Named Entity Recognition and Aspect based Sentiment Analysis

Sangeeta Oswal
Assistant Professor
Department of MCA
Vivekanand Education Society
Institute of technology
Mumbai, India

Ravikumar Soni
Student
Department of MCA
Vivekanand Education Society
Institute of technology
Mumbai, India

Omkar Narvekar
Student
Department of MCA
Vivekanand Education Society
Institute of technology
Mumbai, India

Abhijit Pradha
Student
Department of MCA
Vivekanand Education Society Institute of Technology
Mumbai, India

ABSTRACT
Twitter is a microblogging website where people can share their feelings quickly and spontaneously by sending a tweets limited by 280 characters. You can directly address a tweet to someone by adding the target sign "@" or participate to a topic by adding an hashtag "#" to your tweet. Here specific hashtag (#) based tweets are downloaded using tweepy and they are cleansed for removal of irrelevant data then Entity Recognition is performed using the NER Algorithm which specifies different entities belonging to that tweet eg person, place, organization, etc. and finally sentiment analysis is performed where we analyze the general sentiment that can either be positive, negative or neutral at the entity level.

General Terms
Named entity recognition, Sentimental analysis.

Keywords
Entity Recognition, Tweepy, Vadersentiment, #Mumbaiband.

1. INTRODUCTION
Because of the usage of Twitter, it is a perfect source of data to determine the current overall opinion about anything.It is a rapidly expanding service with over 200 million registered users out of which 326 million are active users and half of them log on twitter on a daily basis - generating nearly 500 million tweets per day. Due to this large amount of usage achieving a reflection of public sentiment by analysing the sentiments expressed in the tweets. is possible. Analysing the public sentiment is important for many applications such as firms trying to find out the response of their products in the market, predicting political elections and predicting socioeconomic phenomena like stock exchange.Although other sentiment analysis researches has been done but mostly all of them are word based classification. The aim of this study is to develop a functional classifier for accurate and automatic sentiment classification of an unknown tweet stream based on aspect of the tweets selected by the machine learning algorithm.

2. LITERATURE

2.1 Named entity recognition
Named entity recognition (NER) is a process of extracting information related to identifying a set of specific named entities from a group of unstructured texts and classify them into predefined categories such as names of persons, organizations, locations, monetary values, etc.Suppose we take a block of text such as Ravi applied for an internship program in Crisil Limited, Mumbai.

Using the above text we can classify names of entities

Ravi = Person

Crisil Limited = Organization

Mumbai = Location.

2.2 Sentimental analysis
Sentiment Analysis also called as opinion mining is a sub task of machine learning specific in research of text mining. Sentiment Analysis uses natural language processing, text analysis to identify people's opinions, attitudes and sentiments towards an entity. Sentiment Analysis is generally done on reviews, surveys social media posts, etc. Using sentiment analysis a user can extract subject information and classify it as positive, negative or neutral.

3. METHOD USED

3.1 Tweepy
Twitter can be a great source of text data and a primary source of data for this research. To extract required textual data from twitter we need tweepy. Tweepy is an open source Python library used to access Twitter API to extract data. The data extracted here for mining are tweets from twitter. Tweepy accesses twitter data via OAuth (Open Authorization). OAuth is an open standard for authentication and authorization using tokens over the internet. OAuth is used in getting end users information without revealing their password. Twitter developer section will provide the authentication access key which can then be used to extract data. Tweepy will download home timeline tweets and print each of them on the console. When invoked an API method a tweepy model class will return which will contain data returned from twitter.

```
import tweepy

auth = tweepy.OAuthHandler(consumer_key, consumer_secret)

auth.set_access_token(access_token, access_token_secret)

API = tweepy.API(auth)

public_tweets = api.home_timeline()

for tweet in public_tweets:

print tweet.text
```

3.2 Natural language toolkit (NLTK)

The natural language Toolkit is a set of NLP libraries and programs which contains packages which help machines to understand human language and when understood reply with an appropriate response. It was developed by Steven Bird and Edward Loper in the Department of Computer and Information Science at the University of Pennsylvania. NLTK acts as a platform which works with python programs that applies statistical natural language processing. Text processing libraries for tokenization, parsing, classification, stemming, tagging, semantic reasoning Punctuation, character count, word count are used in NLTK. It also includes graphical demonstrations of data.

3.3 Valence aware dictionary for sentiment reading (VADER)

The nature of social media content poses serious challenges to practical applications of sentiment analysis.VADER, a simple rule-based model for general sentiment analysis, and compare its effectiveness to eleven typical state-of-practice benchmarks Using a combination of qualitative and quantitative methods, we first construct and empirically validate a gold standard list of lexical features (along with their associated sentiment intensity measures) which are specifically attuned to sentiment in microblog-like contexts. We then combine these lexical features with consideration for five general rules for expressing and emphasizing sentiment intensity. Using parsimonious rule-based model, VADER outperforms individual human raters.

The VADER lexicon performs exceptionally well in the social media domain. The correlation coefficient shows that VADER (r = 0.881) performs as well as individual human raters (r = 0.888) at matching ground truth . Upon further inspecting the classification accuracy, we see that VADER (F1 = 0.96) actually even outperforms individual human raters (F1 = 0.84) at correctly classifying the sentiment of tweets into positive, neutral, or negative classes. VADER retains (and even improves on) the benefits of traditional sentiment lexicons like LIWC: it is bigger, yet just as simply inspected, understood, quickly applied (without a need for extensive learning/training) and easily extended. Like LIWC (but unlike some other lexicons or machine learning models), the VADER sentiment lexicon is gold-standard quality and has been validated by humans. as compared to LIWC, VADER is more sensitive to sentiment expressions in social media contexts while also generalizing more favourably to other domains

Once VADER is imported, the sentiment for a list of sentences can be found using the vaderSentiment() method:

```
FromvaderSentiment import sentiment as vaderSentiment

sentences = [
 "The plot was good, but the characters are uncompelling and the dialog is not great.",

"A really bad, horrible book.",

"At least it isn't a horrible book."]

for sentence in sentences:

        print sentence,

        sentiment = vader Sentiment(sentence)

        print "\n\t" + str(sentiment)
```

The vaderSentiment() method returns the values to represent the amount of negative, positive, and neutral sentiment and also works out the compound sentiment value as a signed value to indicate overall sentiment polarity.[5] The output for the code snippet above:

The plot was good, but the characters are uncompelling and the dialog is not great.
{'neg': 0.327, 'neu': 0.579, 'pos': 0.094, 'compound': -0.7042}

A really bad, horrible book.

{'neg': 0.791, 'neu': 0.209, 'pos': 0.0, 'compound': -0.8211}

At least it isn't a horrible book.

{'neg': 0.0, 'neu': 0.637, 'pos': 0.363, 'compound': 0.431}

3.4 Orange

Orange is a machine learning suite for data analysis using Python scripting and visual programming. We report on the scripting part, which features interactive data analysis and component-based assembly of data mining procedures.While selection and design of components, we focus on the flexibility of their reuse: our principal intention is to let the user write simple and clear scripts in Python. Orange is used by experienced users, programmers, and by students of data mining.

Orange library is a hierarchically-organized toolbox of data mining components. Data filtering, probability assessment and feature scoring, are the low-level procedures at the bottom of the hierarchy,which are assembled into higher-level algorithms, such as classification tree learning. This enables developers to easily add new functionality at any level and fuse it with the existing code. The main branches of the component hierarchy are:

data management and pre processing for data input and output, data filtering and sampling, imputation, feature manipulation (discretization, continuation, normalization, scaling and scoring), and feature selection, classification with implementations of various supervised machine learning algorithms (trees, forests, instance-based and Bayesian approaches, rule induction), borrowing from some well-known external libraries such as LIBSVM (Chang and Lin, 2011), regression including linear and lasso regression, partial least squares regression, regression trees and forests, and multivariate regression splines, ensembles implemented as wrappers for bagging, boosting,which includes k-means and hierarchical clustering approaches, evaluation with cross-validation and other sampling-based procedures, functions for scoring the quality of prediction methods, and procedures for reliability estimation, projections with implementations of principal component analysis, multidimensional scaling and self-organizing maps. Orange's core is a collection of nearly 200 C++ classes that cover the basic data structures and majority of preprocessing and modelling algorithms.[6]

4. SYSTEM DESIGN

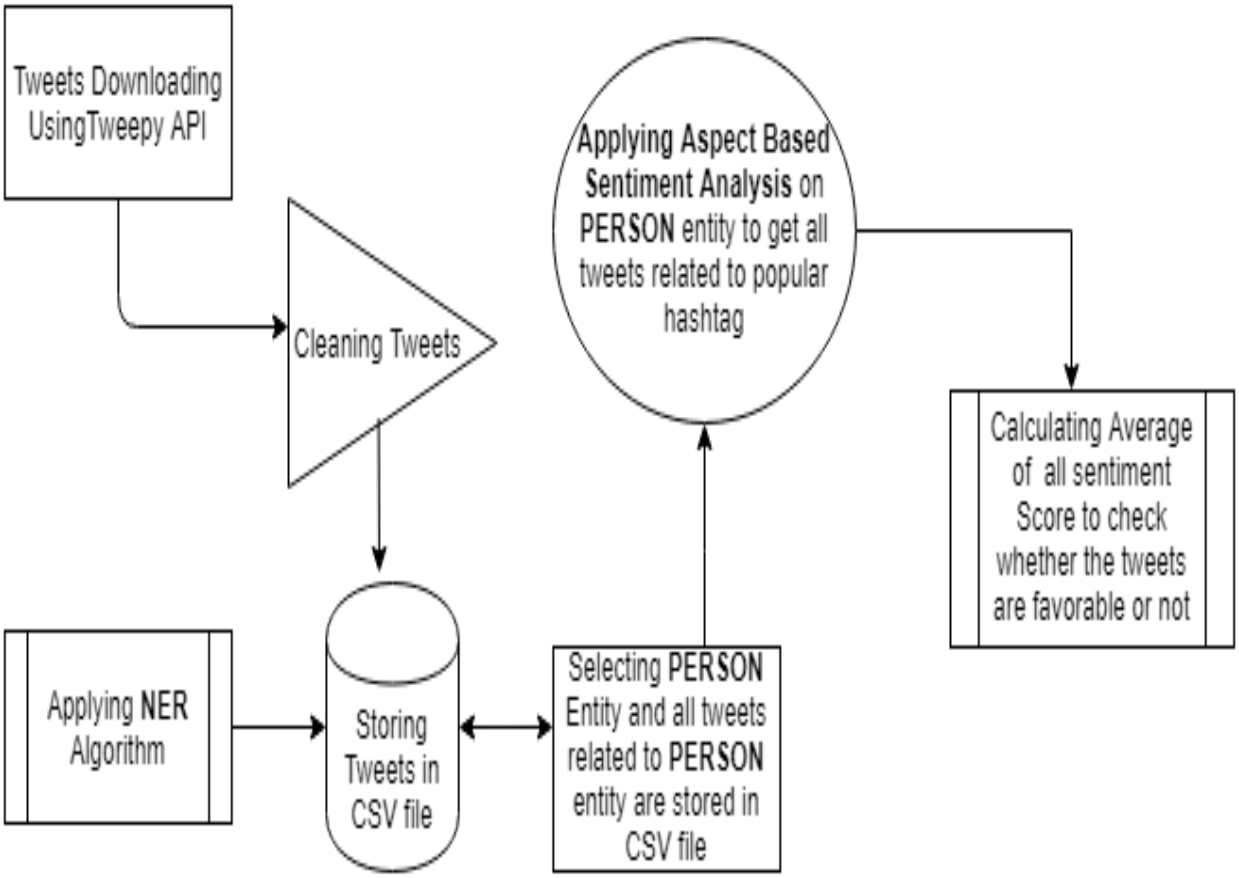

Figure 1: Flow chart of project

- Download specific tweets from twitter using tweepy API for a particular hashtag eg. #mumbaibandh.
- Extract Entity from all downloaded tweets through Named Entity Recognition (NER) using nltk.
- After getting the entities, we have classified each entity into a particular group or title eg. person, organization. We have taken those entities which comes under persons.
- Now Cleaning process is carried out on all the tweets under person group. In cleaning, repeated values are removed .
- Sentiment Analysis is carried out on all tweets of above selected entites.Here aspect based sentiment analysis is done.
- Based on the output provided by sentiment analysis, we have calculated the average score of the tweets under each entity.

4.1 Identifying entities from tweets using NER NLTK

This is the first step of our process. Here tweepy an open source library is used to download tweets from twitter.

Using tweepy we make request to the Twitter API for the tweets of #Mumbaibandh.#MumbaiBandh was trending over twitter in India for reservation for marathas in government and educational institutions.Like this any trending topic over twitter associated with a hashtag can be chosen.

Here OAuth authentication method is used for authentication.After successful authentication, tweets are retrieved from API home timeline.

The following process has been shown in figure 2.

"b'RT @rishabhrai1advo: I had no idea that Marathas who have such a glorified past and who are so proud of it, are on the same footing as sche\xe2\x80\xa6'"

b'RT @SirJadejaaaa: Maratha Kranti Morcha Calls Off Maharashtra Bandh After Violence In Many Parts.\n \nGovt Must Take Strong Action Against Suc\xe2\x80\xa6'

"b'RT @CBhattacharji: Ordinary folk stopped, attacked, traumatised #MumbaiBandh today. Pix via NDTV reporters.

Figure 2: Tweets of #mumbaiband

Above figure shows downloaded tweets using tweepy

4.2 Identifying entities from tweets using NER NLTK

```
PERSON, Marathas
PERSON, Maratha Kranti Morcha Calls Off
Maharashtra Bandh
GPE, Violence
GPE, Many
PERSON, Strong Action Against
ORGANIZATION, MumbaiBandh
ORGANIZATION, NDTV
ORGANIZATION, MumbaiPolice
ORGANIZATION, BeingSalmanKhan
ORGANIZATION, RealShivai
ORGANIZATION, BeingSalmanKhan
ORGANIZATION, RealShivai
GPE, Maharashtra
PERSON, Marathas
ORGANIZATION, BeingSalmanKhan
ORGANIZATION, RealShivai
ORGANIZATION, BeingSalmanKhan
ORGANIZATION, RealShivai
ORGANIZATION, MumbaiPolice
```

Figure 3: Entities of downloaded #mumbaibandh tweets

Downloaded tweets can contain any number of various entities. Here entities can be a person or a place or a thing. Since we have to perform sentiment analysis on specific entity, we first classify entities from tweets.

Here we have identified various entities used in all the tweets downloaded through NER .Now after extracting entities we have to classify them.Here we classified entities into particular groups like person, organisation, GPE etc.

Aspect based sentiment analysis is done on maratha, reservation and kranti morcha entities. For sentiment analysis, vader is used.

4.3 Retrieving only person entities from all the entities

```
PERSON, Marathas
PERSON, Anti Hindu
PERSON, Mumbai
PERSON, Maratha Protester Death
PERSON, Maratha Protester Death
PERSON, Mumbai
PERSON, Click
PERSON, Mumbai
PERSON, Sandeep98227377
PERSON, Marathas
PERSON, Devendra Fadnavis
PERSON, One
PERSON, Maratha Kranti Morcha
PERSON, Marathas
PERSON, Marathas
PERSON, Kal
```

Figure 4: Person entities of downloaded #mumbaiband tweets.

Sentiment analysis is meant to be performed on an entity which will give us meaningful results. Since we want to perform sentiment analysis on entities with meaningful results, we have extracted all entities that belong to the specific class called person.

This specific entity is chosen over others because a person can be well suited to find sentiment of an individual regarding a certain topic.

4.4 Performed aspect based sentiment analysis on entities in orange

After extracting the required tweets, aspect based sentiment analysis is performed on tweets using orange. Orange is an data mining application for data analysis through python scripting and visual programming.

In orange we have used sentiment analysis module and tweet profiler module. Tweet profiler module is used to classify the tweets into emotions like Joy, surprise, fear etc. Aspect based sentiment analysis is done on maratha, reservation and kranti morcha entities. For sentiment analysis, vader is used.

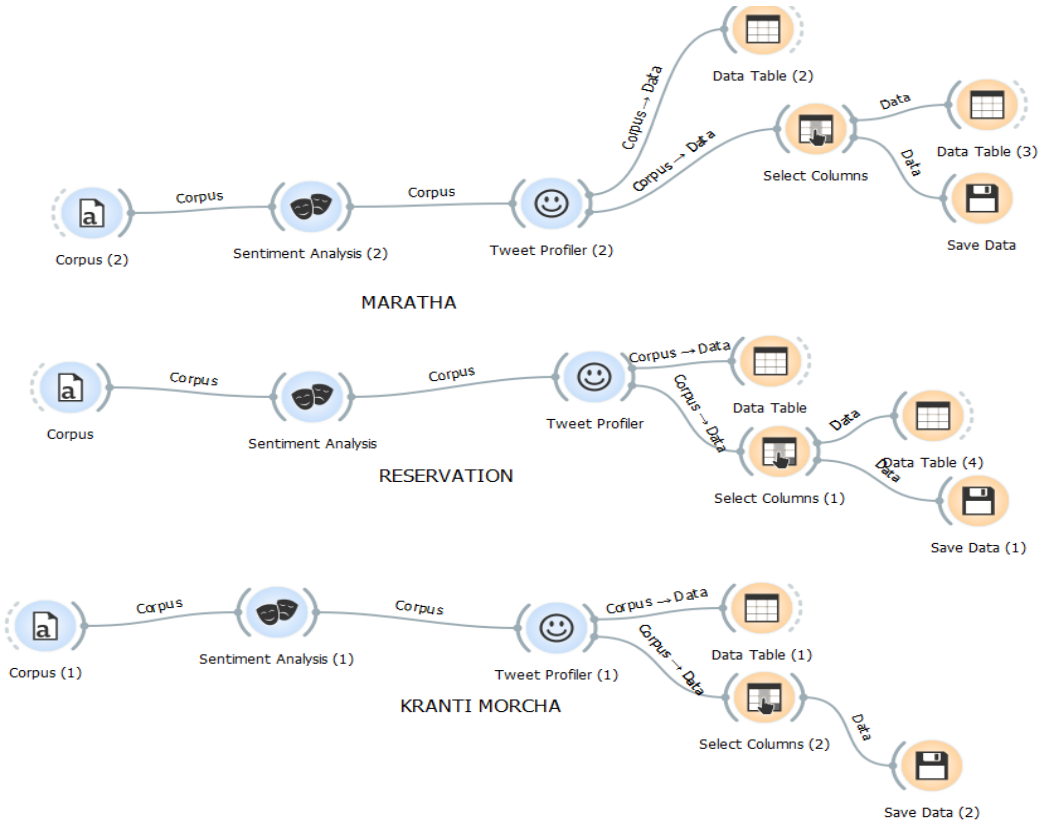

Figure 5: Aspect based sentiment analysis of person entities

4.5 Calculating sentiment scores using vader sentiment analysis.

pos	neg	neu	compound	Emotion	Maratha
continuou	continuou	continuou	continuou	discrete	string
					meta
0.144	0	0.856	0.4019	Joy	"b""RT @KhushbooTweet
0.176	0.11	0.714	0.296	Joy	b'RT @Vaib_Sriv: #Mumba
0	0	1	0	Fear	b'Make changes in the cor
0	0	1	0	Joy	"b""RT @ciril_dsouza: Do
0	0.18	0.82	-0.296	Joy	b'RT @Kumargautamkg: S
0	0	1	0	Surprise	"b""Do you think Maratha
0	0.223	0.777	-0.5106	Joy	b'RT @apnnewsindia: #Ma
0	0.143	0.857	-0.128	Joy	b'Mumbai Bandh \xe2\x80
0.188	0.378	0.434	-0.6037	Sadness	b'RT @adventurewala: #M
0.14	0.069	0.791	0.4215	Joy	"b""RT @PorgiMarathi: I a
0	0.249	0.751	-0.4574	Joy	b'RT @MumbaiLiveNews:
0	0.299	0.701	-0.4574	Surprise	b'#MarathaKrantiMorcha:
0	0.223	0.777	-0.5106	Joy	b'RT @apnnewsindia: #Ma
0	0.194	0.806	-0.34	Fear	"b'#MarathaQuotaStir : 2
0.14	0.069	0.791	0.4215	Joy	"b""RT @PorgiMarathi: I a
0.144	0	0.856	0.4019	Joy	"b""RT @KhushbooTweet

Figure 6: Sentiment scores of all person entities tweets

Using vader we can rated the tweets into 4 groups that is positive, negative, neutral and compound. Vader is a method of getting sentiments of the sentences. we have also calculated the scores of each group based on the keywords. This scores gives the amount of sentiments expressed in tweets.

4.6 Calculating average of popular entities in jupyter notebook.

```
The average of maratha is :  0.889707
The average of reservation is :  0.808187
The average of Krantimorcha is :  0.811375
```

Figure 7: Average scores of popular entities

After sentiment analysis is done, we have calculated the average scores of each person entities i.e. Maratha(0.889) Reservation (0.808), krantimorcha(0.811). The listed entities are the most discussed one in the #mumbaibandh with the overall sentiment which is positive. We can draw the inference that people are in favour for the reservation and the overall Sentiment for all the three major entities extractedfrom the hashtag (#mumbaibandh)is approving the idea for reservation.

5. CONCLUSION

Performing sentiment analysis on specific entities identified in the hashtag makes us aware of general public opinion of not only the term, but also we can do aspect level analysis and draw conclusions. This can be applied to any trending topic on twitter and help organization to get microscopic analysis to understand the overall public opinion.

6. REFERENCES

[1] Alan Ritter, Sam Clark, Mausam and Oren Etzioni "Named Entity Recognition in Tweets: An Experimental Study" Proceeding EMNLP '11 Proceedings of the Conference on Empirical Methods in Natural Language Processing Pages 1524-1534.

[2] D. Albanese, R. Visintainer, S. Merler, S. Riccadonna, G. Jurman, and C. Furlanello. mlpy: Machine learning Python. CoRR, abs/1202.6548, 2012.

[3] Perkins, Jacob. Python Text Processing with NLTK 2.0 Cookbook: over 80 Practical Recipes for Using Python's NLTK Suite of Libraries to Maximize Your Natural Language Processing Capabilities. PACKT Publishing, 2010.

[4] K. Ravi and V. Ravi, "A survey on opinion mining and sentiment analysis: Tasks, approaches and applications," Knowledge-Based Syst., vol. 89, pp. 14–46, 2015.

[5] Kenneth R. Beesley and Lauri Karttunen. 2002. Finite-State Morphology: Xerox Tools and Tech-niques. Studies in Natural Language Processing.Cambridge University Press.

[6] L. Zhang and B. Liu, "Aspect and Entity Extraction for Opinion Mining," pp. 1–40, 2014.

[7] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," In LREc, vol. 10, May. 2010, pp. 1320-1326.

[8] A. K. Jose, N. Bhatia, and S. Krishna, "TwitterSentimentAnalysis". NationalInstituteof TechnologyCalicut,2010.

[9] P. Lai, "ExtractingStrongSentimentTrendfromTwitter". Stanford University, 2012.

[10] E. Kouloumpis, T. Wilson, and J. Moore, "Twitter Sentiment Analysis:The Good the Bad and theOMG!", (Vol.5). International AAAI, 2011.

[11] J. Spencer and G. Uchyigit, "Sentiment or: Sentiment Analysis of Twitter Data," Second Joint Conference on Lexicon and Computational Semantics. Brighton:University of Brighton, 2008.

[12] P. Nakov, Z. Kozareva, A. Ritter, S. Rosenthal, V. Stoyanov, T. Wilson, Sem Eval-2013 Task2:Sentiment AnalysisinTwitter (Vol.2,pp. 312-320 2013.

[13] X. Zheng, Z. Lin, X. Wang, K.-J. Lin, and M. Song, "Incorporating appraisal expression patterns into topic modeling for aspect and sentiment word identification," Knowledge-Based Syst., vol. 61, pp. 29–47, May 2014.

[14] T. C. Chinsha and S. Joseph, "A syntactic approach for aspect based opinion mining," in Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing, IEEE ICSC 2015, 2015, pp. 24–31.

[15] S. Poria, E. Cambria, L.-W. Ku, C. Gui, and A. Gelbukh, "A Rule-Based Approach to Aspect Extraction from Product Reviews," Work. Nat. Lang. Process. Soc. Media, pp. 28–37, 2014.

[16] A. K. Jose, N. Bhatia, and S. Krishna, "TwitterSentiment Analysis".NationalInstituteof TechnologyCalicut,2010.

[17] D. Boyd, S. Golder, & G. Lotan, "Tweet, tweet, retweet: Conversational aspects of retweeting on Twitter," System Sciences (HICSS), 2010 …. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5428313