An unsupervised approach to sentence classification

Shailesh S. Deshpande and Girish Keshav Palshikar and G. Athiappan

Tata Research Development and Design Centre
Tata Consultancy Services Limited
54B Hadapsar Industrial Estate, Pune 411013
India
{shailesh.deshpande, gk.palshikar, athiappan.g}@tcs.com

Abstract

Many real-life databases include columns containing textual data, which need to be processed to produce novel and actionable insights. We consider the novel problem of identifying *specific* sentences in the given text, since they tend to be actionable and novel. We demonstrate that specific sentences in the textual responses collected in an employee satisfaction survey and a product review are useful in understanding concrete issues, grievances and actionable suggestions for improvements. We propose an unsupervised approach for identifying specific sentences: We define and compute several features for each sentence and compute a *specificity score* for each sentence. Top k sentences in terms of this score are identified as specific. Some features are semantic in the sense that they use a supporting ontology (that of WordNet). We use the theory of speech acts to build an unsupervised (knowledge-based) classifier to further classify the identified specific sentences into *suggestion* and *complaint* classes. Our contributions here include treating suggestion and complaint as speech acts as well as the use of sentiment analysis in identifying speech acts. We demonstrate the utility of the proposed work in two real-life domains: employee surveys and product reviews.

1 Introduction

Many databases in real-life applications frequently include columns containing textual data. As an example, in this paper we consider real-life *survey response databases*, in which each row corresponds to the responses given by a particular person (e.g., an employee or a customer) to a set of questions in a questionnaire. Processing such textual data in columns to produce novel and actionable insights is a critical practical requirement. Text classification (also called as text categorization) (TC) is a well-known task in text-mining, which involves assigning a class label to an entire document. TC has wide-ranging applications (e.g., email spam detection); see [25] for a survey of TC techniques.

A closely related, but quite distinct, problem is that of assigning an appropriate class label to individual sentences, rather than to an entire document. Syntactically, a sentence is typically classified into classes such as DECLARATIVE, IMPERATIVE, INTERROGATIVE, EXCLAMATIVE, COMMUNICATIVE, INFORMATIVE etc., with further sub-classes. Other structurally-oriented sentence classes include MAJOR (has subject and predicate), MINOR (without a finite verb; e.g., `The more, the merrier.`), PERIODIC (meaning is not complete until the final clause or phrase; e.g., `Silent and soft, and slow, descends the snow.`) etc. Semantically classifying sentences (based on the sentence's purpose) is a much harder task, and is gaining increasing attention from linguists and NLP researchers [31], [27], [18], [2], [3], [12], [15], [28], [7], [8], [21], [14], [19]. Most work in this area has used supervised learning approaches (e.g., using SVM, decision trees, maximum entropy based classifier, naive Bayes etc.), with the exception of [11] (semi-supervised) and [26], [10] (knowledge-based). Sentence classification has been applied to tasks such as summarization, information extraction, IR, automatic ontology creation [9] and text entailment [29]. Sentence classification has been used on documents in several practical application domains such as biomedical papers, legal judgements, product reviews, customer complaints in helpdesk, emails etc. The sentences classes have also been more domain dependent (Table 1).

In this paper, we consider a more generic sentence classification problem, which is domain-independent. We consider the problem of automatically classifying a given (declarative) sentence as SPECIFIC or NOT-SPECIFIC (i.e., GENERAL). Intuitively, a SPECIFIC sentence is much more "on the ground" whereas a GENERAL sentence is much more "in the air". For example, the sentence `My table is cramped and hurts my knees.` is more SPECIFIC than a rather GENERAL sentence `The work environment needs improvement.` As

International Conference on Management of Data
COMAD 2010, Nagpur, India, December 8–10, 2010

another example, `Travel vouchers should be cleared within 2 working days.` is more SPECIFIC than `The accounts department is very inefficient.` Automatically identifying highly SPECIFIC sentences in a large text corpus (and ranking them in terms of their specificity) is very useful in many applications, particularly because SPECIFIC sentences tend to possess desirable properties such as actionability and novelty. For example, we demonstrate in this paper that SPECIFIC sentences in the textual responses collected in an employee satisfaction survey [22] tend to be very useful in understanding concrete issues, grievances and actionable suggestions for improvements etc. Similarly, SPECIFIC suggestions among the responses collected in a customer satisfaction survey (or a product reviews on a web-site) are useful in understanding problems faced in using the product, different use case scenarios for the product, product limitations, desired features, comparison with other products etc. Such understanding can be used to design an improvement plan that is more closely aligned to specific needs and problems of the stake-holders.

There are several standard ways to build such a sentence classifier. A supervised approach would learn decision rules from a labeled training corpus of sentences, where each sentence is manually assigned a class label (SPECIFIC or GENERAL) by an expert. This approach, while yielding accurate classifiers, has several well-known limitations: large amounts of time and efforts needed by experts in creating a labeled corpus; need for specialized domain knowledge; noise in the corpus; possibilities of class imbalance; disagreements among experts on assigned labels etc. As we will show, there is considerable disagreement about what specificity means among experts even in the same domain (HR or marketing domains, in our work). From our discussion, the primary reason for this phenomenon seems to be that the notion of what constitutes a SPECIFIC sentence is highly subjective and task-dependent. Since we found that labeled training data for SPECIFIC sentences often contains many disagreements among experts, we approach the problem of identifying SPECIFIC sentences in the following manner.

1. Assigning a binary class label appears to be an over-simplification in our application domains, due to expert disagreements. We develop a mechanism to compute a *specificity score* for each sentence. This scoring mechanism is unsupervised (knowledge-based), without the need for any labeled training examples. Briefly, we define a set of *features* and compute their values for each of the given sentences. The features are lexical and some are semantic. The features are context-free in the sense that their values are computed exclusively using the words in the given sentences and they do not depend on any other (e.g., previous) sentences. Then we combine the feature values for a particular sentence into its specificity score.

2. Sentences in the given set are then ranked (i.e., ordered) in terms of their specificity score.

3. The user can choose a suitable threshold so that sentences whose specificity is more (less) than the chosen threshold are SPECIFIC (GENERAL). Alternatively, the user can select top k sentences from this set of score-ordered sentences, for some k.

4. We further sub-classify specific sentences into classes SUGGESTION and COMPLAINT (some sentences may not belong to either class), which are more suitable for our application domains. We use linguistic knowledge (speech acts) to perform this classification in an unsupervised manner.

The paper is organized as follows. Section 2 contains a survey of related work. Section 3 discusses the details of our proposed approach to compute the specificity score which is useful to identify SPECIFIC sentences. Section 4 discusses an unsupervised approach to further sub-classify sentences into our domain-specific classes. Sections 5 and 6 discuss some applications of these techniques in analyzing (1) employee satisfaction survey responses; and (2) product reviews. Section 7 presents our conclusions and further work.

2 Related work

As mentioned earlier, most of the previous work in sentence classification is focused on domain-oriented sentence classes and uses supervised learning approaches. [28] and [18] used SVM to classify sentences occurring in abstracts of biomedical papers into classes indicating rhetorical status, as shown in Table 1. For the same task, [26] and [10] both used linguistic knowledge based on indicator phrases; e.g., `In this paper, we show that ...` usually indicates sentence of class AIM. Like these works, we have also used the unsupervised approach, though we use linguistic knowledge only indirectly through the use of ontologies. [15] also used SVM to classify sentences in help-desk emails and demonstrated the need for feature selection in supervised sentence classification. [7] used various classifiers including SVM and maximum entropy to classify sentences in legal documents into classes shown in Table 1. [2] used machine learning techniques (including SVM) to classify sentences in emails into classes defined by a Verb-Noun pair (see Table 1), such as REQUEST-MEETING and DELIVER-DATA. [3] performed a similar task with of classifying sentences in emails using SVM. [12] addresses a similar analysis for transcripts of Instant Messages for online shopping. [31] used Naive Bayes classifier to classify sentences in biographical articles into classes shown in Table 1. [19] used a language-based model with smoothing along with a

Table 1: Examples of Sentence Classes.

Ref.	Domain	Class labels
[28], [10] [26], [18]	research papers	BACKGROUND, TOPIC, RELATED-WORK, PURPOSE/PROBLEM, HYPOTHESIS, AIM, SOLUTION/METHOD, RESULT, CONCLUSION/CLAIM, FUTURE-WORK
[19]	movies	OPINIONATIVE, FACTOID
[27]	product reviews	RECOMMEND, NOT-RECOMMEND
[15]	help-desk	REQUEST, QUESTION, APOLOGY, INSTRUCTION, SUGGESTION, STATEMENT, SPECIFICATION, THANKS, APOLOGY, RESPONSE-ACK
[7]	legal	FACT, PROCEEDINGS, BACKGROUND, FRAMING, DISPOSAL
[2], [3]	emails	REQUEST, PROPOSE, AMEND, DELIVER, COMMIT for MEETING, DATA, TASK; CHITCHAT, FAREWELL
[31]	biography	BIO, FAME, PERSONALITY, SOCIAL, EDUCATION, NATIONALITY, SCANDAL, PERSONAL, WORK

Bayes classifier to classify sentences as OPINIONATIVE or FACTOID. This classification is close to ours, in the sense of being generic and domain-independent; however, their approach is supervised. [27] used a naive Bayes classifier along with knowledge-based post-processing to classify sentences in product reviews as RECOMMEND or NOT-RECOMMEND. Sentence classification has been used as a building block in text-mining tasks such as text entailment [29] and automated ontology discovery [9]. In [9], sentences likely to contain WordNet ontology relations were identified using linguistic knowledge; e.g., `Agar is a substance prepared from a mixture of red algae, such as Gelidium, for laboratory or industrial use.` contains both IS-A (is kind of) relation (`Agar` IS-A `red algae`) and IS-INSTANCE-OF (`Gelidium` IS-INSTANCE-OF `red algae`) relations.

In this paper, we have used the general-purpose ontology in WordNet [5], which is reasonably detailed for nouns but not for verbs and adjectives. We could have also used other general-purpose ontologies, such as Omega [23], Mikrokosmos [17] or Sensus [16]. Since some of these ontologies (e.g., Omega) explicitly address the limitations of the WordNet ontology, it would be interesting to compare the results obtained via using them, since the ontology plays a crucial role in our unsupervised approach to identify specific sentences.

Following [26] and [10], we have also used the theory of speech acts [24] to further classify SPECIFIC sentences into SUGGESTION and COMPLAINT. See [13] for a good review of this theory. We have used SentiWordNet [4] to check the polarity of words, which has been used to analyze sentiment contents of many types of documents including product reviews.

3 Identifying SPECIFIC sentences

Broadly, our approach to identify SPECIFIC sentences in given text consists of the following steps:

1. Pre-process the input text: spelling correction, cleanup, sentence boundary detection etc.
2. Identify and mark named entities in each sentence.
3. Identify POS tag for each word in each sentence.
4. Remove stopwords and words other than nouns, verbs, adjectives and adverbs. In different experiments, we remove adjectives and/or adverbs.
5. Convert each word to its root form; e.g., `tried` to `try`.
6. Convert each verb and adjective to the associated noun using PERTAINS-TO relation in WordNet; e.g., `financial` to `finance` and `tried` to `try`. This is done to compensate for the fact that WordNet ontology is shallow for verbs and non-existent for adjectives. Note that WordNet does not give any associated noun for some adjectives (e.g., `bad` or `handsome`). We exclude such adjectives from any further consideration. The correct form of the noun for a particular verb may also depend on its sense. We have developed a small logic that attempts to pick the right noun form even when the correct sense of the verb is not known.
7. Compute features for each sentence. Compute the specificity score for each sentence. Rank the sentences in terms of their specificity score and select top k.

3.1 Sentence Features

Average semantic depth $S.ASD$:

It is often possible to use a standard ontology T and *map* a given word w to a particular node u in T. In the simplest case, w is mapped to that node u in T which is either identical or synonymous to w. We use the standard hypernym (ISA) hierarchy in WordNet [5] as T; this hierarchy is fairly detailed for nouns but not for adjectives and verbs. Word u is a *hypernym* of another word v (or v is a *hyponym* of u) if v *is a kind of* u as per some common-sense knowledge; e.g., `fruit` is a hypernym of `apple` (equivalently, `apple` is a hyponym of `fruit`).

We propose to use the *semantic depth* or just *depth* (i.e., the distance in terms of number of edges from the root of T) of u, denoted $SD_T(u)$ or $SD(u)$ if T

is clear, as one measure of the specificity of any word w. The more the depth of w (i.e., the farther the node u in T to which w is mapped is from the root), the more specific is w and vice versa. The *average semantic depth* $S.ASD$ for a sentence $S =< w_1 w_2 \dots w_n >$ containing n content-carrying words (remaining after removing stop-words from the original sentence) is the average of the depths of the individual words:

$$S.ASD = \frac{\sum_{i=1}^{n} SD(w_i)}{n}$$

Optionally, we can assume a constant value (say 4) for the depth of personal pronouns (including `I, you, he, she, we, they, my, her` etc. but excluding `it`), since use of personal pronouns tends to be higher in more specific sentences. The depth of a word may change with its POS tag and with its sense for a given POS tag; e.g., $SD(bank) = 7$ for financial institution sense and $SD(bank) = 10$ for flight maneuver sense. If a word sense disambiguation (WSD) algorithm is applied to pre-process the input text, then the identified sense can be used. Otherwise, we can either use the depth of the word for its most commonly used sense or we can take an average of the depths of the word for all (or top k of) its senses.

As an example, assuming that the correct sense is identified for each content-carrying word (underlined), the average semantic depth of the sentence `My table hurts the knees.` is $(8 + 2 + 6)/3 = 5.3$ and that of `The work environment needs improvement.` is $(6 + 6 + 1 + 7)/4 = 5$. This example raises the possibility that the feature depth alone may not always be sufficient for deciding the relative specificity of sentences, as confirmed by the example given below. WordNet hypernym hierarchies for words `apple` and `fruit` in Fig. 1 show that the depths for `apple` and `fruit` are $SD(apple) = 7$ and $SD(fruit) = 8$ respectively. This is counterintuitive, since clearly `apple` is more specific than `fruit`. Thus the sentence `I like apples.` may get classified as more general than `I like fruits.` Such examples demonstrate a need for more features, which can help in overcoming this limitation.

Average semantic height $S.ASH$:

We analogously define another feature called the *semantic height* (or just *height*) of a word w, denoted $SH(w)$, as the length of the longest path from the word w to a leaf node *under* w in the given ontology T. Lower (higher) values of $SH(w)$ indicate that w is more specific (general). When using the WordNet, $SH(w)$ is the length of the longest path from w to a leaf node in the hyponym tree for w. Fig. 2 shows the WordNet hyponym tree for `apple` produced by the WordNet command `wn apple -treen`. Using this tree, $SH(apple) = 3$ because the longest path from `apple`

```
apple
=> edible fruit
=> produce, green goods, green groceries, garden truck
=> food, solid food
=> solid
=> matter
=> physical entity
=> entity

fruit
=> reproductive structure
=> plant organ
=> plant part, plant structure
=> natural object
=> whole, unit
=> object, physical object
=> physical entity
=> entity
```

Figure 1: WordNet hypernym trees for words `apple` and `fruit`.

```
apple
       => crab apple, crabapple
       => eating apple, dessert apple
            => Baldwin
            => Cortland
            => Delicious
                 => Golden Delicious, Yellow Delicious
                 => Red Delicious
            => Empire
            => Grimes' golden
            => Jonathan
            => McIntosh
                 => Macoun
            => Northern Spy
            => Pearmain
            => Pippin
                 => Cox's Orange Pippin
            => Prima
            => Stayman
            => Winesap
            => Stayman Winesap
            => Granny Smith
       => cooking apple
            => Bramley's Seedling
            => Lane's Prince Albert
            => Newtown Wonder
            => Rome Beauty
```

Figure 2: WordNet hyponym tree for the word `apple`.

to a leaf node has 3 edges (`apple` → `eating apple` → `Delicious` → `Golden Delicious`). In comparison, $SH(fruit) = 5$, thus indicating that `apple` is more specific than `fruit`. The average semantic height $S.ASH$ for a sentence $S =< w_1 w_2 \dots w_n >$ containing n content-carrying words (remaining after removing stop-words from the original sentence) is the average of the depths of the individual words:

$$S.ASH = \frac{\sum_{i=1}^{n} SH(w_i)}{n}$$

Optionally, we can also assume some constant value (say 3) for the height of personal pronouns (including `I, you, he, she, we, they, my, her` etc. but excluding `it`), since use of personal pronouns tends to be higher in more specific sentences. Again, the height of a word may change with its POS tag and with its sense for a given POS tag. If a WSD algorithm is applied to pre-process the input text, then the identified sense can be used. Otherwise, we can either use the height of the word for its most commonly used sense or we can take an average of the heights of the word for all its senses.

Total Occurrence Count $S.TOC$:

It is intuitively obvious that a more specific sentence will tend to include words which occur *rarely* either in the given corpus or in some standard (reference) corpus. More the number of *rare* words in a sentence, more specific it is likely to be. We could use measures like the well-known inverse document frequency (IDF), using which words which occur less in other documents can be considered as *rare*. Alternatively, WordNet provides the count (frequency) of the occurrences of words in its corpus; e.g., `apple` (2), `fruit` (14), `food` (34). Thus more specific words tend to have a lower occurrence count. However, this is not always the case: `produce` occurs only 2 times as a noun in WordNet corpus, even though it is more general than `fruit`; hence the need for other features to cover such limitations. Let $OC(w)$ denote the occurrence count of a word w in WordNet; if w has multiple senses, then $OC(w)$ is the average of the occurrence counts for top k senses of w (we use $k = 3$). Then *total occurrence count* $S.TOC$ for a sentence $S =< w_1 w_2 ... w_n >$ containing n words is the sum of the lowest m occurrence counts of the individual words, where m is a fixed value (e.g., we use $m = 3$). In effect, we order the occurrence counts of the words in the sentence (after removing stop-words) in ascending order and take a sum of the first m values. We assume the occurrence counts of 0 for any stop-words, including personal pronouns. Again, the occurrence count of a word may change with its POS tag and with its sense for a given POS tag. If a WSD algorithm is applied to pre-process the input text, then the identified sense can be used. Otherwise, we simply take the average of the occurrence counts of the given word for top k its senses, as we do for SD and SH.

Number of Named Entities $S.CNE$:

Named entities (NE) are commonly occurring groups of words which indicate specific semantic content, such as person name (e.g., `Bill Gates`), organization name (e.g., `Microsoft Corporation`), location (e.g., `New York`), date, time, amount, email addresses etc. Since each NE refers to a particular object, the presence of an NE is intuitively a good indicator that the sentence contains specific information. Hence another specificity indicating feature for a sentence S is the count of NE occurring in S, denoted $S.CNE$. Any standard NE tagger can be used to identify and count NE in the given sentence (we used the Stanford NER [6] for the experiments).

Number of Proper Nouns $S.CPN$:

Proper nouns - e.g., acronyms (`IBM`), domain terms (`oxidoreductases`) or words like `SQL Server` or `Apple iPhone`) and quantities such as numbers - have very specific information content. Presence of such words in a sentence often indicates that the sentence is likely to be more specific. Hence another specificity indicating feature for a sentence S is the count of proper nouns and numbers occurring in S, denoted $S.CPN$. This feature overlaps with the NE feature, since most NE are also tagged as proper nouns or numbers. Hence the NE feature can be made optional, particularly, to save processing time.

Sentence length $S.Len$:

Sentence length, denoted $S.Len$, is a weak indicator of its specificity in the sense that more specific sentences tend to be somewhat longer than more general sentences. Length refers to the number of content-carrying words (not stopwords) in the sentence, including numbers, proper nouns, adjectives and adverbs, even though these words are not considered for computing features such as ASD, ASH and TOC.

3.2 Specificity Score

We now combine the values of the features into a single specificity score. There are two issues to handle about the feature values before they can be combined. First, the features, as they are defined, have contradictory polarity. In general, we want each feature to have the property that higher values indicate more specificity. However, that is not true for features ASH and TOC; lower values indicate higher specificity for both these features. This can be fixed in several ways; the simplest way is to convert value x of a feature into $K - x$ where K is a fixed constant. For example, suppose the value of ASH for a sentence is $x = 4.5$. Then, assuming $K = 10$ for ASH, this value is converted to $10 - 4.5 = 5.5$; any value $x > 10$ is converted to 0.

Next, the scales of values for various features are not the same, because of which some features may unduly influence the overall combined score. For example, ASD is usually a positive number ≤ 10, whereas TOC might be a much larger integer. We map a number $x \in [a, b]$ to a number $y \in [c, d]$ using:

$$y = c + \frac{(d - c) \times (x - a)}{(b - a)}$$

When $x = a, y = c$ and when $x = b, y = d$. This simple uniform scaling mechanism is used to map each value (of a feature) to an integer between 0 to 10 ($c = 0, d = 10$). The upper and lower limit values a, b are suitably determined for each feature. If the polarity of the feature is to be reversed along with the scaling, then the formula is:

$$y = d - \frac{(d - c) \times (x - a)}{(b - a)}$$

Once all the feature values for each sentence have the same polarity and the same scale, we can combine

Table 2: Example sentences with content-carrying words underlined.

No.	Sentence
1	Experienced associates are forced to work on the projects or areas where they are not interested and of which they do not have any background.
2	I still struggle to find my feet in this town due to financial reasons.
3	An average associate knows the TCS vision very well, however he does not get to know the progress made against the set vision statement.
4	Compensation structure should be in line with the role and responsibility.
5	If I see something is wrong, most of the times I do not know whom to approach.
6	After coming to PNQ, I had to pay for the hotel initially for the first three days of my stay and was later reimbursed but to find a house and shift there I was left with only 5000 rupees which is in no way a handsome amount to settle down in PNQ.
7	Other facilities like Vending machine, drinking water, Air Conditioning are not proper.
8	It would be good if we could be provided radiation free monitors to reduce the stress.
9	Canteen at Rajashree gets overcrowded at peak hours.
10	We need to wait for 10 minutes for the lift.

Table 3: Features for the example sentences.

No.	ASD	ASH	TOC	CNE	CPN	Len	Specificity Score
1	4.22 (6.75)	8.06 (2.52)	9.49 (25.33)	0 (0)	0 (0)	2.67 (8)	24.44
2	4.53 (7.25)	8.81 (1.54)	6.34 (183)	0 (0)	0 (0)	1.67 (5)	21.35
3	4.97 (7.94)	8.21 (2.33)	9.49 (25.67)	2 (1)	1.67 (1)	2.33 (7)	28.66
4	4.17 (6.67)	7.82 (2.83)	9.19 (40.67)	0 (0)	0 (0)	1.67 (5)	22.84
5	5.21 (8.33)	8.21 (2.33)	9.73 (13.67)	0 (0)	0 (0)	0.33 (1)	23.47
6	5.09 (8.14)	8.18 (2.36)	9.82 (9)	2 (1)	6.67 (4)	6 (18)	37.76
7	4.48 (7.17)	8.08 (2.5)	9.78 (11)	0 (0)	5 (3)	2.33 (7)	29.67
8	5.17 (8.27)	8.31 (2.2)	9.59 (20.33)	0 (0)	0 (0)	2 (6)	25.07
9	3.96 (6.33)	8.78 (1.58)	9.16 (42)	2 (1)	3.33 (2)	1.67 (5)	28.9
10	4.27 (6.83)	8.89 (1.44)	9.33 (33.67)	0 (0)	1.67 (1)	1.33 (4)	25.49

them into a single *specificity score* by simply adding them together.

3.3 Example

Table 2 gives a set of 10 example sentences. Nouns and verbs in the sentence are underlined. Table 3 gives the values of all the features for each of these sentences (original feature values are in bracket and values scaled to $[0, 10]$ are shown outside the bracket). We illustrate the computations using the following sentence:

I still struggle to find my feet in this town due to financial reasons.

We convert the verb `struggle` to the corresponding noun (also `struggle`). The SD values for top 3 senses of the noun `struggle` are: 8, 7, 10 whose average is $SD(struggle) = 25/3 = 8.33$. Similarly, $SD(feet) = 6.67$, $SD(town) = 7.33$ and $SD(reasons) = 6.67$, so that $S.ASD = 7.25$. Assuming that ASD values fall in the range $[0, 16]$ ($a = 0, b = 16$) and the target range is $[0, 10]$ ($c = 0, d = 10$), the value 7.25 is scaled to 4.53. Similarly, the SH values for top 3 senses of the noun `struggle` are: 1, 3, 1 whose average is $SH(struggle) = 5/3 = 1.67$. Similarly, $SH(feet) = 2$, $SH(town) = 1.5$ and $SH(reasons) = 1$, so that $S.ASH = 1.54$. The polarity reversed and scaled value corresponding to 1.54 is 8.81. OC for top 3 senses of the noun `struggle` are 16, 11 and 1, whose average is $OC(struggle) = 28/3 = 9.33$. Similarly, $OC(feet) = 402$, $OC(town) = 136$ and $OC(reasons) = 37.7$. The sum of the lowest 3 OC values is $S.ASH = 9.33 + 37.67 + 136 = 183$. This sentence has no NE and no proper nouns/numbers, so that $S.CNE = 0, S.CPN = 0$. Length of this sentence is $S.Len = 5$ (adjective `financial` is included in the length computation). The last column of Table 2 shows the specificity score for each sentence; the score for this sentence is $4.53 + 8.81 + 6.34 + 1.67 = 21.35$. Top 4 sentences in terms of specificity score are: 6, 7, 9 and 3. If the user sets the threshold for specificity score at 28.0 then again the sentences 6, 7, 9 and 3 are identified as specific.

4 Classification of SPECIFIC Sentences

In our application case-studies (survey responses and product reviews, discussed later), the end-users wanted a further classification of the sentences identified as SPECIFIC into classes such as SUGGESTION and COMPLAINT (or PROBLEM). For example, sentences 9

Table 4: Rules for classification of sentences.

No.	Class	Rule	Example
1	C	{QUAL} neg-ADJP	`pathetic, poor, wrong, bad, limited, terrible, costly, late, limited, insulting, impossible, very inconvenient, too congested, highly uncomfortable, so uncooperative, simply bizarre, extremely slow, really small, degrading, rigid, overcrowded, unnecessary, unhappy, strenuous, less time, longer queue, bad quality, worse, worse product, worst, worst food`
2	C	NEG {QUAL} pos-ADJP	`no good, not available, not very friendly, not enough, not sure, hardly sufficient, not approachable`
3	C	MODAL-AUX NEG VERB	`should not claim, do not remain, won't get, does not work`
4	C	NEG NP	`no food, no choice, too much trouble, too many approvals, lack of courtesy, without consideration, tough time, no sense, no gifts, no park, no greenery, no transparency, no variety, no support, no water, any sense`
5	C	no|not VERB	`not like, not given, not considered, not expected, not understand, not work, not pleased, not made, not satisfied`
6	C	COMPLAINT-VERB	`complain, fail, stop, wait, miss, struggle, neglect, unable, face, shout, insult, degrade, ask, concern`
7	C	COMPLAINT-NOUN	`complaint, dissatisfaction, demotivation, trouble, problem, theft, negligence, nobody, no one, callousness, error, jerk inaccessibility`
8	C	neg-ADV	`shabbily, slowly, hastily`
9	C	CD times	`four times`
10	C	general	`why bother, I found, have seen, nobody cares`

and 10 in Table 2 are clearly of class COMPLAINT, while sentences 4 and 8 are of class SUGGESTION (this is only an example; some of these sentences were not chosen as SPECIFIC). Once again, we could use a supervised learning approach to train a classifier (such as decision tree or SVM) which learns decision rules by generalizing from labeled examples of these sentence classes. We have already discussed the well-known limitations of the supervised approach. In our applications, the high cost of creation of labeled training datasets (particularly for surveys in different types of organizations) and wide-spread expert disagreements over labels, required us to explore a linguistic knowledge based approach to perform this classification.

Following [26] and [10], we also use the theory of *speech acts* in linguistics to design a knowledge-based classifier to identify sentences belonging to these two classes. A speech act studies *illocutionary* sentences containing an *act* such as asking, promising, answering etc. Theory of speech acts further classifies illocutionary sentences into *assertive*, *directive*, *commissive*, *expressive* and *declarative*. A rich hierarchical mark-up language called DAMSL has been developed for tagging speech (and dialogue) acts in text. There are several approaches to automatically identify occurrences of various speech acts in given text. *Surface structure analysis* identifies simple speech acts such as YES-NO-QUESTION by syntactic analysis of the sentence. *Plan-inferential interpretation* attempts to analyze the semantics of the sentence (typically in first-order predicate logic) to infer its true purpose and thereby infer its speech act. Supervised learning techniques (e.g., decision tree and HMM) have been used to train classifiers to recognize speech acts by generalizing the labeled training examples. We follow the approach to speech act identification based on indicator (cue) phrases; [26] and [10] used the same approach to classify sentences. Our novel contribution is design of a knowledge-base for detecting occurrences of these new types of speech acts and the use of sentiment analysis. The key observation is that sentences in classes SUGGESTION and COMPLAINT are really two kinds of speech acts, clearly recognizable by cue phrases. This approach is simple, efficient and easy to extend for different domains. It is not the most accurate - it may occasionally miss some sentences. However, in applications such as survey response analysis, the input text may contain hundreds (or thousands) of suggestions or complaints, and missing a few may be acceptable.

Table 4 shows a few of the sample rules for the class COMPLAINT; rules for the class SUGGESTION are quite similar. neg-ADJP indicates an adjective phrase containing a negative polarity adjective, such as `bad`. pos-ADJP indicates an adjective phrase containing a positive polarity adjective, such as `good`. We use SentiWordNet [4] to check the polarity of words. neg-ADV indicates a negative polarity adverb like `shabbily`. NEG is a group of words indicating negation, such as `no, not` etc.; depending on the nature of the next phrase, NEG may also include words line `hardly, without, lack of, lacking, bad, worse, worst` etc. QUAL is a set of qualifiers, quantifiers and modifiers

such as `so, very, extremely, simply, highly, too, really, quite, at all, too much, too many, so few, so little`. In rule (14), !NEG pos-ADJP means that a positive adjective phrase must not be preceded by negation. Terms in curly braces (e.g., {QUAL}) are optional. Rules for the class SUGGESTION are quite similar to those for the class COMPLAINT. Some extra rules for SUGGESTION look for imperative speech acts, because suggestions are sometimes stated in an imperative manner (`Reintroduce reduced working hours.`). Note that the rules for these two classes are not mutually exclusive; e.g., `Need to wait too much for cutlery.` can get classified as both COMPLAINT and SUGGESTION. We have a simple scheme based on rule priority to resolve such ambiguities. Some sentences may genuinely contain both classes; e.g., `IT support is not good and needs to improve.` In such cases, the class label is chosen randomly. The dependence on word polarity can sometimes mislead: `less ventilation` indicates a complaint but `less congested` indicates a suggestion. Sentences without a verb can be missed; e.g., `The elevator facility at Nyati.` Sentence like `There are 4 lines of workstations in each wing.` is missed because it is a complaint about lack of privacy or congested work-space and a deeper semantic analysis is needed to classify it. Despite these limitations, we have found that the speech act based sentence classifier works quite well on the datasets in our application domains. Upon applying this classifier to all 10 sentences in Table 2 (we usually give SPECIFIC sentences as input), sentences 1, 3, 5, 6, 7, 9, 10 are identified as COMPLAINT and sentences 4 and 8 as SUGGESTION. Sentence 2 is not classified into either class.

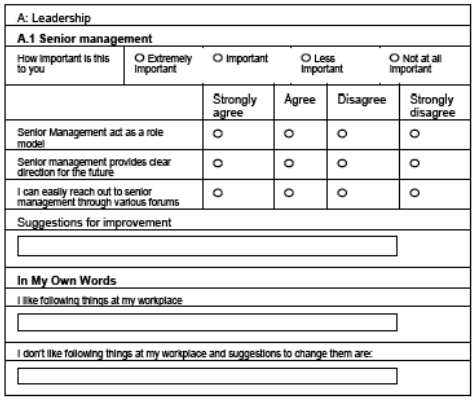
5 Case-Study: Survey Responses

5.1 Overview and Dataset

We present a real-life case study where the specific suggestion identification algorithm discussed in this paper has been successfully used (in part) to answer some business questions of the end-users. The client, a large software organization, values contributions made by its associates and gives paramount importance to their satisfaction. It launches an *employee satisfaction survey (ESS)* every year on its Intranet to collect feedback from its employees on various aspects of their work environment. The questionnaire contains a large number of questions of different types. Each *structured question* offers a few fixed options (called *domain of values*, assumed to be 0 to N for some N) to the respondent, who chooses one of them. *Unstructured questions* ask the respondent to provide a free-form natural language textual answer to the question without any restrictions. The questions cover many categories which include organizational functions such as human resources, work force allocation, compensation and benefits etc. as well as other aspects of the employees work environment. Fig. 3 shows sample questions. The ESS dataset consists of (a) the *response data*, in which each record includes an employee ID and the responses of that particular employee to all the questions, both structured and unstructured; and (b) the *employee data*, in which each record consists of employee information such as age, designation, gender, experience, location, department etc. ID and other employee data is masked to prevent identification. Some columns in the response data table contain textual data (answers to unstructured questions). An employee *satisfaction index (SI)* - a number between 0 and 100 - is computed (and stored in one column) for each employee using the responses of that employee to structured questions only; higher SI values indicate higher "happiness" levels.

5.2 Business Goals for Analysis

The goal is to analyze the ESS responses and get insights into employee feedback which can be used to improve various organization functions and other aspects of the work environment and thereby improve employee satisfaction. There are a large number of business questions that the HR managers want the analysis to answer; see [22] for a detailed discussion. Here, we focus on analyzing the responses to the unstructured questions, in particular identifying specific sentences from the textual responses to such questions. The overall aim of the analysis reported here is to answer two specific business questions: (1) Are there any subsets of employees, characterised by a common (shared) pattern, which are unusually unhappy? and (2) What are the root causes of the unhappiness for each such subset of employees? The answer to question (1) is clearly formulated in terms of interesting subsets. The problem of automatically discovering interesting subsets is well-known in the data mining community as *subgroup discovery*; see [1] for an overview. Each subset of employees (in employee data) can be characterised by a *selector* over the employee attributes; `DESIGNATION = 'ITA'` $\wedge$ `GENDER = 'Male'` is an example of a selector. A subset of employees, characterised by a selector, is an *interesting subset*, if the statistical characteristics of the SI values in this subset are very different (e.g., *significantly* lower or higher) from that of the remaining respondents. Thus we use the SI values (i.e., the column SI) as the measure for interesting subset discovery. If such an interesting subset is large and coherent enough, then one can try to reduce their unhappiness by means of specially designed targeted improvement programmes. We use the interesting subset discovery algorithms in [20] for discovering interesting subsets of unusually unhappy respondents. As an example, this algorithm discovered the following selector (among many others): `customer='X' AND designation='ASC'`. There are 29 employees in this sub-


A: Leadership

A.1 Senior management

How important is this to you | O Extremely Important | O Important | O Less Important | O Not at all Important

	Strongly agree	Agree	Disagree	Strongly disagree
Senior Management act as a role model	O	O	O	O
Senior management provides clear direction for the future	O	O	O	O
I can easily reach out to senior management through various forums	O	O	O	O

Suggestions for improvement

In My Own Words

I like following things at my workplace

I don't like following things at my workplace and suggestions to change them are:

Figure 3: Sample questions in the survey.

set. As another example, the algorithm discovered the following interesting subset `EXPERIENCE = '4_7'`; the average SI for this subset is 60.4 whereas the average SI for the entire set of all employees is 73.8.

5.3 SPECIFIC Suggestion Analysis

To answer question (2), we analyze the textual answers of the employees in each interesting subset to each unstructured question. We have selected a subset of 815 sentences from the responses to the question `Tell us what you don't like about Company XYZ`. While we perform much other analysis of these responses using text mining techniques like sentiment analysis, clustering and classification, following are some of the SPECIFIC sentences identified using the technique described in this paper.

```
1) Though it is not personal vendetta against
anyone, i would like to point out the fact that,
freshers and new people joining the XYZ account
feel very much humiliated and feel disinterested
to begin their work due to a very callous and
arrogant attitude of Ms.  P Q, Global ABC.
2) After coming to PNQ, I had to pay for the
hotel initially for the first three days of my
stay and was later reimbursed but to find a house
and shift there i was left with only 5000 rupees
which is in no way a handsome amount to settle
down in PNQ.
3) Process for installing softwares should be
shorter and access to CTMT facilities at MDS, PNQ
like jogging track, tennis court is denied.
4) Since Ya Pa is a huge complex, suggest
installing an ATM machine in the campus.
5) Either for 1 to 2 hours after office hours
Internet restrictions should not be there or 3
to 4 dedicated pc for internet access should be
placed in every office.
```

The identified SPECIFIC sentences are further partitioned into groups of similar sentences using text clustering techniques; we have modified and re-implemented the CLUTO algorithm [30] for this task. We also classify them into classes such as SUGGESTION and COMPLAINT using the techniques discussed later in this paper. The results of such analysis on textual responses to various unstructured questions is combined into a coherent *root-cause analysis*, which forms an answer to question (2).

5.4 Inter-rater Agreement

The selected set of 815 sentences were given to 3 experts, who labeled each sentence as either SPECIFIC or GENERAL. The pairwise inter-rater agreement computed using the Kappa statistic is: $\kappa(1,2) = 0.22, \kappa(1,3) = 0.20, \kappa(2,3) = 0.36$. Clearly, the agreement levels among the experts are rather low. We found similar (low) agreement levels for other datasets as well. From our discussions, the primary reason for this phenomenon seems to be that the notion of what constitutes a SPECIFIC sentence is highly subjective and task-dependent.

5.5 Further Experiments

The approach discussed so far can be called as *sense neutral* because we do not perform WSD on the input text. Since the correct sense of `struggle` is not available, we use its top k senses. For example, the SD values for top $k = 3$ senses of the noun `struggle` are: 8, 7, 10 whose average is $SD(struggle) = 25/3 = 8.33$. Similar computations are performed for features ASH and TOC. To study the effect of k on the quality of the results (i.e., specific sentences identified), we conducted several experiments in which we vary k as $k = 1$ (use only the most frequent sense of each word), $k = 3$ (use only the top 3 most frequent senses of each word) and $k = 30$ (use all senses of each word). For example, the top 5 SPECIFIC sentences from Table 2 identified with different values of k are as follows:
$k = 1$ {6, 7, 3, 9, 5}
$k = 3$ {6, 7, 9, 3, 10}
$k = 30$ {6, 7, 9, 3, 10}
There is considerable overlap among the results produced for different values of k.

In the example dataset of 815 sentences, we labeled those sentences as SPECIFIC where at least two experts agreed (*majority voting*). Fig. 4 shows the accuracy results (precision P, recall R and F-measure) obtained by comparing the specific sentences selected by our algorithm with the sentence labels in this training dataset. For these experiments, we varied k as 1, 3 and 30; also, we selected either top 5%, 10%, 15%, 20% or 25% sentences (ranked as per their scores) as SPECIFIC. As seen, the overall accuracy (F) does not vary significantly with k (for a fixed % threshold), thus demonstrating the sense-neutral nature of the proposed approach. That is, considering more senses does not improve the accuracy. The overlaps $\lambda(k_1, k_2)$ between the sentences identified as SPECIFIC by the algorithm

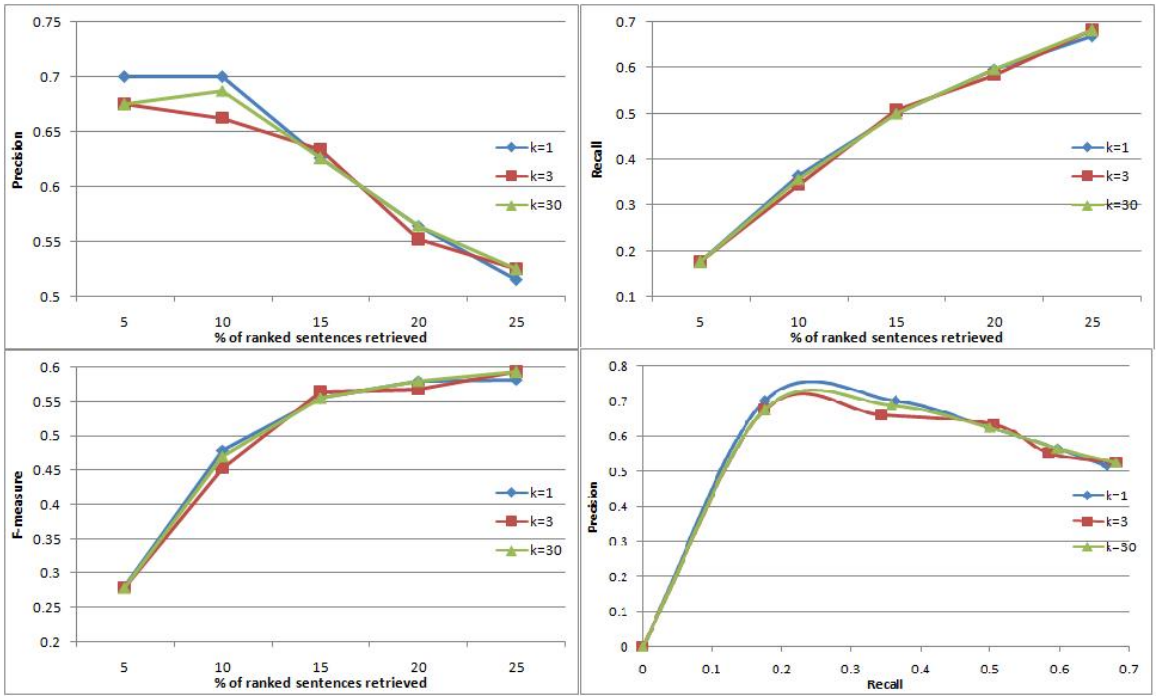


Figure 4: Precision, Recall and ROC curves.

for two values k_1 and k_2 of k (at 5% sentence threshold) are as follows: $\lambda(1,3) = 85\%$, $\lambda(1,30) = 85\%$ and $\lambda(3,30) = 100\%$. For example, 34 sentences were common among the two sets of 40 sentences (top 5% sentences out of 815) identified as specific by using $k = 1$ and $k = 3$, giving $\lambda(1,3) = 34/40 = 85\%$. This high overlap % again indicates that the proposed approach sense neutral.

6 Case-Study: Product Reviews

Many web-sites and blogs allow customers to post product reviews, which typically contain experiences, comments, feedback, complaints and suggestions by customers. Such reviews are a valuable source for improving quality and reach of the product. We consider a dataset containing 220 sentences from 32 reviews of a product by Kelty one of the leading manufacturers of outdoor gears (http://www.outdoorreview.com/cat/outdoor-equipment/backpacking-camping-hiking/internal-frame-backpacks/kelty/PRD_76963_2957crx.aspx). The product is Kelty Red Cloud, an internal frame sack built for recreational backpackers. The reviewers are a mix of backpackers, campers and people from other adventure sports. A review has common information such as reviewer's name, review date, overall rating, value rating, price paid etc. The review description has sections such as summary (wherein users experience with the backpack is described) and comments on customer service and information on similar products used. The average rating of this product is 4.16, with predominantly positive reviews. A sample review is a shown in Fig. 5. Following sentences (among others) were identified as SPECIFIC in this dataset.

```
1) Hiked 25 miles from North to South Rim of the
Grand Canyon, spent 4 nights in the canyon, the
pack worked great.
2) At times I've had to carry 60 to 70 lbs.loads
and the pack performed extremely well without
killing me; I took this pack to Europe for three
months and it did well.
3) i didn't really like the color though, red and
black with yellow straps and i think they should
have made the pick up strap a brighter color so
you can see it to grab it when you are taking the
pack off which is usually a fast motion.
4) I'd also prefer two ice ax loops as I've used
it for ice climbing and nobody carries one ax
unless mountaineering, in which case, it is often
in your hand.
5) Anyone looking for an inexpensive pack that
acts as a jack of-all trades would do well with a
Kelty Red Cloud.
```

7 Conclusions and Further Work

The crucial importance of analyzing textual data in columns in tables to produce novel and actionable insights is well-understood in many real-life applications. In this paper, we considered the problem of identifying *specific* sentences in the given text, since specific

Figure 5: A sample product review.

sentences tend to possess desirable properties such as actionability and novelty. We demonstrated that specific sentences in the textual responses collected in an employee satisfaction survey and a product review are useful in understanding concrete issues, grievances and actionable suggestions for improvements. We reported an experiment which shows considerable disagreement among experts on identifying specific sentences. Hence we proposed an unsupervised approach for identifying specific sentences. The approach defines and computes several features for each sentence and then computes a *specificity score* for each sentence. Top k sentences in terms of this score are identified as specific. Some features are semantic in the sense that they use a supporting ontology (that of WordNet). We use the theory of speech acts from linguistics to build an unsupervised (knowledge-based) classifier to further classify the identified specific sentences into *suggestion* and *complaint* classes. Our novel contribution here includes treating suggestion and complaint as speech acts as well as the use of sentiment analysis in identifying speech acts. We demonstrate the utility of the proposed work in two real-life domains: employee surveys and product reviews.

For further research, we are working on improving the quality of the results produced (i.e., specific sentences identified) by the technique described in this paper. In particular, we are exploring the possibility of including additional features based on information-theoretic measures. The reason for this idea is the observation that the specificity of a sentence is clearly related to the information content of a sentence. As of now, we compute the specificity score of all sentences, which is wasteful. To improve efficiency, we are exploring ways of eliminating sentences which are unlikely to be in the top-k set of specific sentences. Currently, the specificity score does not allow us to assign weights to individual features. A mechanism that automatically learns weights for various features might be useful. As mentioned earlier, WordNet generic ontology is rather shallow for verbs. We want to explore the possibility of using different generic ontologies. We would also like to use additional (e.g., domain-specific) ontologies, in addition to a generic ontology. We continue our experiments in applying the tool to other application domains (such as financial news and reports) to extract specific sentences. We are also looking at the generalization of the speech act based sentence classifier to cover other classes required for survey response analysis; e.g., further classifying the COMPLAINT sentences into sub-classes such as HR-COMPLAINT, FINANCE-COMPLAINT, PROJECT-COMPLAINT, PERSONAL-COMPLAINT, WORK-ENV-COMPLAINT, etc. (and a similar sub-classification for SUGGESTION).

Acknowledgements. The authors would like to thank Prof. Harrick Vin, Vice President and Chief Scientist, Tata Consultancy Services Ltd., for his support and encouragement. Thanks also to colleagues in TRDDC and TCS for their help. We thank the reviewers for their helpful comments.

References

[1] M. Atzmuller. *Knowledge Intensive Subgroup Mining: Techniques for Automatic and Interactive Discovery.* Aka Akademische Verlagsgsellschaft, 2007.

[2] W. Cohen, V. Carvalho, and T. Mitchell. Learning to classify email into "speech acts". In *Proc. Empirical Methods in Natural Language Processing (EMNLP-04)*, pages 309–316, 2004.

[3] S. Corston-Oliver, E. Ringger, M. Gamon, and R. Campbell. Task-focused summarization of email. In *Proc. ACL-04 Workshop on Text Summarization Branches Out*, pages 43–50, 2004.

[4] A. Esuli and F. Sebastiani. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of the LREC-2006 Conference*, 2006.

[5] C. Fellbaum. *WordNet: An On-line Lexical Database and Some of its Applications.* MIT Press, 1998.

[6] J. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proc. 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pages 363–370, 2005.

[7] B. Hachey and C. Grover. Sentence classification experiments for legal text summarisation. In *Proc. 17th Annual Conference on Legal Knowledge and Information Systems (Jurix-2004)*, pages 29–38, 2004.

[8] Q. He, K. Chang, and E.-P. Lim. Anticipatory event detection via sentence classification. In *Proc. IEEE Int. Conf. on Systems, Man and Cybernetics (SMC-06)*, pages 1143–1148, 2006.

[9] M. A. Hearst. Automated discovery of wordnet relations. In *in C. Fellbaum (Ed.), WordNet: An Electronic Lexical Database and Some of its Applications*. MIT Press, 1998.

[10] F. Ibekwe-SanJuan, S. Fernandez, E. SanJuan, and E. Charton. Annotation of scientific summaries for information retrieval. In *Proc. ESAIR-08*, 2008.

[11] T. Ito, M. Shimbo, T. Yamasaki, and Y. Matsumoto. Semi-supervised sentence classification for medline documents. *IEIC Technical Report*, 104(486):51–56, 2004.

[12] E. Ivanovic. Using dialogue acts to suggest responses in support services via instant messaging. In *Proc. 2006 Australasian Language Technology Workshop (ALTW2006)*, pages 159–160, 2006.

[13] D. Jurafsky and J. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*. Prentice-Hall, 2000.

[14] Y. Kadoya, K. Morita, M. Fuketa, M. Oono, E.-S. Atlam, T. Sumitomo, and J.-I. Aoe. A sentence classification technique using intention association expressions. *Int. J. Computer Mathematics*, 82(7):777–792, 2005.

[15] A. Khoo, Y. Marom, and D. Albrecht. Experiments with sentence classification. In *Proc. 2006 Australasian Language Technology Workshop (ALTW2006)*, pages 18–25, 2006.

[16] K. Knight and S. Luk. Building a large-scale knowledge base for machine translation. In *Proc. AAAI-94*, 1994.

[17] K. Mahesh and S. Nirenberg. A situated ontology for practical NLP. In *Proc. Workshop on Basic Ontological Issues in Knowledge Sharing at IJCAI-95*, 1995.

[18] L. McKnight and P. Srinivasan. Categorization of sentence types in medical abstracts. In *Proc. American Medical Informatics Association Annual Symposium*, pages 440–444, 2003.

[19] S. Momtazi and D. Klakow. Language model-based sentence classification for opinion question answering systems. In *Proc. International Multiconference on Computer Science and Information Technology*, pages 251–255, 2009.

[20] M. Natu and G. Palshikar. Discovering interesting subsets using statistical analysis. In G. Das, N. Sarda, and P. K. Reddy, editors, *Proc. 14th Int. Conf. on Management of Data (COMAD2008)*, pages 60–70. Allied Publishers, 2008.

[21] M. Naughton, N. Stokes, and J. Carthy. Sentence-level event classification in unstructured texts. *Information Retrieval*, 13(2):132–156, 2010.

[22] G. Palshikar, S. Deshpande, and S. Bhat. Quest: Discovering insights from survey responses. In *Proc. 8th Australasian Data Mining Conf. (AusDM09)*, pages 83–92, 2009.

[23] A. Philpot, M. Fleischman, and E. Hovy. Semi-automatic construction of a general purpose ontology. In *Proc. International Lisp Conference (ILC-03)*, 2003.

[24] J. Searle. *Speech Acts*. Cambridge University Press, 1969.

[25] F. Sebastiani. Machine learning and text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[26] S. Teufel and M. Moens. Sentence extraction and rhetorical classification for flexible abstracts. In *Proc. AAAI-98*, 1998.

[27] C. Wang, J. Lu, and G. Zhang. A semantic classification approach for online product reviews. In *Proc. IEEE/WIC/ACM International Conference on Web Intelligence*, pages 276–279, 2005.

[28] Y. Yamamoto and T. Takagi. Experiments with sentence classification: A sentence classification system for multi biomedical literature summarization. In *Proc. 21st International Conference on Data Engineering Workshops*, pages 1163–1168, 2005.

[29] F. Zanzotto and L. DellArciprete. Efficient kernels for sentence pair classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP-09)*, pages 91–100, 2009.

[30] Y. Zhao and G. Karypis. Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery*, 10(2):141–168, 2005.

[31] L. Zhou, M. Ticrea, and E. Hovy. Multi-document biography summarization. In *Proc. Empirical Methods in Natural Language Processing (EMNLP-04)*, pages 434–441, 2004.