SMS Classification Based on Naïve Bayes Classifier and Apriori Algorithm Frequent Itemset

Ishtiaq Ahmed, Donghai Guan, and Tae Choong Chung

***Abstract*—In this paper, we propose a hybrid system of SMS classification to detect spam or ham, using Naïve Bayes classifier and Apriori algorithm. Though this technique is fully logic based, its performance will rely on statistical character of the database. Naïve Bayes is considered as one of the most effectual and significant learning algorithms for machine learning and data mining and also has been treated as a core technique in information retrieval. However, by applying user-specified minimum support and minimum confidence, we gain significant improvement on effective accuracy 98.7% from the traditional Naïve Bayes approach 97.4% experimenting on UCI Data Repository.**

***Index Terms*—Short message service (SMS), Naïve Bayes classifier, Apriori algorithm, spam, ham, minimum support, minimum confidence.**

I. Introduction

As the mobile phone market is rapidly expanding and the modern life is heavily dependent on cell phones, Short Message Service (SMS) has become one of the important media of communications [1]. This media of communication has been considered as one of the fundamental and primitive way of connection for its cheapness, more convenient for advanced to novice users of cell phone, mobility, individualization and documentation. The number of junk SMS is increasing day by day and according to Korea Information Security (KISA), this amount of junk SMS is more than the email spam. Besides this, the cell phone users in US got 1.1 billion spam SMS and Chinese users also received 8.29 spam SMS per week [2].

Constructing efficacious classification is one of the most challenging tasks in machine learning and data mining. Previously many techniques are invented, decision trees [Q92], k-NN [3], Neural Network [4], Centroid-based approaches [5], `SVM, Rocchio Classifier [6], Regression Models [5], Bayesian probabilistic approaches [7], inductive rule learning, online learning [8], rule learning [CN89, C95] and Naïve Bayes classification [DH73]. Besides these there are some other systems C4.5 [Q92], CN2 [CN89], and RIPPER [c95]

In the Naïve Bayes classification, all words a in a given SMS are considered as mutually independent. It is the simplest form of Bayesian network which can be interpreted as conditional independent [8]. In our proposed algorithm we have incorporated the frequent item idea which effectively increases the overall accuracy. We have not only considered each and every word as independent and mutually exclusive but also frequent words as a single, independent and mutually exclusive. The main contribution of this paper is better accuracy than the state of the art method of classifying text.

This paper is organized as follows. In Section II addresses related work like how the SMS is classified to spam and ham by Naïve Bayes classifier. In Section III our proposed method is described. In Section IV the performance analysis of our suggested method is discussed. The last section addresses our conclusions and future work.

II. Background Study and Related Work

There has been numerous numbers of studies on active learning for text classification using machine learning techniques [9]-[11], probabilistic models [12], [13]. The query by committee algorithm (Seung *et al.* 1992, Freund *et al.*, 1997) used priori distribution than hypothesis. The popular techniques for text classifications are decision trees [14], [15], Naïve Bayes [14]-[16], rule induction, neural networks [14]-[16], nearest neighbors and later on Support Vector Machine [17]. Though there is lot of techniques and algorithms which have been proposed so far, the text classification is not yet accurate and faultless and still in demand of improvement.

Two types of SMS classification exists in the current mobile phones and they are enlisted as Black and White [18]. These kinds of techniques are based on the previously known keywords and patterns. These techniques are currently available to the numerous number of cell phone operating systems. These techniques are also recalled as Spam SMS blocker in Google android phones and SMS spam runner in Symbian Operating Systems. As these techniques are based on limited number of keywords, the accuracy levels are not quite satisfactory as compared to human satisfaction.

Naïve Bayes is one of the simplest probabilistic classifiers which are based on Bayes theorem with strong naïve independence assumption. This assumption treated each and every word as a single, independent and mutually exclusive. This model can be described as "Independent Feature Model" [9]. As the complexity for learning Bayesian Classifier is

Manuscript received August 20, 2013; revised December 10, 2013. This work was supported a grant from the NIPA (national IT Industry Promotion Agency) in 2013. (Global IT Talents Program), South Korea and Development machine leaning and applications for avoiding obstacles of mobile robots in dynamic environments.

Ishtiaq Ahmed is with the Department of Computer Engineering, School of Electronics and Information, Kyung Hee University, Giheung-gu, Yongin-si, Gyeonggi-do 446-701, Republic of Korea (e-mail: Ishtiaq.khu@khu.ac.kr).

Donghai Guan is with the Faculty of Department of Computer Engineering, Kyung Hee University, Republic of Korea (e-mail: donghai@oslab.khu.ac.kr).

Tae Choong Chung is with the Faculty of Department of Computer Engineering, Kyung Hee University, Republic of Korea. He is also with the Artificial Intelligence Lab, Kyung Hee University (e-mail: tcchung@khu.ac.kr).

DOI: 10.7763/IJMLC.2014.V4.409

colossal, there must be some ways which can reduce the complexity and thus Naïve Bayes classifier is introduced. The Naïve Bayes Classifier does this by making a conditional independence assumption that dramatically reduces the number of parameters to be estimated when modeling $P(X|Y)$, from $2(2^n - 1)$ to just 2^n [14].

The Naïve Bayes algorithm is a classification algorithm based on Bayes rule, that assumes all the attributes $X_1,\ldots,X_n$ are conditionally and mutually independent given Y. The value of this assumption dramatically simplifies and reduces the complexity and representation of $P(X|Y)$ [19] and the problem of estimating it from the training data. Considering the case where $X = (X_1, X_2)$.

$$P(X|Y) = P(X_1, X_2|Y) = P(X_1|X_2,Y)P(X_2|Y)$$
$$= P(X_1|Y)P(X_2|Y)$$

This can be represented as

$$P(X_1\ldots X_n|Y) = \prod_{i=1}^{n} P(Xi|Y)$$

Let, Y is any discrete-valued variable and the attributes $X_1\ldots X_n$ are any discrete or real valued attributes, the equation for the probability that Y will take the k^{th} possible value, according to Bayes rule, is

$$P(Y=y_k|X_1\ldots X_n) = \frac{P(Y=y_k)\,P(X_1..X_n|Y=y_k)}{\sum_j P(Y=y_j)P(X_1..X_n|Y=y_j)}$$

Assuming the X_i is conditionally independent given Y, the equation can be rewrite as

$$P(Y=y_k|X_1\ldots X_n) = \frac{P(Y=y_k)\,\prod_i P(X_i|Y=y_k)}{\sum_j P(Y=y_j)\prod_i P(X_i|Y=y_j)}$$

Let we have five SMSs and among them two messages are ham: “good.” And “very good.” and the rest of them are considered as spam: “bad.”, “very good”, “very bad.” and “very bad, very bad!” in Table I. However, for training the system construction of vector table is very important and need to train the system through the vector table. Initially we have only one feature extraction process which breaks down each SMS into individual words and produces 5 words by separating the words by space or comma(,) or full stop(.) or exclamatory sign(!). So after the feature extraction process the words become the word vocabulary: “good”, “very”, “bad”.

TABLE I: VECTOR TABLE

SMS No.	Type	Word attributes		
		Good	Very	bad
1	Ham	1	0	0
2	Ham	1	1	0
3	Spam	0	0	1
4	Spam	0	1	1
5	Spam	0	2	2

As the Naïve Bayes is the probabilistic classifier, we don’t need to know the total number of words in each SMS, thus the vector table can be replaced by the word occurrence table which is demonstrated in Table II,

TABLE II: WORD OCCURRENCE TABLE

Word attributes	Ham Occurrences	Spam Occurrences
good	2	0
very	1	3
bad	0	4
Total	3	7

So, after the construction of the table, if an unknown SMS suddenly needs to analyze whether it is spam or ham described as “good? bad! very bad boy!,”Then by applying the feature extraction process we have three words as the boy is not enlisted in the word occurrence table. So finally the words are: “good”, “very”, “bad,”Therefore, to classify the unknown incoming SMS we can demonstrate the Naïve Bayes classification as:

$$P(ham|good, very, bad)$$
$$= P(ham) \times P(good|ham)$$
$$\times P(very|ham) \times P(bad|ham)$$
$$P(spam|good, very, bad)$$
$$= P(spam) \times P(good|spam)$$
$$\times P(very|spam) \times P(bad|spam)$$

Calculating the final probability of ham and spam we can finally make the decision of being ham or spam depending on their majority value. If the proportion of ham exceeds the proportion of spam, then it has a greater chance to be a “ham” and vice versa.

Beside this, R. Agrawal and R. srikant[20], describes Apriori algorithm in their paper discovering association rules between items in a large database sales transactions. In our proposed algorithm we integrate these two concepts with little modification and adding with extra computation, which successfully produces better result than the state of art algorithm.

III. OUR PROPOSED METHODOLOGY

In this paper we present a method to build a categorization system that integrates association rule mining with the classification problem [21]. However, we need to perform SMS collection, preprocessing, feature selection, vector creation, filtering process and updating the system. The whole overall process is described below, which significantly produces better result with adequate accuracy than the state of the art algorithm. There are several steps for text classification and each of them is discussed below:

A. *Loading Database*

This step collects various SMSs from different incoming messages and for our experiment we have collected data from UCI Machine learning repository “SMS Spam collection Data Set” which consists of 5574 SMSs of spam and ham. At the beginning, we have divided this database into two subclasses as collection of ham and spam. Initially we have considered the first 1000 lines only for our experiment only.

B. *Feature Extraction*

In the traditional Naïve Bayes approach, each and every word is considered as an independent word. However, in our approach we have also considered words are independent to each other, but in modified concept. Additionally, we have also treated the high frequency words as a single and mutually independent also. As a simple example, let we have nine SMSs consisting of ham and spam. Among them five SMSs are considered as spam and the rest of them are ham. Spam SMSs are: “word1, word2, word5”, “word2, word3”, “word1, word3”, “word1, word3”, “word1, word2, word3”; similarly the ham SMSs are “word2, word4”, “word1, word2, word4”, “word2, word3” and “word1, word2, word3, word5,”Considering the spam and ham SMSs we have built two separate databases. Now, by applying Apriori algorithm, we have separated the frequent individual items. However, considering the minimum confidence as 2 in spam SMSs, we have three different frequent items which are “word1, word2”, “word1, word3”, “word2, word3,”These words are considered as individual and single words. So after the feature extraction process for spam SMSs we have 7 words including the frequent items which are generated by the Apriori algorithm and these are: “word1”, “word2”, “word3”, “word5”, “word1, word2”, “word1, word3”, “word2, word3,”Similarly for ham SMSs database, we have 8 words as well and these are: “word1”,”word2”, “word3”, “word4”, “word5”, “word1, word2”, “word2, word3” and “word2, word4,”

C. Vector Creation and Training

Vector creation is an important factor for the Naïve Bayes classification system. A dataset is imbalanced if the classification categories are not approximately equally represented. As this procedure depicts the performance issue of the whole system, this is considered as the core part and influence the overall operation. We propose to use word occurrence table as its simple to demonstrate and use also. Let, we have SMS as “word1, word2, word2, word1, word3, word5” and we have high frequency words length as 3 which means three words together form a single word. First of all, we have to separate the unique words as “word1, word2, word3, word5,”Then, we have to make the combination of these words and this combination will be at most three words together as “word1, word2”, “word1, word3”, “word1, word5”, “word2, word3”, “word2, word5”, “word1, word2, word3”, “word1, word3, word5”, “word1, word2, word5”, “word2, word3, word5,”Then we have to count the frequencies of individual and high frequency words. According to the previous description, we have separated the dataset into two sub categories as spam and ham and thus create the vector table for spam SMSs only (see Table III and Table IV).

TABLE III: VECTOR TABLE FOR SPAM SMS

SMS No	Word Attributes						
	W1	W2	W3	W5	W1, W2	W1, W3	W2, W3
1	1	1	0	1	1	0	0
2	0	1	1	0	0	0	1
3	1	0	1	0	0	1	0
4	1	0	1	0	0	1	0
5	1	1	1	0	1	1	1

TABLE IV: VECTOR TABLE FOR HAM SMS

SMS No	Word Attributes							
	W1	W2	W3	W4	W5	W1,W2	W2,W3	W2,W4
1	0	1	0	1	0	0	0	1
2	1	1	0	1	0	1	0	1
3	0	1	1	0	0	0	1	0
4	1	1	1	0	1	1	1	0

So, after making the vector tables, we have formed the word occurrence table combined with spam and ham word frequencies as like bellow:

TABLE V: WORD OCCURRENCE TABLE

Word attributes	Ham occurrences	Spam occurrences
Word1	4	2
Word2	3	4
Word3	4	2
Word4	0	2
Word5	1	1
Word1, word2	2	2
Word1, word3	3	0
Word2, word3	2	2
Word2, word4	0	2
Total	**19**	**17**

D. Running the Naïve Bayes System

After building the word occurrence table successfully, we will run the system to classify a SMS whether the SMS is spam or ham. Before having the classification of SMS using naïve Bayes, we should say how an individual SMS is processed for the system. Let, we have SMS: “word1, word1, word2, word2, word3,” Then we have to make all possible combination to form conjugal words i.e. high frequency conjugal words which have been processed and calculated by running the association rule mining technique Apriori algorithm. The maximum number of words that has formed the conjugal high frequency word would be the same as the training session example. Before going to have the combination, we need to separate the unique words as word1, word2, and word3. In the above example the all possible combinations would be “word1”, “word2”, “word3”, “word1, word2”, “word1, word3”, “word2, word3,”Here I haven’t made the words which are formed more than 2 words as there are no frequent words which are formed more than two words in the above frequency table. Since the Naïve Bayes classifier works on the probability of words, we have to calculate the probability in little bit different way. We will not only consider the individual words occurrence only, but also consider the high frequency conjugal words also. We also have to calculate occurrence of each individual words and the high frequency words which will make significant impact on the overall performance. After having those values if we observe that the probability of being ham is greater than the spam, then it could have more chance of being ham and vice versa. So, from the example we have demonstrated so far we can these following data:

- Prior probability of ham P(ham) = 4/9
- Prior probability of spam P(spam) = 5/9
- Total number of vocabulary $|v|= 9$
- Total number of ham words $N_{ham} = 19$
- Total number of spam words $N_{spam}= 17$

Therefore, we can classify the SMS as:

P(ham, word1, word1, word2, word2, word3) = P(ham) ×P(word1|ham)^2 × P(word2|ham)^2 × P(word3|ham) × P(word1, word2|ham)^2 × P(word1, word3|ham) × P(word2, word3|ham)

P(spam, word1, word1, word2, word2, word3) = P(spam) ×P(word1|spam)^2 × P(word2|spam)^2 × P(word3|spam) × P(word1, word2|spam)^2 × P(word1, word3|spam) × P(word2, word3|spam).

To obtain a better accuracy we have applied the Laplace estimator to avoid the zero probability for SMS. As we are already familiar with the prior probability of spam and ham, now we will compare with the individual probability factor of each and every words and high frequency words we mentioned earlier.

P(word1|ham) = (4 + 1)/(19 + $|v|$) = 5/28
P(word2|ham)= (3 + 1)/(19 + $|v|$) = 4/28
P(word3|ham)= (4 + 1)/(19 + $|v|$) = 5/28
P(word1, word2|ham)= (2 + 1)/(19 + $|v|$) = 3/28
P(word1, word3|ham)= (3 + 1)/(19 + $|v|$)= 4/ 28
P(word2, word3|ham)= (2 + 1)/(19 + $|v|$)= 3/28
P(word1|spam) = (2 + 1)/(17 + $|v|$) = 3/26
P(word2|spam)= (4 + 1)/(17 + $|v|$) = 5/26
P(word3|ham)= (2 + 1)/(17 + $|v|$) = 3/26
P(word1, word2|spam)= (2 + 1)/(17 + $|v|$) = 3/26
P(word1, word3|spam)= (0 + 1)/(17 + $|v|$)= 1/26
P(word2, word3|spam)= (2 + 1)/(17 + $|v|$) = 3/26

Finally applying these values the above equation we get,

P(ham, word1, word1, word2, word2, word3) = 4/9 ×(5/28)^2 × (4/28)^2 × (5/28) × (3/28)^2 × (4/28) × (3/28) = 9.075049e -9

P(spam, word1, word1, word2, word2, word3) =(5/9)×(3/26)^2×(5/26)^2×(3/26)×(3/26)^2×(1/26)×(3/26) =1.86481e-9

Now by observing these values we can predict that the mentioned SMS has greater probability of being ham than spam. Besides this, we can use logarithm rule to have better precision and thus could avoid underflow problem as:

$$\log(\alpha\beta) = \log(\alpha) + \log(\beta)$$

IV. Results and Discussion

For our experiment we have used Intel Core™ i5 machine with 3GB ram, the whole system is implemented by Java SE language and UCI data repository putting constraint minimum support value [22] 5, is used for training the system. Firstly we have considered the first 1000 lines of SMS of the database instead of considering the whole database for training and testing the system.

We have segmented the database as follows. At first we train our system by first 900 SMSs and then test our system by next 100 SMSs, depicted in Table VI.

We have done this procedure several times and produced better accuracy than the state of the art algorithm (Naïve Bayes Classifier). For the first iteration, we have considered 1~900 SMSs as training data and 901~1000 SMSs as testing data. These procedure is again applied in the system and this time the training SMSs are 101~1000 and testing SMSs are 1~100 SMSs. These procedures are repeatedly done for 10 times. As we have noticed from the table, the overall accuracy is much better than the state of the art algorithm and significantly depicts steady performance and never degrades accuracy than the state of art algorithm. We also come to know from table that the improvement is made from the avg. accuracy 97.4% to 98.7%, which depicts 1.3% improvement than the traditional approach.

TABLE VI: Accuracy Comparison

No. of Test SMSs	Proposed System Accuracy (%)	State of the art algorithm (Naïve Bayes Classifier) Accuracy (%)	Difference (%)
1~100	98.0	97.0	+1.0
101~200	100.0	97.0	+3.0
201~300	100.0	100.0	0.0
301~400	98.0	97.0	+1.0
401~500	100.0	99.0	+1.0
501~600	99.0	99.0	0.0
601~700	98.0	96.0	+2.0
701~800	96.0	96.0	0.0
801~900	99.0	96.0	+3.0
901~1000	99.0	97.0	+2.0
Avg. Accuracy	98.7	97.4	+1.3

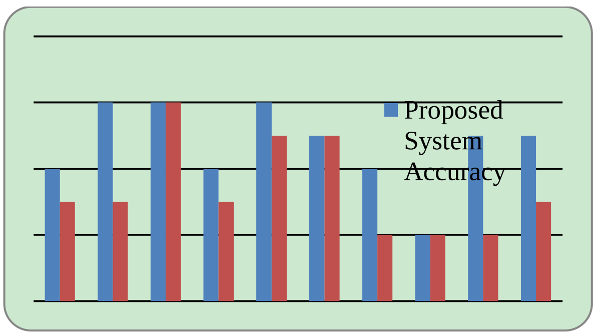

Fig. 1. Numerical comparison between our system and state of art algorithm.

Though training the system for the first time requires little bit more time than the state of the art, it increases the accuracy significantly in Fig.1. Once the system is trained, then classifying single SMS takes almost same time as the state of the art algorithm. In our system the avg. time which is needed for classifying the text is 0.13 sec, whereas the state of the art takes around 0.00007 sec. In blank eyes we hardly understand that our system slightly takes more times.

V. CONCLUSION

Automatic text categorization is the task of assigning level of different categorization. In our paper it's between spam and ham and to make this procedure in reality we have incorporated Apriori algorithm with Naïve Bayes classification but in little bit modification. Although this technique is logic based, but the result id depended with dataset. By applying our strategy we depicted significant improvement than the state of the art algorithm. Our supervised machine learning system for handling and organizing spam system and by performing our proposed strategy this SMS spam detection technique have reached accuracy levels that can outperform even the state of the art algorithm.

REFERENCES

[1] P. J. Denning, "Electronic Junk," *ACM Communications*, vol. 25, no. 3, Mar. 1982, pp. 163–165.

[2] W. Qian, H. Xue, and W. Xiayou, "Studying of Classifying Junk Messages Based on The Data Mining," in *Proc. International Conference on Management and Service Science*, IEEE Press, Sept. 2009, pp. 1-4.

[3] S. Gao, W. Wu, C. H. Lee, and T. S. Chua. "A maximal .gure-of-merit (MFoM)-learning approach robust classifier design for text categorization," *ACM Transactions on Information Systems*, vol. 24, no. 2, pp. 190–218, 2006.

[4] R. Schapire, Y. Singer, and A. Singhal. "Boosting and Rocchio applied to text clustering," in *Proc. the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, Melbourne, Australia, 1998, pp. 215–223.

[5] R. Klinkenberg and T. Joachims. "Detecting Concept Drift with Support Vector Machines," in *Proc. the Seventeenth International Conference on Machine Learning*, 2000, pp. 487–494.

[6] Z. Cataltepe and E. Aygun. "An improvement of centroid-based classi.cation algorithm for text classification," in *Proc. IEEE 23rd International Conference on Data Engineering Workshop*, 2007, pp. 952–956.

[7] S. Weiss, C. Apte, F. Damerau, D. Johnson, F. Oles, T. Goetz, and T. Hampp, "Maximizing text-mining performance," *IEEE Intelligent Systems*, pp. 63–69, 1999.

[8] A. McCallum and K. Nigam. "A comparison of event models for naive bayes text classification," presented at AAAI-98 Workshop on Learning for Text Categorization, 1998.

[9] W.-W. Deng and H. Peng, "Research On A Naïve Bayesian Based Short messaging Filtering System," in *Proc. Fifth International Conference on Machine Learning and Cybernetics*, Dalian, August 13-16, 2006.

[10] J. M. G. Hidalgo *et al.*, "Content based SMS spam filtering," in *Proc. the 2006 ACM Symposium on Document Engineering*, Amsterdam, The Netherlands, October 10-13, 2006.

[11] G. V. Cormack *et al.*, "Feature engineering for mobile (SMS) spam filtering," in *Proc. the 30th Annual international ACM SIGIR Conference on Research and Development in information Retrieval*, Amsterdam, The Netherlands, July 23-27, 2007.

[12] M. T. Nuruzzaman and C. Lee "Independent and Personal SMS Spam Filtering," presented at 11th IEEE International Conference on Computer and Information Technology, 2011.

[13] Cormack *et al*., "Spam filtering for short messages," in *Proc. the Sixteenth ACM Conference on Conference on Information And Knowledge Management*, November 06-10, 2007, Lisbon, Portugal .

[14] T. M. Mitchell, *Machine Learning*, McGraw-Hill.

[15] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006,

[16] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*.

[17] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *Journal of Machine Learning Research*, pp. 45-66, 2001.

[18] M. Taufiq, M. F. A. Abdullah, K. Kang, and D. Choi, "A survey of preventing, blocking and filtering short message services (SMS) Spam," in *Proc. International Conference on Computer and Electrical Engineering*, IACSIT, Nov. 2010, vol. 1, pp. 462-466.

[19] T. Michell, "Generative and discriminative classifiers: naive bayes and logistic regression," *Machine Learning*, ch. 1.

[20] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," presented at VLDB, 1994.

[21] P. Madadi, "Text Categorization based on apriori algorithm's frequent itemsets," MSc. thesis, School of Computer Sceince., Howard R. Hughes College of Engineering, University of Nevada, Las Vegas, 2009.

[22] J. Han and M. Kamber, *Data mining Concepts and Techniques*, 2nd Edition, China Machine Press, 2006

Ishtiaq Ahmed received his B.S. degree in computer science and engineering from the University of Dhaka, Bangladesh in 2011. At present, he is pursuing his MS degree in artificial intelligence Lab, Dept. of Computer Engineering, Kyung Hee University, South Korea. His current research interests include Machine Learning, Data Mining, Pattern Recognition and Bioinformatics.

Donghai Guan received his Ph.D. degree in Computer Science from Kyung Hee University, South Korea in 2009. From 2009, he was a post doctoral Fellow at Computer Engineering Department, Kyung Hee University. Since February 2011, he has been an assistant professor in Harbin Engineering University, China. Since March 2012, he has been an assistant professor in Kyung Hee University, Korea. His research interests are machine learning, data mining, activity recognition, and trust management.

Tae Choong Chung received the B.S. degree in electronic engineering from Seoul National University, Republic of Korea, in 1980, and the M.S. and Ph.D. degrees in computer science from KAIST, Republic of Korea, in 1982 and 1987, respectively. Since 1988, he has been with Department of Computer Engineering, Kyung Hee University, Republic of Korea, where he is now a Professor. His research interests include machine learning, meta search, and robotics.