8-6-2011

Suggestion Mining from Customer Reviews

Amar Viswanathan
Infosys Labs, Infosys Technologies Limited, Amar_Viswanathan@infosys.com

Prasanna Venkatesh
Infosys Labs, Infosys Technologies Limited, Prasanna_Venkatesh01@infosys.com

Bintu Vasudevan
Infosys Labs, Infosys Technologies Limited, Bintu_Vasudevan@infosys.com

Rajesh Balakrishnan
Infosys Labs, Infosys Technologies Limited, RajeshB@infosys.com

Lokendra Shastri
Infosys Labs, Infosys Technologies Limited, Lokendra_Shastri@infosys.com

Suggestion Mining from Customer Reviews

Amar Viswanathan
Infosys Labs,
Infosys Technologies Limited
Bangalore 560100
Amar_Viswanathan@infosys.com

Prasanna Venkatesh
Infosys Labs,
Infosys Technologies Limited
Bangalore 560100
Prasanna_Venkatesh01@infosys.com

Bintu Vasudevan
Infosys Labs,
Infosys Technologies Limited
Bangalore 560100
Bintu_Vasudevan@infosys.com

Rajesh Balakrishnan
Infosys Labs,
Infosys Technologies Limited
Bangalore 560100
RajeshB@infosys.com

Lokendra Shastri
Infosys Labs,
Infosys Technologies Limited
Bangalore 560100
Lokendra_Shastri@infosys.com

ABSTRACT

The increasing online content has influenced users' buying behavior. It has triggered a paradigm shift in marketing strategies, as the consumer is no longer swayed by marketers, instead relying on user comments for a particular product or service. This paper focuses on extracting information from feedbacks like suggestions and recommendation by the users that is often present along with the sentiment. While Sentiment Analysis looks at extraction of consumer sentiment, our focus is on extracting actionable feedback present in the text for use by different stakeholders like business analysts and the customer. Our focus is on mining the key suggestions present in text which would benefit the product developer. We present our results and observations in the paper.

Keywords

Suggestion mining, Text Analytics, Natural Language Processing, Ontology

INTRODUCTION

The availability of a huge dataset of customer reviews on the internet in different forms of social media such as blogs, tweets and product review forums has facilitated an exponential increase in the number of techniques used to mine the customer sentiments from such unstructured text. Sentiment Analysis or Opinion Mining commonly focus on extracting the polarity of products or individual features, which is expressed as either positive, negative or neutral; these techniques can be accomplished either through linguistic, probabilistic or statistical means. Natural Language Processing has enabled Sentiment Analysis to be implemented on a very large scale, providing a number of algorithms and AI based methodologies to process text from product reviews into more manageable units which can be worked on by machines. However, existing mining systems all share one common limitation. They work well only for those sentences where the customer's sentiment or opinion about the product under discussion is defined explicitly; implicit opinions, sarcasm, suggestions, Figure of speech and indirect references to individual entities are not been handled in such systems, and further exploration is needed to spot the underlying sentiment expressions in addition to the identifiable structures of the text. To illustrate the limitations of these systems, consider the example *"Nokia is the king of all mobiles"*. This sentence is an example of a metaphor – a Figure of speech which expresses one thing in terms of another thing in a different context. Here, the brand name '*Nokia*' is recognized, but there is no evident sentiment word or phrase that accompanies it in the sentence; therefore, the entire sentence is ignored by the system, although the human mind can interpret this line as a very strong case of positive. sentiment, arising from the fact that our cognitive capabilities can make out the importance of the word '*king*' with respect to the context in which the sentence is made. It is an uphill task to replicate such mental abilities in machines, since they are mainly designed to work with straight-forward expressions written in English. In the light of such difficulties, ontologies can play a crucial role in deconstructing complex sentences. (Gruber, Tom 1993) define ontology as a specification of a conceptualization in a manner that can be understood by both human and machines with ease. We utilize ontology to assign the associated properties of '*king'* to the brand '*Nokia'*, which in this case equates Nokia to be the top-most in the mobile domain in the same way we understand, a king to be top man of the kingdom. Therefore, it is possible to treat sentiment oriented expressions containing Figure of speech by using the inherent reasoning capabilities provided by ontologies, provided the expression constitutes a certain syntactic structure which defines it as a simile, metaphor or any other indirect form of speech. Figures of speech have long been one of the key areas of research in sentiment analysis, given their apparent lack of grammar structure and the various meanings that could be conveyed by them. Methods for identifying metaphors and application of ontologies to determine relationships between different class items have already been accomplished in research areas like linguistics and knowledge management. Consider the sentence *"I think Nokia should improve the sound quality of the Music Player"*. Existing techniques extract the manufacturer '*Nokia'*, product *'Music Player'* and feature *'sound quality'*, but the inherent suggestions *'improve sound quality'* present in the text is not conveyed. Our approach makes use of linguistic rules to identify and extract Figure of speech in the sentiment expression, and then obtain inferences from the extracted text with the help of ontology to arrive at the intended sentiment for the corresponding product. In addition, we examine the effectiveness of this approach and compare the results with existing sentiment analysis techniques.

RELATED WORK

A lot of methods and approaches are already in place for finding the polarity of individual products and their features. (Pang, Lee, 2008 and Haji et al., 2009) identified the key tasks of sentiment mining, namely extraction of features, derivation of feature sentiments and comparison with other features. At the document level, sentiment analysis typically involves classification based on overall polarity as an aggregation of the sentiments on related topics and parts of the document, while the sentiment on individual features can be accomplished through topic analysis, feature generalization or usage of parse trees. On the opinion mining front (Lee, Jeong, and Lee 2008) identifies two kinds of opinion mining systems- those which do not use linguistic resources like ReviewSeer (Dave et al., 2003) system based on 'thumbs up/down' and RedOpal which extracts features from customer reviews and assigns sentiments based on star ratings (Scaffidi et al., 2007), and linguistic based systems such as Opinion Observer (George A. Miller, 1990; Hu et al., 2004 and Liu et al., 2005) which presents extracted opinions use only adjectives as opinion words and assign prior polarity positive and negative sentiments in a graph format; WebFountain, which use base noun phrases to list sentiment bearing sentences for a given product features (Yi, Niblack, 2005) a high-precision sentiment analysis system (Hiroshi et al., 2004) which utilizes full parsing and top down tree matching, using a syntactic parser with matching patterns and polarity lexicons, to extract sentiments. OPINE, build based on (Popescu, Etzioni 2005), which applies PMI (Point-wise Mutual Information) method to extract features and syntactic parse trees to derive the corresponding opinion, presenting the results as a tuple feature, ranked opinion list. All the above papers mainly deal with opinion mining on particular product features in the customer review comments, but they work only if these opinions are explicitly expressed. We propose a mechanism to extract the inherent explicitly as well as implicit suggestions in the same dataset, which are ignored by current systems.

SUGGESTION MINING PROBLEM

A suggestion is a statement made by a person, usually as a word of advice that has a tendency to influence the choices and decisions of the listener. In the context of online product forum discussions, a suggestion made by a user indicates what one should know about something being discussed before an informed decision can be made by the reader. Suggestions can range from guiding people in choosing the product that suit their preferences, to the various features or characteristics of the product or service that need to be looked at to make full use of it. Moreover, a suggestion can also be a wish or an indirect request for additional features by the user, which is not communicated in a direct manner to the business; it can also be a form of recommendation for a particular product or service, which provides a big boost to the business' brand management. The presence of suggestions in opinionated text can trigger a noticeable change in the overall degree of polarity of the review. For instance, the statement *"Nokia could have included a flash with the camera"* implies a negative connotation with the camera, but it is rather implicit and does not actually indicate a negative opinion about the camera. On the other hand, a line like *"I suggest the Nokia 5130 for your expected budget"* contributes to a strong positive polarity. Therefore, a system that can capture user suggestions and *'wish lists'*, in addition to extracting the expressed sentiment about products, will enable the business to identify the areas of concerns expressed by the users and work on them for future releases. On an average, about twenty to thirty percent of the product reviews in consumer forums have been found to contain one or more suggestive statements in these, as well as a number of recommendations for particular products. For a very large dataset, running into hundreds of distinct reviews, the proportion of such feedback will be significantly huge, providing a valuable source of information to be mined. However, capturing these suggestions will be a challenge, given the known complexity of the English language, the ever-increasing occurrence of SMS lingo and slang, the presence of spelling and grammatical mistakes in reviews, which users never bother to correct, and various other factors like disjointed sentences or missing punctuation. This necessitates the application of effective natural language processing techniques to handle all kinds of sentences and to identify certain patterns in the given input text which fit the definition of suggestions, recommendations or wishes. The captured suggestion can be presented in a format that is easily assimilated by the user.

CORPUS STUDY

In the our study, we analyzed a large collection of user reviews for mobile phones from India's top review website www.mouthshut.com, with the goal of identifying the various ways in which user suggestions and inputs can be conveyed. Our findings revealed that every one out of five user reviews included some form of implicit user feedback, which is generally not construed as a sentiment bearing sentence by most opinion mining systems. Based on our observations, we have derived a set of patterns associated with user feedback, which can be converted into feedback rules for usage in a feedback mining application. There are three different ways in which these patterns can be identified usage of explicit keywords, the presence of queries and modal verbs. For our research, we use some basic terminologies. Entity refers to a product, a brand, a company, a location etc. '*Feature'* represents the features of the entity '*Product'*, and '*Attribute'* is the characteristic of the feature for which a suggestion is made.

Patterns with explicit keywords

Some of the user reviews surveyed during the course of our research were found to contain direct indicators of suggestions, based on the keywords like 'suggest', 'recommend' and 'go for'. Since these keywords could possibly be used in some context other than the product being reviewed, we zeroed in on those sentences which included a mention of the product or its associated features co-existing with that keyword in the same sentence. When a user suggests in his/her review that other users should buy a particular product, it implies that the business has met the requirements of a targeted customer base, thereby conveying a stronger degree of positivity towards the product. An example of a suggestion rule is as follows:

> Pattern: '*suggest' <Product> 'for' <Features>*
> Example: "I suggest you purchase the Nokia 5130 for looks, music and value for money"

Recommendations are similar to suggestions, but they carry a stronger positive meaning than the latter, and hence lend more weightage to the business' brand image. The frequency of the word '*recommend'* in product review forums is higher compared to other explicit keywords, and these reviews can be used by the business to identify the *'unofficial brand ambassadors'* of its product. One of the rules for identifying a sentence as a recommendation phrase is shown below.

> Pattern: '*recommend' <Product> 'for/because' <Features>*
> Example: "I recommend everyone to buy nokia 1100 because it is cheap, durable, functional, light, compact, good quality sound, nice design and value for money"

The presence of modifiers like '*strongly'*, '*highly'* or '*definitely'* along with these suggestion keywords serves to elevate the overall positive meaning associated with the product, and can be used as an indicator of customer satisfaction. The sentences *"I strongly suggest the 1100 as an entry level cheap handset"* and *"I highly recommend this phone if you have 5k in your pocket"* are examples of such instances. In addition, we consider phrases containing the keyword *'go for'*, which imply a form of suggestion or recommendation for a particular product. One such example is shown below.

> Pattern: '*if' <Features> 'go for' <Product>*
> Example: "If you are a music freak and have more money to spend, go for Nokia 5310 XpressMusic"

In some other cases where the user is satisfied with his/her purchase, but still '*wishes'* for new features or improvements over the existing features in the same product. A wish is an indirect form of suggestion which is aimed solely at the manufacturer of the product, and therefore merits the attention of the business. The following rule is one way of capturing user wishes.

> Pattern: *'I wish' <Product> 'had/contain/include' <Feature>*
> Example: "I wish Samsung Corby Pro included a trackball joystick."

Patterns containing queries

In some of the reviews which we analyzed in our dataset, we were able to identify certain instances where some user queries could be interpreted as an implicit form of suggestion to the business, given the context in which they were asked. With the right set of patterns at hand, we were able to single out these examples for processing. Below is an example of a query phrase.

> Pattern: '*Why' <Feature> in <Product>?*
> Example: "Uploading songs or downloading pics via cable is headache and time consuming. Why don't they give memory card reader with phone instead?"

Patterns containing modal verbs

Modal Verbs are a special class of auxiliary verbs (Jurafsky and Martin 2009), which place a condition on the verb form that follows them. The most common modal verbs are (*can, could, shall, should, may, might, must, will and would).* English speakers use modal verbs to express the mood of the verb such as (*suggestion, necessity, possibility, desire or request*). Modal Verbs have the property to change the overall meaning of a sentence, and their usage in online customer reviews can be extrapolated as implicit suggestions for or expectations from a particular product, provided the sentence in which they are present contain any of the patterns that we have derived as a consequence of our findings. The combination of modal verbs with different participles, like *'should have'*, *'could have been'* etc., induces varying degrees of suggestion strength; that is, they determine whether the suggestion made by the user in his review is just a mere observation *'could be'* or demands urgent notice *'must be'*. *'Should'* generally indicates a slightly strong suggestion or a form of advice or expectation, while *'shall'* relates to a weak suggestion. *'Can'* and *'could'* makes the phrase a request or a suggestion, depending on whom the sentence is intended for (business or other users). *'May'* and *'might'* indicates that the phrase is either a possibility or suggestion oriented one. *'Will'* and *'would'* attribute to the sentence being a request or a necessity, or a query as well; these convey a greater degree of suggestion than *'can'* and *'could'*.

> Pattern: <Product> *'should have'* <Feature>
> Example: "The E71 should have a 3.5 mm jack at least!" (Type: Suggestion), (Strength: Strong)
>
> Pattern: <Product> *'would'* <Feature>
> Example: "N 73 is a perfect phone for intermediate users who would like to use it for good camera with carl zeiss lens, good sound quality in both headphone and speakers" (Type: Recommendation), (Strength: Strong)
>
> Pattern: <Feature> *'could have been'* <Suggestion>
> Example: "Camera quality could have been better" (Type: Suggestion), (Strength: Medium)

SUGGESTION MINING APPROACH

In this section, we propose a system that segregates feedback phrases from the re-view dataset, processes them to extract the elements of the feedback and presents them in a suitable format to the user. This is accomplished through a combination of Natural Language Processing techniques, ontology and rule lookup from a knowledge repository and general inferencing methods. Figure 1, illustrates the functional architecture of the Suggestion Mining (SM) application. We describe each module of the SM in detail.

Knowledge Repository (KR): The KR comprises of a collection of lexicons, ontologies, feedback rules and other useful features like an SMS dictionary and a collection of slang words. The lexicons include domain-specific entities like brand or product names, as well as features and attributes of features in the same domain. The lexical information is added to the corpus either by a domain expert or through manual sifting of large quantities of data and adding words which meet the domain criteria, or through an aggregation of seed words. The ontology information present in KR encapsulates the relationship between different entities in the domain, such as which products come under which brand and other similar links. Feedback rules refer to a set of patterns which confirm the presence of a feedback type in the sentence under process. They are added to the repository based on an exhaustive analysis of large quantities of customer review data to identify the common expressions of customer feedback.

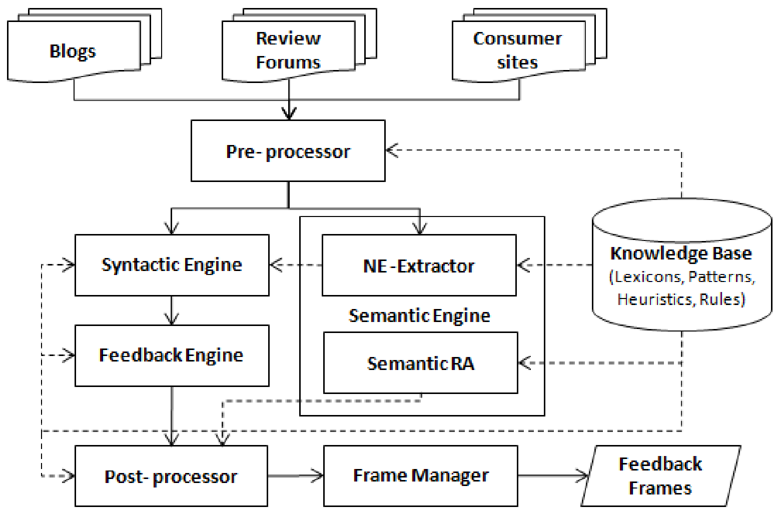

Figure 1. Functional architecture of the Suggestion Mining

Pre-processor (PreP): This module fine-tunes the input text for smoother manipulation in the subsequent modules. It takes input data from multiple sources of opinionated text such as blogs, product review forums or other media, and performs an array of diverse functionalities on it. These include spell checking to correct the common grammatical mistakes generally present in reviews, converting words in SMS lingo to the correct representation using the SMS dictionary present in the corpus as well as handling slang words, and splitting the rectified text into a set of sentences, which is then fed into the syntactic engine and semantic engine.

Syntactic Engine (SynE): The core of this engine is a statistical parser (Dan and Christopher 2003). The parser initially invokes a POS tagger to assign parts of speech to the tokens in the sentence. The Named Entity Extractor (NEE) in the Semantic Engine identifies (and tags) domain relevant entities and features and passes them to the parser as single tagged named entities with the POS tag for a noun phrase. (For example, in the mobile domain, the parser would treat the feature *'screen display'* as a single token rather than two separate tokens *'screen'* and *'display'*, tagging it with an 'NN' assignment). The parser also has the capability to interpret modal verbs in sentences, based on the 'MD' tag assigned to them; this would enable the Feedback Engine to detect feedback phrases, using the patterns described in section 4.3.

Semantic Engine (SemE): The NEE of SemE plays a role early on in the processing of sentences by the parser. It tags entities and features or attributes of features from preprocessed text through lexicalized lookup augmented with limited pattern matching. It alerts the POS tagger in the SynE to treat tagged named entities as a single token with a POS tag 'NN' (for noun phrase). It assigns semantic roles to them based on the mapping indicated by the Feedback Rules (FRs). In order to do so, it also checks that the 'potential filler' for a semantic role satisfies the requisite semantic type constraint (e.g., the entity should be a 'brand or product'). In addition, the SemE contains a lookup dictionary of suggestion phrases which are mapped to either of the primary suggestion actions (*improve, add, remove, modify, increase, decrease, choose*), which is invoked by the post-processor. This dictionary contains a list of all possible phrases which are synonymous with each of the generic suggestion actions. For example, the lookup dictionary for the general suggestion *'remove'* would comprise the following list of phrases:

[do away with; done away with; ditched; ditch; drop; dropped; discard; discarded; throw away; thrown out; bullied out; shooed away; butted out; dislodge; dislodged; dismiss; dismissed; carted away; carry away; carried; expel; exterminate; expunged; eject; ejected; remove; removed; taken; dump; dumped; taken down; detach; detached; isolate; extract; withdraw; withdrawn; eliminate; eliminated; separate; separated; polish; polished; wipe out; wiped out; get rid of; erase; erased; exclude; excluded; eradicate; eradicated; dispose; disposed; . . .]

We build a dictionary which captures most of the possible phrases which can be classified under a single action, and the post-processor module uses it determine the suggestion to which the suggestion phrase is mapped. Feedback Engine (FE): The FE provides the critical linkage between the syntactic structure of a sentence and its meaning. It does so by identifying the mapping between the syntactic constituents of a sentence and the roles of the semantic frame that constitutes the meaning of the sentence, and comparing the relationships with the FRs stored in the KR. If the sentence is found to fit the definition of a feedback phrase, it is passed on to the Post-processor.

Post-processor (PosP): This module processes to extract the relevant tokens and rearranges them as a feedback frame template (Brand, Product, Feature, Attribute, Type of feedback, Suggestion Phrase, Suggestion, Strength of feed-back). Here, ‘Type of feedback’ classifies the feedback sentence as a *‘suggestion’, ‘request’, ‘necessity’, ‘recommendation’, ‘demand’, ‘query’, ‘possibility’*, ‘Suggestion phrase’ refers to the sequence of words which identify the sentence as a suggestion-oriented one, ‘Suggestion’ indicates the inferred meaning of the suggestion phrase, and ‘Strength of feedback’ is the strength of the feedback type (Strong, medium, or low). The PosP extracts the Suggestion phrase by looking for mentions of explicit feedback keywords or queries. If they are not found, it then analyzes the parsed sentence for modal verbs, tagged as ‘MD’; if the ‘VP’ part of the sentence contains a ‘MD’ tag, the entire ‘VP’ part of the sentence is taken as the suggestion phrase. From the suggestion phrase, the PosP determines the generic suggestion present in the sentence, using the suggestion lookup feature present in the SemE. Therefore, a sentence like *“Nokia should dispose the cheap stylus for 5233”* returns the suggestion phrase *‘should dispose’*, from which the suggestion *‘remove’* is inferred based on the matching phrase present in the suggestion dictionary. Feedback strength is determined based on property of the modal verb or suggestion keyword present in the extracted suggestion phrase, using heuristics present in the KR. This process of mapping suggestion phrase to the generic suggestion can be done in a better way, using common sense reasoning techniques such as the one described for iSEE system (Shastri et al., 2010)

Frame Manager (FM): Once the PosP takes care of the feedback frame template creation, the FM is responsible for converting the template into a frame in a suitable format. It does so by generating unique frames which contain semantic labels mapped to all the items. If a product has different features for which different forms of feedback has been dug up, the manager generates a number of frames equal to the number of features, each containing the same format as the template but with different feature information.

ONTOLOGY AN KNOWLEDGE REPRESENTATION

We have modeled ontology based knowledge representation for our system, Techniques of automated reasoning allow a computer system to draw conclusions from knowledge represented in a machine-interpretable form as describe by (Stephan et al., 2007). Snippet of the KB used in our system as shown in Figure 2, Here mobile phone in general has a set of features like Camera, Music Player, <features> which in turn have their own set of attributes. These have been formed by taking into account the generic set of features of all mobile phones. The attributes of these features serve to enhance the inferencing capability of the engine. The snapshot shows one set of the features of a Mobile Phone. The music player has attributes like bass, loudness, which in turn has certain logical properties. When any review contains comments on an attribute, the suggestion mining system can, using this ontology figure out the associated feature. In addition the ontology also provides a means for identifying ‘good’ or ‘bad’ suggestions on the attribute based on sentiment lexicons. For example loudness has values of ‘high’ and ‘low’. Inferences can be made as to whether Loudness ‘high’ is considered as a positive suggestion or a negative suggestion.

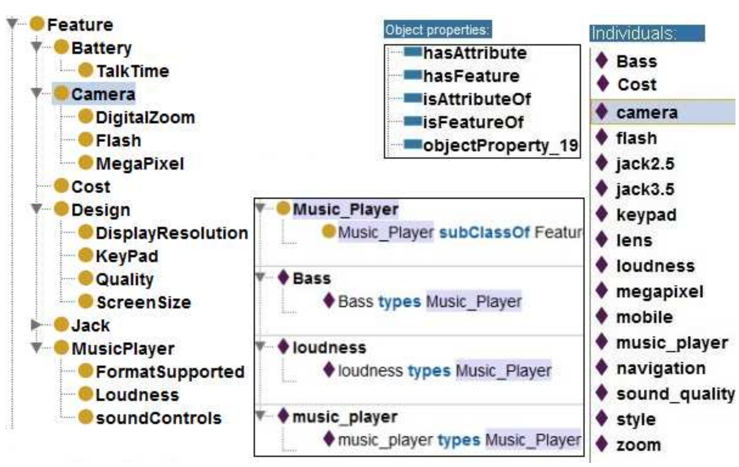

Figure 2. Snippet of Knowledge representation (Ontology)

EXPERIMENTAL WALKTHROUGH AND RESULTS

In this section, we examine the various stages of the SM system and the output from each stage.

Input text: "The design quality of 1100 is gud but it could be better. Perhaps the one button Navi key should have been done away with"

Pre-processor: The spell checker and sentence detector are applied to input text by the pre-processor; (i) The design quality of 1100 is ***good*** but it could be better. (ii) Perhaps the one button ***Navigation key*** should have been done away with.

NEE (Semantic Engine) + Syntactic Engine: Each sentence is converted into parsed text as shown in Figure 3. Semantic Engine: The module adds class information to the parse tree as shown in Figure 4.

```
(i)
  (S
    (NP
      (NP (DT The) (NN design) (NN quality))
      (PP (IN of)
        (NP (NN 1100))))
    (VP (VBZ is)
      (VP (VBN good)
        (SBAR (CC but)
          (S
            (NP (PRP it))
            (VP (MD could)
              (VP (VB be)
                (ADJP (JJR better)))))))
  )
```

```
(ii)
  (S
    (ADVP (RB Perhaps))
    (NP (DT the) (NN one button)
    (NN Navigation key))
        (VP (MD should)
          (VP (VB have)
            (VP (VBN been)
              (VP (VBN done)
                (PRT (RP away))
                (PP (IN with))))))
  )
```

Figure 3. Output of NEE (Semantic Engine) + Syntactic Engine

```
(i)
(S(NP(NP (DT The) (NN design/FE)(NN quality/ATR))(PP (IN of)(NP (NN 1100/PRD))))
(VP (VBZ is)(VP (VBN good)(SBAR (CC but)(S(NP (PRP it))(VP (MD could)(VP (VB be)
(ADJP (JJR better)))))))))
(ii)
(S(ADVP (RB Perhaps))(NP (DT the) (NN one button/ATR)(NN Navigation key/FE)
(VP (MD should)(VP (VB have)(VP (VBN been)(VP (VBN done)(PRT (RP away))
(PP (IN with)))))))
```

Figure 4. Output of Semantic Engine

Feedback Engine: Figure 5, shows the sentences which fit the FRs. (i) Confirms to the pattern <feature><attribute> of <product> could be <suggestion>, while (ii) can be derived as <attribute> <feature> should have been <suggestion>.

```
(i)  (NN design/FE), (NN quality/ATR)), (PP(IN of)(NP (NN 1100/PRD),
     (VP (MD could)(VP(VB be)(ADJP (JJR better))
(ii) (NN one button/ATR), (NN Navigation key/FE), (VP (MD should)(VP
     (VB have)(VP (VBN been)(VP(VBN done)(PRT (RP away))(PP(IN with))
```

Figure 5. Output of Feedback Engine

Post-processor: The feedback frame template for above sentence are (i) (Nokia, 1100, design, quality, (suggestion, *could be* better, **improve**, moderate). (ii) Nokia, 1100, Navigation key, one button, (suggestion, *should have been done away with,* **remove**, strong). The suggestion marked in bold is derived by applying the SemE RA on the suggestion phrases 'could be better' and 'should have been done away with'. The items marked in italics are filled in by the PosP based on ontology information in the KR, while the product name in the second sentence is obtained from analysis of the review topic.

Frame Manager: The Frame Manager translates the frame templates generated by post-processor into unique frames as shown in Figure 6.

Feedback : Frame 1	
Brand	Nokia
Product	1100
Feature	Design
Attribute	Quality
Type	Suggestion
Suggestion Phrase	Could be better
Suggestion	Improve
Strength	Moderate

Feedback : Frame 2	
Brand	Nokia
Product	1100
Feature	Navigation key
Attribute	One button
Type	Suggestion
Suggestion Phrase	Should have been done away with
Suggestion	Remove
Strength	Moderate

Figure 6. Suggestion Frames

Results: We chose 350 customer reviews posts in the mobile phone domain from our suggestion-mining corpus described in Section 4, and conducted cross validation on that dataset. The results are summarized in Table 1. The first three phrases are explicit forms of suggestions, while the last three are implicit forms. We evaluated the results by recall and precision.

SM Phrase	Occurrence	No of suggestions	suggestions retrieved	Precision	Recall
Suggest	10	8	7	0.87	0.70
Recommend	34	31	29	0.93	0.85
Go for	51	40	33	0.82	0.64
Should	35	27	18	0.66	0.52
Would	96	71	53	0.74	0.55
Can	227	138	72	0.52	0.31

Table 1. The precision and the recall for Suggestion Mining

CHALLENGES

Our proposed system is designed to take care of many forms of feedback, but there are still some instances where it falls short of capturing them. The presence of gures of speech, sarcastic comments and implicit references to domain rele- vant entities in reviews are too complex to be processed. Secondly, the system cannot handle some implicit suggestions, which require human intuition to grasp the meaning. The sentence “They should ensure that the phone doesn’t heat up quickly” talks about the feature battery, but it is ignored by our system. In the case of complex and long sentences, the suggestion pattern rules may not be identied or captured correctly, and therefore the application will treat them as non-suggestion bearing sentences.

CONCLUSION AND FUTURE WORK

In this paper, we have described the techniques for identifying and segregating mainly customer suggestions and recommendations from customer feedback in product reviews form online sources, and the fundamental architecture of the system to carry out these tasks. Using the proposed system, we were able to identify and collate various suggestion patterns from a large review dataset. The current version makes use of the SemE suggestion lookup dictionary to derive the general suggestion action for the extracted suggestion phrase in the post-processor. We can improvise the inferencing process by using Common Sense Reasoning (CSR) to infer the suggestion implicitly from the suggestive phrases, based on inference rules present in KR and class information in the SemE; the CSR module defined in iSEE (Shastri et al., 2010) can be extended to accomplish this. We can extend the scope of using this application to integrate it with existing sentiment analysis tools as an enhancement.

REFERENCES

1. Dave, K., Lawrence, S., Pencock, D.M., (2003) Mining the peanut gallery: Opinion ex- traction and semantic classification of product reviews. *In Proceedings of the 12th International Conf. World Wide Web (WWW'03),* 519-528
2. Haji Binali, Vidyasagar Potdar, Chen Wu, (2009) A state of the art opinion mining and its application domains. *In Proceedings of the IEEE International Conference on industrial Technology.*
3. George A. Miller, (1995) WordNet: *A Lexical Database for English Communications of the ACM* Vol. 38, 11, 39-41.
4. Gruber, T. R. (1993). Toward principles for the design of ontologies used for knowledge sharing *International Workshop on Formal Ontology, Padova, Italy. revised August.*
5. Hiroshi, K., Tetsuya, N., Hideo, W. (2004), Deeper sentiment analysis using machine translation technology. *In COLING '04: Proceedings of the 20th International Conf. on Computational Linguistics*, Morristown, *NJ, USA,* 494
6. Hu, M., B, B.L., (2004) Mining opinion features in customer reviews. *In: Proceedings of the 19th National Conference, on Arti_cial intelligence, The AAAI Press,* 755-760
7. Jurafsky, D., Martin, J.H., (2009) Speech and Language Processing: An Introduction to Natural Language Processing Computational Linguistics, and Speech Recognition. *Prentice Hall, Second Edition*
8. Lee, D., Jeong, O.R., Lee, S. G., (2008) Opinion mining of customer feedback data on the web. *In Proceedings of the 2nd International Conference on Ubiquitous information management and communication (ICUIMC'08),* 230-235
9. Liu, B., Hu, M., Cheng, J., (2005) Opinion observer: Analyzing and comparing opinions on the web. *In Proceedings of the 14th International Conference WWW'05*, 342-351
10. Dan Klein and Christopher D. Manning., (2003). Accurate Unlexicalized Parsing. *Proceedings of the 41st Meeting of the Association for Computational Linguistics*, pp. 423-430. NLP Parser. http://nlp.stanford.edu/software/lex-parser.shtml
11. Pang, B., and Lee, L., (2008) Opinion mining and sentiment analysis *Foundations and Trends in Information Retrieval* 2(1-2), 1-135
12. Popescu, A., Etzioni, O. (2005) Extracting product features and opinions from reviews. *In Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language*, 339-346
13. Scaffidi, C., Bierho_, K., Chang, E., Felker, M., Ng, H., Jin, C. (2007) Red opal: product feature scoring from reviews. *In Proceedings of the 8th ACM Conference on Electronic Commerce, (San Diego, California, USA),* 182-191
14. Shastri, L., Parvathy, A., Kumar, A., Wesley, J., Balakrishnan, R. (2010) Sentiment extraction: Integrating statistical parsing, semantic analysis, and common sense reasoning. *In Innovative App. of Artificial Intelligence (AAAI/IAAI).*
15. Stephan Grimm and Pascal Hitzler and Andreas Abecker (2007) Knowledge Representation and Ontologies , *Book*: *Semantic Web Services: Concepts, Technology and Applications,* 51-106
16. Yi, J., Niblack, W. (2005) Sentiment mining in web fountain. *In Proceedings* of the 21st *International Conference*. on *Data Engineering (ICDE'05), IEEE Computer Society,* 1073-1083