Feature-based Sentiment Analysis on Android App Reviews Using SAS® Text Miner and SAS® Sentiment Analysis Studio

Jiawen Liu[1], Mantosh Kumar Sarkar[2], and Dr. Goutam Chakraborty[3]
[12] Management Information Systems, Oklahoma State University, Stillwater, OK 74078
[3] Marketing, Oklahoma State University, Stillwater, OK 74078

Introduction

Sentiment analysis is a popular technique for summarizing and analyzing consumers' textual reviews about products and services. There are two major approaches for performing sentiment analysis; statistical model based approaches and Natural Language Processing (NLP) based approaches to create rules. In this study, we first apply text mining to summarize users' reviews of Android Apps and extract features of the apps mentioned in the reviews. We then use NLP approach for writing rules. We use reviews of two recent apps; a widget app from Brain& Puzzle category and a game app from Personalization category. We extracted six hundred textual reviews for each app from Google Play Android App Store. SAS® Enterprise Miner™ 7.1 is used for summarizing reviews and pulling out features, and SAS® Sentiment Analysis Studio 12.1 is used for performing sentiment analysis. Our results show that for both apps, carefully designed NLP rule-based models outperform the default statistical models in SAS® Sentiment Analysis Studio 12.1 for predicting sentiments in test data. NLP rule based models also provide deeper insights than statistical models in understanding consumers' sentiments.

Keywords: *Natural Language Processing (NLP), Sentiment Analysis, Text Mining, Android Apps.*

DATA

We collected data from Google Play Android App Store. Google Play Android App Store has a large and varied collection of Android Apps with rankings and user reviews. We extracted textual reviews having rich content from the App Store site. We chose to use two app categories for this research: Personalization and Brian& puzzle. We've chosen the most popular app, "Beautiful widgets" and "Where is my Perry", from each category for our research purpose. 600 rich text reviews for each app are collected. 500 reviews were used as corpus for text mining, building sentiment models, and writing sentiment rules; 100 reviews were held back as testing dataset. We categorized each textual review into positive and negative directory based on overall ratings. Google Play Android App Store uses a 5-star scale for rating: Greater than or equal to 4 stars are considered as positive; less than or equal to 2 stars are considered as negative for the purpose of this research.

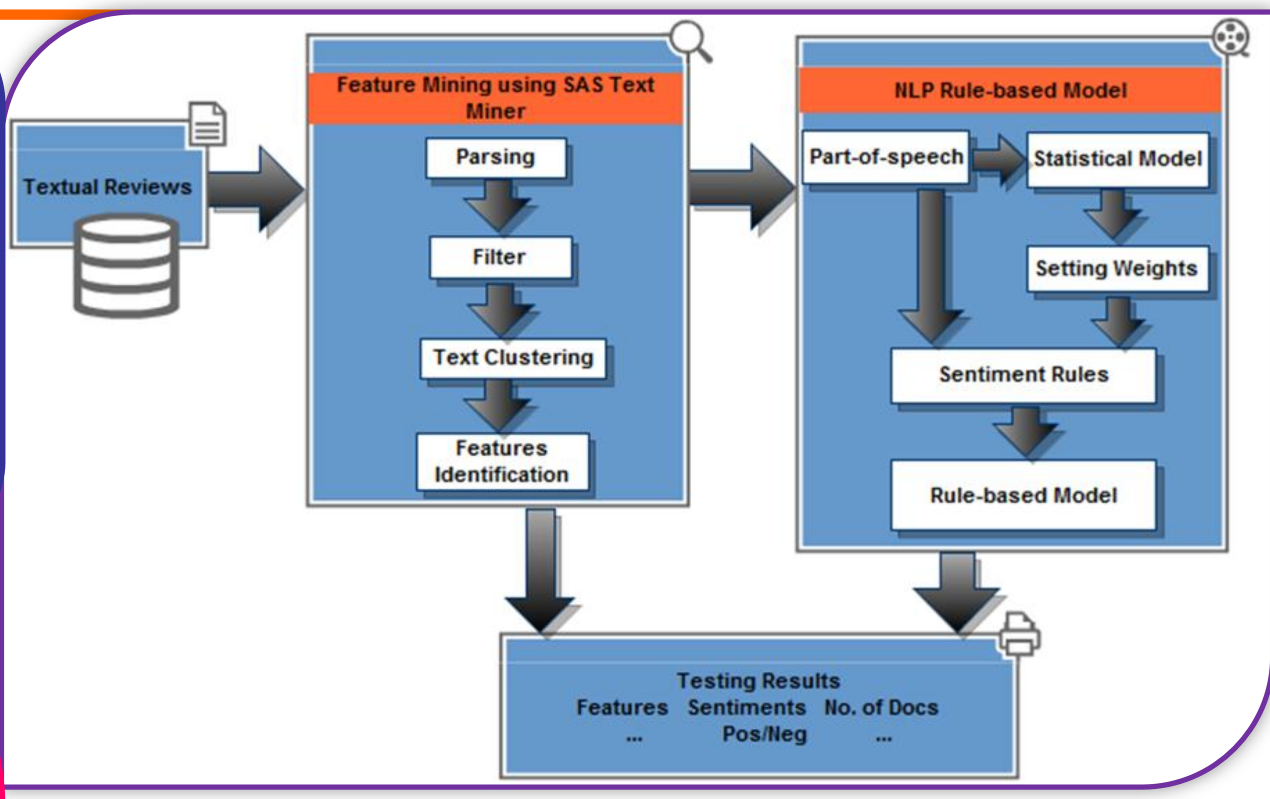

Figure 1 Methodology and Process Flow Chart

TEXT MINING

Figure 2 is node process flow created in SAS® Enterprise Miner™ 7.1. It starts with text parsing node. In parsing node, each comment is broken into tokens (terms). The identified tokens are listed in a "term by frequency" matrix. In this node, we ignored abbr, aux, conj, det, interj, num, part, prep, pron, and prop in the part-of-speech.

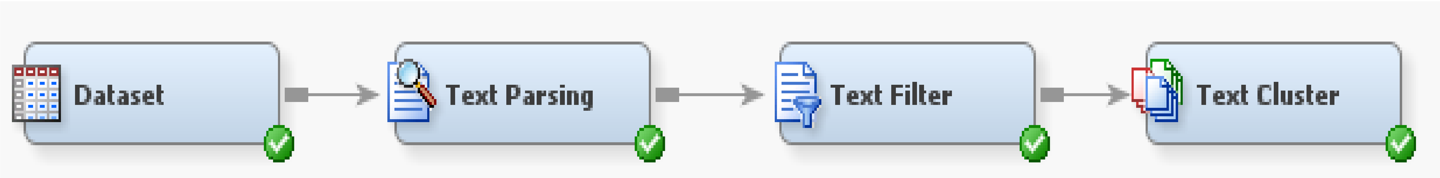

Figure 2 Diagram Flow in SAS® Enterprise Miner™ 7.1

In the text clustering node, we used SVD dimensions (k) of 40 (Figure 3). Singular Value Decomposition (SVD) is used to reduce dimensionality by converting the term frequency matrix into a lower dimensional form. Smaller values of k (2 to 50) are thought to generate better results for text clustering using short textual comments.

Property	Value
General	
Node ID	TextCluster
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Transform	
SVD Resolution	Low
Max SVD Dimensions	40
Cluster	
Exact or Maximum Number	Maximum
Number of Clusters	40
Cluster Algorithm	EXPECTATION-MAXIMIZATION
Descriptive Terms	8

Figure 3 Text Cluster property panel

Text clustering is used for unsupervised grouping of textual documents. In this research, we used default setting for clustering and run it separately for positive versus negative comments to identify most frequently mentioned features from each group. Figure 4 and figure 5 show the clustering results of Widget App. Figure 4 shows negative textual reviews clustering results. The clustered topics are majorly featuring to "battery usage/ widget", "permissions/privacy settings", "GPS widget", "overall performance", and "update". Figure 5 illustrates that "design/ graphics", "clock widget", and "price" are commonly mentioned features. Figure 6 and figure 7 show the clustering results of Game App. From figure 6, we identified "price", "overall performance", "network connection", "graphics", and "number of levels" as commonly mentioned features from negative app user reviews. Figure 7 pictures "design/graphics", "battery", and "addictiveness" features.

Descriptive Terms		Frequency
+'battery widget' +fix stopped galaxy s2 time working keeps	...	23
+late refund temperature bugs purchasing looks fixed useless	...	8
+suck accounts permissions remove reason +account +buy +permission	...	7
problems 'beautiful widgets' +problem +year beautiful few +add location	...	22
free 'fancy widgets' fancy battery widgets better +version options	...	26
+lock +unlock broken +phone animations +appear annoying +look	...	15
+setting settings crap manually transparency stars +buy +problem	...	9
rating fixed issues loved months galaxy +fix working	...	10
'anonymous statistics' +star anonymous statistics sucks left ugly +option	...	9
nexus properly geolocation shows updating +love useless +reboot	...	16
'home screen' +happen icons larger +screen home right weather	...	18
great freezes +day freezing works installed +load keeps	...	18
failed waste money developers +install +thing 'a lot of' geolocation	...	10
closes force constant uninstall +bug latest updates bugs	...	9
skins +skin download downloaded default installed always anymore	...	30
found standard +find toggle widgets +load +fail purchasing	...	17

Figure 4

Descriptive Terms		Frequency
+style phones different clocks clock +phone +match customize	...	11
'beautiful widgets' +'beautiful widget' beautiful widgets +'weather widget' +widget +find better	...	17
+love customizations +phone loved +fix apps 'a lot' animations	...	19
+keep fixes +developer always +work developers problems want	...	22
skins options +clock +match love +add different accurate	...	20
+issue fixed fixing update latest +good problems clock	...	33
great works fine working time years issues apps	...	33
awesome choose excellent highly battery weather love time	...	20
'home screen' home screen +problem +find highly +easy customize	...	6
favorite version +old wallpaper paid +update devices found	...	12
+animation nice +'weather widget' weather +widget +clock animations want	...	21
good +look customizable looking looks found worth +easy	...	25

Figure 5

Descriptive Terms		Frequency
different water good buying better easier loved thing	...	28
+install keeps freezes purchased +remove screen +fix play	...	24
device graphics +load games updated loading desire galaxy	...	20
levels +update updates level money back disappointed worth	...	28
'internet connection' internet opening connection great star thing +play	...	9
+close payed starts closes paid force +time refund	...	14
+pay 'fun game' fun stopped game crashing thing times	...	17
fix played playing +play bought stars 'great game' +open	...	43
+bug fixes locks +phone bugs +love able buy	...	17
original +work crashes good paid stars fixed nexus	...	18
37mb +download exactly loved updating kids +time disappointed	...	12
nexus +good works +work game +keep updated +open	...	13

Figure 8

Descriptive Terms		Frequency
puzzles challenging +'great game' similar +puzzle tools added funny	...	20
lots great fun love +challenge physics easy highly	...	15
playing +love played easy original +play +easy loves	...	36
+watch entertained keeps +keep hours +year play addictive	...	11
good different 'good game' +level adults challenges tools water	...	14
stars little son physics times +time +pass similar	...	9
loved perry love water adults better kids cute	...	19
boring hour passes best 'good game' +good hours pretty	...	19
graphics 'a bit' highly wonderful pretty entertaining play +puzzle	...	11
levels add +easy addicted +level tools addicting hard	...	16
+update fixed updates works +level cute +pass funny	...	10
bored nice great +'great game' challenges hard funny +time	...	12
'awesome games' +'awesome game' awesome greatest addictive games game +play	...	22
'fun game' fun times hard time +challenge added kids	...	23
old years +year daughter loves watching entertaining physics	...	10

Figure 7

NLP BASED SENTIMENT ANALYSIS

Part-of speech (POS) tagging is often the most time consuming and challenging task before doing sentiment analysis of any text documents. Online textual reviews are often short, non-grammar sentences and contain slangs, abbreviations, and symbols which make the POS tagging even more difficult. However, SAS® Enterprise Miner™ 7.1 gave us a good head start to have a sense what are app users' languages.

For sentiment analysis, we use SAS® Sentiment Analysis Studio 12.1. In the rule based mode, we started with defining new products and features. Here, products are the two apps. Features are identified as the ones identified from text clustering. For example, consider the following statement. "The game is good. I love its graphics design and I can play it for hours." In this comment, "game" is tagged as product and "graphics design" is tagged as feature. Products and features are tagged as nouns. One of many great things about SAS® Sentiment Analysis Studio 12.1 is that, we can define the synonym list of products and features. We are using this feature because of uncertain and non-grammar online reviews. For example, consider the following comment. "I love the high res". Here "res" likely refers to resolution, and resolution is a word which is similar to graphics.

For other Part-of speech tagging, we need to input all identified terms. To do this, we started with Statistical model in SAS® Sentiment Analysis Studio 12.1. We imported "learned features" from built statistical models to rule-based model to start part-of-speech tagging. "Import Learned Features" automatically extract keywords (terms) from corpus directory. We do not directly include those given weights into our model, but we consider it as importance and frequency indicator. We tagged Adverb (ADV), negative adjective (NEGADJ), positive adjective (POSADJ), and common verbs (VERB). We implemented this by using these as intermediate entities. In our model, NEGADJ1 is the list with all negative adjective words that we considered having less negative sentiments than the words in NEGADJ2. For example, we believe "terrible" is a stronger expression of negative sentiment than the word "bad". So, "terrible" will be in the NEGADJ2 list, and "bad" is in the NEGADJ1 list. Similarly, we believe the word "Awesome" has stronger positive sentiments than the word "nice". Therefore, we included the word "Awesome" in POSADJ2 and the word "nice" in POSADJ1. We also create more sophisticated rules by recognizing the combination of an adverb to an adjective. For example, consider the following comment. "I am so addicted to this game. Graphics is amazingly vivid." in this case, "so" and "amazingly" will be considered as adding higher positive sentiments. One of the difficulties that researchers often face in doing sentiment analysis is with textual reviews that contain mixture sentiments such as the following comment. "I love the graphics, but it drains battery a lot". Because we are doing feature based sentiment analysis, we are able to easily handle such reviews. In this case, the sentiment is positive on "graphics/design" and negative on "battery".

Final step is to implement all above rules into program. In this paper, we used CLASSIFIER, CONCEPT, CONCEPT_RULE, and PREDICATE_RULE rules. Figure 8 gives five examples that we used for matching negative sentiment rules for weather feature in widget app.

Type	Body	Weight
PREDICATE_RULE	(DIST_7, "_def{BWweather}","_a{_def{VERB}}", "_b{_def{NEGWORD}}")	1
CONCEPT_RULE	(SENT,"_c{_def{NEGADJ1}}","_def{BWweather}")	1
PREDICATE_RULE	(DIST_7,"_def{BWweather}","_a{_def{ADV}}","_b{_def{NEGADJ2}}")	2.5
PREDICATE_RULE	(DIST_7,"_def{BWweather}","_a{_def{ADV}}","_b{_def{NEGADJ1}}")	1.5
CONCEPT_RULE	(SENT,"_c{_def{NEGADJ2}}","_def{BWweather}")	2

Figure 8 Examples of PREDICATE_RULE and CONCEPT_RULE

RESULT

After we wrote the sentiment rules, we apply the rule-based models on testing datasets. Figure 9 and figure 10 are the results for testing widget app dataset. These show 86 percent precision on positive directory and 94 percent precision on negative directory. Figure 11 and figure 12 are the results for testing game app dataset. These show 94 percent precision on positive directory and 90 percent precision on negative directory. In Table 1, we compared overall performance of Statistical Model and Rule-based Model for two apps. It clearly shows that rule-based models outperformed the statistical models for both apps.

App	Statistical Model			Rule-based Model		
	Positive Precision	Negative Precision	Overall Precision	Positive Precision	Negative Precision	Overall Precision
Widget	64%	96%	80%	86%	94%	90%
Game	88%	74%	81%	94%	90%	92%

Table 1. Overall performance comparison

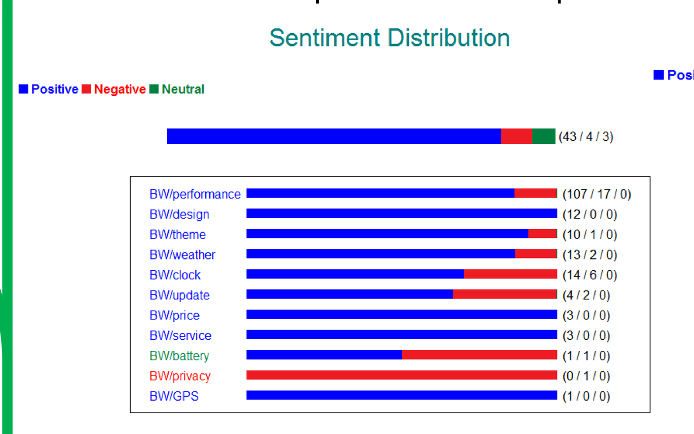

Figure 9 Widget App testing results from Positive directory

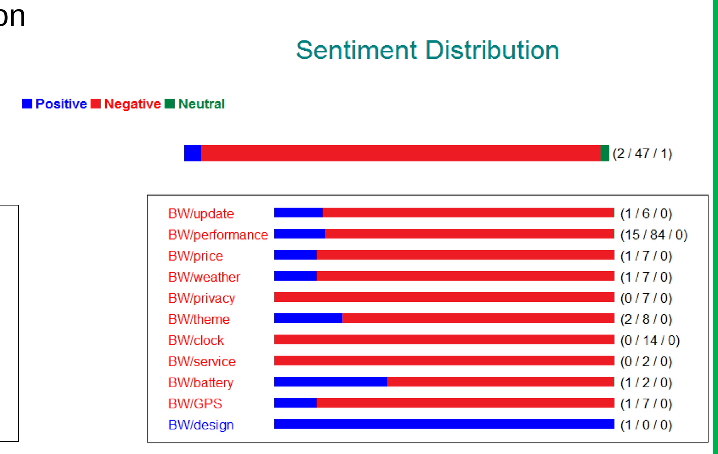

Figure 10 Widget App testing results from Negative directory

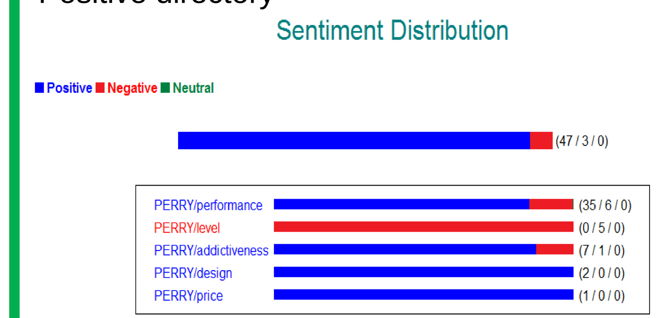

Figure 11 Game App testing results from Positive directory

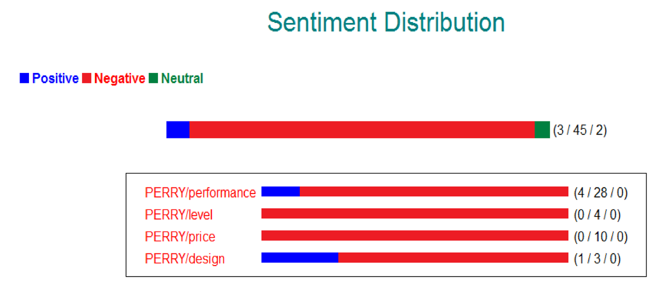

Figure 12 Game App testing results from Negative directory

DISCUSSION

We find that, that for both apps, carefully designed NLP rule-based models outperform the default statistical models in SAS® Sentiment Analysis Studio 12.1 for predicting sentiments in test data. The current default versions for statistical models in SAS® Sentiment studio do not allow for much customization. This may have contributed to the poorer performance of statistical models than NLP models in this study. The NLP rule based models also provide deeper insights than statistical models in understanding consumers' sentiments. For example, we find that app users are very addicted to the game app, but not happy for being charged more as they play more; they are pleased with graphics design of the widget app, but not ok with the app accessing their personal information.

No spelling check and no stop list were used in this study. These could be considered in future research to perhaps get better text mining and sentiment mining results.

CONTACT INFORMATION

Your comments and questions are valued and encouraged.
Jiawen Liu, Oklahoma State University, Stillwater OK, USA
Email: Jiawen.liu@okstate.edu
Mantosh Kumar Sarkar, Oklahoma State University, Stillwater OK, USA
Email: Mantosh.sarkar@okstate.edu
Dr. Goutam Chakraborty, Oklahoma State University, Stillwater OK, USA
Email: goutam.chakraborty@okstate.edu