RESEARCH

Neural Architecture-based Treatment Side Effect Prediction from Online User-generated Content and User Credibility Analysis

Van-Hoang Nguyen*, Kazunari Sugiyama, Min-Yen Kan and Kishaloy Halder

*Correspondence:
vhnguyen@u.nus.edu
School of Computing, National University of Singapore, 13 Computing Drive, 117417, Singapore
Full list of author information is available at the end of the article

Abstract

Background: With Health 2.0, patients and caregivers increasingly seek information regarding possible drug side effects during their medical treatments in online health communities. These are helpful platforms for non-professional medical opinions, yet pose risk of being unreliable in quality and insufficient in quantity to cover the wide range of potential drug reactions. Existing approaches which analyze such user-generated content in online forums heavily rely on feature engineering of both documents and users, and often overlook the relationships between posts and user credibility within a common discussion thread. Inspired by recent advancements, we propose a neural architecture that models the textual content of user-generated documents and user experiences in online communities to predict side effects during treatment.

Results: Evaluations show that our proposed architecture is capable of capturing user credibility and encode online health content effectively. Learned credibility is shown to be highly representative of the common notion for trustworthiness. In the task of predicting treatment side effect from health discussion, the system is able to achieve the F_1 score of 0.793 and outperforms baseline models. Ablation study also shows that different components of the architecture – *i.e.,* User Credibility Learning, User Expertise Representation, and Cluster Attention, are crucial in achieving the highest performance.

Conclusions: Our solution proposes a novel method for jointly capturing both document contents and user features of online health communities. Other than the solving the task of treatment side-effect prediction, we also foresee many powerful applications of the architecture not only in health communities but also in general online discussion platforms.

Keywords: online health communities; credibility analysis; treatment side effect prediction; natural language processing

1 Background

Seeking medical opinions from online health communities has become commonplace: 71% of age 18–29 (equivalent to 59% of all U.S. adults) reported consulting online health opinion [1]. These opinions come from an estimated twenty to one hundred thousand health-related websites [2], inclusive of online health communities that network patients with each other to provide information and social support [3].

Drugs	Side effects
Lexapro	chills, constipation, cough, decreased appetite, **decreased sexual desire**, **diarrhea**, **dry mouth**, joint pain, muscle ache, tingling feeling, **sleepiness or unusual drowsiness**, unusual dream, **sweating**, ...
Xanax	abdominal or stomach pain, muscle weakness , **changed behavior**, chills, cough, decreased appetite, decreased urine, **diarrhea**, difficult bowel movement, cough, **dry mouth**, tingling feeling, **sleepiness or unusual drowsiness**, slurred speech, sweating, yellow eye...
Zoloft	**changed behavior**, decreased sexual desire, **diarrhea**, **dry mouth**, heartburn, **sleepiness or unusual drowsiness**, **sweating**...

Table 1: Side effects of anti-depressants.

Platforms such as HealthBoards[1] and MedHelp[2] feature users reporting their own health experiences, inclusive of their self-reviewed drugs and medical treatments. Hence, they are valuable sources for researchers [4, 5].

Although readers use these platforms to get valuable information about potential drug reactions during treatment, this is not potentially serious problems. There is lexical variation: users do refer side effects differently: "*dizziness*" can be expressed as "*giddiness*" or "*my head is spinning*". More concern is that discussions rarely cover all possible prescribed drugs and their side effects during a treatment, and some topics refer to a condition without mentioning any particular drug. Relying on such information could lead to adverse reactions.

It is important to note that a tool that looks up mentioned drugs' side effects from a static database would not return answers with sufficient coverage. There are also common concerns regarding credibility of user-generated contents – Impicciatore *et al.* [6] have shown that online health information is of variable quality and approached with caution.

Having these caveats in mind though, experienced users can provide valuable expertise. For instance, while reporting expected side effects for a specific treatment, patients with long-term use of certain drugs can be valuable authorities as follows:

While my experience of 10 years is with Paxil, I expect that Zoloft will be the same. You should definitely feel better within 2 weeks. One way I found to make it easier to sleep was to get lots of exercize. Walk or run or whatever to burn off that anxiety. – User 3690.

This is an answer to a thread asking for expected side effects for depression treatment with Zoloft. User 3690's history of actively discussing about other anti-depressants such as Lexapro and Xanax gives insights in predicting potential drug reactions during the treatment of depression. Table 1 shows that Zoloft (mentioned in the thread) shares many common side effects with the other two anti-depressants: "*changed behavior*," "*dry mouth*," and "*sleepiness or unusual drowsiness*."

A method that can differentiate trustworthy user-generated content from others would be valuable, allowing us to macroscopically harness a large amount of online information that would pave the way to many critical tasks such as digital pharmacovigilance [7] and disease monitoring [8]. Even on the microscopic level of individual posts, such a tool offers users' suggestions for drug reactions and improves the quality of user-generated content.

We address this need in our work. We build a neural architecture that models each post's textual content and its author's experience to predict expected side

[1] `https://www.healthboards.com/`
[2] `https://medhelp.ord/`

effects during treatments. Crucially, our supervised neural approach *jointly* learns content of each post and experience level of each user within a thread. Our key observation is that users can be grouped into clusters that share the same expertise or interest in certain drugs, possibly due to their common treatment or medical history. We leverage this expertise by embedding it into a low dimensional vector learned by the model, and subsequently predict side effects that are unmentioned in the discussion. We believe that our model represents trustworthiness more robustly when compared with representations such as a single weights [9] and traditional drug side effect extraction [10]. Furthermore, inspired by [11], we train a cluster-sensitive attention mechanism that allows our model to emphasize various parts of the post. We also follow general definition of truth discovery and let the model learn a credibility score that is unique to every user and reflective of her trustworthiness. Our experimental results show that integrating the above components outperforms baseline text classification models.

The contributions of our work are summarized as follows:

- We propose a neural network architecture that can capture user expertise, user credibility, individual post's and overall thread's semantic content.
- We formulate the task of side effect prediction during treatment as supervised multi-label classification and apply our proposed method to the task of side effect prediction during treatment.
- We record and analyze the performance of our proposed model through a set of progressively designed experiments. Additionally, we compare the obtained results with traditional text encoding algorithms.

2 Related Work

Our approach learns the representation of posts, threads and users, and then integrates them to apply to the task of drug side effect prediction during treatment. We thus review works on the representation of fundamental objects in online communities, and the discovery of drug side effects.

2.1 Modeling Objects in Online Communities

Post content modeling. In statement credibility prediction, linguistic features of a post are strong indicators for reliability.

Mukherjee *et al.* [12] adopted stylistic features – *i.e.*, the number of strong/weak modals, conditionals or negations – and affective features – *i.e.*, words that depict an author's attitude and emotion to represent a post's content. Such feature engineering requires a great amount of correlation analysis when applied to a novel problem or dataset.

Linguistic features also often fail to fully capture document content, as most do not account for distinctive words in exchange of scalability. Its counter parts, bag of words and per-vocabulary features loosely capture textual content but disregard semantics and suffer scalability with sparsity issues.

To address this, state-of-the-art architectures feature complex modeling to model subtle dependencies and rely on word embeddings to address scalability issues, achieving robust results in text classification [13], neural machine translation [14], among others.

User ID	Post	Drug mentioned	Aggregated side effects
3690	While my experience of 10 years is with Paxil, I expect that Zoloft will be the same. You should definitely feel better within 2 weeks. One way I found to make it easier to sleep was to get lots of exercize. Walk or run or whatever to burn off that anxiety.	Zoloft	changed behavior, decreased sexual desire, diarrhea, dry mouth, heartburn, sleepiness or unusual drowsiness,...
26521	I've heard of people going "cold turkey" and having withdrawal at 6 months! Please, get in contact with a doctor ASAP! "common symptoms include dizziness, electric shock-like sensations, sweating, nausea, insomnia, tremor, confusion, nightmares and vertigo"		

Table 2: A sample thread, including its list of post–user pairs, mentioned drugs, and side effects.

Inspired by the success of their approaches, we adopt the recurrent neural network architecture (RNN) for post content modeling. Coupled with an attention mechanism [15], our approach adaptively weights the importance of parts in each post.

Thread content modeling. Most of works on thread-level modeling usually obtain thread content representation by aggregating each content of its posts [16]. However, we hypothesize that each post has different contribution to thread content and should be variously weighted to reflect specific factors, such as its author's level of credibility.

User modeling. Statement credibility prediction often represents users by a single scalar that indicates their trustworthiness. The intuition is that users who provide trustworthy information frequently will be assigned high reliability scores [17]. Such representation is effective yet insufficient. Recent works have shown that encoding users into high-dimensional embeddings can improve system performance [18], which we have adopted in our model.

2.2 Side Effect Discovery

Most drug reaction discovery methods focus on extracting mentioned side effects. A common technique is to apply Named Entity Recognition (NER) and Relation Extraction (RE) systems in a supervised manner. sampathkumar *et al.* [19] demonstrated its effectiveness in detecting drugs and side effects that appear in a target document (in-context), and predicting if they are related.

However, in our side effect prediction during treatment, our model is required to cover potentially encountered reactions, many of which are not explicitly mentioned in the given post (out-of-context). Hence, we do not identify our task with traditional task of adverse drug side effect extraction [20]. Our approach overcomes the limitations of the existing works by first modeling user experience, credibility during post, and thread encoding, then subsequently predicting both in- and out-of-context side effects.

3 Preliminaries

Basic Terminologies. To ensure a consistent representation, let us first define some terminology:

- A *drug* d has a set of side effects, $S_d = \{s_1, s_2, \ldots, s_k\}$
- A *post* p, which is written by a *user* u and belongs to a *thread* t, is the most basic document, containing a sequence of sentences.
- A *user* u is a member of an online community. She participates in certain threads, *i.e.,* $T_u = \{t_1, t_2, \ldots, t_l\}$ by writing at least one post in each thread. We use the terms *user* and *author*, as well as *user experience* and *user expertise* interchangeably.
- A *thread* t (see Table 2) is an ordered collection of post–user pairs, $Q_t = [(p_1, u_1), (p_2, u_2), \ldots, (p_n, u_n)]$.
  Every thread discusses the treatment of a particular condition and entails a list of prescribed drugs $D_t = \{d_1, d_2, \ldots, d_m\}$. Hence, every thread has a list of aggregated side effects defined as $S_t = S_{d_1} \cup S_{d_2} \cdots \cup S_{d_m}$, which is also the list of potential side effects experienced during the treatment.

Task Definition. *Drug side effect prediction during treatment* is the task of assigning the most relevant subset of side effects to threads discussing certain treatment, from a large collection of potential side effects. We view the drug side effect prediction problem as a multi-label classification task. In our setting, an instance of item–label is a tuple $(\boldsymbol{x}_t, \boldsymbol{y})$ where $\boldsymbol{x}_t$ is the feature vector of thread t derived from its list of post–user pairs Q_t and $\boldsymbol{y}$ is the side effect label vector *i.e.*, $\boldsymbol{y} \in \{0,1\}^S$, where S is the number of possible side effect labels. Given training instances, we train our classifier to predict the list of treatment side effects in unseen threads.

Formal Hypothesis. Given a thread t with Q_t, we hypothesize that considering the credibility and experience of user $u \in (p, u) \in Q_t$ improves the quality of feature representation in thread t, resulting in better treatment side effect prediction.

4 Proposed Method

We propose a neural network-based supervised learning approach to predict a subset of side effects from a set of possible side effects. The network has 3 major components: user expertise representation with rich multi-dimensional vectors; cluster-sensitive attention being capable of focusing on relevant phases for post content encoding improvement; and credibility weighting mechanism which effectively learns to assign credibility score to each user based on his content and enhances thread encoding. Their implementation will be discussed in the followings together with an ablation study as baseline systems for our comparative evaluation. Figure 1 shows the detailed network architecture of our model.

User Expertise Representation (UE): We embed each user $u \in U$ as a vector $\boldsymbol{v}_u$ so that the vector captures user u's experience with certain drugs. As each user u participates in the threads T_u, entailing a list of experienced drugs, we derive user drug experience vector $\boldsymbol{v^*}_u \in \mathbb{R}^{|D|}$ where D is the set of all possible drugs and $v^*_{u_i} = n_{u_i}$ where user u has mentioned i^{th} drug in n_{u_i} threads. We obtain a user drug experience matrix $\boldsymbol{M}^* \in \mathbb{R}^{|U| \times |D|}$ where j^{th} row of $\boldsymbol{M}^*$ denotes user drug experience vector of j^{th} user $u_j \in U$. Since the average number of drugs experienced

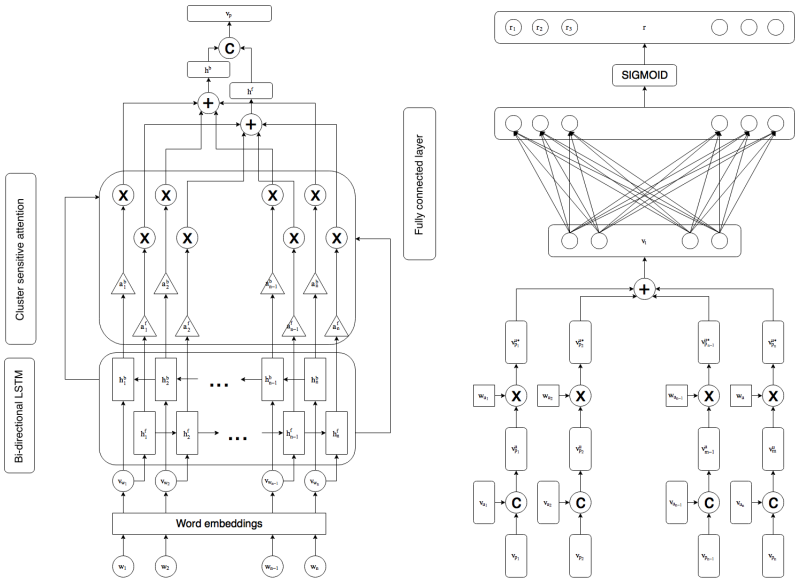

Figure 1: Full model architecture: (l, 1^{st} half) word embeddings and cluster sensitive attention, (r, 2^{nd} half) thread representation and multi label side effect prediction.

by each user is much fewer than the total number of drugs (see Table 3), M^* suffers from data sparsity and limited scalability. Without dimensionality reduction, the model learns at least $|D|$ parameters for every user, amounting to $|D| \times |U|$ when aggregated for all users. Data sparsity causes a large number of insufficiently tuned parameters, resulting in significant increase of training time, storage, and decline of the system's robustness.

We apply Principal Component Analysis (PCA) [21] to $\boldsymbol{M}^*$ obtained from training set. Figure 2 shows percentage of variance explained versus number of included principal components (PCs) to determine the number of PCs, g. Since our PCA plots do not show added explanation percentage beyond 50 components, we use $g = 50$ components, reducing our original $\boldsymbol{M}^* \in \mathbb{R}^{|U| \times |D|}$ to user expertise matrix $\boldsymbol{M} \in \mathbb{R}^{|U| \times g}$.

User Clustering: To model per-user expertise, in a naive setting, we train $\approx |U| \times g$ parameters. Given limited data, this is infeasible as it faces sparsity issues. We make a second, key assumption that our set of users U can be grouped into a set of meaning clusters C of size k where $k \ll |U|$. Users within a cluster would have experience with similar drugs, and hence representable using a single vector, reducing the number of learned parameters to $k \times g$.

We apply K-means clustering algorithm [22] to cluster the users into k groups. To determine the number of clusters k, we analyze the total distance to the nearest centroid versus the number of potential clusters in set C – as in Figure 3, where $D(C)$ is defined as follows:

$$D(C) = \frac{\sum_{c \in C} \sum_{u \in c} dist(\boldsymbol{v}_c, \boldsymbol{v}_u)}{argmax D(C)}, \tag{1}$$

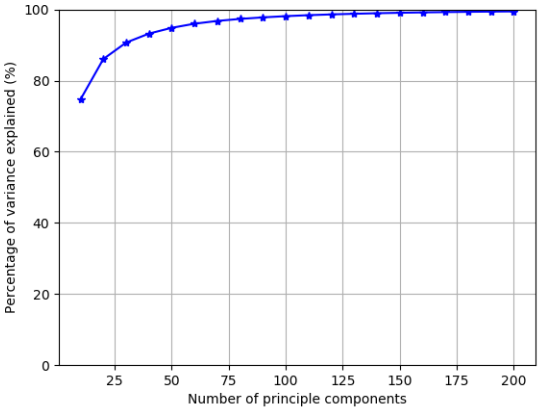

Figure 2: Principal component analysis.

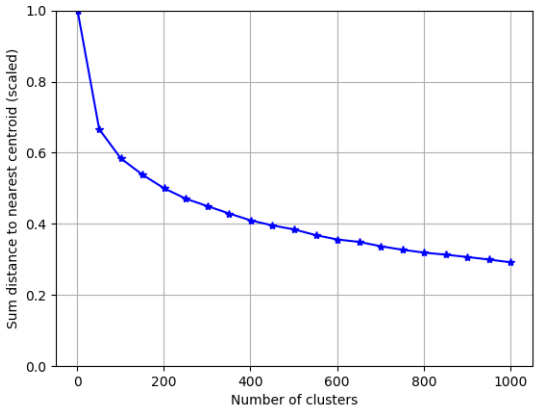

Figure 3: Cluster analysis obtained by K-means.

where $argmaxD(C)$ is the maximum total distance obtained when $|C| = 1$.

Since clustering does not gain significant reduction in total distance beyond 100 clusters, we sort each user to a cluster $c \in C$ where $|C| = k = 100$. For each user, we take the vectors of cluster's centroid assigned by each user as the user's expertise vector.

Post Content Encoding: The network takes the content of a thread t as input, which is a list of post–user pairs Q_t. Post p_i of pair $(p_i, u_i) \in Q_t$ consists of a sequence of words $(w_1, \ldots, w_n)$. We seek to represent a post p_i as vector $\boldsymbol{v}_p$ that effectively captures its semantics. We embed each word into a low dimensional vector and transform the post into a sequence of word vectors $\{\boldsymbol{v}_{w_1}, \boldsymbol{v}_{w_2}, \ldots, \boldsymbol{v}_{w_n}\}$. Each word vector is initialized using Google's pre-trained word2vec [23]. Additionally, while each out-of-vocabulary word vector is initialized randomly, we keep it tunable during training to capture domain-specific meanings. Such model adaptation is necessary, as the model needs to learn the embeddings for the drug names, most of

which are not included in the pre-trained embeddings but are critical to predict the side effects.

We employ long-short term memory (LSTM) [24] to encode the textual content. A bi-directional LSTM encodes the word vector sequence, outputting two sequences of hidden states: a forward sequence, $H^f = \boldsymbol{h}_1^f, \boldsymbol{h}_2^f, \ldots, \boldsymbol{h}_n^f$ that starts from the beginning of the text; and a backward sequence, $H^b = \boldsymbol{h}_1^b, \boldsymbol{h}_2^b, \ldots, \boldsymbol{h}_n^b$ that starts from the end of the text. For many sequence encoding tasks, knowing both past (left) and future (right) contexts has proven to be beneficial [25]. The states h_i^f and h_j^b in the forward and backward sequences are computed as follows:

$$\boldsymbol{h}_i^f = LSTM(\boldsymbol{h}_{i-1}^f, \boldsymbol{w}^i),\ \boldsymbol{h}_j^b = LSTM(\boldsymbol{h}_{j+1}^b, \boldsymbol{w}^j),$$

where $\boldsymbol{h_i^f}, \boldsymbol{h_j^b} \in \mathbb{R}^e$, and e are the number of encoder units.

Cluster-sensitive Attention (CA): Inspired by [11], we initialize an attention vector, $\boldsymbol{v}_{a_i} \in \mathbb{R}^e$ for each cluster c_i. Given a forward sequence $H^f = \boldsymbol{h}_1^f, \boldsymbol{h}_2^f, \cdots, \boldsymbol{h}_n^f$ and backward sequence $H^b = \boldsymbol{h}_1^b, \boldsymbol{h}_2^b, \cdots, \boldsymbol{h}_n^b$ of hidden post states p written by user u belonging to cluster c_i, the corresponding w_{a_j} weights each hidden state $\boldsymbol{h}_j^f$ and $\boldsymbol{h}_j^b$ of both sequences based on their similarity with the attention vector are:

$$w_{a_j} = \frac{exp(\boldsymbol{v}_{a_i}\boldsymbol{h}_j)}{\sum_{l=1}^{n} exp(\boldsymbol{v}_{a_i}\boldsymbol{h}_j)}. \tag{2}$$

The intuition behind Equation (2), inspired by [15], is that hidden states which are similar to the attention vector $\boldsymbol{v}_{a_i}$ should be paid more attention to; hence are weighted higher during document encoding. $\boldsymbol{v}_{a_i}$ is adjusted during training to capture hidden states that are significant in forming the final post representation. w_{a_j} is then used to compute forward and backward weighted feature vectors:

$$\boldsymbol{h}^f = \sum_{j}^{n} w_{a_j}\boldsymbol{h}_j^f, \quad \boldsymbol{h}^b = \sum_{j}^{n} w_{a_j}\boldsymbol{h}_j^b. \tag{3}$$

We concatenate the forward and backward vectors to obtain a single vector, following previous bi-directional RNN practice [26].

Thread Content Encoding with Credibility Weights (CW): For every post–user pair (p_i, u_i) of thread t, we first compute feature vector $\boldsymbol{v}_{p_i}$ for post p_i.

It is then concatenated with user u_i's expertise vector $\boldsymbol{v}_{u_i}$ to form post–user complex vector $\boldsymbol{v}_{u_i}^p$. This user-post complex is weighted by a user credibility $e^{-w_{u_i}}$, where w_{u_i} initially randomized per user and updated while training, to obtain final post–user pair representation $\boldsymbol{v}_{u_i}^{p*}$. We implement credibility learning according to the general intuition from the truth discovery literature: users who give quality posts, on which the model can solely base to make correct predictions, are given a higher credibility. We also exploit this credibility score to encode the thread representation more effectively by placing emphasis on the content of credible users.

# Users	14,388
# Threads	99,682
Avg. # of words per post	73.65
Avg. # of posts per thread	8.16
Avg. # of threads per user	26.21
# Side effects (SE)	1,500
Avg. # of SEs per thread	90.47
# Drugs	1869
Avg. # of drugs per user	19.72

Table 3: Dataset statistics.

A representation of thread that meets the above description is the weighted sum of each post–user complex vector:

$$\boldsymbol{v}_t = \sum_{i=1}^{n} \boldsymbol{v}_{u_i}^{p*} = \sum_{i=1}^{n} e^{-w_{u_i}} \boldsymbol{v}_{u_i}^{p} \tag{4}$$

Multi-label Prediction: We feed the thread content representation v_t through a fully connected layer whose outputs can be computed as follows:

$$\boldsymbol{s}_t = \boldsymbol{W} tanh(\boldsymbol{v}_t) + \boldsymbol{b}, \tag{5}$$

where $\boldsymbol{W}$ and $\boldsymbol{b}$ are weights and biases of the layer. The output vector $\boldsymbol{s}_t \in \mathbb{R}^{|S|}$ is finally passed through a sigmoid activation function $\sigma(\cdot)$, and trained using cross-entropy loss L defined as follows:

$$\begin{aligned} L = \frac{1}{T} \sum_{t=1}^{T} \{ \boldsymbol{y}_t \cdot log(\sigma(\boldsymbol{s}_t)) + (1 - \boldsymbol{y}_t) \cdot log(1 - \sigma(\boldsymbol{s}_t)) \} \\ + \lambda_1 \sqrt{\sum_u \boldsymbol{v}_u^2} + \lambda_2 \sum_i |\boldsymbol{w}_{u_i}| \end{aligned} \tag{6}$$

We adopt regularization that penalizes the training loss with the user experience matrix's $L2$ norm by a factor of λ_1 and the user score vector $\boldsymbol{w}_u$'s $L1$ norm by a factor of λ_2. The loss function is differentiable, thus trainable with Adam optimizer [27]. During our gradient-based learning, user u_i's credibility score w_{u_i} can be updated by calculating $\frac{\partial L}{\partial w_{u_i}}$ by back-propagation (see Appendix).

5 Experiments

We conduct experiments to validate the effectiveness of our proposed model. More specifically, we (1) compare our architecture with text encoding baselines, (2) highlight performance improvements incrementally, and (3) evaluate and analyze the obtained results, both at the macroscopic and microscopic levels.

5.1 Dataset

We conduct our experiments on the same dataset as [12] including 15,000 users and 2.8 million posts extracted from 620,510 HealthBoards[1] threads.

Ground truth possible side effects experienced during treatment are defined as the side effects of the drugs which are mentioned in the discussion. As annotating such

amount of posts is expensive, drug side effects are extracted from Mayo Clinic's Drugs and Supplements portal[3] and are used as surrogates for potential reactions of treatments.

5.2 Baselines

As a competitive baseline from prior work, CNN-KIM [13] first constructs a document matrix that incorporates word embeddings, and then applies a convolution filter to obtain feature maps. These feature maps are passed through a max-pooling filter to construct a document representation. During prediction, the representation is fed through a fully connected layer. We replace the final softmax layer of the author's model with sigmoid to make it work in a multi-label prediction setting.

We employ the following baselines to perform an ablation test of our model.

- **RNN**: We implement a bi-directional LSTM baseline, which is equivalent to our proposed method without CA, UE and CW.
- **Weighted Post Encoder (WPE)**: We construct thread representation by summing each of its post–user complex vector weighted by user credibility. This is equivalent to our proposed methodology without CA and UE.
- **Weighted Post Encoder with User Expertise (WPEU)**: We concatenate user expertise with post vector to create post–user complex vector. This is equivalent to our proposed method without CA.

5.3 Experimental Settings

We applied a standard natural language preprocessing — Snowball stemming [28] and stop-word elimination — before representation modeling. From the original dataset, we only extract threads that are annotated with drugs and their side effects, along with the lists of contained posts and corresponding users. Table 3 shows the dataset statistics. We divide our data into 10 folds to perform cross-validation (8,1, and 1 folds for training, validation, and testing, respectively). We perform PCA and K-means clustering on training set, using scikitlearn's built-in modules [29], 50 principal components ($g = 50$) and 100 clusters ($k = 100$).

For CNN-KIM, we experiment with filters with varying window sizes from 2 to 5, and set the number of feature maps for each filter to 256 and dropout to 0.5. For our proposed model and baseline models using the RNN architecture, when performing post content encoding, we set the number of units in the LSTM cell to 128. We use dropout rates of 0.2 and 0.5 in our LSTM cells and fully-connected layers, respectively. Cluster attention vectors and user credibility values are initialized with values ranging from -1.0 to 1.0. We initialize each user u's expertise vector with the value of v_u obtained in Section 4 and allow training to fine-tune. All models are trained using Tensorflow[4] library.

We conducted the following two separate experiments:

- **Experiment 1**: We keep the text as-is. Any mentioned drugs are retained inside the thread.
- **Experiment 2**: We remove all mentions of any drug in our drug list. This is a more aggressive experiment which requires the model to predict the treatment's side effects without any mention of the experienced drugs.

[3] `https://www.mayoclinic.org/drugs-supplements`
[4] `https://www.tensorflow.org/`

System	Components			Experiment 1			Experiment 2		
	CW	UE	CA	Pre.	Rec.	F_1	Pre.	Rec.	F_1
1. CNN-KIM				0.818	0.677	0.751	0.813	0.503	0.614
2. RNN				0.810	0.657	0.735	0.808	0.484	0.599
3. WPE	✓			**0.873**	0.678	0.773	**0.859**	0.507	0.638
4. WPEU	✓	✓		0.865	0.705	0.781	0.819	0.537	0.643
5. Our model	✓	✓	✓	0.844	**0.730**	**0.793**	0.788	**0.573**	**0.659**

Table 4: Experimental results obtained by both actual (Experiment 1) and Strict (Experiment 2) settings. In the "Component" columns, "CW", "UE", "CA" denote "Credibility Weights", "User Expertise" and "Cluster Attention module components", respectively.

6 Results and Discussion

Table 4 shows the precision, recall, and F_1 obtained by our method and the four baselines.

Macroscopic Analysis: Firstly, all of the three models that apply credibility weighting (CW) – WPE, WPEU, and our model – outperform both RNN and CNN baselines in both experiments. Specifically, in Experiment 1, weighting each post by its author credibility improves the performance of naive post encoder by 6.32%, 2.15% and 3.86% on precision, recall, and F_1, respectively. We observe similar trends in Experiment 2. These demonstrate the effectiveness of accounting for author credibility when encoding thread content, improving side effect prediction.

Improvements by incorporating user experience (UE) are less pronounced. In Experiment 1, adding UE (WPEU vs. WPE) improves recall by 2.65% and 0.8% in F_1. Again, Experiment 2, which is stricter than Experiment 1, shows similar performance trends. On a macro scale, these statistics indicate that our model successfully learns to include more side effects in its prediction, where many are relevant to the ground truth. This is consistent with our hypothesis that considering author experience of each post is effective in predicting out-of-context side effects.

Applying cluster-sensitive attention (CA) in combining RNN's hidden states also improves the performance. In Experiment 1, we observe that adding CA (our model vs. WPEU) also improves recall and F_1, where again, Experiment 2 demonstrates similar but slightly more pronounced performance changes. These indicate that the attention mechanism is more effective when the drugs are present since the drug names in our documents are the phrases that receive greater emphasis.

As settings in Experiment 1 start with more information compared with those in Experiment 2, the task is easier and thus performance is improved (12.7% to 14.15% in F_1). The margin for improvement for Experiment 2 is larger, which explains why absolute score improvements are larger in Experiment 2. When measuring relative improvement, the gains are comparable.

Generally, according to the macroscopic analysis of results in Table 4, we conclude that all of the three components in our proposed architecture, namely, CW, UE, and CA have a positive impact on the overall performance of the model. We observe consistent improvements in F_1 after adding each component is consistent with our stated hypotheses, in both experimental settings.

Microscopic Analysis: We also analyze our model performance at per-sample level to check whether they are consistent with the macroscopic results. We aim to

User ID	Experienced drugs	Top common experienced side effects
24296	rifampin, **vitamin**, clarithromycin, aciphex, a zithromax, **plaquenil**, **flagyl**, minocycline, levaquin, tetracycline, **tinidazole**, advil	**diarrhea**, bad breath, **headache, heartburn, unusual tiredness and weakness, nausea, fever**
1537	**vitamin**, rocephin, hydroquinone, **plaquenil**, **flagyl**, minocycline, levaquin, **tinidazole**	**diarrhea, skin rash, headache, dizziness, heartburn**, bad breath, **sleepiness**
5232	doxycycline, prozac, **vitamin**, norvasc, tylenol, **flagyl**, questran, biotin, cefuroxime, **plaquenil**	bad breath, **diarrhea, nausea, dizziness, unusual tiredness and weakness**
16248	celexa, prilosec, **vitamin**, rocephin, klonopin, nexium, fumarate, elidel, citrate, prozac	**diarrhea**, sneezing, **nausea**, excessive gas, body pain, loss voice, **heart burn**

Table 5: Experienced drugs and common side effects among users.

User ID	Posts	Output side effects					
		CNN-KIM	RNN	WPE	WPEU	Our model	Ground truth
24296 (credibility: 0.11)	[...] **little red rashes all over my body** that resembled vasculitis. [...] I was diagnosed and treated with the "standard treatment" twice, to not much effect), a very stiff neck, really bad brain fog and **confusion**. [...]	**diarrhea, skin rash**	**skin rash**	**headache, diarrhea, skin rash**	**headache, diarrhea, unusual tiredness and weakness, dizziness, sleepiness fever, nausea**, bad breath	**headache, diarrhea, unusual tiredness and weakness, dizziness, sleepiness, fever, nausea, heartburn, belching, indigestion, acid stomac, difficult bowel movement**, bad breath, bone joint pain	headache, diarrhea, unusual tiredness and weakness, dizziness, fever, nausea, loss appetite, chills, heartburn, belching, indigestion, acid stomach, confusion, skin rash, weight loss, difficult bowel movement, shakiness
1537 (credibility: 0.32)	[...] now last month my symptoms including joint pains, twitching and tremors and **bug crawling under my skalp sensations reappeared** [...]						
5232 (credibility: 0.36)	[...] I don't know about **cysts in the brain per se** [...]						
16248 (credibility: 0.21)	[...] I've been growing increasingly **sensitive to more foods** over the last year [...] How do you know that you had **damage to your intestines** from Lyme? [...] I'm curious because I am in the process of getting a Lyme work up and my **intestines are messed up**, but all GI tests came back negative.						

Table 6: A sample thread in the test set, mentioning drugs *Flagyl, Tinidazole, Plaquentil,* and *Vitamins*.

confirm three hypotheses: (1) Considering author expertise improves prediction on out-of-context side effects, (2) Considering author credibility improves the extraction of both in- and out-of-context side effects from trustworthy users' content, and (3) Placing attention on different parts of the document enhances the performance of in-context side effect extraction. Tables 5 and 6 show a sample testing thread, its users' commonly experienced drugs, and its side effects.

We observe that CNN-KIM and the simple, RNN-based post encoding can capture side effects that are mentioned both directly (*e.g.*,"*skin rash*") as well as indirectly

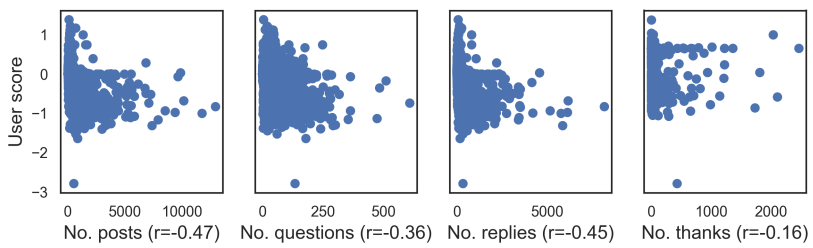

Figure 4: Correlation between user score and online interaction.

(*e.g.*, "*diarrhea*"), but fail to capture the remaining symptoms, many of which are out-of-context.

Considering User 1537's credibility shows performance improvements. In her posts, User 1537 indirectly refers to "*headache*" by mentioning "*bug crawling under my skalp sensations*". The calculated higher credibility score weights User 1537 experiences with "*sleepiness*" higher in the WPEU (CW + UE) baseline prediction, which is correct. These observations are consistent with our hypothesis about user credibility.

User experience is effective in predicting out-of-context symptoms. In the illustrated sample training set, all of the four users have experience with similar drugs with common side effects such as "*unusual tiredness and weakness*", "*nausea*", and "*fever*". As "*bad breath*" is also a shared side effect, it is comprehensible that the model outputs "*bad breath*". Nonetheless, it is intuitive for the model to pick up such commonness among users and compute relevant results. These observations are consistent with our hypothesis on user experience.

Finally, the model with CA can learn different parts of the documents. Especially for User 16248's posts that mentioned digestive problems, hidden states encode phrases such as "*increasingly sensitive to more foods*", and "*damage to your intestines*" receive higher attention, resulting in the prediction of "*heartburn*", "*belching*", "*indigestion*", "*acid stomach*", and "*difficult bowel movement*". This functionality is consistent with our original purpose and expectation for adding attention to the post encoder architecture.

User Credibility Analysis: In this section, we discuss how representative are the credible users assigned by the model to our common notion of trust-worthy users in online communities. Although the dataset (see Section 5.1) does not provide any credibility evaluation, it includes features of user's interaction on the forums. We believe that an active and enthusiastic contributor to the online platforms should significantly express trustworthiness. Therefore, we examine the correlation between our learned credibility and community interaction, specifically by plotting the correlation between our learned user score w_u and the interaction indicators: the number of user's interacted posts, raised questions, replies, and received thanks.

Figure 4 shows the scatter plot of the learned user's scores w_u and the four features. The credibility score, which weights each post-user complex vector in constructing weighted thread representation, is computed as $e^{-w_{u_i}}$. Thus, note that the lower the scores, the higher the credibility. Based on Pearson's correlation (denoted

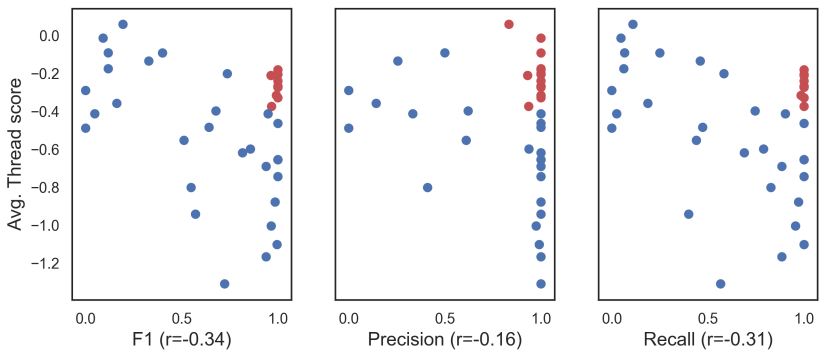

Figure 5: Correlation between average thread score and our model evaluated with F_1, precision, and recall, in which Pearson correlations are -0.34, -0.16, and -0.31, respectively.

as r), we can observe that the learned user scores correlate with their interaction level in the online communities.

We will later offer potential explanations for why this correlation is not perfect, especially with the number of thanks, as one of our model's limitations.

We believe that trustworthy users are those whose opinion are valuable and can help other user's decision. Therefore, we hypothesize that if given threads contributed by many trust-worthy users, our model would gain valuable information and give more accurate predictions, reflected by high F_1, Precision and Recall scores. We verify if credible users can help our model's prediction by examining the correlation between the average user score of a thread and the model's performance in term of the above metrics. The average thread score of thread t is defined as $\frac{1}{N}\sum_i^N w_{u_i}$, where t's size $= N$. The lower the average thread score of t, the higher the average credibility of users in t. Figure 5 shows a negative Pearson's correlation and indicates that our model tends to perform better, especially in term of F_1 and Recall, on threads with low average thread score or whose users are highly credible. From the scatter plots, we can also observe that there are many examples (red dots in Figure 5) where our model gives highly accurate prediction, although the average thread score of them are relatively high. As explained at "Microscopic Analysis" in Section 6, our model can obtain the credible information from the trustworthy users, even in the thread with overall low credibility. The analysis on these "outliers" also justifies why the correlation between our model's performance and the average thread's credibility cannot be perfect. The high correlation between user credibility and the recall score of our prediction also indicates that our model can successfully identify "uncommon" side effects from trustworthy users, and effectively reduces the number of missed-out side effects.

Limitations: We foresee a limitation arisen from our design of user's credibility defined by Equation (4). A user's credibility can be damaged if their posts do not directly help with predicting the correct side effects. These posts are questionable when users are asking for some information instead of giving answers. We also observe the cases where users receive thanks for giving helpful information such as suggesting nutritious diet or healthy lifestyle without mentioning any relevant

side effects. We recognize this kind of limitation in our model when users without malicious intent can also be assigned a low credibility score. This case of "failure" can explain the few "false negative" points in the plot of user score and number of thanks (Figure 4) where those points are assigned low to moderate credibility despite their high number of thanks. However, our definition makes sure that the credibility learning mechanism does not express the opposite adverse behavior of assigning high credibility to untrustworthy users.

Overall, our analysis suggests that user credibility scores, although learned in an unsupervised manner, can capture the expected notion of credibility and are representative of trustworthiness. Every component of our architecture is also shown to be vital in achieving the highest performance.

7 Conclusion

We have addressed the importance of user experience and credibility in modeling thread contents of online communities, specifically through the task of drug side effect prediction during treatment. Our approach suggests a subset of side effects relevant to the mentioned treatment in the given discussion, taking into account the each post content and its author expertise in certain treatments. Mainstream models for online communities fail to fully capture post content semantically and user experience with previous drugs.

We modeled users' expertise by examining their experience with different drugs, and then group the users with similar experience into clusters that share a common experience vector representation. We also proposed an unsupervised method which assigns credibility to users based on the correctness of their contents and overall improves thread representations.

We have also discussed how representative is the learned credibility of the common notion of trustworthiness, as well as acknowledged the limitations of our unsupervised credibility learning. However, experimental results show that our proposed thread content encoder outperforms state-of-the-art document encoders, and that our neural components, including the learned user credibility, play a significant role in improving task performance. Our analysis also shows that the learned credibility scores correlate with other indicators of active interaction in the communities.

We believe that our model is applicable to other domains. We plan to use it for downstream application in online health community such as credibility analysis and thread recommendation in the future.

Appendix

Updating user score with stochastic gradient descent and back-propagation

The overall loss function in Equation (6) can be rewritten as logistic loss on a single training example and a single label s as follows:

$$L = log(1 + exp(-y_t(\boldsymbol{w}_s^T tanh(v_t) + b_s))), \tag{7}$$

where y_t is the binary truth for label s, b_s is a classification bias, and $w_s \in \mathbb{R}^{g \times 1}$ is a row of $\boldsymbol{W}$ in Equation (5). $\boldsymbol{w}_s$ is the classification weights of a single label s.

In back-propagation, we update the score w_{u_i} of user u_i based on the gradient calculated by taking the derivative of the loss L with regard to w_{u_i}:

$$\frac{\partial L}{\partial w_{u_i}} = \frac{(1 - tanh^2(v_t)) y_t \boldsymbol{w}_s^T \boldsymbol{v}_{u_i}^p e^{-w_{u_i}}}{1 + exp(y_t(\boldsymbol{w_s}^T tanh(v_t) + b_s))} = \nabla_{w_{u_i}} L. \tag{8}$$

User score w_{u_i} is updated as follows:

$$w_{u_i}^{t+1} = w_{u_i}^t - \eta \nabla_{w_{u_i}^t} L, \tag{9}$$

where η is the learning rate.

When the prediction is correct, y_t and $(\boldsymbol{w}_s^T tanh(v_t) + b_s)$ share the same sign and $y_t(\boldsymbol{w}_s^T tanh(v_t) + b_s))$ is highly positive, making the denominator highly positive and the overall gradient small. The user score w_{u_i} is minimally updated.

When the prediction is incorrect, y_t and $(\boldsymbol{w_s}^T tanh(v_t) + b_s)$ have different signs and the denominator approaches 1. In the nominator, $\boldsymbol{w}_s^T \boldsymbol{v}_{u_i}^p$ is the prediction if we solely consider the post vector $\boldsymbol{v}_{u_i}^p$ of user u_i.

- If this prediction is correct, which fits our definition of credible user, $y_t \boldsymbol{w}_s^T \boldsymbol{v}_{u_i}^p$ is positive, making the overall gradient positive. User score w_{u_i} is updated in the negative direction and the credibility score $e^{-w_{u_i}}$ used to weight user u_i's content increases.
- On the other hand, when the prediction from solely considering the post vector $\boldsymbol{v}_{u_i}^p$ of user u_i is incorrect, indicating a not credible user, $y_t \boldsymbol{w}_s^T \boldsymbol{v}_{u_i}^p$ is negative, and the overall gradient is negative. w_{u_i} is updated in the positive direction and the credibility score $e^{-w_{u_i}}$ used to weight user u_i's content decreases.
- The magnitude of the gradient is proportional to $e^{-w_{u_i}}$. This indicates that users who are currently learned as credible are most affected by back-propagation when the model's prediction is incorrect.

Abbreviations

Avg.: Average
CA: Cluster-sensitive Attention
CNN: Convolutional Neural Network
CNN-KIM: Convolutional Neural Network developed by Yoon Kim [13]
CW: Thread Content Encoding with Credibility Weights
F1 score: The harmonic average of the precision and recall
LSTM: Long-short Term Memory

PCA: Principal Component Analysis
R: Pearson's correlation
RNN: Recurrent Neural Network
UE: User Expertise Representation
#: Number of

Declarations

Acknowledgements
None declared.

Funding
None declared.

Authors' contributions
All authors conceived of and planned the reported work. KS proposed the problem statement and suggested literature for review. VHN mainly designed the model architecture with the help from MYK and KH. VHN implemented the experiments, and interpreted the results. VHN, KS, MYK took the lead in writing the manuscript with support from KH. All authors discussed the results and commented on the manuscript.

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
All authors declare that they have no competing interests.

Availability of data and materials
The dataset used in this paper was first published in [12] and is publicly available at https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/impact/peopleondrugs/

References

1. Fox, S., Duggan, M.: Health Online 2013. Health **2013**, 1–55 (2013)
2. Diaz, J.A., Griffith, R.A., Ng, J.J., Reinert, S.E., Friedmann, P.D., Moulton, A.W.: Patients' Use of the Internet for Medical Information. Journal of General Internal Medicine **17**(3), 180–185 (2002)
3. Johnston, A.C., Worrell, J.L., Di Gangi, P.M., Wasko, M.: Online Health Communities: an Assessment of the Influence of Participation on Patient Empowerment Outcomes. Information Technology & People **26**(2), 213–235 (2013)
4. Leyens, L., Reumann, M., Malats, N., Brand, A.: Use of Big Data for Drug Development and for Public and Personal Health and Care. Genetic Epidemiology **41**(1), 51–60 (2017)
5. Martin-Sanchez, F., Verspoor, K.: Big data in medicine is driving big changes. Yearbook of Medical Informatics **9**(1), 14–20 (2014)
6. Impicciatore, P., Pandolfini, C., Casella, N., Bonati, M.: Reliability of Health Information for the Public on the World Wide Web: Systematic Survey of Advice on Managing Fever in Children at Home. BMJ **314**(7098), 1875 (1997)
7. Salathé, M.: Digital Pharmacovigilance and Disease Surveillance: Combining Traditional and Big-Data Systems for Better Public Health. The Journal of Infectious Diseases **214**(suppl_4), 399–403 (2016)
8. St Louis, C., Zorlu, G.: Can Twitter Predict Disease Outbreaks? BMJ: British Medical Journal (Online) **344**(e2353) (2012)
9. Li, Y., Gao, J., Meng, C., Li, Q., Su, L., Zhao, B., Fan, W., Han, J.: A survey on truth discovery. ACM SIGKDD Explorations Newsletter **17**(2), 1–16 (2016)
10. Aramaki, E., Miura, Y., Tonoike, M., Ohkuma, T., Masuichi, H., Waki, K., Ohe, K.: Extraction of Adverse Drug Ffects from Clinical Records. MedInfo 2010 **160**, 739–743 (2010)
11. Halder, K., Poddar, L., Kan, M.-Y.: Cold Start Thread Recommendation as Extreme Multi-label Classification. In: Proc. of the Workshop on Extreme Multilabel Classification for Social Media Co-located with the Web Conference (WWW'18 Companion), pp. 1911–1918 (2018)
12. Mukherjee, S., Weikum, G., Danescu-Niculescu-Mizil, C.: People on Drugs: Credibility of User Statements in Health Communities. In: Proc. of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'14), pp. 65–74 (2014)
13. Kim, Y.: Convolutional Neural Networks for Sentence Classification. In: Proc. of the 2014 Conference on Empirical Methods on Natural Language Processing (EMNLP 2014), pp. 1746–1751 (2014)
14. Luong, M.-T., Le, Q.V., Sutskever, I., Vinyals, O., Kaiser, L.: Multi-task Sequence to Sequence Learning. In: Proc. of the 4th International Conference for Learning Representations (ICLR2016) (2016)
15. Luong, M.-T., Pham, H., Manning, C.D.: Effective Approaches to Attention-based Neural Machine Translation. In: Proc. of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015), pp. 1412–1421 (2015)

16. Yang, D., Piergallini, M., Howley, I., Rose, C.: Forum Thread Recommendation for Massive Open Online Courses. In: Proc. of the 7th International Conference on Educational Data Mining (EDM 2014), pp. 257–260 (2014)
17. Li, Y., Du, N., Liu, C., Xie, Y., Fan, W., Li, Q., Gao, J., Sun, H.: Reliable Medical Diagnosis from Crowdsourcing: Discover Trustworthy Answers from Non-Experts. In: Proc. of the 10th ACM International Conference on Web Search and Data Mining (WSDM 2017), pp. 253–261 (2017)
18. Yu, Y., Wan, X., Zhou, X.: User Embedding for Scholarly Microblog Recommendation. In: Proc. of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016), pp. 449–453 (2016)
19. Sampathkumar, H., Chen, X.-W., Luo, B.: Mining Adverse Drug Reactions from Online Healthcare Forums using Hidden Markov Model. BMC Medical Informatics and Decision Making **14**(1), 91–108 (2014)
20. 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: Proc. of the 2010 Workshop on Biomedical Natural Language Processing (BioNLP 2010), pp. 117–125 (2010)
21. Jolliffe, I.T.: Principal Component Analysis and Factor Analysis. Statistical Methods in Medical Research **1**(1), 115–128 (1986)
22. MacQueen, J.: Some Methods for Classification and Analysis of Multivariate Observations. In: Proc. of the 5th Berkeley Symposium on Mathmatical Statistics and Probability, pp. 281–297 (1967)
23. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: In Proc. of the Advances in Neural Information Processing Systems (NIPS 2013), pp. 3111–3119 (2013)
24. Hochreiter, S., Schmidhuber, J.: Long Short-Term Memory. Neural Computation **9**(8), 1735–1780 (1997)
25. Dyer, C., Ballesteros, M., Ling, W., Matthews, A., Smith, N.A.: Transition-Based Dependency Parsing with Stack Long Short-Term Memory. In: Proc. of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015), pp. 334–343 (2015)
26. Ma, X., Hovy, E.: End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In: Proc. of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016), pp. 1064–1074 (2016)
27. Kingma, D.P., Ba, J.: Adam: A Method for Stochastic Optimization. In: Proc. of the 3rd International Conference for Learning Representations (ICLR2015) (2015)
28. Porter, M.F.: An Algorithm for Suffix Stripping. Program **14**(3), 130–137 (1980)
29. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research (JMLR) **12**(2011), 2825–2830 (2011)