Implementation of Human Cognitive Bias on Naïve Bayes

Hidetaka Taniguchi
Tokyo Denki University
School of Science and Engineering
Hatoyama, Hiki,
Saitama 350-0394, Japan
+81-49-296-5416
htdendai@gmail.com

Tomohiro Shirakawa
National Defense Academy of Japan
Department of Computer Science
1-10-20 Hashirimizu Yokosuka,
Kanagawa 239-8686, Japan
+81-46-841-3810
sirakawa@nda.ac.jp

Tatsuji Takahashi
Tokyo Denki University
School of Science and Engineering
Hatoyama, Hiki,
Saitama 350-0394, Japan
+81-49-296-5416
tatsujit@mail.dendai.ac.jp

ABSTRACT
We propose a human-cognition inspired classification model based on Naïve Bayes. Our previous study showed that human-cognitively inspired heuristics is able to enhance the prediction accuracy of the text classifier based on Naïve Bayes. In the study, our classification model that addresses n-dimensional feature vectors of both categories, showed higher performance than the conventional Naïve Bayes under specific conditions. In this paper, to investigate the mechanism that realizes the higher performance of classification, we further tested our model and its modified variant. As a result, our two models showed slightly different behaviors, but both of them achieved higher performance than the conventional Naïve Bayes.

Categories and Subject Descriptors
I.2.0 [**Artificial Intelligence**]: General - *Cognitive simulation.*

General Terms
Algorithm.

Keywords
Naïve Bayes, Text Classification, Attribute Independence Assumption, Cognition-Inspired model, Bayesian Spam Filtering.

1. INTRODUCTION
Naïve Bayes classifier is one of the most successful machine-learning methods that is widely used for spam-detecting tasks and its conditional assumption is suitable for text data mining. This "naive" assumption sets the conditions that all features are independent given the class and each distribution is estimated as a one-dimensional distribution [1]. And thus, the parameters for each attribute will be learned separately and this greatly simplifies the learning. Therefore Naïve Bayes algorithm is frequently used for the classification with a feature vector of high dimensionality due to its independence assumption simplifies the algorithm especially when the number of attributes is large [2]. Although the independence of attributes is unrealistic, Naïve Bayes classifier shows the superior performance in the text-classification. However, Bayesian classifier will not be optimal when attribute independence does not hold [3]. In such a situation, the assumption of Naïve Bayes is likely to be violated by missing data or uncertainty in feature selection [4,5] and thus the prediction accuracy would be decreased. This problem is triggered by many kinds of factors (e.g. the number of sample data is too small, or data is too much biased to apply the assumption) and difficult to detect the cause of problem among them.

Meanwhile, some studies [6,7,8,9] have indicated that the Human-Cognitively inspired bias is able to enhance the prediction accuracy of machine learning algorithms. This Human-Cognitively inspired model called "Loosely Symmetric (LS) model" introduced by Shinohara et.al. [7] was designed to flexibly adjust the two biases of $symmetry$ and$mutual\ exclusivity$ and exhibits an optimal property breaking the usual trade-off between speed and accuracy. Oyo et.al. [8] reported that LS model is able to describe the human evaluation of co-occurrence information for inductive inference of causal relationships, and developed an excellent heuristics for evaluating options for two-armed bandit problems and Naïve Bayes [9]. Furthermore, we thus assume that the LS model can smoothly adjust biases between classes and factors more than the conventional Naïve Bayes model. In this paper, we propose two kinds of human-cognition inspired classification model named Loosely Symmetric Naïve Bayes (LSNB) model and its variant incorporated with stronger bias named enhanced Loosely Symmetric Naïve Bayes (eLSNB) model.

2. Naïve Bayes Text Classifier
Naïve Bayes is a classification method based on Bayes' theorem. For the text classification, each message is represented as a n-dimensional vector $F = \langle f_1, f_2, \dots, f_n \rangle$ that belongs to category $C = \{c_1, c_2, \dots, c_{|C|}\}$, and the probability $P(c_i|F)$ is calculated as in (1).

$$P(c_i|F) = \frac{P(c_i)P(f_1, f_2, \dots, f_n|c_i)}{P(F)} \quad (1)$$

Where $P(F)$ can be ignored and regarded as a constant because it takes same value for all categories and does not affect the relative values of their probabilities [3].

$$P(c_i|F) = P(c_i)P(f_1, f_2, \dots, f_{|F|}|c_i) = P(c_i)\prod_{j=1}^{n} P(f_j|c_i) \quad (2)$$

Naïve Bayes requires an assumption that every feature in texts is conditionally independent [3]. But this assumption is clearly incorrect because some words are likely to co-occur at the same time (e.g. the word "Roulette" is likely to co-occur with "Casino") [10]. However, this "Naïve" assumption enables the improvement of processing-speed, simplification of the algorithm and reliable performance. For the spam-classifying tasks, the n-dimensional word vector $W = \langle w_1, w_2, \dots, w_n \rangle$ in text T that belongs to

BICT 2015, December 03-05, New York City, United States
Copyright © 2016 ICST
DOI 10.4108/eai.3-12-2015.2262494

category $C = \{spam, ham\}$. Naïve Bayes text classification is performed based on (3). And table 1 shows the criterion of the classification.

$$P(c_i|T) = P(c_i)P(w_1, w_2, \dots, w_n|c_i) = P(c_i)\prod_{j=1}^{|W|} P(w_j|c_i) \quad (3)$$

Table 1. Criterion of the Spam Classification

Text type	*Criterion*
$spam$	$P(spam\|T) > P(ham\|T)$
ham	$P(spam\|T) < P(ham\|T)$

3. Human-Cognitively Inspired NB-Model

3.1 Loosely Symmetric Model

Previous researches [6,7,8,9] have shown the capability of implementing human-cognition inspired model for machine-learning tasks. The well-used model called LS model flexibly adjusts the two biases ($symmetry$ and $mutual\ exclusivity$) and has correlation to human-cognition [7]. The LS model shows the superior performance on machine learning tasks including two-armed bandit problems [8] and Naïve Bayes [9]. It is known that human has illogical symmetric cognitive biases that induces from a proposition "if p then q" its converse "if q then p" and inverse "if $\bar{p}$ then not $\bar{q}$". The LS model quantitatively represents these tendencies [6]. Takahashi et.al. [7] suggested that LS formula can be applied to every area that involves the use of conditional probability. In Table 2, the cells a, b, c and d represent each co-occurrence of p, q,that is, probabilities of co-occurrence; pq, qp, $p\bar{q}$, $\bar{p}q$. LS model describes the relationship between p and q as in (4). Therefore, LS model estimates each distribution as a one-dimensional distribution from a set of n-dimensional feature vectors. We adopted a probabilistic model using the LS model to enhance the prediction accuracy and applied the flexibility to spam-classifier with cognitive features.

Table 2. A 2 × 2 contingency table for causal inference

	q	$\bar{q}$
p	a	b
$\bar{p}$	c	d

$$LS(p|q) = \frac{a + P(p|\bar{q})d}{a + b + P(p|q)c + P(p|\bar{q})d} = \frac{a + \frac{bd}{b+d}}{a + b + \frac{ac}{a+c} + \frac{bd}{b+d}} \quad (4)$$

3.2 Loosely Symmetric Naïve Bayes Model

We implemented Loosely Symmetric model using Naive Bayes classifier. Table 3 shows the co-occurrence table of LSNB. Where $P(w_j|c_i)$ is the co-occurrence of class c_i and word w_j in a document and $P(w_j|\neg c_i)$ is the co-occurrence of the counterpart class of c_i and w_j.

Table 3. A 2 × 2 co-occurrence table of LSNB

	w_j	$\neg w_j$
c_i	a	b
$\neg c_i$	c	d

For example, if c_i is $spam$, $P(w_j|c_i)$ is a word co-occurrence of $spam$, and $P(w_j|\neg c_i)$ is the word co-occurrence of ham, and $P(\neg w_j|c_i)$ and $P(\neg w_j|\neg c_i)$ are the probability that w_j was not observed in c_i or $\neg c_i$. Each co-occurrence a – d is set as in (5)-(8) and the posterior probability calculated by LSNB is as in (9)-(11).

$$a = P(w_j|c_i) \quad (5)$$

$$b = 1 - P(w_j|c_i) \quad (6)$$

$$c = P(w_j|\neg c_i) \quad (7)$$

$$d = 1 - P(w_j|\neg c_i) \quad (8)$$

$$P_{LS}(w_j|c_i) = \frac{a + \frac{bd}{b+d}}{a + b + \frac{ac}{a+c} + \frac{bd}{b+d}} \quad (9)$$

$$P_{LS}(W|c_i) = \prod_{j=1}^{n} P_{LS}(w_j|c_i) \quad (10)$$

$$P_{LS}(c_i|T) = P(c_i)P_{LS}(W|c_i) \quad (11)$$

3.3 Enhanced LSNB Model

We developed a new classification model named enhanced LSNB (eLSNB) model that derived from LSNB model. The eLSNB model has greater symmetric biases than LSNB and is formalize as in (12) to (19). $N(c_i \cap w_j)$ is the frequency of a word, or the number of "counts" that indicates the number of appearance of w_j in c_i. As in (12), the word density of w_j in c_i is represented by $WD(c_i \cap w_j)$ [11]. The purpose of this modification is to enhance the probability of each word on the feature vector that co-occurred in c_i. For example, if w_j was only observed in $spam$ class much more frequently than ham class, w_j should be considered a $spam$ related word, and vice-versa. And thus, eLSNB model is designed to maintain stronger biases for the binary classification. Each co-occurrence is strongly biased by $WD(c_i \cap w_j)$ as in (13)-(16), and the posterior probability calculated by LSNB is as in (17)-(19).

$$WD(c_i \cap w_j) = \frac{N(c_i \cap w_j)}{\sum_{k=1}^{|W|} N(c_i \cap w_k)} \quad (12)$$

$$a = P(w_j|c_i)WD(c_i \cap w_j) \quad (13)$$

$$b = \left(1 - P(w_j|c_i)\right)WD(\neg c_i \cap w_j) \quad (14)$$

$$c = P(w_j|\neg c_i)\, WD(\neg c_i \cap w_j) \quad (15)$$

$$d = \left(1 - P(w_j|\neg c_i)\right)WD(c_i \cap w_j) \quad (16)$$

$$P_{eLS}(w_j|c_i) = \frac{a + \frac{bd}{b+d}}{a + b + \frac{ac}{a+c} + \frac{bd}{b+d}} \quad (17)$$

$$P_{eLS}(W|c_i) = \prod_{j=1}^{n} P_{eLS}(w_j|c_i) \quad (18)$$

$$P_{eLS}(c_i|T) = P(c_i)P_{eLS}(W|c_i) \quad (19)$$

4. Experimental Settings

4.1 Benchmark Corpora

We tested our LSNB models and Naïve Bayes model using six email corpuses, Ling-Spam, SpamAssassin, PU1, PU2, PU3 and PUA. Table 4 shows the number of spam and ham messages and the spam ratio of each corpus. The Ling-Spam [17,18] corpus consists of 2412 of non-spam messages (ham) and 481 of spam messages from Linguist list. We used *lemm* version of Ling-Spam corpus for the experiment. SpamAssassin [19] corpus consists of 3900 of non-spam messages and 1897 of spam messages. These messages are divided into 4 directories; 2442 of non-spam messages for *easy_ham* directory, 493 of spam-messages for *spam* directory, 1401 messages for *easy_ham2* directory and 1397 spam messages for *spam2* directory. We used *easy_ham* and *spam* directories as sample data and *easy_ham2* and *spam2* directories for test data, and the spam ratio of sample data was 26%. PU corpus [18] consists of non-spam and spam messages that have been tokenized due to the security policy, and each word is expressed as numbers.

Table 4. Corpuses used in the experiments

Corpus	Ham	Spam	Spam Ratio
Ling-Spam	2412	481	17%
SpamAssassin	3900	1897	33%
PU1	618	481	44%
PU2	579	142	20%
PU3	2313	1826	44%
PUA	571	571	50%

4.2 Class Prior Probability

The prior probability is typically estimated by dividing the number of training examples of category c_j by the total number of training examples [12,13]. However, since we partly used the limited numbers of training examples for the experiment, the prior probability hardly affects the classification and assuming uniform priors can improve the classification accuracy [14]. Therefore, the prior probability for the binary classification set to be equivalent: $P(spam) = 0.5$, $P(ham) = 0.5$.

4.3 Data Preparation

For text classification, feature selection is a necessary step due to the high dimensionality of feature vector. First, we removed punctuation and words that occurred only once, and that are in a standard stop word list [15] from the feature vector. Also we removed numbers from the feature vector except for PU corpus that expressed as integers. We only use "White Space" and "Line Break" as separators between the words. For the treatment of missing values, we adopt a simple method, replacing by a default value as "missing". This is really simple, however, Robert [16] reported that the model handling missing values by treating "missing" as a legitimate value showed better results than the models with more difficult rules.

5. Results and Discussion

We tested NB, LSNB and eLSNB models using 6 corpuses. For the SpamAssassin test, each model classified the entire email of *easy_ham2* and *spam2* directories. For the tests using Ling-Spam and PU, we combined all directories for the experiment and used randomly chosen data as sample, and classified the rest of data. The number of training data is given by $spam = 4 + t * 8$, $ham = 20 + t * 40$, where $[t|0 \leq t \leq 17]$ and to simplify the experiments, the spam ratio of sample data was always 17%. The scores displayed in Figures 1-24 indicate the average of 30 results. The classification results from 6 corpuses are shown in Figures 1-18. They indicate the accuracy of the $spam$ classification, ham classification and the average of them by NB, LSNB and eLSNB with each database. Figure 19-24 indicate the values of F-measure in each test.

LSNB and eLSNB showed better performances in $spam$ classification than NB in almost all the experiments. We suppose this is because LSNB and eLSNB classifiers can refer each word w_j from both categories and this implementation yields more biases between words and categories than the NB classifier. Such an effect is enhanced in eLSNB, and this seems to increase the performance of eLSNB in $spam$ classification compared to LSNB. Also the F-measure scores of LSNB and eLSNB showed better learning efficiency than the NB classifier. This fact indicates that LSNB and eLSNB can learn more effectively from a small number of sample data compared to the NB classifier.

Meanwhile, LSNB and eLSNB did not significantly improve the prediction accuracy of ham classifications on every corpus except SpamAssassin, though LSNB has slightly improved performance than the other two models. We suppose this is because ham documents did not contain "trigger words" like $spam$ documents, and LSNB and eLSNB could not effectively adjust biases between categories and documents. It is particularly prominent in the results from eLSNB that is supposed to mistakenly adjust biases of harmless words from $spam$ category, and thus the prediction accuracy did not improve significantly. However, LSNB and eLSNB models substantially improved average of the classification accuracy and F-measure on every corpus and the results indicate that the implementation of human cognitive bias contributes to the enhancement of the prediction.

6. Conclusion

We have introduced a modified Naïve Bayes model by implementing the human-like causal inference, and our model showed its effectiveness in text classification. The main purpose of this study was to extensively test the performance of LSNB in our previous study and improve the prediction accuracy of LSNB model by some modifications. As a result, our new model named eLSNB performed the best score in Ling-Spam, Spam Assassin and PUA classifications. Both of LSNB and eLSNB showed better score than NB, and between them, eLSNB was better in spam classification and LSNB was better in ham classification.

In future work, we will elucidate the reason why our model did not improve the prediction accuracy in ham classification and detect the composition difference between $spam$ and ham documents. Also, we will try to minimize the number of sample data since LSNB and eLSNB are supposed to be learned effectively from a small number of sample data as compared to NB classifier [9] and measure execution time and resource consumption. In order to improve the prediction accuracy of both categories, we will modify and improve our models to adapt to any training corpus.

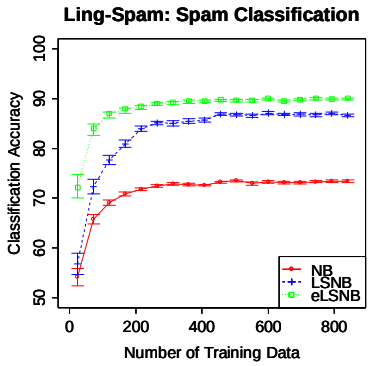

Figure 1. Spam classification results of Ling-Spam

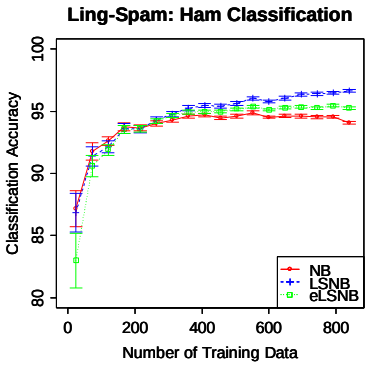

Figure 2. Ham classification results of Ling-Spam

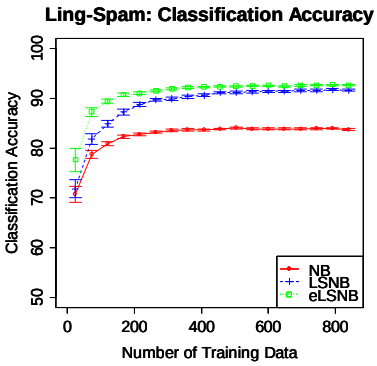

Figure 3. Average of Spam and Ham results of Ling-Spam

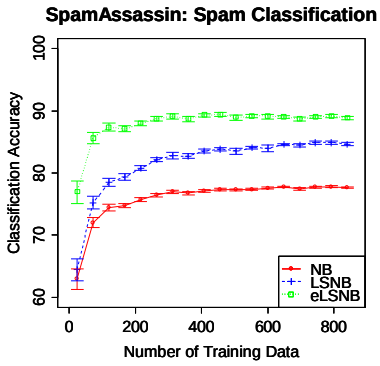

Figure 4. Spam classification results of SpamAssassin

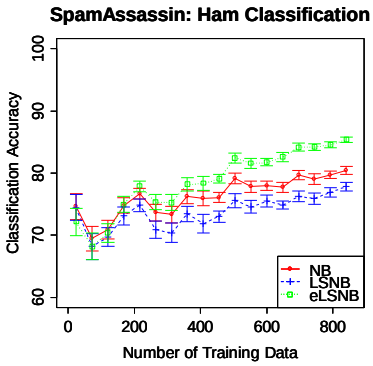

Figure 5. Ham Classification results of SpamAssassin

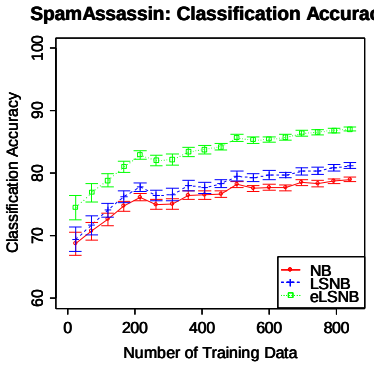

Figure 6. Average of Spam and Ham results of SpamAssassin

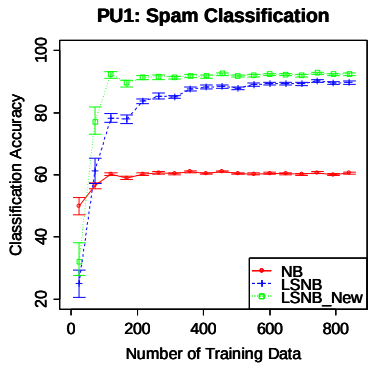

Figure 7. Spam classification results of PU1

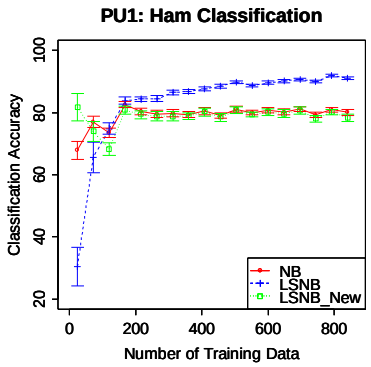

Figure 8. Ham classification results of PU1

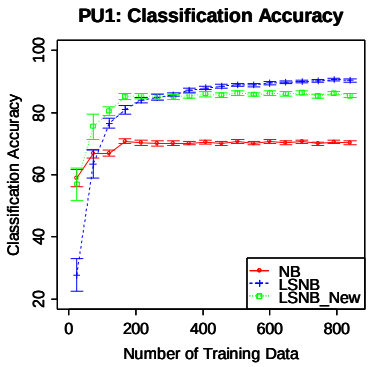

Figure 9. Average of Spam and Ham results of PU1

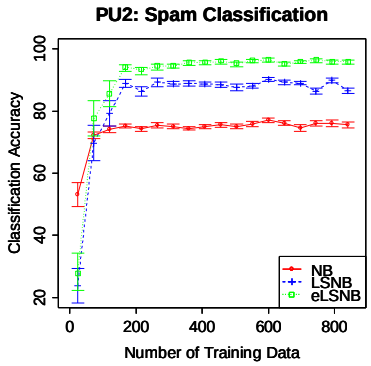

Figure 10. Spam classification results of PU2

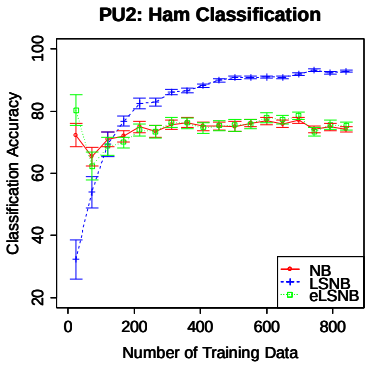

Figure 11. Ham classification results of PU2

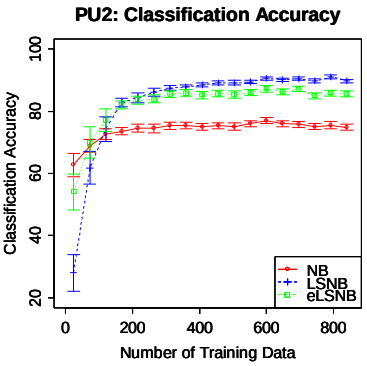

Figure 12. Average of Spam and Ham results of PU2

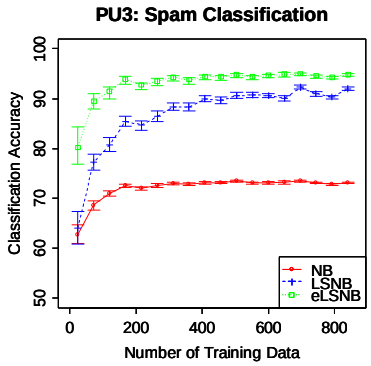

Figure 13. Spam classification results of PU3

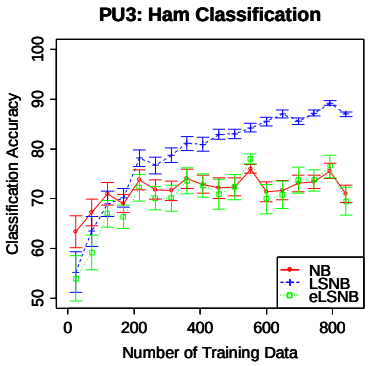

Figure 14. Ham classification results of PU3

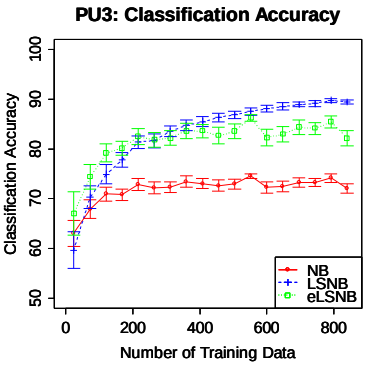

Figure 15. Average of Spam and Ham results of PU3

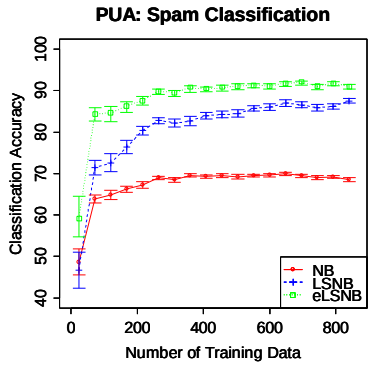

Figure 16. Spam classification results of PUA

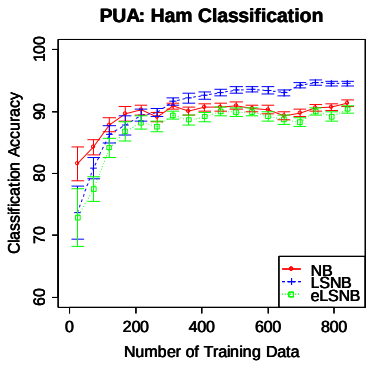

Figure 17. Ham classification results of PUA

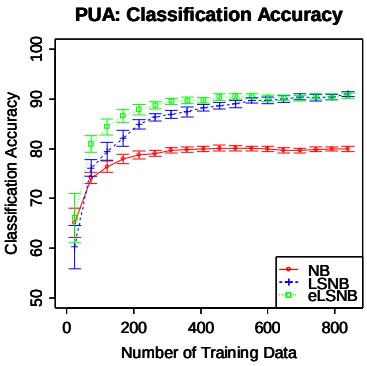

Figure 18. Average of Spam and Ham results of PUA

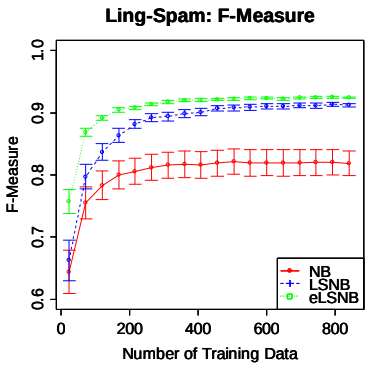

Figure 19. F-measure of Ling-Spam

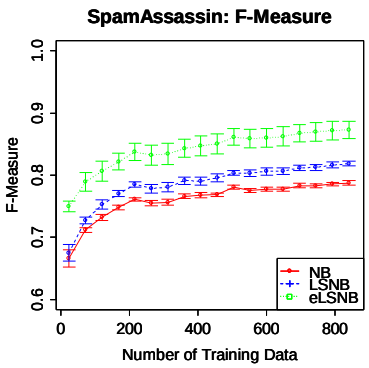

Figure 20. F-measure of SpamAssassin

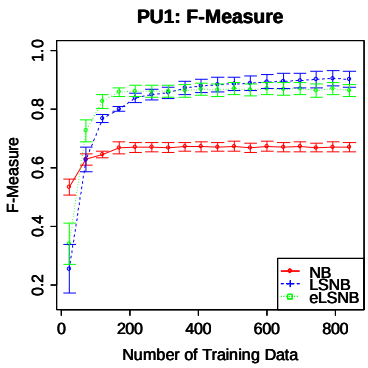

Figure 21. F-measure of PU1

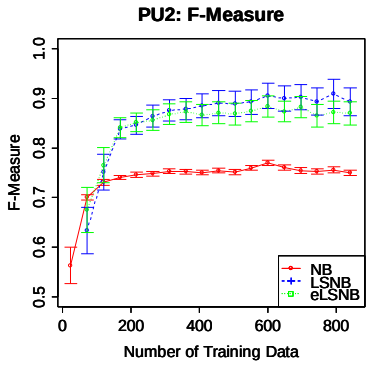

Figure 22. F-measure of PU2

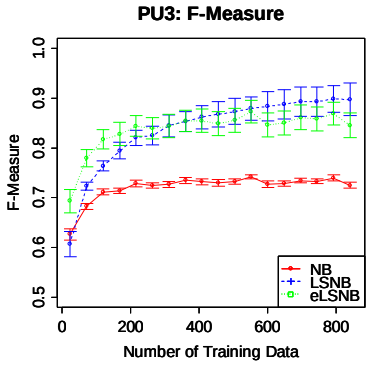

Figure 23. F-measure of PU3

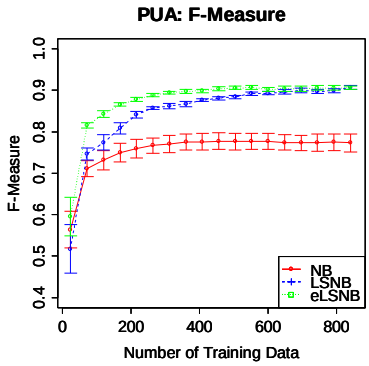

Figure 24. F-measure of PUA

7. ACKNOWLEDGMENTS

We would like to thank Dr. Hiroshi Sato of National Defense Academy of Japan and his students for their encouraging supports and comments.

8. REFERENCES

[1] Geoff Dougherty. 2013. *Pattern Recognition and Classification-An Introduction*. Springer., New York, USA.

[2] Andrew McCallum and Kamal Nigam. 1998. *A comparison of event models for Naive Bayes text classification*. In *Proceedings of the AAAI-98 Workshop on Learning for Text Categorization*, pp. 41-48.

[3] Pedro Domingos and Michael Pazzani. 1997. *On the Optimality of the Simple Bayesian Classifier under Zero-One Loss*. In *Machine Learning*, v.29 n.2-3, Nov./Dec., pp.103-130.

[4] M. C. Monard and G. E. A. P. A. Batista. 2002. *Learning with Skewed Class Distribution*. In *Advances in Logic, Artificial Intelligence and Robotics*, Sao Paulo, SP, IOS Press, pp. 173-180.

[5] Daniele Soria, Jonathan M. Garibaldi, Federico Ambrogi, Elia M. Biganzoli and Ian O. Ellis. 2011. *A 'non-parametric' version of the naive Bayes classifier*. In *Knowledge-Based Systems*, Volume 24, Issue 6, pp. 775–784.

[6] Shuji Shinohara, Ryo Taguchi, Kouichi Katsurada and Tsuneo Nitta. 2007. *A Model of Belief Formation Based on Causality and Application to N-armed Bandit Problem*. In *Trans. Jpn. Soc. Artif. Intell*, 22(1), pp. 58–68.

[7] Tatsuji Takahashi, Kuratomo Oyo and Shuji Shinohara. 2011. *A Loosely Symmetric Model of Cognition*. In *Advances in Artificial Life. Darwin Meets von Neumann*, Volume 5778 of the series Lecture Notes in Computer Science, pp. 238-245.

[8] Kuratomo Oyo and Tatsuji Takahashi. 2013. *A Cognitively Inspired Heuristic for Two-armed Bandit Problems: The Loosely Symmetric (LS) Model*. In *Procedia Computer Science*, Volume 24, pp. 194–204.

[9] Hidetaka Taniguchi, Kuratomo Oyo, Yu Kohno and Tatsuji Takahashi. 2015. *Causal cognition and spam classifier*. In *PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON NUMERICAL ANALYSIS AND APPLIED MATHEMATICS 2014 (ICNAAM-2014)*, Volume 1648, pp. 580002-1-4.

[10] Jon Kågström. 2005. *IMPROVING NAIVE BAYESIAN SPAM FILTERING*. In *Master thesis of Mid Sweden University Department for Information Technology and Media*.

[11] Ashwini Madane and Devendra Thakore. 2012. *An Approach for Extracting the Keyword Using Frequency and Distance of the Word Calculations*. In *International Journal of Soft Computing and Engineering (IJSCE)*, Volume-2, Issue-3, ISSN: 2231-2307.

[12] Tom Mitchell. 1997. *Machine Learning*. McGraw Hill., New York, ISBN: 0070428077.

[13] Vangelis Metsis, Ion Androutsopoulos and Georgios Paliouras. 2006. *Spam filtering with Naive Bayes-Which Naive Bayes?* In *Third Conference on Email and Anti-Spam (CEAS)*.

[14] Karl-Michael Schneider. 2005. *Techniques for Improving the Performance of Naive Bayes for Text Classification*. In *Proceedings of CICLing*.

[15] Gerard Salton. 1989. *Automatic text processing: the transformation, analysis, and retrieval of information by computer*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[16] Robert C. Holte. 1993. *Very Simple Classification Rules Perform Well on Most Commonly Used Datasets*. In *Machine Learning*, April 1993, Volume 11, Issue 1, pp. 63-90.

[17] Ion Androutsopoulos, John Koutsias, Konstantinos V. Chandrinos, George Paliouras and Constantine D. Spyropoulos. 2000. *An evaluation of naive bayesian anti-spam filtering*. In *Proceedings of the Workshop on Machine Learning in New Information Age*, Barcelona, Spain.

[18] Ion Androutsopoulos, Georgios Paliouras and Eirinaios Michelakis. 2004. *Learning to filter unsolicited commercial e-mail*. In *Technical Report 2004/2, NCSR "Demokritos". Revised version*.

[19] "SpamAssassin Public Corpus", [online] 2003, http://spamassassin.apache.org/publiccorpus (Accessed: 24 October 2015).