applied sciences

MDPI

Article

Gender Classification Using Sentiment Analysis and Deep Learning in a Health Web Forum

Sunghee Park and Jiyoung Woo *

Department of Future Convergence Technology, Soonchunhyang University, Asan-si, Korea; sunghee@sch.ac.kr
* Correspondence: jywoo@sch.ac.kr

Received: 15 February 2019; Accepted: 19 March 2019; Published: 25 March 2019

Featured Application: This work can be applied to detect the gender of online users who do not disclose this information.

Abstract: Sentiment analysis is the most common text classification tool that analyzes incoming messages and tells whether the underlying sentiment is positive, negative, or neutral. We can use this technique to understand people by gender, especially people who are suffering from a sensitive disease. People use health-related web forums to easily access health information written by and for non-experts and also to get comfort from people who are in a similar situation. The government operates medical web forums to provide medical information, manage patients' needs and feelings, and boost information-sharing among patients. If we can classify people's emotional or information needs by gender, age, or location, it is possible to establish a detailed health policy specialized into patient segments. However, people with sensitive illness such as AIDS tend to hide their information. Especially, in the case of sexually transmitted AIDS, we can detect problems and needs according to gender. In this work, we present a gender detection model using sentiment analysis and machine learning including deep learning. Through the experiment, we found that sentiment features generate low accuracy. However, senti-words give better results with SVM. Overall, traditional machine learning algorithms have a high misclassification rate for the female category. The deep learning algorithm overcomes this drawback with over 90% accuracy.

Keywords: sentiment analysis; gender classification; machine learning; deep learning; medical web forum

1. Introduction

Sentiment analysis is contextual mining of text that identifies and extracts subjective information in source material, helping a business to understand the social sentiment of their brand, product, or service while monitoring the online conversation. With recent advances in machine learning, text mining techniques have improved considerably. Creative use of advanced artificial intelligence techniques can be an effective tool for doing in-depth research.

Sentiment analysis of web content is becoming increasingly important due to augmented communication through Internet sources such as e-mail, websites, forums, and chat rooms. By collecting these articles and analyzing people's emotions expressed in them, we can figure out people's feelings and opinions about polices, products, brands, and so on. Compared to traditional surveys and other research techniques, this rich information can be obtained with less cost and effort. People can search for particular information based on their individual needs. Patients, patients' significant others, or caregivers also use health-related web forums to get health and medical information and to get comfort from people who are similar to themselves. They also ask questions about the disease and find information that is easy to understand [1]. The medical forum also reflects

their feelings [2]. The government relies on web forums to act as a helpdesk to promote people's well-being. Some governments directly operate medical web forums to provide medical information, get to know patients' needs, manage their emotions, and help people share information [3]. Users divulge some pieces of personal information when they sign up, and this information can be used to establish national health care policies. However, for sensitive disease such as AIDS, patients tend not to expose their personal information. In terms of health policy, it is important to know who is looking for more information. If we can guess the information from the non-disclosed data, we can understand who is suffering from a specific disease and what types of worries they might have. Especially, in sensitive diseases, by understanding patients we can help them deal with their situation properly and adapt to society. The analysis of AIDS patients' communication could be useful from a government perspective, too.

Categorizing and analyzing according to demographic information such as gender, age, and region is essential to obtain information such as consumers' emotions, values, and attitudes in all areas of marketing. In particular, it is important to distinguish gender for detailed policy establishment because the situation and necessity can be different by gender. It can also help users to see what topics are most talked about by males and females, and what services are liked or disliked by men and women [4]. Knowing this information is crucial for market intelligence because the information can be used in targeted advertising and service improvement.

Our study is based on a real-life application of a web forum. In this study, we collect data from the AIDS-related bulletin board at Healthboards.com [5], which is one of the top 20 health websites according to Consumer Reports Health Web Watch. Under the premise that "Emotions expressed by men and women will be different," we propose a model that extracts words and emotions from text messages and distinguishes the gender. By establishing a learning model using gender information, we figure out unclassified gender information and identify the gender awareness of AIDS patients by gender.

2. Related Works

In most previous studies on gender classification, the various features fed to machine learning algorithms. At the Islamic Women's political forum, Zhang et al. built a machine learning model that classifies genders by properly combining the characteristics of vocabulary, syntax, structure, uni-gram, and bi-grams [6]. In the study of Ryu et al. [7], they used logistic regression and SVM (a support vector machine) to estimate the gender, age, and location of Twitter users. Wang et al. [8] suggested research based on the Latent Dirichlet Allocation (LDA) model using text data from Facebook or Twitter, in which women have been shown to use a lot of personal themes while men tend to post a lot of philosophical or informative text. In the study of Na and Cho [9], they surveyed the emotions of male and female college students and showed that gender can be distinguished by analyzing emotions using Fastcluster.

In the study of Yan et al. [10], they presented a naïve Bayes classification approach to identify the gender of weblog authors. They used weblog-specific features such as webpage background colors and emoticons. They report an F-measure of around 64% using their features. Mukherjee and Liu. [11] proposed two new techniques. They used their own POS Sequence mining algorithm and an Ensemble feature selection technique, and achieved an accuracy of 88%.

Pennacchiotti and Popescu [12] proposed a user classification model applying machine learning algorithm to the feature set including user profile, user tweeting behavior, linguistic content of user messages, and user social network features. They explored the feature importance and found that linguistic features are consistently effective in user classification.

Dwivedi et al. [13] present two systems, a manual feature extraction system and a deep learning method, to automatically classify the gender of blog authors. For the deep-learning-based model, they apply a Bidirectional Long Short-Term Memory Network. Barlte and Zheng [14] report an

accuracy of 86% in gender classification on blog datasets by applying deep learning models based on the Windowed Recurrent Convolutional Neural Network (WRCNN).

Filho et al. [15] proposed textual meta-attributes, taking into account the characters, syntax, words, structure, and morphology of short, multi-genre, content-free text posted to Twitter to classify an author's gender via three different machine learning algorithms. The novel contribution of this work is to employ a word-based meta-attribute such as Ratio between hapax dislegomena (a word that appears only twice in a whole text) and the total number of words. Furthermore, they developed a textual morphology based on the meta-attributes from textual structures such as the ratio between the number of pronouns and the total number of words. This work achieved 81% accuracy, but the performance by each class is not presented. Garibo-Orts [16] proposed a statistical approach to the task of gender classification in tweets. The statistical features include skewness, kurtosis, and central moments; these statistical features make the learning algorithm context-free and language-independent.

Recently, Convolutional Neural Networks (CNNs) have also been successful in various text classification tasks. Kim [17] showed that a simple CNN with little hyper-parameter tuning and static vectors achieves excellent results. Severyn et al. [18] proposed a deep learning model for Twitter sentiment classification. They advanced a CNN model to adjust word embedding using unsupervised learning.

Based on previous works, females and males are found to have differences in terms of the vocabulary used and the emotional expression. During a literature review, we found some limitations in the current literature. First, most recent works focus on Twitter or weblogs. Even though the gender information disclosed in the medical web forum, especially on sexually transmitted diseases, is important, we hardly found any work dealing with medical web forums. Secondly, studies using sentiment features derived from text are lacking, while word features are explored a lot. Thirdly, deep learning is applied a lot to text classification, but its application to gender classification from textual information is rare.

In this work, we aim to figure out how the sentiment features derived from emotions expressed in articles work in gender classification. We will do this by using both machine learning and deep learning methods.

3. Experiment

We retrieved the users' gender information from an AIDS-related medical web forum and presented a gender detection model that classifies gender based on the emotions expressed in posts and comments. We developed the sentiment feature set that expresses how often the posts contain emotions, and assessed the emotional complexity, which is the number of emotion categories shown in a post. Using vocabulary characteristics and emotions from the disclosed data, we built machine learning models and measured the accuracy, varying the feature sets to select the best model. The proposed framework is presented in Figure 1.

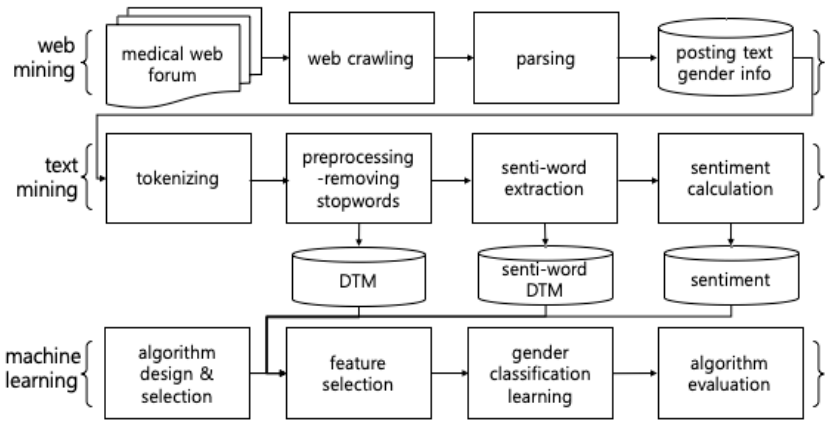

Figure 1. Gender classification model for medical web forum.

3.1. Dataset

From 14 November 2000 to 25 February 2010, we collected 3451 posts and 18,944 comments—in total, 22,395 messages—from an AIDS-related bulletin board from HealthBoard.com. It is one of the first independent health community websites and has more than 280 bulletin boards for communication about diseases. As shown in Table 1, this dataset has 8954 male-written messages and 6078 female-written messages. A total of 3282 users participated in the AIDS bulletin board, with 849 female users and 1054 male users. This implies that 1379 users have not disclosed their gender. The average post length is 113 words for men and 106 words for women. The percentage of users who disclosed gender information is 59% for males and 41% for females; 33% of the total has no gender information. As for the Alzheimer's-related board, only 1708 (6.2%) out of 27,874 (posts: 2486, comments: 25,134) have not disclosed their gender. The number of users in the Alzheimer's bulletin board is 1498. The number of female users is 924 and male users 185. In total, 1109 users have gender information. Compared to the users in the Alzheimer's-related board, users in the AIDS-related board have a low ratio of disclosed gender information.

Table 1. The gender information disclosed in an AIDS-related web forum.

Gender	Numbers of Messages	Ratio (%)
Male	8954	40
Female	6078	27
N/A	7363	33
Total	22,395	100

3.2. Feature Extractions

Before feature extraction, we first perform data preprocessing. We first converted everything to lowercase, removing punctuations, numbers, and stopwords. Stopwords are words that have no special meaning but are frequently used. For example, in English, "a," "the," and "is" are examples of stopwords. In this work, we excluded these words to reduce the feature set in text mining, with the assumption that the stopwords are no different in distribution between women and men.

Then, we build the tokenizer only on the training data. We could add the testing data, but results may go up or down. Testing is needed to determine whether or not adding the testing data will help. In most text classification studies, words are the most important features. Here are the features that we considered in this work.

3.2.1. Unigram

We extracted the word features from the text to construct a learning model. The first step was to derive words after preprocessing. We used the Bag-of-Words technique to tokenize each word into a DocumentTermMatrix that indicates how many times the word appears in each article. The Bag-of-Words technique transforms the sentence into a numeric vector as follows. First, the word set is built by collecting words from all sentences in all documents. Then, a sentence is expressed as the word counts in word order. For example, in "symptoms appear after at least two" and "it was two years of unnecessary stress," these two sentences generate the word set, {symptoms, appear, after, at, least, two, it, was, years, of, unnecessary, stress}. Two sentences are expressed as the counts of each word as {1,1,1,1,1,1,0,0,0,0,0,0} and {0,0,0,0,0,1,1,1,1,1,1,1}, respectively. The numeric vector is derived for a document by aggregating the word count for all sentences in a document. The document term matrix (DTM) is represented in Figure 2.

DocumentS	Terms								
		Condom	Get	Give	Good	Help	Hiv	however	impossible	...
	Post1	1	2	1	1	1	1	1	1	
	Post2	0	1	0	0	1	1	0	0	
	Post3	0	1	0	0	2	11	0	0	
	Post4	0	5	0	2	1	3	0	0	
	...									

Figure 2. Representation of DocumentTermMatrix.

We extracted the document term matrix in two ways, the word occurrence and the occurrence frequency, to test which feature extraction method is efficient.

To reduce the feature set, we set the threshold value to the number of appearances for extracting word features. We incrementally increased the threshold value from 5 to 15 by 5s and checked the classification performance.

3.2.2. Sentiment Features

To derive the sentiment features, we used tidytext, which is a dictionary package built in R, to measure the rate of emotions. There are several dictionaries, but we used NRC [19] and BING. The NRC dictionary has 10 categories: trust, fear, negative, sadness, anger, surprise, positive, disgust, joy, and anticipation. Based on this, we developed sentiment-related features: the number of emotion types expressed in a message, objectivity (1-number of used emotion types/total emotion (10)), and emotional complexity (the number of emotion types/10).

The Bing dictionary classifies words only into positive or negative, and has 6788 words. Based on the Bing dictionary, we calculated the positive rate (the number of positive words/the total number of words), negative rate (the number of negative words/the total number of words that contain emotions), and the total number of words.

As shown in Tables 2 and 3, women use the words 'shine,' 'thank,' 'bless,' and 'glad' (positive words) and 'problem,' 'scary,' and 'illness' (negative words) about twice as often as men. On the other hand, men use the words 'accurate,' 'important,' 'receptive' (positive words) and 'issue,' 'fever,' and 'aches' (negative words) more than twice as often as woman.

Table 2. Differences in positive words between men and women.

Word	Men	Women
Shine	1	419
Thank	655	714
Glad	154	233
Bless	168	267
Accurate	350	166
Important	196	76
Correct	149	54

Table 3. Differences in negative words between men and women.

Word	Men	Women
Problem	246	337
Scary	80	147
Illness	144	211
Issue	565	179
Fever	358	140
Aches	184	39

3.3. Classification Algorithms

We employed a representative machine learning algorithm for text classification.

3.3.1. Naïve Bayes

Naïve Bayes is the first approach we tried. Let $C = (c_1, c_2)$ be the gender class, and $F = (f_1, f_2, \ldots f_n)$ are features, according to Bayes's theorem:

$$P(c|F) = \frac{P(c)P(F|c)}{P(F)}. \quad (1)$$

The naïve Bayes assumption is that:

$$\hat{P}(F|c) = \prod_{i=1}^{n} \hat{P}(f_i|c). \quad (2)$$

Based on the naïve Bayes assumption, the probability of belonging to each class is as follows:

$$\hat{P}(F|c) = \underset{c}{\text{argmax}}\text{P(C = c|F)} = \underset{c}{\text{argmax}}\text{P(C|c)} \prod_{i=1}^{n} P(f = f_i|C = c). \quad (3)$$

The basic assumption is that all features are independent. This assumption is too strong, but it works well in reality.

3.3.2. Support Vector Machine

The support vector machine is known as a powerful algorithm until deep learning algorithm replaced it [20]. Various lines can be drawn between classes A and B. If new data comes in and overlaps with a line that is on one side, it is hard to judge whether those data belong to class A or class B. Therefore, the best way to increase the classification accuracy is to deploy a line in a center between groups with the maximum distance from the line to groups. The maximum distance from the line to groups is called the margin. At the maximum margin, the middle boundary is called the optical hyper-plane.

3.3.3. Random Forest

Random forest is an algorithm that generates multiple decision trees by randomly sampled data for learning and assembles the results of the decision trees by majority voting. It is a type of ensemble leaning method that implements classification, regression, and clustering based on group learning. Random forest trains the model using the subset of features and the subset of data in a repetitive way. This operation reduces the risk of overfitting. This algorithm has been widely used in various applications in recent years and has been proved to outperform other algorithms.

3.4. Deep Learning Approach

The deep learning algorithm is a kind of machine learning. Machine learning focuses on solving real-world problems with neural networks created by mimicking the decision-making processes that take place in the human brain by adopting some of the main ideas of artificial intelligence. Deep learning focuses more on the tools and techniques of machine learning. The difference between machine learning and deep learning is that, in the course of machine learning, people are still involved in the process of extracting features from the given data to train the machine, but deep learning uses the given data as input data. It does not feature an engineering process to extract the features of training by human intervention, but automatically performs learning of important features in the data itself. So deep learning is called end-to-end machine learning. This is because the machine learns from beginning to end, by itself, without human intervention.

3.4.1. Convolution Neural Network

CNNs are widely used on image datasets [21,22], but researchers have found that they work great with text, too. Previous works showed that CNN outperforms the state-of-art algorithm in many cases [21,22] That is because text can be considered a 1D image. Now, the explanation for why CNNs work for images is quite complicated, but they (partly) revolve around something called 'spatial locality'. What this means is that CNNs can find patterns not just from each feature (e.g., a pixel) but from locations of pixels (e.g., neighborhoods of pixels correlated with another). This makes sense because images are not a random collection of pixels, but pixels are connected across the image.

3.4.2. Convolution Neural Network (CNN) for Text Classification

To make the sentences in text into an image, we need to encode words with numeric values. Word embedding is employed to express words as numeric values. The method that assigns a unique integer value to each word would cause the dimension explosion. Word embedding set the dimensionality, k, and expresses the word using k-dimensional values. For example, when k is set to 5, "hospital" is expressed as {0.1,0.2,0.3,0.8,0}. This embedding method dramatically reduces the dimensions. However, this method does not consider the relationship between words when placing a word in a matrix. The advanced technique is to consider the relationship between words when transforming the input data into an input matrix like an image. Word2Vec considers that words that appear in a similar context have similarities in semantic meaning. We adopted Google's word2Vec [23], which is a pretrained model from the GoogleNews dataset.

We performed the experiment using word-embedding method and Word2Vec method as well.

The architecture of the text classification is shown in Figure 3. We transformed words into a k-dimensional vector. We set the number of words to make an input matrix, denoted as n. A sentence is tokenized and n words organize the input data. At the end of the sentence, there could not be enough words to fill the input. In this case, we pad the rows with zeros until the number n is filled.

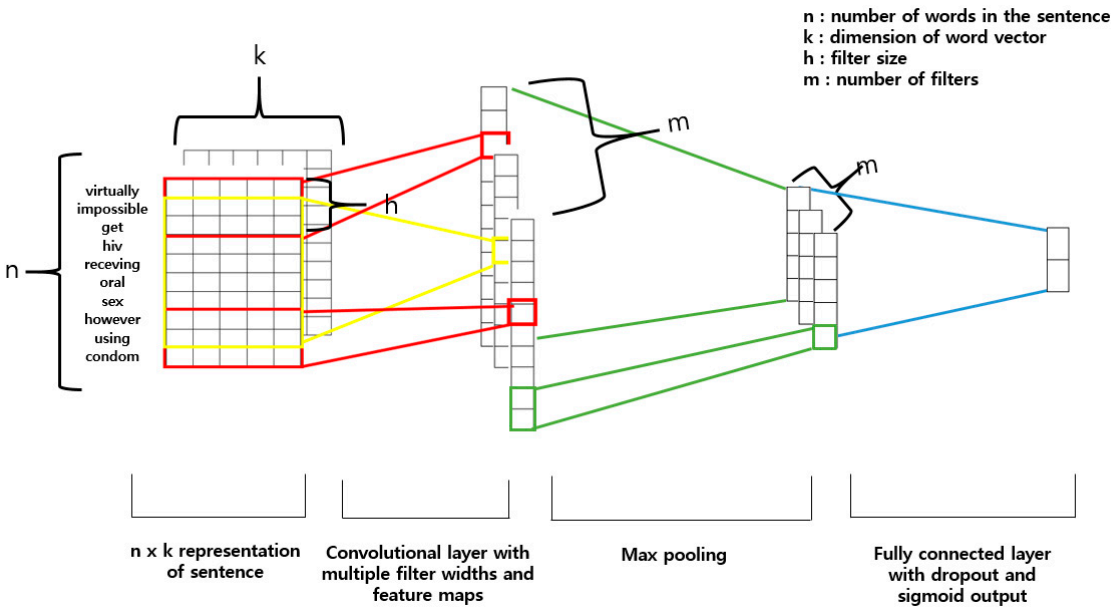

Figure 3. Model architecture with an example sentence.

To derive abstract features from input data and make the algorithm derive useful features automatically, we employ a hidden layer and a convolutional layer to the network.

In a convolution layer, we feed the input matrix to the filter of k*h size, then the input of h*k size is mapped to a cell in the convolutional layer. In Figure 2, the rectangle with a red line is a 3*k filter. The filter works as an aggregator of h words. The filter needs to keep the word meaning when

aggregating words, so the horizontal size of the filter is set to k. As a result, the convolutional layer is one-dimensional. The number of nodes in a convolution layer is calculated as n-h + 1.

The vertical size of the filter can be varied. In this work, we test two sizes of filter, 3 (in red) and 8 (in yellow). The number of filters depends on the user's decision. More filters are needed to enrich features. The max pooling layer is also employed to avoid overfitting a result of too much features. The max pooling layer adopts a filter with the size of (n-h + 1)*1 and finds the maximum value for all values in a precedent convolutional layer. The max pooling layer generates one-dimensional m rows. Finally, m rows are fully connected to the output. The output layer has two nodes to express bi-class, gender in our case. Two nodes are filled with 1 or 0, it indicates male or female respectively.

4. Results

4.1. Traditional Machine Learning Results

Initially, 70% of data, including gender information, were used as training data and 30% as test data. We experimented with naïve Bayes (NB), support vector machine (SVM), and random forest (RF), which are representative algorithms proven to outperform other algorithms. The precision measure and the recall measure are calculated for each class. For the female class, the precision is calculated as TP/(TP + FP) and recall is calculated as TP/(TP + FN). For the male class, the precision and recall are calculated as FN/(FN + TN) and TN/(FP + TN), respectively. The positive class is defined as the class of interest. In this work, the female class and male class are treated equally, but the positive class is set as the female class. The accuracy, the true positive rate, and the true negative rate are calculated considering two classes. Accuracy indicates how well a binary classification test correctly identifies each class. False positive rate is calculated as FP/(FP + TN) and indicates the ratio of the wrongly classified females among the males. False negative rate is calculated as FN/(FN + TP). Using only sentiment features, overall, the female class has poor performance. Three algorithms did not recognize the female classes well and classified most of the users into the male class. This indicates that sentiment features such as emotion types and their density are not effective for classifying authors' gender on their own.

To test the effects of sentiment features, we performed three experiments as follows. At first, we only built a sentiment feature set including the ratio of 10 types of emotional occurrences, objectivity, and emotional complexity. The accuracy of them are 58.33% (NB), 60.86% (SVM), and 58.66% (RF), respectively, as shown in Table 4. Compared to the male class, the female class has a poor performance. Over 50% of female users were classified into the male class.

Table 4. The accuracy of gender detection using sentiment feature set.

Algorithm / Real	NB		SVM		RF	
	Female	**Male**	**Female**	**Male**	**Female**	**Male**
Female	174 (True positive: TP)	200 (False negative: FN)	81	42	402	413
Male	1679 (False positive: FP)	2456 (True negative: TN)	1723	2663	1451	2243
Precision	9.39%	92.47%	4.49%	98.45%	21.69%	84.45%
Recall	46.52%	59.40%	65.85%	60.72%	49.33%	60.72%
Accuracy	58.33%		60.86%		58.66%	
False positive rate	40.60%		39.28%		39.28%	
False negative rate	53.48%		34.15%		50.67%	

The second experiment was performed using senti-word features. Instead of calculating sentiment value from senti-word, we checked the senti-word occurrence. Among the 7954 words that appeared more than five times each, we selected only 1594 senti-words as features that are in the NRC dictionary. The accuracy increased a little bit with the SVM and RF algorithms, as shown in Table 5. The precision for the female class improved. The senti-words used to express the author's emotion were comparatively effective for gender classification. Among the three algorithms, SVM had the highest performance for the sentiment feature set and the senti-word feature set.

Table 5. The accuracy of gender detection using senti-word feature set.

	NB		**SVM**		**RF**	
	Female	**Male**	**Female**	**Male**	**Female**	**Male**
Female	1706	2434	701	236	514	210
Male	117	252	1122	2450	1309	2470
Precision	93.58%	9.38%	38.45%	91.21%	28.20%	92.16%
Recall	41.21%	68.29%	74.81%	68.59%	70.99%	65.36%
Accuracy	43.42%		69.88%		66.27%	
False positive rate	31.71%		31.41%		34.64%	
False negative rate	58.79%		25.19%		29.01%	

The final experiment was performed using the entire feature set, combining word features and sentiment features. When we assessed all 7954 words that occurred more than five times along with the sentiment features, the accuracy improved. However, the performance of SVM rather decreased. In this case, RF has the highest performance and the precision for the female class somewhat improved. However, the female class still has a poor performance as shown in Table 6.

Table 6. The accuracy of gender detection using entire feature set (word features + sentiment features).

	NB		**SVM**		**RF**	
	Female	**Male**	**Female**	**Male**	**Female**	**Male**
Female	1838	2623	1	0	754	135
Male	14	34	1822	2686	1069	2551
Precision	99.24%	1.28%	0.05%	100.00%	41.36%	94.97%
Recall	41.20%	70.83%	100.00%	59.58%	84.81%	70.47%
Accuracy	41.52%		59.55%		73.33%	
False positive rate	29.17%		40.42%		29.53%	
False negative rate	58.80%		0.00%		15.19%	

Based on the three experiments varying the feature set, we obtained two findings, one is the algorithm perspective and the other is the feature perspective. We can conclude that SVM underperforms other algorithms in the case of a few features. On the other hand, RF works well with more features. In addition, we can conclude that NB is not appropriate for text classification task. For feature perspective, words are good features for gender classification in the web forum. Types of emotion and their density are not enough, although these sentiment features reduce the computational complexity with a small number of features. However, when using with SVM, the senti-word feature set generates the comparable performance to the word feature set.

Even though words are good features, we cannot employ all words because we need to consider the computational complexity and cost to make the algorithm work. Machine learning algorithms support a limited number of features, unlike the deep learning algorithm, and we need to reduce the number of words included in the feature set. For feature reduction, we explored the threshold value to extract words from 5 to 100, as shown in Table 7. The number of words involved in learning is displayed at the end of Table 7. We also tested which feature extraction method among the occurrence with boolean and the term frequency is useful. From the experiments results, the occurrence with boolean outperforms the term frequency. In naïve Bayes, which assumes the independence of the variables, the number of words decreases when setting a higher threshold value, and the performance improves. SVM has the opposite behavior with NB. The best performance is achieved when applying the RF model to the term frequency features with a threshold value of 10, resulting in 4822 words. However, the performance of RF begins to decline beyond a certain number of features from the first two rows of Table 7. From all the experiments, we could conclude that traditional machine learning algorithms are

not good at gender classification, resulting in a large performance gap between two classes. In the next section, we will see how much the deep learning algorithm improved the performance.

Table 7. The accuracy of gender detection using different feature set.

Feature | Accuracy	NB (%)	SVM (%)	RF (%)
TF/Boolean (5) [1] + senti	41.52/41.32	59.55/59.57	73.3/72.79
TF/Boolean (10) [2] + senti	44.95/45.13	70.68/71.66	**74.41/73.59**
TF/Boolean (15) [3] + senti	49.48/51.08	71.72/**72.79**	73.16/73.05
TF/Boolean (100) [4] + senti	**61.5/66.16**	**71.99**/72.52	72.9/73.52

[1] including 7588 words, [2] including 4822 words, [3] including 4732 words, [4] including 1108 words.

4.2. Deep Learning Results

We tested two methods of embedding words into vectors, random word-embedding and the word2Vec method. We also varied the CNN structure with two filter sizes of length 3 or 8, which implies that three words or eight words are aggregated. The number of filters is set to 10. For parameter optimization, we used ADAM [24]. This uses squared gradients to scale the learning rate and takes advantage of momentum by using the moving average of the gradient instead of the gradient itself. The experiment results show that the random word-embedding method outperforms the word2Vec method.

We guess that word2Vec is built with formal text from GoogleNews, but our text dataset is written in spoken language. Thus, word2Vec does not work well in our case. The accuracy of the random word embedding method ranges between 88% and 91% depending on the CNN structure. On the other hand, the accuracy with the word2Vec ranges between 67% and 71%.

Secondly, we varied the feature set from word features to senti-word features. The sentiment feature set is low-dimensional with a small number of features, so we judged that deep learning is not necessary for this feature set. Thus, we tested CNN for the reduced feature consisting of senti-words and the full feature set consisting of word features. We achieved 88.7% with senti-word features and 90.6% with word features. The number of senti-word features is at 20% of the number of word features, but it generates a comparable performance. Compared to traditional machine learning algorithms, CNN dramatically increases the performance. For the female class, the precision and recall are much improved and the performance for the two classes is equally good.

We changed the CNN structure to find the best structure for our context. Starting from a simple structure, we designed more complicated structures. The combination of two convolution layers proceeded by the pooling layer, the dropout layer after that, and the fully-connected layer at last, outperformed other structures. A more complicated model with two convolutional layers outperforms the simplest model with one convolutional layer. However, the accuracy increases by at most 0.004. Slight modifications to the baseline model, in the second row, degrade the accuracy. From the other models, we found one fully connected layer is enough for comparing the second through fourth models. The max pooling layer employed to reduce feature dimension is best deployed right after each convolutional layer. In future work, we will test more complicated structures and more diversity in terms of the number of filters and sizes. Table 8 indicates the result of CNN performance according to each structure.

Table 8. Comparison of CNN performance according to the structures.

CNN Structure	Accuracy	Female		Male	
		Precision	Recall	Precision	Recall
Conv + Pool + Dropout + FC	0.906	0.93	0.91	0.87	0.90
Conv + Pool + Conv + Pool + Dropout + FC	**0.910**	0.93	0.92	0.88	0.89
Conv + Pool + Conv + FC + Dropout + FC	0.905	0.94	0.89	0.86	0.92
Conv + Conv + Pool + FC + Dropout +FC	0.887	0.91	0.90	0.85	0.87

5. Discussion and Conclusions

In this study, we constructed a model to detect gender information by using machine learning algorithms based on word and sentiment feature sets. We developed various sentiment feature sets including the number of emotion types expressed in a message, objectivity, and emotional complexity.

To check the effectiveness of sentiment features in gender classification, we varied the feature sets, including sentiment, senti-word, sentiment+word feature sets and using machine learning algorithms or deep learning algorithms. From the experiments, we found that sentiment features with a small number of features and domain-independent are not enough for classifying gender in the medical web forum. Words should be incorporated to classify the author's gender. We found that it is possible to construct gender classification models by using a sentiment+word feature set. In cases of a lightweight system with a limited number of features, SVM performed moderately with senti-word features.

However, overall, the traditional machine learning algorithm failed, having low precision in the female class. Many male authors were misidentified as female authors. When we applied the deep learning algorithm, both gender classes were identified well. The senti-word feature set and word feature set both worked well, with a 2% accuracy gap. The number of senti-word features is 20% of the number of word features, but generates a comparable performance.

As a design issue, to embed text into a matrix form, we tested word-embedding methods, one random and the other pre-built word2Vec. To find the better model, we varied the CNN structure by adding and deleting hidden layers, the convolutional layer, and the max pooling layer.

This work poses remaining challenges as follows. We tested the word2Vec built by Google and found that it does not work well for our dataset. In future work, we will build our own word2Vec suitable for medical web forums by adopting Google's training method. A consideration of word similarity would improve the detection performance. Secondly, in the current work, we eliminated stopwords, numbers, and special characters to reduce the feature set. However, previous work has suggested that these words could be helpful in the text classification task. In the deep learning model, feature reduction is not necessary, so we will check their effectiveness. In addition, we intend to add features in the context of social networks by analyzing the relationship between the posts/comments in order to improve the accuracy of the proposed model. The social network features are also generic across disease types and forums.

Author Contributions: S.P. and performed the experiments and writing. J.W. collected the dataset and participated in writing.

Funding: This work was supported by the research fund of Soonchunhyang University (project number 20180118) and a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP; Ministry of Science, ICT & Future Planning (NRF-2017R1D1A3B03036050)).

Acknowledgments: The authors gratefully acknowledge the support by the research fund of Soonchunhyang University (project number 20180118) and a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP; Ministry of Science, ICT & Future Planning (NRF-2017R1D1A3B03036050)).

Conflicts of Interest: The authors have no conflicts of interests.

References

1. Weaver, J.B., III; Mays, D.; Weaver, S.S.; Hopkins, G.L.; Eroğlu, D.; Bernhardt, J.M. Health information–seeking behaviors, health indicators, and health risks. *Am. J. Public health* **2010**, *100*, 1520–1525. [CrossRef] [PubMed]
2. Woo, J.; Lee, M.J.; Ku, Y.; Chen, H. Modeling the dynamics of medical information through web forums in medical industry. *Technol. Forecast. Soc. Chang.* **2015**, *97*, 77–90. [CrossRef]
3. Denecke, K.; Nejdl, W. How valuable is medical social media data? Content analysis of the medical web. *Inf. Sci.* **2009**, *179*, 1870–1880. [CrossRef]
4. Sullivan, C.F. Gendered cybersupport: A thematic analysis of two online cancer support groups. *J. Health Psychol.* **2003**, *8*, 83–104. [CrossRef] [PubMed]
5. Healthboard. Available online: https://www.healthboards.com/ (accessed on 25 March 2019).

6. Zhang, Y.; Dang, Y.; Chen, H. Gender classification for web forums. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2011**, *41*, 668–677. [CrossRef]
7. Ryu, K.; Jeong, J.; Moon, S. Inferring Sex, Age, Location of Twitter Users. *J. KIISE* **2014**, *32*, 46–53.
8. Wang, Y.-C.; Burke, M.; Kraut, R.E. Gender, topic, and audience response: An analysis of user-generated content on facebook. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Paris, France, 27 April–2 May 2013; pp. 31–34.
9. Na, Y.; Cho, G. Grouping preferred sensations of college students using sementic differential methods of sensation words. *Korean J. Sci. Emot. Sensib.* **2002**, *5*, 9–16.
10. Yan, X.; Yan, L. Gender Classification of Weblog Authors. In Proceedings of the AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs, Palo Alto, CA, USA, 27–29 March 2006; pp. 228–230.
11. Mukherjee, A.; Liu, B. Improving gender classification of blog authors. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Cambridge, MA, USA, 9–11 October 2010; pp. 207–217.
12. Pennacchiotti, M.; Popescu, A.-M. A machine learning approach to twitter user classification. In Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, Barcelona, Spain, 17–21 July 2011.
13. Dwivedi, V.P.; Singh, D.K.; Jha, S. Gender Classification of Blog Authors: With Feature Engineering and Deep Learning using LSTM Networks. In Proceedings of the 2017 Ninth International Conference on Advanced Computing (ICoAC), Chennai, India, 14–16 December 2017; pp. 142–148.
14. Bartle, A.; Zheng, J. *Gender Classification with Deep Learning*; Stanford cs224d Course Project Report; The Stanford NLP Group: Stanford, CA, USA, 2015.
15. Lopes Filho, J.A.B.; Pasti, R.; de Castro, L.N. Gender classification of twitter data based on textual meta-attributes extraction. In *New Advances in Information Systems and Technologies*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 1025–1034.
16. Garibo-Orts, O. A big data approach to gender classification in twitter. In Proceedings of the Ninth International Conference of the CLEF Association (CLEF 2018), Avignon, France, 10–14 September 2018.
17. Kim, Y. Convolutional neural networks for sentence classification. *arXiv* **2014**, arXiv:1408.5882.
18. Severyn, A.; Moschitti, A. Unitn: Training deep convolutional neural network for twitter sentiment classification. In Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), Denver, CO, USA, 4–5 June 2015; pp. 464–469.
19. Mohammad, S.M. Challenges in sentiment analysis. In *A Practical Guide to Sentiment Analysis*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 61–83.
20. Nayak, J.; Naik, B.; Behera, H. A comprehensive survey on support vector machine in data mining tasks: Applications & challenges. *Int. J. Database Theory Appl.* **2015**, *8*, 169–186.
21. Zhang, Y.D.; Dong, Z.; Chen, X.; Jia, W.; Du, S.; Muhammad, K.; Wang, S.H. Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation. *Multimedia Tools Appl.* **2019**, *78*, 3613. [CrossRef]
22. Wang, S.H.; Sun, J.; Phillips, P.; Zhao, G.; Zhang, Y.D. Polarimetric synthetic aperture radar image segmentation by convolutional neural network using graphical processing units. *J. Real-Time Image Process.* **2018**, *15*, 631. [CrossRef]
23. word2Vec. Available online: https://code.google.com/archive/p/word2vec/ (accessed on 25 March 2019).
24. Kingma, D.P.; Ba, J.L. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980v9.