider range of user generated contents like reviews and micro-posts to predict subjectivity.

Acknowledgement

This research is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

References

Chan, Y. S., and Ng, H. T. 2008. Maxsim: A maximum similarity metric for machine translation evaluation. *Proceedings of ACL-08: HLT* 55–62.

Cong, G.; Wang, L.; Lin, C.-Y.; Song, Y.-I.; and Sun, Y. 2008. Finding question-answer pairs from online forums. In *Proceedings of ACM SIGIR conference on Research and development in information retrieval*.

Esuli, A., and Sebastiani, F. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of LREC*, volume 6, 417–422.

Han, J.; Pei, J.; Mortazavi-Asl, B.; Pinto, H.; Chen, Q.; Dayal, U.; and Hsu, M. 2001. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In *Proceedings of the 17th International Conference on Data Engineering*, 215–224.

Islam, A., and Inkpen, D. 2008. Semantic text similarity using corpus-based word similarity and string similarity. *ACM Trans. Knowl. Discov. Data* 2(2):10:1–10:25.

Jindal, N., and Liu, B. 2006a. Identifying comparative sentences in text documents. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 244–251. ACM.

Jindal, N., and Liu, B. 2006b. Mining comparative sentences and relations. In *Proceedings of the National Conference on Artificial Intelligence*, volume 21, 1331. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

Ku, L.-W.; Liang, Y.-T.; and Chen, H.-H. 2007. Question analysis and answer passage retrieval for opinion question answering systems. *ROCLING*.

Kuhn, H. W. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly* 2(1-2):83–97.

Li, B.; Liu, Y.; Ram, A.; Garcia, E. V.; and Agichtein, E. 2008. Exploring question subjectivity prediction in community qa. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*.

Li, B.; Liu, Y.; and Agichtein, E. 2008. Cocqa: co-training over questions and answers with an application to predicting question subjectivity orientation. In *Proceedings of the conference on empirical methods in natural language processing*.

Liu, B. 2010. Sentiment analysis and subjectivity. In *Handbook of Natural Language Processing, Second Edition. Taylor and Francis Group, Boca*.

Matsumoto, S.; Takamura, H.; and Okumura, M. 2005. Sentiment classification using word sub-sequences and dependency sub-trees. In *Advances in Knowledge Discovery and Data Mining*. Springer. 301–311.

Munkres, J. 1957. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial & Applied Mathematics* 5(1):32–38.

Ng, V.; Dasgupta, S.; and Arifin, S. 2006. Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews. In *Proceedings of the COLING/ACL on Main conference poster sessions*.

Pang, B., and Lee, L. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval* 2(1-2):1–135.

Riloff, E.; Wiebe, J.; and Phillips, W. 2005. Exploiting subjectivity classification to improve information extraction. In *Proceedings of the National Conference On Artificial Intelligence*, volume 20, 1106.

Stoyanov, V.; Cardie, C.; Litman, D.; and Wiebe, J. 2006. Evaluating an opinion annotation scheme using a new multi-perspective question and answer corpus. In *Computing Attitude and Affect in Text: Theory and Applications*. Springer. 77–91.

Stoyanov, V.; Cardie, C.; and Wiebe, J. 2005. Multi-perspective question answering using the opqa corpus. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*.

Surdeanu, M.; Ciaramita, M.; and Zaragoza, H. 2008. Learning to rank answers on large online qa collections. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL 2008)*, 719–727.

Taskar, B.; Lacoste-Julien, S.; and Klein, D. 2005. A discriminative matching approach to word alignment. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*.

Wang, K., and Chua, T.-S. 2010. Exploiting salient patterns for question detection and question retrieval in community-based question answering. In *Proceedings of the 23rd International Conference on Computational Linguistics*.

Wiebe, J., and Riloff, E. 2005. Creating subjective and objective sentence classifiers from unannotated texts. In *Computational Linguistics and Intelligent Text Processing*. Springer. 486–497.

Wiebe, J.; Wilson, T.; Bruce, R.; Bell, M.; and Martin, M. 2004. Learning subjective language. *Computational linguistics* 30(3):277–308.

Wilson, T.; Wiebe, J.; and Hoffmann, P. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*.

Wilson, T.; Wiebe, J.; and Hoffmann, P. 2009. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational linguistics* 35(3):399–433.

Yan, X., and Han, J. 2002. gspan: Graph-based substructure pattern mining. In *Proceedings of the 2002 IEEE International Conference on Data Mining*, ICDM '02. Washington, DC, USA: IEEE Computer Society.

Yu, H., and Hatzivassiloglou, V. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*.