A Novel Approach For Generating Rules For SMS Spam Filtering Using Rough Sets

Ashima Wadhawan, Neerja Negi

Abstract: Spam is defined as unwanted commercial messages to many recipients. Email Spamming is a universal problem with which everyone is familiar. This problem has reached to the mobile networks also now days to a great extent which is referred to as SMS Spamming. A number of approaches are used for SMS spam filtering like blacklist-white list filter, Content based filter, Bayesian filtering, checksum filter, heuristic filter. The most common filtering technique is content based spam filtering which uses actual text of messages to determine whether it is spam or not. Bayesian method represents the changing nature of message using probability theory. Bayesian classifier can be trained very efficiently in supervised learning. We have used a new mathematical approach Rough set Theory. Rough Set Theory is a new methodology which is used to cluster the objects of a decision system with a large data set. In this dissertation, the Naïve Bayes and the RST method are implemented.

Index Terms: Bayesian Filtering, Classification, Checksum Filter,Content Based Filtering Heuristic Filtering, Rough set,SMS Spam Filtering

1 INTRODUCTION

With the development of Internet and the rapid increase of network bandwidth, spam mail or also call Unsolicited Commercial Email(UCE) is increasingly becoming a great problem today. One of the reasons for the exponential growth of spam is where the email which has provides a cheap and neat instantaneous mode of communication world-wide. Spam has caused some serious problem that alert email user nowadays Spam can be defined as unsolicited (unwanted, junk) electronic message in which the number of recipient are in bulk, hence making the context impertinent and where the recipient has not granted the permission for it to be sent[1].Spam is distributed in a widely variety of forms including email spam, instant messaging spam, SMS spam, image spam. SPAM stands for Short pointless annoying message that describe sort of things. SMS has certain characters that are different from mails. A mail consists of certain structured information such as subject, mail header, salutation, sender's address etc. but SMS lacks such structured information. These make the SMS classification task much difficult. This situation makes the necessity for developing an efficient SMS filtering method.

2 RELATED WORK

Before 1990, some Spam prevention tools began to emerge in response to the Spammers who started to automate the process of sending Spam email. The first Spamprevention tool has used simple approach, based on language analysis by simply scanning emails for some suspicious senders or phrases like "click here to buy" and "free of charge". In late 1990s, blacklisting and white- listing methods were implemented at the Internet Service Provider (ISP) level.

- *Ashima Wadhawan is currently pursuing masters degree program in computer engineering inManav Rachna International University Faridabad, PH-01123456789.*
  E-mail: aashimawadhawan8@gmail.com
- *Neerja Negi is currently pursuing masters degree program in computer engineering in YMCA, Faridabad PH-01123456789. E-mail: neerja.fet@mriu.edu.in*

However, these methods suffered from some maintenance problems. There are many efforts underway to stop the increase of Spam that plagues almost every user on the mobile network. Various techniques have been used to filter the Spam messages. Bayesian [1] classifier is a simple probabilistic classifier. Its main advantage is that naïve Bayes classifiers can be trained very efficiently in a supervised learning Bayesian classifiers are used for parameter estimation in numerous practical applications. In supervised learning, the parameters are estimated by Maximum Likelihood Estimation (MLE) method. Decision Tree [2] is one of the most famous tools of decision- making theory. Decision tree is a classifier in the form of a tree structure that shows the reasoning process. Rough Sets[3] is a new methodology which is used to cluster the objects of a decision system with a large data set. An Information System is represented as $IS = (U, A)$, where, U is the Universal set of objects and C is a set of condition attributes. Here, we deal with a Decision System, which is represented as $DS = (U, AU\{d\})$,where d is a decision attribute. An Indiscernibility Relation is defined on a subset B of A $(B \subseteq A)$ as $RB = \{(x, y)\epsilon U \times U \mid a(x) = a(y), for\ all\ a\epsilon A\}$, where $a(x)$ is the value of object x for attribute a. The set U is partitioned into different sets based on the decision classes of a decision attribute and the equivalence classes are obtained based on B. Let there be k decision classes, $d_1, d_2, \ldots, d_k$. The equivalence classes based on the decision attribute, d, are represented as $[U]_d$. Clearly, $[U]_d$ is a subset of U. Let $[U]_d$ be denoted as X i.e. $X \subseteq U$. Let the equivalence classes obtained from the Indiscernibility relation be denoted by $[x]_B$. There is no work done for Rough set SMS Spam filtering yet and it is much more necessary to start the work

3 PROPOSED WORK

The proposed system framework contains four steps: Data set, preprocessing, Bayesian filtering classification, Decision rules.

3.1 DATA SET AND PREPROCESSING:

Firstly take a data set. The purpose of preprocessing is to transform messages in SMS in to a uniform format. It can take some attributes and also taken a corpus set(training set)if every attribute which you have taken that can match with every message of the corpus set then that consider 1otherwise 0.

3.2 BAYESIAN FILTERING CLASSIFICATION

sms	Se_kn0w	webms	Longst	thanks	Congr	win	free	sorry	urgent	private	please	finally	service	offer	great	oops	reminder	call
1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

MEAN

spam	0	0	0	0	0	0	o.75	0	0	0	0	0	0	0	0	0	0	0	0.5
ham	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

VARIANCE

spam	0	0	0	0	0	0	0.25	0	0	0	0	0	0	0	0	0	0	0.333
Ham	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Testing:-Urgent we are trying to contact u. Todays draw show that you have won a £800 prize guarnteed.call 09050001808 from landine.claim M95. valid 12hrs only.

Sample	sender	Webmsg	Longstrg	thanks	congratu	win	free	sorry	urgent	private	please	finally	service	offer	great	oops	reminder	call	
-	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	

We wish to determine which posterior is greater ham or spam for classification as spam the posterior is given by posterior(spam)=P(spam)p(sender_known|spam)p(webmsg|spam)p(longstring|spam)p(thanks|spam)p(congratulations|spam)p(win|spam)p(free|spam)p(sorry|spam)p(urgent|spam)p(private|spam)p(please|spam)p(finally|spam)p(service|spam)p(offer|spam)p(great|spam)p(oops|spam)p(reminder|spam)p(call|spam)

posterior(ham)=P(ham)p(sender_known|ham)p(webmsg|ham)p(longstring|ham)p(thanks|ham)p(congratulations|ham)p(win|ham)p(free|ham)p(sorry|ham)p(urgent|ham)p(private|ham)p(please|ham)p(finally|ham)p(service|ham)p(offer|ham)p(great|ham)p(oops|ham)p(reminder|ham)p(call|ham

$p(free|spam)=1/\sqrt{2\prod\sigma 2}*exp(x-\mu)2/2\sigma 2$

for every attribute|spam or ham i.e we can apply the same formula...

posterior(spam)=0.5*1*1*0.3678*1*1*0.3678*0.2587*1*0.3678*1*1*1*1*1*1*1*1*0.6870=0.0044

posterior(ham)=0.5*0.3678*1*0.3678*1*1*0.3678*1*1*0.3678*1*1*1*1*1*1*1*1*0.3678=0.0033

so posterior(spam) is greater than posterior(ham)

we predict the sample is **spam**

3.3 DECISION RULES USING RST
RST is a mathematical tool that used to find the decision rules. It convert the data in to required format(.isf) for applying Rough set theory. It can generate the decision rules using rst

3.3.1Approximations

Class	**No. of objects**	**Lower approximation**	**Upper approximation**	**Accuracy**
0	674	665	684	0.9722
1	126	116	135	0.8593

3.3.2 Reduct

#	Reduct	Length
1	Sender Known, web msg, long String, win, free, sorry, service, offer, great, call	10

3.3.3 Core Viewer

Quality of classification

For all condition attribute	0.9762
For all Condition attribute in core	0.9762

Attributes in CORE

Core sender_known
Core web msg
Core longstring
Core win
Core free
Core sorry
Core service
Core offer
Core great
Core call

3.3.4 RULES
\# ModLEM with Entropy
\# C:\Program Files\ROSE2\examples\smsspam.isf
\# objects = 800
\# attributes = 19
\# decision = spam
\# classes = {0, 1}
\# Sun May 11 14:17:38 2014
\# 0

Rule 1:
(sender_known = 1) & (sorry = 0) => (spam = 0); [657, 657, 97.48%, 100.00%][657, 0]
[{1, 2, 4, 5, 7, 8, 11, 14, 15, 17, 18, 19, 21, 22, 23, 24, 26, 30, 35, 37, 38, 40, 41, 44, 45, 46, 51, 53, 54, 56, 57, 58, 59, 60, 62, 63, 64, 65, 67, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 95, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 116, 117, 119, 120, 123, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 137, 138, 139, 141, 142, 143, 144, 145, 146, 147, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 162, 163, 164, 167, 169, 170, 171, 172, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 190, 191, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 225, 227, 229, 230, 231, 232, 233, 235, 237, 238, 239, 240, 242, 243, 244, 245, 246, 247, 248, 249, 250, 252, 253, 254, 255, 256, 257, 258, 259, 261, 262, 263, 264, 266, 267, 268, 270, 272, 273, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 298, 299, 300, 301, 302, 303, 304, 305, 307, 308, 309, 311, 312, 314, 315, 316, 317, 318, 319, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 335, 337, 338, 339, 341, 342, 343, 344, 345, 346, 347, 348, 349, 351, 352, 353, 355, 356, 357, 360, 361, 362, 363, 364, 365, 366, 367, 370, 371, 372, 373, 374, 375, 377, 378, 379, 380, 381, 382, 383, 384, 385, 387, 388, 389, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 417, 418, 420, 422, 424, 426, 427, 428, 429, 430, 431, 432, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 457, 458, 459, 460, 461, 462, 463, 466, 467, 468, 469, 470, 471, 473, 474, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 489, 490, 491, 495, 496, 497, 499, 500, 501, 502, 503, 504, 505, 507, 508, 509, 510, 511, 512, 513, 514, 515, 517, 520, 521, 522, 523, 524, 525, 527, 529, 531, 533, 534, 535, 536, 537, 538, 539, 540, 541, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 578, 579, 580, 582, 583, 584, 585, 586, 587, 591, 592, 593, 594, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 610, 611, 612, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 627, 628, 629, 630, 633, 634, 635, 636, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 686, 687, 688, 689, 690, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 712, 713, 715, 716, 717, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 733, 734, 735, 736, 737, 738, 740, 741, 742, 743, 744, 746, 747, 748, 750, 751, 754, 755, 756, 757, 758, 759, 760, 761, 763, 765, 766, 768, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 786, 787, 788, 791, 792, 793, 794, 795, 796, 797, 799, 800}, {}]

Rule 2:
(great = 1) => (spam = 0); [12, 12, 1.78%, 100.00%][12, 0]
[{1, 38, 43, 294, 325, 342, 351, 406, 441, 463, 467, 593}, {}]

Rule 3:
(sender_known = 1) & (offer = 1) => (spam = 0); [3, 3, 0.45%, 100.00%][3, 0]
[{27, 182, 400}, {}]

Rule 4:
(sender_known = 1) & (call = 1) => (spam = 0); [38, 38, 5.64%, 100.00%][38, 0]
[{75, 76, 81, 82, 86, 130, 133, 138, 149, 173, 177, 206, 227, 248, 289, 290, 314, 340, 341, 388, 398, 422, 444, 460, 465, 494, 496, 521, 541, 567, 575, 587, 681, 689, 703, 727, 765, 769}, {}]

Rule 5:
(sender_known = 0) & (web_msg = 0) => (spam = 1); [103, 103, 81.75%, 100.00%][0, 103]
[{}, {3, 6, 9, 10, 12, 20, 25, 28, 29, 31, 32, 33, 34, 36, 39, 50, 55, 61, 66, 68, 69, 94, 96, 115, 118, 121, 122, 124, 135, 136, 140, 148, 160, 161, 166, 168, 189, 228, 241, 260, 265, 269, 271, 297, 310, 313, 320, 334, 336, 350, 359, 368, 376, 386, 390, 402, 416, 421, 423, 425, 433, 456, 464, 472, 475, 493, 506, 516, 518, 526, 528, 530, 532, 565, 577, 581, 589, 590, 595, 609, 613, 632, 650, 661, 673, 674, 685, 691, 711, 714, 718, 732, 739, 749, 752, 753, 762, 764, 767, 785, 789, 790, 798}]

Rule 6:
(sender_known = 0) & (free = 1) => (spam = 1); [29, 29, 23.02%, 100.00%][0, 29]
[{}, {3, 6, 10, 13, 39, 96, 140, 148, 189, 228, 269, 271, 297, 358, 368, 386, 402, 419, 456, 464, 488, 493, 581, 595, 609, 631, 785, 790, 798}]

Rule 7:
(sender_known = 0) & (offer = 1) => (spam = 1); [8, 8, 6.35%, 100.00%][0, 8]
[{}, {260, 297, 368, 464, 528, 581, 637, 798}]

Rule 8:
(service = 1) => (spam = 1); [18, 18, 14.29%, 100.00%][0, 18]
[{}, {33, 39, 61, 94, 140, 160, 166, 189, 269, 369, 376, 416, 423, 595, 661, 739, 749, 753}]

Rule 9:
(sender_known = 0) & (please = 1) => (spam = 1); [10, 10, 7.94%, 100.00%][0, 10]
[{}, {25, 29, 33, 39, 52, 66, 94, 124, 160, 189}]

Rule 10:
(sender_known = 0) & (congratulations = 1) => (spam = 1); [3, 3, 2.38%, 100.00%][0, 3]
[{}, {251, 358, 506}]

Rule 11:
(sender_known = 0) & (win = 1) => (spam = 1); [18, 18, 14.29%, 100.00%][0, 18]
[{}, {12, 94, 115, 135, 168, 189, 274, 313, 320, 336, 358, 390, 506, 565, 577, 588, 718, 767}]

Rule 12:
(long_string = 1) => (spam = 1); [4, 4, 3.17%, 100.00%][0, 4]
[{}, {16, 20, 48, 711}]

Rule 13:
(sender_known = 0) & (web_msg = 1) & (long_string = 0) & (congratulations = 0) & (win = 0) & (free = 0) & (please = 0) & (service = 0) & (offer = 0) => (spam = 0) OR (spam = 1); [10, 10, 52.63%, 100.00%][1, 9]
[{49}, {42, 165, 192, 226, 236, 306, 519, 542, 710}]

Rule 14:
(sender_known = 1) & (sorry = 1) & (offer = 0) & (great = 0) & (call = 0) => (spam = 0) OR (spam = 1); [9, 9, 47.37%, 100.00%][8, 1]
[{47, 193, 224, 234, 354, 492, 498, 745}, {626}]

4 Experimental set up and Results

Matlab language is used for the implementation of the proposed framework. Rose2 software is used for High level results like reduct and decision rules .Naïve Bayes and Rough set algorithms have implemented for the Spam filtering task. Extensive tests have been performed with varying numbers of data set sizes. The success rates reach their maximum using all the messages and all the words in training corpus.

5 CONCLUSION AND FUTURE SCOPE

In this dissertation Naive Baye's has been implemented and test data is giving the desired results. The Naive Baye's based on Supervised learning technique. We can get association rules from Naive Baye's but in SMS spam data set we need to find the rules whether an incoming message is spam or not. To implement this is a new mathematical tool rough set has been used. In this dissertation the rudiments of rough set are implemented and high level results i.e reduct and rule induction are obtained by using Rough set tool ROSE2.By the implementation it can be seen that more desired results are obtained by using Rough set theory. In the future improve the structural data the size of Corpus can be implemented by Collecting more SMS and can implement the high level results in the rough sets.

Acknowledgment

Th **Acknowledgements** Authors would like to thank Ms.Richa Arora, Lecture Delhi institute of engineering, Smalkha and Asst. Prof. Ms.Neerja Negi for their supervision during the completion of this work.

References

[1] Sarah Jane Delany , Mark Buckley , Derek Greene, "Sms Spam Filtering: methods and data" , Expert Systems with Applications , proc ELSEVIER 2012 pp 899–908.

[2] Noemí Pérez-Díaz, David Ruano-Ordás, Florentino Fdez-Riverola, José R. Méndez, "SDAI: An integral evaluation methodology for content-based spam filtering models" ,Expert Systems with Applications ,proc ELSEVIER 2012 pp487–500

[3] Zbigniew Suraj **"**An Introduction to Rough Set Theory and Its Applications : A tutorial"proc ICENCO 2004, Cairo, Egypt pp27-30

[4] José María Gómez Hidalgo, Guillermo Cajigas Bringas, Enrique Puertas Sánz , "Content Based SMS Spam Filtering", proc ACM pp373-380

[5] A.K Uysal,S.Ergin, E. Sora Gunal, "The Impact of Feature Extraction and Selection on SMS Spam Filtering" ,proc.IEEE,2010, pp-1392-1412

[6] Tiago A. Almeida, José María Gómez Hidalgo, Tiago P. Silva" Towards SMS Spam Filtering: Results under a New Dataset" proc INTERNATIONAL JOURNAL OF INFORMATION SECURITY SCIENCE pp 1-18

[7] Jan Komorowski, Lech Polkowski and Anderzej Skowron, "Rough Sets: A Tutorial",pp 1-8

[8] Zdzisław Pawlak and Andrzej Skowron Information Sciences, "Rudiments of Rough Sets", pp 3 -27, 2007