*information*

Article

# Adverse Drug Event Detection Using a Weakly Supervised Convolutional Neural Network and Recurrent Neural Network Model

**Min Zhang [1,2] and Guohua Geng [1,*]**

[1] School of Information Science and Technology, Northwest University, Xi'an 710127, China
[2] School of Engineering and Technology, Xi'an Fanyi University, Xi'an 710127, China
* Correspondence: ghgeng@nwu.edu.cn

Received: 16 July 2019; Accepted: 29 August 2019; Published: 4 September 2019

**Abstract:** Social media and health-related forums, including the expression of customer reviews, have recently provided data sources for adverse drug reaction (ADR) identification research. However, in the existing methods, the neglect of noise data and the need for manually labeled data reduce the accuracy of the prediction results and greatly increase manual labor. We propose a novel architecture named the weakly supervised mechanism (WSM) convolutional neural network (CNN) long-short-term memory (WSM-CNN-LSTM), which combines the strength of CNN and bi-directional long short-term memory (Bi-LSTM). The WSM applies the weakly labeled data to pre-train the parameters of the model and then uses the labeled data to fine-tune the initialized network parameters. The CNN employs a convolutional layer to study the characteristics of the drug reviews and active features at different scales, and then the feed-forward and feed-back neural networks of the Bi-LSTM utilize these salient features to output the regression results. The experimental results effectively demonstrate that our model marginally outperforms the comparison models in ADR identification and that a small quantity of labeled samples results in an optimal performance, which decreases the influence of noise and reduces the manual data-labeling requirements.

**Keywords:** adverse drug reactions (ADRs); CNN-LSTM; sentiment classification; weakly supervised

## 1. Introduction

Adverse drug reactions (ADRs) are part of the leading cause of morbidity and mortality in public health. Research has indicated that death and hospitalizations due to ADRs number in the millions (up to 5% hospitalizations, 28% emergency treatments, and 5% death), and the related consumption is approximately 75 billion dollars annually [1–3]. Post-marketing drug safety monitoring is therefore essential for pharmacovigilance. Regulatory agencies (e.g., the Food and Drug Administration (FDA)) establish and support spontaneous reporting systems (SRS) to monitor the most current pharmacovigilance activities in the United States. Suspected ADRs may be raised by patients and healthcare providers through these surveillance systems. However, biased and underreported events limit the effectiveness of these systems, which report an estimated ADR rate of approximately 10% [4].

Social media, especially health-related social networks (e.g., DailyStrength (http://www.dailystrength.org) and AskaPatient (https://www.askapatient.com/)), enable both the patients and nursing staff to share and obtain comments regarding drug safety. Drug reviews of patient feedback on social media are a potential and timely source for ADR identification [5,6]. User reviews contain sentiment information (i.e., positive, negative or neutral expressions) to provide important features for ADR identification [7], and sentiment features can marginally improve ADR detection in health-related forum reviews [8].

In this study, based on the intuition of patient reviews about adverse drug reactions (ADRs) expressing negative sentiments, we aim to recognize ADRs through sentiment classification, which is commonly used to complete ADR identification through social media reviews [9]. The current sentiment classification methods are typically divided into three categories: (1) lexicon-based methods, (2) traditional machine learning methods, and (3) deep learning methods. Lexicon-based methods have implemented a string-matching method that matches the detected terms to predefined drug adverse event lexicons [10,11]. However, lexicon matching cannot easily distinguish whether a drug-related event is related to an ADR or to an indication for a medication. In addition, the characteristics of social media language (e.g., informal, vernacular, abbreviations, symbols, misspellings, and irregular grammar) further limit the precision of the lexicon matching method in ADR identification.

Traditional machine learning classifiers (e.g., conditional random fields (CRFs)) [12,13] combine knowledge bases with sentiment-related text features. However, the fixed-width window mechanism of CRFs only considers the target word and its neighbouring words in the scope of their input; therefore, important information associated with more distant words may be excluded.

Deep learning models (e.g., convolutional neural networks CNNs) [14–16] may limit CRF's. Hierarchical CNNs specialize in extracting position-invariant features. Given the specificity of social media user reviews, an entire sentence may describe a positive sentiment, but the phrases that contain a negative sentiment (e.g., "don't" and "miss") may appear. Thus, the long-short-term-memory (LSTM) network (specifically a class of recurrent neural networks (RNNs) [17,18] with a sequential architecture can be used to correctly process long sentences. The LSTM 'memory mechanism, which is well suited for marking tasks, has a hidden state to remember previous labeling decisions and then labels the current token. However, LSTM does not perform well in the emotional classification of social media to complete a key-phrase recognition task [19].

Furthermore, a deep learning model is an end-to-end model, allowing the computer to automatically learn sentiment features, thereby reducing feature-extracted complexity and incompleteness. However, a successful deep learning model depends on large-scale labeled data, and obtaining massive labeled training data manually is time-consuming and expensive. The lack of large-scale labeled data has become a bottleneck for deep learning in ADR identification-related research [20].

To reduce the limitations of deep learning, researchers mine the information from the data generated by users (e.g., sentiment ratings, tweets, reviews, and emoticons), which is helpful in the training of sentiment classifiers. However, the behaviour of labeling texts, which users designate as predefined labels for each review, is arbitrary and has no uniform standard. These labeled data are noisy (a high score with a negative review) and are called weakly labeled data [21]. The classification model influenced by noise data in weakly labeled data will lower the accuracy [22].

In this work, we propose a deep learning framework for the sentiment classification of drug reviews. The framework utilizes a weakly supervised mechanism (WSM) that applies weakly labeled data to pre-train the parameters of the model and then uses the labeled data to fine-tune the initialized parameters. First, we attempt to leverage a large quantity of weakly labeled data to pre-train a deep neural network that reflects the drug reviews' sentiment distribution in the neural network. Second, we utilize a small quantity of labeled data to fine-tune the network and learn the target prediction function. In contrast, previous training methods, usually based on weakly labeled data, directly learn the target prediction function, which can impact the prediction function because of the noise in the data. CNN is better at classifying sentences with simple syntactic structure. LSTM can capture long-distance dependencies in comment statements and is better at "understanding" the semantics of sentences as a whole. Through the training framework of "weak supervised pre-training + supervised fine-tuning", the influence of noise on the model training process is reduced, and a large amount of useful information in the weak labeled data is better "remembered" in the depth model. The time efficiency of CNN, LSTM and CNN_LSTM are not very different when we use our small datasets. Our method performs well in ADR recognition.

We propose a model that applies the WSM combining the strength of the CNN and bi-directional long-short-term memory (Bi-LSTM) [23–25] (named WSM-CNN-LSTM) to complete the sentiment classification task of ADR reviews. The WSM-CNN-LSTM model includes two parts: the CNN employs a convolutional layer to study and extract the characteristics of the drug review and active features of different scales within the drug reviews. Then, the Bi-LSTM seizes past and future information by the forward and backward networks, respectively, and utilizes the sentence sequence information to compose features sequentially and output the regression results.

To effectively train the WSM-CNN-LSTM model, we collect drug reviews identified as weakly labeled datasets, containing 61,263 comments from the AskaPatient.com forum to pre-train a deep neural network. Additionally, a manually labeled dataset containing 11,083 comments is used to fine-tune the network to learn the target prediction function. Sufficient experiments are designed and implemented to validate the effectiveness of the WSM-CNN-LSTM model.

In this work, our contributions are as follows:

We propose a novel method that uses a WSM for the sentiment analysis of ADR reviews to avoid a large amount of manually labeled data. The WSM greatly reduces the influence of noise on the model in the weakly labeled data. To our knowledge, this is the first work in the health forum, particularly in the field of drug review sentiment analysis.

We propose a novel architecture named WSM-CNN-LSTM to complete the task of ADR identification. This model reports that the stand-alone CNN model performs poorly in the characteristics of the long text of most drug reviews, while adding feed-forward and feed-back neural networks dramatically improves the classification effects.

We validate that the WSM-CNN-LSTM model presents superior performance in ADR identification through experiments, in which a large amount of weakly labeled data is utilized to pre-train a deep neural network and a small quantity of labeled data is used to fine-tune the network and learn the target prediction function. Our proposed training method avoids the direct use of a weakly labeled data training target prediction function, which can partly reduce the influence of noise on the prediction function.

This paper is organized as follows. The weakly supervised multi-channel CNN-LSTM model proposed in this paper is introduced in Section 2. In Section 3, the experimental process and results are discussed. Finally, Section 4 is conclusions and presents directions for future work.

## 2. Related Work

In recent years, some researchers used potential resources from social media to detect ADR. Leaman et al. [26] applied Lexicon-based approach and used 450 comments for Concept/relation extraction system development. Akhtyamova et al. [27] proposed a CNNs model based on varied structural parameters. The majority vote determines the prediction of the model. Santiso et al. [28] proposed a deep model based on the LSTM to discover ADRs from Electronic Health Records (EHRs). Embeddings are created using lemmas to reinforce lexical variability of EHRs. However, due to the lack of labeled data, the accuracy of prediction results needs to be improved.

Fortunately, although there is a lack of large-scaled of labeled data, there is still a large amount of weakly labeled data on social networks, such as comment containing sentiment orientation. Tutubalina Elena et al. [29] proposed the method based on ADR review scores to predict demographic. The weak-tagged text corpus is used to generate dictionary. However, in their work, the generated lexicon using weakly labeled is still not escape the limitations of domain knowledge.

## 3. Methods

### 3.1. Word Embedding

As the input of our model, we normally needed to generate high-dimensional word vectors that capture information regarding the words of morphology, syntax, and semantics in the word embedding layer. We trained every word as a k-dimensional (300 dimensions) word vector using the publicly available GloVe toolkit [30], where k represents the dimension of the word vector. The sentence matrix is achieved by connecting the word vectors together after pre-training. Let $W_i \in R^k$ be the $i$-th k-dimensional word vector in a sentence; therefore, a drug review with $n$ word vectors is encoded as the sentence matrix W $\in R^{n \times k}$, which is composed of a sequence of word vectors denoted as:

$$W = [w_1, w_2, \ldots, w_n]^T. \tag{1}$$

## 3.2. Framework of the WSM-CNN-LSTM Model

We propose a novel architecture named WSM-CNN-LSTM, which introduces a WSM that combined the strengths of CNN-LSTM to complete the task of three labels (positive, neutral, and negative) for drug reviews, which is a variation in the CNN-LSTM model in [31,32].

Figure 1 indicates the architecture of the WSM-CNN-LSTM model. There were six varieties of layers in this model: input layer, convolutional layer, max-pooling and dropout layer, Bi-LSTM layer, fully connected layer, and softmax layer. First, the pre-trained word vectors were input into the convolutional layer perform a convolution via linear filters with different lengths. The effect of a convolution was to extract features from word vectors and generate feature maps. Second, the max-pooling layer extracted salient features from feature maps generated by the convolution and then input them into the forward and backward LSTM network. In the LSTM layer, these salient features were used to output the regression results. Finally, the fully connected layer and softmax layers extracted regression results from LSTM and output the final classification results.

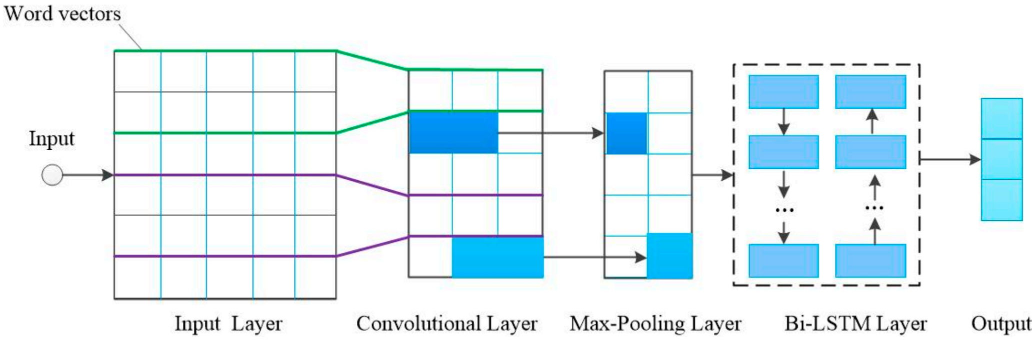


**Figure 1.** Architecture of the WSM-CNN-LSTM model.

## 3.3. CNN-LSTM Model

### 3.3.1. Convolutional Layer

The convolutional layer was used to effectively extract features from the sentence matrix through a set of convolution filters $F \in R^{h \times k}$, where $h$ is the length of the filter. This method convolutes the sentence matrix W input by the word embedding layer to obtain the feature map M$\in R^{n-h+1}$, in which the vector has one column. Different sizes of the feature map are produced from the different filter sizes. The $i$-th result output element of each filter m is generated as:

$$m_i = f\ (w_{i:i+h-1} \otimes F + b)(i : i + h - 1 \leq n), \tag{2}$$

and the feature map M $\in R^{n-h+1}$ is produced as:

$$M = [m_1, m_2, \ldots, m_{n-h+1}], \tag{3}$$

where $b$ is a bias, $\otimes$ is the convolutional operator, and $f$ is a nonlinear function (e.g., tanh). We used the activation function ReLU [33] for a fast calculation, and $w_{i:i+h-1}$ denotes the word vectors, represented as:

$$W_{i:i+h-1} = w_i, w_{i+1}, \ldots, w_{i+h-1}. \tag{4}$$

### 3.3.2. Max-Pooling and Dropout Layer

The max-pooling layer, in which the most salient feature was further extracted from the previous different filters using the maximum mechanism, down-sampled the features learned in the convolutional layer. This method took the most salient feature and reduced the computation by choosing a maximum value, which eliminated the non-maximal values. Because the maximum value represents the most distinguishing salient feature of a drug review in a filter, we chose max-pooling rather than average pooling. In this layer, we applied multiple convolutional filters to extract the different features that were fed into the Bi-LSTM layer.

At the same time, in our model, a dropout layer [34] was introduced after the max-pooling layer because of the inevitable over-fitting in the CNN.

### 3.3.3. Bi-LSTM Layer

The RNN was applied to suitably process sequence data, whose hidden layer's input combined the output of the input layer and the output of the hidden layer at the preceding moment, and the neuron had a memory ability. However, the vanishing gradient problem will produce very small numbers in a simple RNN [35]. Bi-LSTM, with the capacity to catch long-term dependencies, introduced a gate mechanism to effectively address this problem.

LSTM (long short term memory) is specially designed to solve the long-term dependence problem of general RNN, which is added memory units to the neurons of the hidden layer on the basis of RNN. As shown in Figure 2, the LSTM cell consisted of three gates, namely, the input gate $i$, the forget gate $f$, and the output gate $o$, to control the memory length. At each step time $t$, the three gates, input vector, and state update of a memory cell were calculated as follows.

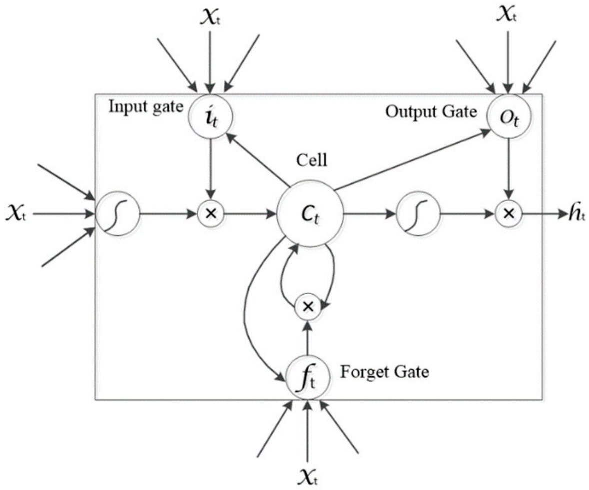


**Figure 2.** Architecture of the LSTM memory.

Three gates:

$$i_t = \sigma(W_{ij}x_t + V_{ij}h_{t-1} + b_{ij}), \tag{5}$$

$$f_t = \sigma(W_{fj}x_t + V_{fj}h_{t-1} + b_{fj}), \tag{6}$$

$$o_t = \sigma(W_{oj}x_t + V_{oj}h_{t-1} + b_{oj}). \tag{7}$$

Input vector:

$$d\_in_t = tanh(W_{dj}x_t + V_{dj}h_{t-1} + b_{dj}). \tag{8}$$

State update:

$$c_t = f_t \otimes c_{t-1} + i_t \otimes d\_in_t, \tag{9}$$

$$h_t = o_t \otimes tanh(c_t), \tag{10}$$

where $x_t$ is the input vector; $W$ and $V$ represent the weight matrix of the input $x_t$ and hidden state $h_{t-1}$, respectively; $b$ is the bias matrix for the input cell and three gates; $d\_in_t$ is the dimension of the word vector for the input cell; $i_t$, $f_t$, and $o_t$ denote the input gate, and forget gate, output gate, respectively; $c_t$ is the memory cell; $p_t$ is the hidden state; ⊗ is the element-wise multiplication; and $\sigma$ is the sigmoid activation function.

In the bi-directional LSTM, the model learned the output weights of the previous moment and the input of each sequence at the current time. Additionally, a forward network and a backward network were beneficial for simultaneously capturing the past (backward direction) and future (forward direction) information of sentence sequences to obtain the contextual information for many sequential tagging tasks during sentence sequence modeling. Therefore, this approach was utilized to capture all the information during sentence sequence modeling [36].

### 3.3.4. Fully Connected Layer

Fully connected layers playing the role of classifiers mapped the distributed feature representation to the sample space to feature vectors that contained the combination information of the characteristics of the input reviews. Finally, these vectors were input to the output layer to complete the classification task.

### 3.3.5. Softmax Layer

In the softmax layer, we used the softmax activation function [37] to compute classification, which was converted by the outputs of the fully connected layer. A vector is output in this layer and is calculated by (11),

$$P(c = i|z) = \frac{e^{z^T w_i}}{\sum_{n=1}^{N} e^{z^T w_i}}, \tag{11}$$

where $N$ is number of classes, z is the input vector from the previous layer, and w is the parameter vector. The final classification labels, namely, positive, neutral, and negative, were output in this layer. The classification result $\hat{c}$ is calculated by (12):

$$\hat{c} = \underset{i}{argmax} P(c = i|z). \tag{12}$$

## *3.4. Weakly Supervised Mechanism*

The WSM-CNN-LSTM model, trained by a scheme called unsupervised pre-training appended supervised fine-tuning, was first pre-trained by a large amount of weekly labeled data and then fine-tuned by a small amount of labeled data via manual labeling.

First, our model was pre-trained by a considerable amount of weakly labeled data from the drug rating reviews obtained from the AskaPatient forum. Second, to improve the accuracy of the pre-trained model by a large amount of weekly labeled data with noise, we manually labeled a small amount of labeled data that was used to fine-tune the pre-trained model. The parameters of the pre-trained model were used as the initial parameters of the supervised training. The labeled data were used to supervise the training and testing of the model, and finally, the classification model was trained.

## 4. Experiments and Discussion

### *4.1. Dataset*

In our work, a dataset was collected from the drug ratings and health care opinions forum named AskaPatient, where actual patients who have previously taken the drug share their treatment experience. The drug reviews were gathered from 1 May 2012 to 31 December 2017. The drug reviews from this forum with comments by patients are shown as eight fields, namely, review the rating of the drug, reason for taking this drug, and side effects that were experienced with the drug. Additional reviews include gender, age, duration/dosage, and date added. The general meaning for the ratings is displayed in Table 1.

**Table 1.** Review Rating for AskaPatient.com.

| Drug Ratings | Satisfied Level | General Meaning |
|---|---|---|
| 1 | Dissatisfied | I would not recommend taking this medicine |
| 2 | Not satisfied | This medicine did not work to my satisfaction |
| 3 | Somewhat Satisfied | This medicine helped somewhat |
| 4 | Satisfied | This medicine helped |
| 5 | Very Satisfied | This medicine cured me or helped me a great deal |

Our target was a multi-classification problem for the sentiment classification of the drug reviews on the AskaPatient forum. We regarded the reviews of the 4 and 5 ratings as positive weakly labeled data and divided them into class 2. The 3 ratings reviews were regarded as neutral and divided into class 1. Finally, the reviews of the 1 and 2 ratings were regarded as negative and were divided into class 0.

In the forum of AskaPatient.com, we captured 63,782 reviews on 2000 publicly available drugs containing prescription medicines currently approved by the FDA, along with many over-the-counter medicines. The remaining 61,263 reviews were non-null comments. The labeled data containing 11,083 drug reviews took one month for two authors to manually label. The composed proportion of weakly labeled data are labeled data are shown in Figure 3. We note that the datasets were roughly balanced and that the labeled data were approximately one-fifth of the weakly labeled data.

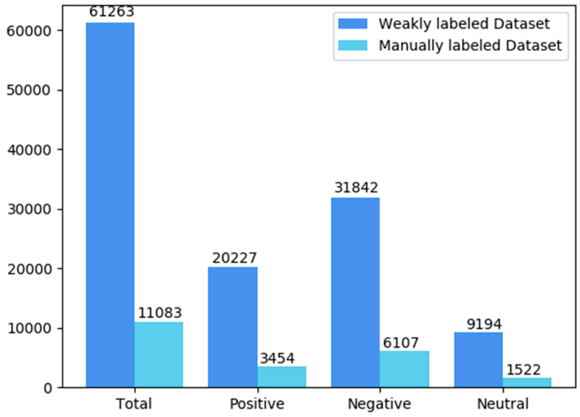


**Figure 3.** Sizes of the weakly labeled and manually labeled datasets.

### *4.2. Experimental Setup*

Seventy percent of the weakly labeled data was randomly leveraged to pre-train the deep neural network, and 30% of the data were utilized for testing. Every drug review was trained as an embedding matrix by the publicly available GloVe toolkit with 300 dimensions, using the TensorFlow model of the Python module [38]. The matrix was composed of a sequence of word embeddings. We prepared an embedding matrix and initialized the words that were not found in the embedding index to be all-zeros. Then, pre-trained word embeddings were loaded into the embedding layer. The batch size was 64, the dropout was 0.5 and the activation function was softmax. The output of the one-dimensional (1D) CNN with global max-pooling was the input of the Bi-LSTM.

According to the characteristics of the drug reviews and to facilitate implementation convenience, we restricted the number of words in each drug review to within 100 words. For a drug review with $k$ words, if $k < 100$, then we appended it to 100 with a zero vector. The model truncated the vector, leaving only the first 100 words, when a drug review had more than 100 words. No drug reviews contained more than 100 words.

## *4.3. Comparison Models*

In our experiments, we specifically compared the performance of our model, SVM [39,40], WSM-CNN-LSTM, with the CNN, LSTM, and CNN-LSTM-rand models and the WSM-CNN and WSM-LSTM models. The compared models were as follows:

- SVM. Support vector machines. We used trigrams and Liblinear classifier;
- CNN-rand. We trained the CNN on of the labeled dataset and randomly initialized the network parameters;
- Weakly supervised mechanism CNN model (WSM-CNN). The weakly labeled data were utilized to train the network model based on the CNN, and the labeled data were used to fine-tune the initialized network parameters;
- LSTM-rand. We trained the LSTM on the labeled dataset and randomly initialized the network parameters;
- Weakly supervised mechanism LSTM model (WSM-LSTM). The weakly labeled data were utilized to train the network model based on LSTM, and the labeled data were used to fine-tune the initialized network parameters;
- CNN-LSTM-rand. We trained the combined CNN and LSTM on the labeled dataset and randomly initialize the network parameters.

## *4.4. Experimental Results and Discussion*

### 4.4.1. Weakly Supervised Model Performance

Table 2 shows the preliminary experimental results of WSM-CNN-LSTM and the comparison baseline model for the dataset. Except for the overall accuracy, we employed micro-F1 [40], precision, and recall as evaluation metrics. They are computed as follows:

$$Precision = \frac{TP}{TP + FP}, \tag{13}$$

$$Recall = \frac{TP}{TP + FN}, \tag{14}$$

$$\mathrm{micro} - F1 = \frac{2 \times P \times R}{P + R}. \tag{15}$$

Importantly, the experimental results demonstrate that WSM-CNN-LSTM improved the comparison models with regard to accuracy and F1 during classification. It is likely that the CNN is good at classifying simple sentence structures, and the LSTM layer can capture the long-distance dependencies in the drug reviews. The WSM-CNN-LSTM model, utilizing the WSM and combining the strengths of both the CNN and LSTM, understood the semantics of the sentence as a whole and improved the classification performance of the model in the sentiment analysis of drug reviews.

**Table 2.** ADR identification performance percentages when testing different comparison models.

| Method | Accuracy | F1 | Precision | Recall |
|---|---|---|---|---|
| SYM | 80.69 | 75.93 | 81.56 | 71.02 |
| CNN-rand | 79.17 | 75.18 | 80.21 | 70.74 |
| WSM-CNN | 83.29 | 81.01 | 82.47 | 79.60 |
| LSTM-rand | 80.71 | 79.77 | 81.86 | 77.78 |
| WSM-LSTM | 84.92 | 82.82 | 83.02 | 82.62 |
| CNN-LSTM-rand | 83.78 | 83.12 | 85.3 | 81.05 |
| WSM-CNN-LSTM | **86.72** | **86.81** | **87.92** | **85.73** |

Note: Statistically significant improvements over comparison models are bolded.

#### 4.4.2. Rand Compared with the WSM

In the comparison experiments, we deliberately used two mechanisms for the same model. The *-rand mechanism trained the network model based on randomly initialized network parameters with labeled datasets, and the WSM-* mechanism was a WSM in which weakly labeled data were used to pre-train the network model and parameters. Then, the small amount of labeled data was used to fine-tune the pre-trained model. Clearly, all WSM-* model results are slightly higher than the *-rand model results in Table 2. This increase is likely due to the expression of the WSM, which uses pre-training to record the prior knowledge of the emotional distribution, and fine-tuning the parameters of the model reduces the effect of noise data on the model training process.

#### 4.4.3. Macro-F1 Result of Our Model

Macro-F1 is the average of the F1 of each class. In order to verify the performance of our model in each class, we present the F1 of each class in Table 3, thus macro-F1 is 86.64. As can be seen from the results in the Table 4, F1 of the negative and positive class are higher than the neural class, which prove that our model is more effective in capturing negative and positive words in drug reviews.

**Table 3.** ADR identification performance percentages when testing different compared models. Stratified 10 × 10-fold cross validation results.

| Method | Accuracy | F1 | Precision | Recall |
|---|---|---|---|---|
| SVM | 80.42 | 77.68 | 82.01 | 73.78 |
| CNN-rand | 78.36 | 74.76 | 79.76 | 70.97 |
| WSM-CNN | 82.94 | 80.41 | 81.91 | 78.19 |
| LSTM-rand | 80.32 | 79.63 | 81.28 | 77.29 |
| WSM-LSTM | 84.56 | 82.01 | 82.36 | 81.92 |
| CNN-LSTM-rand | 82.79 | 82.83 | 85.12 | 80.15 |
| WSM-CNN-LSTM | **85.67 *** | **85.57 *** | **86.88 *** | **84.16 *** |

Note: Statistically significant improvements over comparison models are bolded and marked with an asterisk (*).

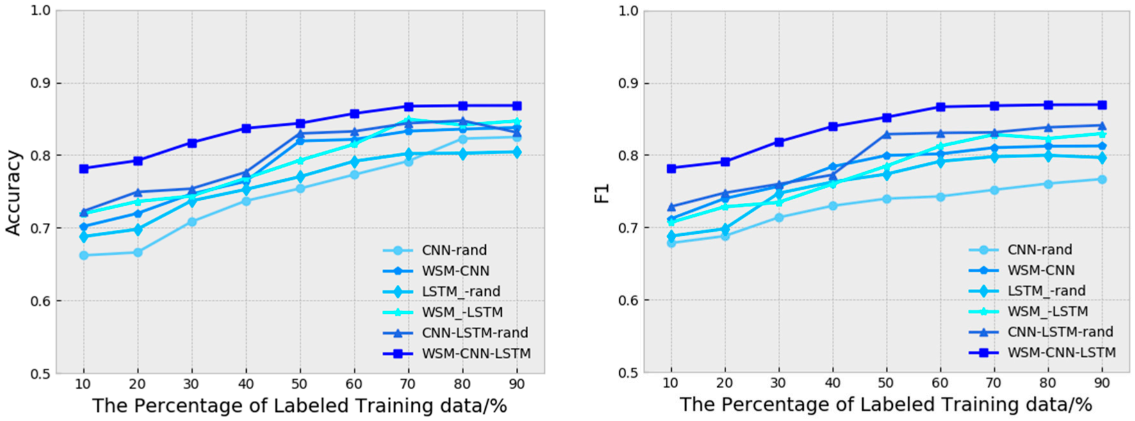
**Table 4.** F1 percentages of each class individually.

| Negative | Neural | Positive |
|---|---|---|
| 89.73 | 78.94 | 91.25 |

#### 4.4.4. Impact of the Labeled Training Data Size on Our Model

It was important to identify the sensitivity of the data sample to the weakly supervised machine learning model, especially the influence of sample size on the model. To investigate this issue, we examined the influence of the labeled training data size on each model. D% of the labeled data, where D ranged between 10 and 90, was chosen to fine-tune our experiments. The model learning curves are shown in Figure 4. Our model reached more than an 80% accuracy and an F1 score from the 30%

training set and appeared to be stable from the 70% training set. The experimental results prove that our model was not influenced by the size of the manually labeled data. It is therefore likely that a small amount of labeled data, which was used to fine-tune the WSM-CNN-LSTM model, is more suitable for the sentiment analysis of drug reviews. Furthermore, this finding significantly reflects the advantages of a small amount of manual labor in our work. Although 90% of the labeled data can achieve a better result, a 70% partition ratio is common and reasonable. In our experiments, we chose 70% labeled data as a training set.



(**a**) Accuracy curves (**b**) F1 curves

**Figure 4.** Impact of labeled training data size on each model.

Stratified 10 × 10-fold cross validation results. Experiments of stratified 10 × 10-fold cross validations were conducted to further verify the statistical significance of the improvements. We combined the training and test data and then distributed them randomly to 10 folds, ensuring that all folds had approximately the same proportion of positive, negative, and neutral drug reviews. We repeatedly used randomly generated folds for training and verification, each time training on nine folds and testing on one fold. The average results after 100 experiments are shown in Table 3. The cross validation results further demonstrate that the *-rand models exhibited no substantial improvement in accuracy and precision due to the influence of noise data on the model functions. The WSM-CNN-LSTM model was relatively effective at avoiding the impact of noise and statistically discriminating long and short sentences to improve the accuracy.

## 5. Conclusions and Future Work Discussion

In this work, we proposed a weakly supervised deep learning model named WSM-CNN-LSTM for identifying ADRs, utilizing the drug reviews of customers on health forums through multiple classification. Our model was an effective combination of a CNN and LSTM, along with a WSM that employed both weakly labeled data to pre-train the model and the use of labeled data to fine-tune the initialized network parameters. Experiments on the drug reviews collected from the AskaPatient forum indicated that the effect of our model on ADR identification was significantly superior to the contrast model in accuracy and F1 performance, which reflects the effectiveness of our model for the sentiment classification of drug review data. ADR identification through drug reviews by customers on health forums was remarkably enhanced by our model. We also observed that the WSM only required a small amount of labeled samples to attain optimal performance, which decreased the influence of noise and reduced the manual data-labeling requirements.

Drug review data in social media and health forums offer us valuable resources. In future work, our continuing research will focus on investigating the potential relationships of the drug reviews and exploring the impact of other features of the drug reviews for ADR identification, so that considerable online review data can better serve the healthy life of individuals.

**Author Contributions:** Conceptualization, G.G.; Methodology, M.Z.; Data collection, M.Z.; experiment, M.Z.; writing—original draft preparation, M.Z.; writing—review and editing, G.G.; supervision, G.G.

**Funding:** This research was funded by National Natural Science Foundation of China, grant number 61731015, 61673319 and 61802311. National Key Research and Development Program of China, grant number 2017YFB1402103. Scientific Research Team of Xian Fanyi University, grant number XFU17KYTDB02.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Lazarou, J.; Pomeranz, B.H.; Corey, P.N. Incidence of adverse drug reactions in hospitalized patients: A meta-analysis of prospective studies. *JAMA* **1998**, *279*, 1200–1205. [CrossRef] [PubMed]
2. Hakkarainen, K.M.; Hedna, K.; Petzold, M.; Hägg, S. Percentage of Patients with Preventable Adverse Drug Reactions and Preventability of Adverse Drug Reactions—A Meta-Analysis. *PLoS ONE* **2012**, *7*, e33236. [CrossRef] [PubMed]
3. Xu, R.; Wang, Q. Large-scale combining signals from both biomedical literature and the FDA Adverse Event Reporting System (FAERS) to improve post-marketing drug safety signal detection. *BMC Bioinform.* **2014**, *15*, 17. [CrossRef] [PubMed]
4. Hazell, L.; Shakir, S.A. Under-Reporting of Adverse Drug Reactions. *Drug Saf.* **2006**, *29*, 385–396. [CrossRef] [PubMed]
5. Pirmohamed, M.; James, S.; Meakin, S.; Green, C.; Scott, A.K.; Walley, T.J.; Farrar, K.; Park, B.K.; Breckenridge, A.M. Adverse drug reactions as cause of admission to hospital: Prospective analysis of 18820 patients. *BMJ Br. Med. J.* **2004**, *329*, 15–19. [CrossRef] [PubMed]
6. Curcin, V.; Ghanem, M.; Molokhia, M.; Guo, Y.; Darlington, J. Mining Adverse Drug Reactions with E-Science Workflows. In Proceedings of the Cairo International Biomedical Engineering Conference, Cairo, Egypt, 18–20 December 2008; pp. 1–5.
7. Sarker, A.; Gonzalez, G. Portable automatic text classification for adverse drug reaction detection via multi-corpus training. *J. Biomed. Inform.* **2015**, *53*, 196–207. [CrossRef] [PubMed]
8. Korkontzelos, I.; Nikfarjam, A.; Shardlow, M.; Sarker, A.; Ananiadou, S.; Gonzalez, G.H. Analysis of the effect of sentiment analysis on extracting adverse drug reactions from tweets and forum posts. *J. Biomed. Inform.* **2016**, *62*, 148–158. [CrossRef] [PubMed]
9. Ji, X.; Chun, S.A.; Geller, J. Monitoring Public Health Concerns Using Twitter Sentiment Classifications. In Proceedings of the IEEE International Conference on Healthcare Informatics (ICHI), Philadelphia, PA, USA, 9–11 September 2013; pp. 335–344.
10. Sarker, A.; Ginn, R.; Nikfarjam, A.; O'Connor, K.; Smith, K.; Jayaraman, S.; Upadhaya, T.; Gonzalez, G. Utilizing social media data for pharmacovigilance: A review. *J. Biomed. Inform.* **2015**, *54*, 202–212. [CrossRef] [PubMed]
11. Freifeld, C.C.; Brownstein, J.S.; Menone, C.M.; Bao, W.; Filice, R.; Kass-Hout, T.; Dasgupta, N. Digital Drug Safety Surveillance: Monitoring Pharmaceutical Products in Twitter. *Drug Saf.* **2014**, *37*, 343–350. [CrossRef]
12. Lafferty, J.; McCallum, A.; Pereira, F.C. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In Proceedings of the 18th International Conference on Machine Learning, Williamstown, MA, USA, 28 June–1 July 2001; pp. 282–289.
13. Wang, W. Mining Adverse Drug Reaction Mentions in Twitter with Word Embeddings. In Proceedings of the Social Media Mining Shared Task Workshop at the Pacific Symposium on Biocomputing, Kohala Coast, HI, USA, 4–8 January 2016.
14. Limsopatham, N.; Collier, N. Modeling the Combination of Generic and Target Domain Embeddings in a Convolutional Neural Network for Sentence Classification. In Proceedings of the 15th Workshop on Biomedical Natural Language Processing, Berlin, Germany, 12 August 2016; pp. 136–140.
15. Magge, A.; Scotch, M.; Gonzalez, G. CSaRUS-CNN at AMIA-2017 Tasks 1, 2: Under Sampled CNN for Text Classification. In Proceedings of the CEUR Workshop Proceedings, Honolulu, HI, USA, 27 January 2017; pp. 76–78.
16. Odeh, F. A Domain-Based Feature Generation and Convolution Neural Network Approach for Extracting Adverse Drug Reactions from Social Media Posts. Ph.D. Thesis, Birzeit University, Birzeit, Palestine, 22 February 2018.

17. Gupta, S.; Pawar, S.; Ramrakhiyani, N.; Palshikar, G.K.; Varma, V. Semi-Supervised Recurrent Neural Network for Adverse Drug Reaction mention extraction. *BMC Bioinform.* **2018**, *19*, 212. [CrossRef]
18. Comfort, S.; Perera, S.; Hudson, Z.; Dorrell, D.; Meireis, S.; Nagarajan, M.; Ramakrishnan, C.; Fine, J. Sorting Through the Safety Data Haystack: Using Machine Learning to Identify Individual Case Safety Reports in Social-Digital Media. *Drug Saf.* **2018**, *41*, 579–590. [CrossRef] [PubMed]
19. Yin, W.; Kann, K.; Yu, M.; Schütze, H. Comparative study of cnn and rnn for natural language processing. *arXiv* **2017**, arXiv:170201923.
20. Cocos, A.; Fiks, A.G.; Masino, A.J. Deep learning for pharmacovigilance: Recurrent neural network architectures for labeling adverse drug reactions in Twitter posts. *J. Am. Med. Inform. Assoc.* **2017**, *24*, 813–821. [CrossRef] [PubMed]
21. Guan, Z.; Chen, L.; Zhao, W.; Zheng, Y.; Tan, S.; Cai, D. Weakly-Supervised Deep Learning for Customer Review Sentiment Classification. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI), New York, NY, USA, 9–15 July 2016; pp. 3719–3725.
22. Qu, L.; Gemulla, R.; Weikum, G. A Weakly Supervised Model for Sentence-Level Semantic Orientation Analysis with Multiple Experts. In Proceedings of the Joint Conference on Empirical Methods in Natural language Processing and Computational Natural Language Learning. Association for Computational Linguistics, Jeju Island, Korea, 12–14 July 2012; pp. 149–159.
23. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
24. Paliwal, K.; Schuster, M. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681.
25. Graves, A.; Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw.* **2005**, *18*, 602–610. [CrossRef]
26. Leaman, R.; Wojtulewicz, L.; Sullivan, R.; Skariah, A.; Yang, J.; Gonzalez, G. Towards Internet-Age Pharmacovigilance: Extracting Adverse Drug Reactions from User Posts to Health-Related Social Networks. In Proceedings of the 2010 Workshop on Biomedical Natural Language Processing, Uppsala, Sweden, 15 July 2010; pp. 117–125.
27. Liliya, A.; Ignatov, A.; Cardiff, J. A Large-scale CNN ensemble for medication safety analysis. In Proceedings of the International Conference on Applications of Natural Language to Information Systems, Liège, Belgium, 21–23 June 2017; pp. 247–253.
28. Sara, S.; Perez, A.; Casillas, A. Exploring Joint AB-LSTM with embedded lemmas for Adverse Drug Reaction discovery. *IEEE J. Biomed. Health Inform* **2018**, 2168–2194.
29. Tutubalina, E.; Nikolenko, S. Demographic Prediction Based on User Reviews about Medications. *Comput. y Sist.* **2017**, *21*, 227–241.
30. Pennington, J.; Socher, R.; Manning, C. Glove: Global Vectors for Word Representation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
31. Yoon, J.; Kim, H. Multi-Channel Lexicon Integrated CNN-BiLSTM Models for Sentiment Analysis. In Proceedings of the 29th Conference on Computational Linguistics and Speech Processing (ROCLING), Taipei, Taiwan, 27–28 November 2017; pp. 244–253.
32. Zhang, Y.; Yuan, H.; Wang, J.; Zhang, X. YNU-HPCC at EmoInt-2017: Using a CNN-LSTM Model for Sentiment Intensity Prediction. In Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Copenhagen, Denmark, 8 September 2017; pp. 200–204.
33. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 27th international conference on machine learning (ICML-10), Haifa, Israel, 21–24 June 2010; pp. 807–814.
34. Tobergte, D.R.; Curtis, S. Improving neural networks with dropout. *J. Chem. Inf. Model.* **2015**, *5*, 1689–1699.
35. Bengio, Y.; Simard, P.; Frasconi, P. Learning long-term dependencies with gradent descent is difficult. *IEEE Trans. Neural Netw.* **1994**, *5*, 157–166. [CrossRef]
36. Mesnil, G.; He, X.; Deng, L.; Bengio, Y. Investigation of Recurrent-Neural-Network Architectures and Learning Methods for Spoken Language Understanding. In Proceedings of the 14th Annual Conference of the International Speech Communication Association (INTERSPEECH), Lyon, France, 25–29 August 2013; pp. 3771–3775.
37. Nasrabadi, N.M. *Pattern Recognition and Machine Learning*; Springer International Publishing: Cham, Switzerland, 2007; Volume 16, p. 049901.

38. Owoputi, O.; O'Connor, B.; Dyer, C.; Gimpel, K.; Schneider, N. Part-of-Speech Tagging for Twitter: Word Clusters and Other Advances. Available online: http://www.cs.cmu.edu/~{}ark/TweetNLP/owoputi+etal.tr12.pdf (accessed on 30 August 2019).
39. Ikonomakis, M.; Kotsiantis, S.; Tampakas, V. Text classification using machine learning techniques. *WSEAS Trans. Comput.* **2005**, *4*, 966–974.
40. Guo, S.X.; Sun, X.; Wang, S.X.; Gao, Y.; Feng, J. Attention-Based Character-Word Hybrid Neural Networks with semantic and structural information for identifying of urgent posts in MOOC discussion forums. *IEEE Access* **2019**, 1–9. [CrossRef]