Predicting User Reported Symptoms Using a Gated Neural Network

Lahari Poddar, Wynne Hsu, Mong Li Lee
School of Computing
National University of Singapore
{lahari, whsu, leeml}@comp.nus.edu.sg

***Abstract*—Detection of Adverse Drug Events (ADE) or side effects of different treatments are necessary to minimize potential health risks of patients. Given the prevalence of user reported content on the web, recent research has focused on the automatic discovery of potential side effects from these online platforms. However, it is not clear whether the symptoms that a patient experiences, are solely side effects of a particular treatment (or a combination of them), or there are other confounding factors influencing them. In this work, we characterize the reported symptoms along with their severity for patients, based on their past interactions with various treatments and their pre-existing medical conditions. We analyze a large dataset from a symptoms tracking app, and observe a strong correlation between a patient's existing health condition(s) and the symptoms he or she experiences across different treatments. We develop a multi-objective neural network with gating mechanism, to predict the possible symptoms and their overall severity level for a set of treatments, for a given patient. Experimental results demonstrate the effectiveness of our model over state-of-the-art approaches. Furthermore, our adaptation of the gating mechanism imbues the network with the ability of justifying its predictions.**

I. Introduction

In the last decade, reporting health information online has become widespread via social networking sites (e.g., Twitter), health forums (e.g. WebMD, HealthBoards), health monitoring apps (e.g. Flaredown, Symple) and so on. Online health resources constitute an important source of medical information, with 59% of the adult US population seeking health-related information online [1], and nearly half of US physicians relying on them for professional use [2]. Lately with the advent of health 2.0, users across the globe not only look up professional health information online but also play an active role by self-reporting their clinical experiences with various treatments. This has led to a surge of research interest to discover medical insights, such as identifying potential side effects of drugs [3], [4], [5]. While these large amount of user-reported health information can help complement existing medical knowledge and speed up discoveries of potential drug reactions [6], [7], there remains a widespread concern of whether the reported side effects are truly due to the drugs [8], [9]. In a realistic scenario, patients experience a set of clinical symptoms which could potentially stem from multiple confounding factors. This makes it difficult to claim if the symptoms are side effects of a drug, by a patient with little medical training. Furthermore, a patient is often under the influence of multiple drugs, and the experienced symptom could be a synergistic effect caused by a combination of the concurrent drug use, instead of being side effects of one of them. A few sample reports are shown in Table I from a real world dataset. We can observe that various symptoms are experienced by different people with varying severity even when they are on the same drug.

We focus on analyzing these user reported symptoms to understand the role of different treatments, and characteristics of the user in triggering them. Our preliminary investigation on a real-world dataset shows that among the reported symptoms, there exists a significant percentage of *unsubstantiated* side effects (not associated with the drug as per expert medical knowledge). Many of these symptoms are, in fact, more correlated to the underlying medical condition(s) of the user than the treatments.

With more and more people seeking health-related information online [1], it is important that these sources provide accurate information, tailored according to individual user's condition, to prevent unnecessary anxiety [10], [11]. This will help in reducing the number of users who might be reluctant to take a drug due to the long list of reported side effects associated with it, even though many of those are not applicable to her. This motivates us to develop a framework to better characterize the complex relationship between user–condition–treatment and personalize the prediction of possible symptoms and their severity for a specific user. Such a system would allow the patient to make an informed decision when choosing between alternate treatments, by weighing in the impact of potential side effects on the expected quality of life.

We formulate a multi-objective learner to predict both the set of symptoms and the severity rating that a user reports while being administered with a set of treatments. We design a novel deep neural network architecture called **M**ulti **o**bjective **M**ixture of **Ex**perts (MoMEx) to encode the complex relationship between user–condition–treatment combination and the target variable of symptoms. MoMEx uses a gating network inspired from the mixture of experts model [12], [13]. It probabilistically combines the predictions from three *local expert* networks that are built to predict symptoms based on the user, the set of medical conditions, and the combination of treatments. The gating network has an added advantage in that we are able to use the probabilities assigned to each of the local experts, which indicates the basis on which the model predicts a certain symptom. This transparency of the predictive

Treatments	User	Severity Rating	Reported Symptoms
Clonazepam	u1	3	decreased appetite, paralyzing anxiety
	u2	1	diarrhea
	u3	4	dizziness, nausea, vomiting, fatigue, tiredness
Levothyroxine	u4	4	nausea, dizziness, dissociation
	u5	3	weight gain
	u6	3	weight gain, hair loss, quivering, insomnia

TABLE I: Sample symptom reporting by different patients for two treatments in Flaredown app.

framework is crucial for a user to make a better health choice decision with confidence.

The key contributions of this work are as follows:

- Systematically investigating the nature of self reported symptoms in an online health tracking app and their correlation with the user and her pre-existing medical condition(s) apart from the treatment(s);
- Designing a multi-objective neural architecture, called MoMEx, for predicting symptoms and their severity score, based on the interaction between user, treatments, and conditions;
- Conducting extensive evaluation of MoMEx on a real-world dataset, to demonstrate its effectiveness compared to state-of-the-art baselines and architectural variants;

The remainder of the paper is organized as follows. We start with conducting an initial analysis on a real world dataset and formally defining our problem statement in Section II. In Section III, we proceed to describe the technical details of our proposed MoMEx framework. Section IV presents the effectiveness of MoMEx in comparison to state-of-the-art baselines. We discuss relevant research works in Section V before concluding in Section VI.

II. Preliminaries

We first describe the dataset, highlighting different signals available, and present an initial analysis to illustrate the challenges and motivate our approach. We use a public dataset available on Kaggle[1] from the Flaredown (FD) app[2]. The app users can 'check-in' each day to record their treatment(s), and the experienced symptoms, along with their severity scores (in the range of 0 to 4). Note that this also includes 'check-in' from users who did not experience any side effects for their treatment(s) and hence their list of side effects is nil and the severity score is 0.

The conditions, treatments and symptoms are pre-defined medical terms in the app, which users need to select from a drop-down list. Treatments are not necessarily prescribed drugs, but could also be alternative medicine or supplements, vitamins, physiotherapy, exercise and so on. For the severity rating, although the app allows users to report severity for each symptom, we assume the maximum reported severity in a 'check-in' to be the representative for the reported set of symptoms. We believe this assumption is reasonable since typically users report many (10 on an average) symptoms in a 'check-in', and might not meticulously note down the severity of each one of them. We filter out those symptoms and treatments which have been mentioned less than 5 times in the whole dataset. We collect the set of medical conditions mentioned by a user across all her 'check-ins'. Statistics of the dataset are shown in Table II.

Number of treatments	1693
Number of users	3461
Number of unique conditions	1895
Number of unique symptoms	2521
Number of evaluations ('check-in')	14,879

TABLE II: Statistics of the dataset.

A. Preliminary Study

To understand the nature of user reported symptoms, we first carry out an initial study to answer a few questions.

Q1. Can all user reported symptoms be substantiated by authoritative medical source as treatment side effects?

We compare the reported symptoms in the FD dataset with those published on Mayo Clinic portal[3], which contains curated expert information about drugs and their side effects categorized into *common, less common*, and *rare*. For each treatment in the FD dataset, we obtain the set of all its symptoms reported in a 'check-in' across all users. Since a 'check-in' might mention multiple treatments, we associate a symptom to all the treatments mentioned in a 'check-in'. This ensures that even if the symptom occurred solely because of a single treatment, it is still considered as substantiated. We match treatment name to a drug-family in the Mayo Clinic portal and consider the listed side effects as the ground truth.

Table III shows that only 33.29% of reported symptoms are known *common* side effects of a drug, while 18.76% and 7.60% are *less common* and *rare* side effects, respectively. This indicates that comparatively lesser known side effects of a drug are indeed reported by users and their discovery can help augment the existing medical knowledge base. However, we also note that an alarming 40.35% of reported symptoms do not match with any known side effects of any of the administered drug. This motivates us to further analyze the reported symptoms for potential confounding factors.

[1] https://www.kaggle.com/flaredown/flaredown-autoimmune-symptom-tracker

[2] http://flaredown.com/

[3] mayoclinic.org/drugs-supplements/

Category	Percentage
Common	33.29%
Less Common	18.76%
Rare	7.60%
Unsubstantiated	40.35%

TABLE III: Percentage breakdown of reported symptoms in the different categories of side effects for a drug.

Q2. Do the pre-existing conditions of patients have any correlation to the symptoms they report across drugs?

We analyze whether pre-existing conditions of a user influence the symptoms she experiences. For e.g., a patient suffering from *insomnia* may experience *fatigue* or *drowsiness*, and report them as side effects of her current treatment.

For each reported symptom in the FD dataset, we examine its association with various treatments and medical conditions. We define three sets of users:

- U_s : Set of users who have reported the symptom s
- U_c : Set of users who suffer from condition c
- U_t : Set of users who have taken treatment t

For each symptom s, we quantify its association with condition c and treatment t, using Jaccard similarity coefficient

$$J(s,c) = \frac{intersection(U_s, U_c)}{union(U_s, U_c)}; J(s,t) = \frac{intersection(U_s, U_t)}{union(U_s, U_t)}$$

We consider a symptom s is more correlated with a condition than a treatment, if there exists a condition c, for which $J(s,c) > J(s,t)$ for all $t \in T$, where T is the total number of treatments in the dataset. We find that around $\mathbf{48.15}\%$ of symptoms are more correlated with a condition than with a treatment, indicating that the pre-existing conditions of a user are linked to the symptoms reported.

B. Problem Formulation

Our preliminary study shows that the reported symptoms are not solely from the treatments but could be correlated with some underlying medical conditions. This motivates us to propose an approach towards predicting the symptoms that a patient might report while undergoing a set of treatments.

We formulate the problem as a multi-objective prediction task. For a user u and a set of treatments τ, we want to make two predictions:

- **Severity of Symptoms:** a numerical rating $r_{u\tau}$, real-valued number in the range $[0, 4]$.
- **List of Symptoms:** a sparse S dimensional binary vector $\mathbf{s}_{u\tau}$, indicating the outcome symptoms where S is the total number of unique symptoms.

III. Proposed MoMEx Framework

We design a neural network architecture, called MoMEx (Multi-objective Mixture of Experts), for predicting user reported symptoms along with their severity rating. The input signals to MoMEx are user, a set of treatments and her pre-existing medical conditions represented as sparse binary one-hot vectors. The architecture is depicted in Figure 1.

We use three separate embeddings to map these inputs to a lower dimensional vectors of dimension k. Let $\mathbf{x_u}$, $\mathbf{y_t}$, $\mathbf{z_c}$ denote the latent feature vectors of user u, treatment t, and condition c respectively. Consider a user u, associated with a set of conditions ϕ, has evaluated a set of treatments τ in a 'check-in'. In order to encode these sets, we employ Deep Averaging Network (DAN) [14], which has proven to be an effective modeling technique for un-ordered sequences. This enables us to capture the dependencies between co-existing conditions (and simultaneous treatments).

We first embed each treatment $t \in \tau$ using treatment embedding to receive a collection of latent vectors $\{\mathbf{y_t}\}$. Then we take an average of the latent vectors of all the treatments in the treatment set (τ) to encode their combination. Thereafter, this vector is passed through multiple feed-forward layers to capture more abstract representations of the concurrent treatments. The output of the last feed-forward layer gives us a k dimensional vector $\mathbf{q}_\tau$, denoting a latent representation of the combination of treatments. We similarly encode the set of conditions to a k dimensional vector, $\mathbf{v}_\phi$ denoting the set of pre-existing conditions of user u.

Given the latent representations of user, treatments and conditions, we next describe the prediction tasks.

A. Predicting Severity of Symptoms

Given the user and treatments embeddings, we learn to predict the severity rating $r_{u\tau}$ by a user u for a set of treatments τ. In order to incorporate the characteristics of both user and treatments, we combine the corresponding latent features by concatenating their embedding vectors $\mathbf{x_u}$ and $\mathbf{q}_\tau$. However, a simple concatenation is unable to capture the complex structure implied in the users' historical interactions. We overcome this by adding multiple fully connected layers on the concatenated vector, introducing flexibility and non-linearity in the model. The output of the last hidden layer L is transformed to a real valued rating $\hat{r_{u\tau}}$.

$$\hat{r_{u\tau}} = f(\mathbf{W_L h_{L-1}} + b_{L-1}) \tag{1}$$

where $\mathbf{W}$ and $\mathbf{b}$ are the weight matrix and bias vectors and f is an activation function, for which we use $tanh$. We obtained comparable results with $Relu$ and slightly worse results for $sigmoid$, as activation functions in our experiments.

We formulate this prediction task as a regression problem and the loss function is constructed as:

$$\mathcal{L}^r = \sum_{(u,\tau)\in\mathcal{X}} (r_{u\tau} - \hat{r_{u\tau}})^2 \tag{2}$$

where $\mathcal{X}$ represents the training set, $r_{u\tau}$ represents the ground truth rating and $\hat{r_{u\tau}}$ represents the predicted severity rating for treatment set τ by user u.

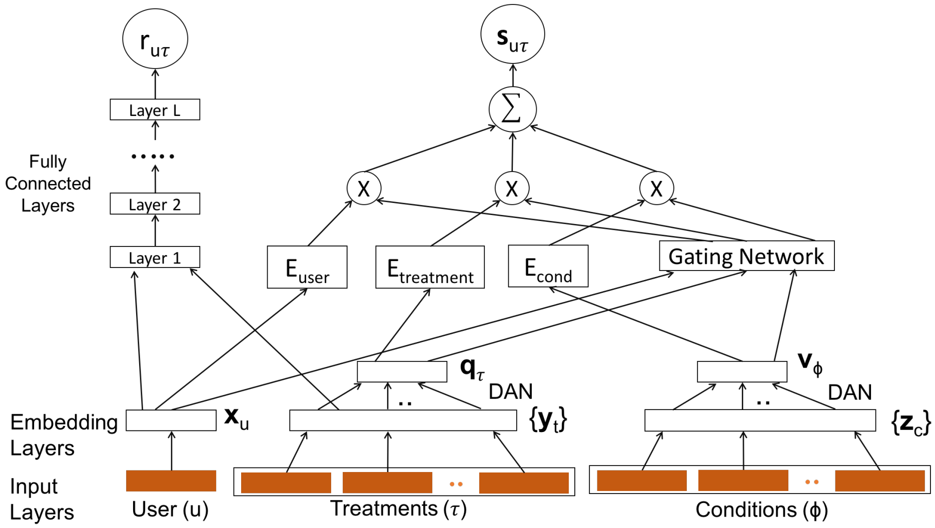


Fig. 1: Proposed neural network architecture for severity rating and symptom prediction.

B. Predicting List of Symptoms

Now, we describe our approach for predicting the list of symptoms $\mathbf{s_{u\tau}}$, reported by user u for treatments τ. This is a sparse binary vector, where the m^{th} entry indicates whether the m^{th} symptom has been reported. We consider this as multiple individual binary classifications, which had been shown as an effective technique in the past [15], where the correlation among labels is exploited by the latent space in the model.

From our initial analysis (recall Section II-A) we realize that the reported symptoms $\mathbf{s_{u\tau}}$, could be due to the treatments τ, or caused by the pre-existing conditions ϕ of the user u. Hence, we learn a model that predicts $\mathbf{s_{u\tau}}$, given the embeddings of user, treatment set and a user's conditions that is $\mathbf{x_u}$, $\mathbf{q}_\tau$ and $\mathbf{v}_\phi$ respectively.

A plausible approach could be concatenating all the three vectors and using a multi-layer perceptron to get a binary prediction for each symptom. However, in such a network it will be difficult to rationalize why a certain symptom was predicted - whether it was because of the treatments or condition of the user or a complicated non-linear combination of them.

Inspired by Mixture of Experts models [12], [13], [16], we develop three simpler local experts namely, $E_{treatment}$, E_{user}, and E_{cond}, taking as input the treatment feature ($\mathbf{q}_\tau$), user feature ($\mathbf{x_u}$), and condition feature ($\mathbf{v}_\phi$) respectively. The predictions from the local experts $E_{treatment}$, E_{user} and E_{cond}, are denoted as $\mathbf{\hat{s}^{treatment}_{u\tau}}$, $\mathbf{\hat{s}^{user}_{u\tau}}$, and $\mathbf{\hat{s}^{cond}_{u\tau}}$ respectively. The m^{th} entry of the vector $\mathbf{\hat{s}^{treatment}_{u\tau}}$ denotes the probability of occurrence of the m^{th} symptom according to the treatment expert classifier.

Finally, we need to combine the predictions from these individual experts to output a single prediction $\mathbf{\hat{s}_{u\tau}}$. One way of doing that could be just averaging their predictions, but that does not make sense for this problem. When we average the output of multiple classifiers and try to match it to a target value, we force each of the classifier to compensate for the combined error made by the other classifiers. However, in our scenario, there are certain symptoms that can be explained by only a single expert classifier (e.g., treatment) and we can ignore the results of the other classifiers for that case. This motivates us to develop a gating function, where for each input an expert is selected with some probability. The final prediction is a weighted average of the local expert predictions, where the weights are the probabilities assigned by the gating function to the experts.

For a particular input from user u for a set of treatments τ, the gating function takes as input the concatenation of user, treatment and condition vectors ($\mathbf{x_u}, \mathbf{q}_\tau$ and $\mathbf{v}_\phi$), and outputs a probability distribution, $\mathbf{w^{treatment}_{u\tau}}$, $\mathbf{w^{user}_{u\tau}}$, and $\mathbf{w^{cond}_{u\tau}}$ for treatment, user and condition experts respectively. The final prediction is computed as

$$\hat{\mathbf{s_{u\tau}}} = \mathbf{w^{treatment}_{u\tau}} \cdot \mathbf{\hat{s}^{treatment}_{u\tau}} + \mathbf{w^{user}_{u\tau}} \cdot \mathbf{\hat{s}^{user}_{u\tau}} + \mathbf{w^{cond}_{u\tau}} \cdot \mathbf{\hat{s}^{cond}_{u\tau}} \tag{3}$$

where $\mathbf{w^{treatment}_{u\tau}}, \mathbf{w^{user}_{u\tau}}$, and $\mathbf{w^{cond}_{u\tau}}$ are vectors of dimension S, the total number of symptoms. Since they denote probability distributions for weighting the three classifiers, for a particular symptom they sum to one i.e. for symptom $s \in \{1, \cdots, S\}$, we have

$$w^{treatment}_{u\tau}(s) + w^{user}_{u\tau}(s) + w^{cond}_{u\tau}(s) = 1 \tag{4}$$

where $w^{treatment}_{u\tau}(s)$ denotes the probability of selecting the prediction of the treatment expert classifier for the s^{th} symptom, others are defined similarly. The gating network helps us in examining our predictions more closely. For a symptom, we can look at the predictions of the three classifiers and their corresponding weights to understand the likely reason for it.

We need to define the structures of our expert networks and the gating network. We choose similar structures, consisting of a stack of fully connected layers, for the three expert classifiers but with different parameters. The gating network multiplies its input with a trainable weight matrix and applies a $sigmoid$ non-linearity to convert it to a vector of dimension S. This transforms the input from latent feature space to the symptom dimension. By multiplying this vector with a second trainable weight matrix, we transform the value in each symptom dimension, to a 3-dimensional vector representing the weights for each of the three experts. With $softmax$ activation on these vectors, its elements are converted to values in the range [0, 1] that add up to 1. We train the gating network by back-propagation, along with the rest of the model. Gradients are also back-propagated through the gating network to its inputs. Following the Mixture of Experts paradigm, we define this loss function as

$$\mathcal{L}^s = \sum_{(u,\tau)\in\mathcal{X}} \Big(\mathbf{w}_{\mathbf{u}\tau}^{\mathbf{user}} \cdot \mathrm{BCE}(\mathbf{s}_{\mathbf{u}\tau}, \hat{\mathbf{s}}_{\mathbf{u}\tau}^{\mathbf{user}}) + \mathbf{w}_{\mathbf{u}\tau}^{\mathbf{treatment}} \cdot \mathrm{BCE}(\mathbf{s}_{\mathbf{u}\tau}, \hat{\mathbf{s}}_{\mathbf{u}\tau}^{\mathbf{treatment}}) + \mathbf{w}_{\mathbf{u}\tau}^{\mathbf{cond}} \cdot \mathrm{BCE}(\mathbf{s}_{\mathbf{u}\tau}, \hat{\mathbf{s}}_{\mathbf{u}\tau}^{\mathbf{cond}}) \Big) \quad (5)$$

where $\mathcal{X}$ represents the training set, $\mathbf{s}_{\mathbf{u}\tau}$ represents the ground truth symptom vector of treatments τ by user u and BCE is the binary cross-entropy loss. A loss function like this will encourage specialization, since we are comparing the prediction of each expert separately with the target and then training to reduce the weighted average of all these discrepancies, where the weights are the probabilities of selecting the experts through the gating network.

C. Multi-Objective Learning

We integrate both the prediction tasks into a unified multi-objective framework with a weighted summation of the losses of its components

$$\mathcal{L} = \sum_{(u,\tau)\in\mathcal{X}} \lambda_r \mathcal{L}^r + \lambda_s \mathcal{L}^s \quad (6)$$

where $\mathcal{L}^r$ and $\mathcal{L}^s$ are the losses for severity prediction and symptoms prediction respectively and λ_r and λ_s are the weights. In our experiments, we set them to be equal but one could vary them depending on which task is more important. The whole network is trained using back-propagation in an end-to-end paradigm.

IV. Evaluation

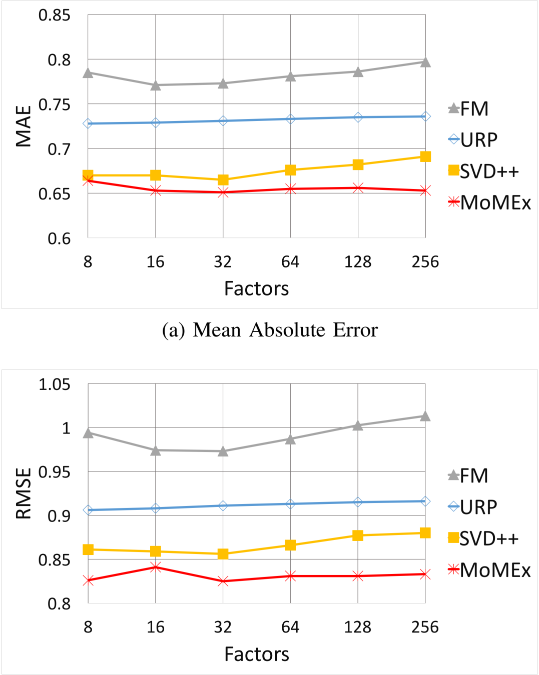
We carry out our experiments to evaluate the effectiveness of the proposed MoMEx framework.

We divide the FlareDown dataset into training (80%), validation (10%) and test (10%) sets using five fold cross validation. The hyper-parameters are tuned via grid search on the validation set. The embedding dimensions are 64 unless otherwise mentioned. The number of fully connected layers, in the DAN for encoding the treatments and conditions is 2, in the rating predictor component is 3 for encoding the user-treatment interaction, in the local expert models and gating network are 3 and 2 respectively consisting of 500 neurons. The network is optimized using Adam [17] optimizer and is implemented using Keras[4]. The learning rate is set to 0.001.

A. Prediction of Severity

We first evaluate our model on severity rating prediction using the most commonly used metrics, Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). We compare MoMEx with a number of state-of-the-art rating prediction models, namely, URP [18] , SVD++ [19] and FM [20]. URP is a generative collaborative filtering model for rating prediction and learns a rating profile for modeling a user's ratings. It has been shown to outperform other mixture based generative models for the task of rating prediction. SVD++ is one of the most competitive rating prediction algorithms that merges two powerful concepts in collaborating filtering, namely, latent factorization and neighborhood approach. Factorization Machines (FM) combine the advantages of Support Vector Machines with that of latent factorization models for general prediction tasks, including rating prediction.

Note that, in a traditional recommendation setting, a rating is available for a user-item pair. However, in our scenario, the severity rating is not always associated with a single treatment but with a set of treatments that a user has mentioned during the 'check-in'. Therefore, for URP and SVD++, we consider each unique treatment-set to be an item. FM can consider any number of real valued features for making the prediction, therefore its input is similar to MoMEx.



(a) Mean Absolute Error

(b) Root Mean Square Error

Fig. 2: Performance comparison for rating prediction with varying number of latent factors.

[4] http://keras.io/

The number of latent factor is important in determining a model's capability. We vary this in the range {8, 16, 32, 64, 128, 256} and compute accuracy for competing models. For MoMEx, we vary the dimensions of latent user, treatment vectors as they are similar in spirit with the latent factors of a CF model for predictive capability [21].

Figure 2 shows that MoMEx consistently achieves the best performance. It outperforms the second best method SVD++ with a 4.07% and 3.09% improvement on an average, in terms of MAE and RMSE respectively. Furthermore, it is more robust to variations in number of latent factors, as SVD++ starts over-fitting with higher number of factors.

B. Prediction of Symptoms

The prediction of the list of symptoms reported by a user is a challenging task, as the class distribution is highly skewed. In each 'check-in', only a few symptoms are reported among a huge list of symptoms. We use the standard precision, recall, and F1-score of the positive class (i.e. of the reported symptoms) as evaluation metrics.

To first understand the contribution of each of the input signals, we perform an ablation study with our MoMEx model. Table IV shows the results. Unsurprisingly, MoMEx achieves the best F1-score when it takes into consideration all the three input signals, instead of taking a subset of them. This proves the necessity of modeling all the three contributing factors in symptoms reporting.

Input Signals for MoMEx	Precision	Recall	F1-Score
user + treatment	0.874	0.739	0.801
treatment + condition	0.836	0.728	0.778
user + condition	0.880	0.764	0.818
user + treatment + condition	**0.901**	**0.794**	**0.843**

TABLE IV: Performance of ablation study using different subset of input signals in MoMEx.

We next compare MoMEx with the following baselines using other neural architectural variants:

- **Multi-Objective Multi Layer Perceptron (MoMLP) :** We replace the mixture of experts network with Multi Layer Perceptron. We concatenate the user-, treatment-, condition-latent vectors and use MLP layer to predict the list of symptoms. We experimented with $1-3$ number of fully connected layers for the MLP, and reported the best results.
- **Single Objective Mixture of Experts (SoMEx) :** We predict only the symptoms using a single loss function

Method	Precision	Recall	F1-Score
MoMLP	0.854	0.753	0.801
SoMEx	0.879	0.779	0.826
MoMEx	**0.901**	**0.794**	**0.843**

TABLE V: Comparison among baseline neural architectures. All competitive models use all three input signals.

Method	Precision	Recall	F1-Score
XGBoost	0.291	0.732	0.415
K Nearest Neighbor	0.821	0.580	0.679
Random Forest	0.815	0.660	0.729
MoMEx	**0.901**	**0.793**	**0.843**

TABLE VI: Comparison with state-of-the-art traditional ML classifiers. All competitive models use all three input signals.

Table V shows that using the Mixture of Experts gives superior performance compared to Multi-Layer Perceptron. Furthermore, using a single objective loss function results in a slightly worse performance compared to MoMEx. This indicates that the joint modeling of both the severity rating and symptoms using multi-objective learning benefits the symptom prediction task, as both of them essentially constitute a single 'check-in' by a user and are therefore connected. When a user gives a severity rating of 0, we learn that the symptom experienced by this user is likely to be nil. On the other hand, when a user gives a high severity rating, the list of symptoms to be predicted is likely to be long.

Finally, we compare MoMEx with few of the most popular and high-performing traditional machine learning based classifiers, namely, Gradient Boosting Machine, K Nearest Neighbour, and Random Forest classifiers. We use the implementations in scikit-learn python package [5] and XGBoost library [6]. Table VI shows that MoMEx clearly outperforms these methods. XGBoost achieves a comparable recall but at a very low precision, whereas K Nearest Neighbour and Random Forest suffer in recall due to the highly skewed distribution. MoMEx is able to exploit the correlation between symptoms using the weights of the shared hidden layers and hence can achieve the best scores.

C. Case Study

A major advantage of a mixture of experts framework is that the gating network outputs a probability distribution over the local experts, E_{user}, $E_{treatment}$, and E_{cond} built using user, treatment, and condition, respectively. This distribution provides insight to the predicted symptoms.

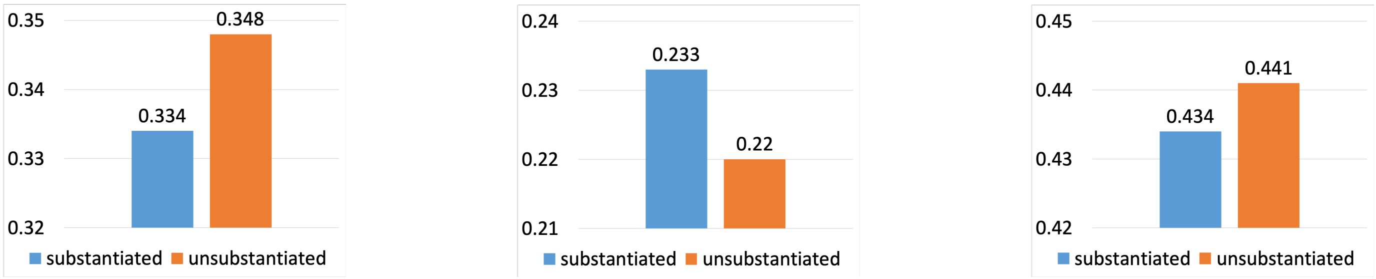
As noted in our preliminary study of the dataset (in Section II-A), while many of the reported symptoms are substantiated side effects of one of the treatments, a significant percentage of them are not substantiated. We first characterize the difference between probability distributions of substantiated versus unsubstantiated side effects (recall section II-A). Figure 3 shows the average probability with which the predictions of the local expert models are weighted to generate the final prediction for these two types of symptoms.

Firstly, Figure 3 shows that the probabilities assigned to E_{cond} (Figure 3c) are higher for both types of side effects, compared to $E_{treatment}$ (Figure 3b). This is consistent with our initial analysis that many of the side effects are correlated to users' medical conditions rather than to the treatments.

From the weights assigned to $E_{treatment}$ (see Figure 3b), we observe that a higher probability is assigned in case of

[5] http://scikit-learn.org/stable/index.html

[6] https://github.com/dmlc/xgboost



(a) E_{user}

(b) $E_{treatment}$

(c) E_{cond}

Fig. 3: Average Probabilities assigned to E_{user}, $E_{treatment}$ and E_{cond} for substantiated vs. unsubstantiated side effects

User	Conditions	Treatments	Predicted Symptoms	Local Experts Probability		
				E_{user}	$E_{treatment}$	E_{cond}
u1	Ehlers-Danlos syndrome, POTS	Bupropin	Out of breath	0.281	**0.373**	0.346
u2	Chronic fatigue syndrome, Crohn's disease, Hashimoto's disease	7-keto-dhea, Prednisolone, Vitamin D	Anxiety	0.285	**0.377**	0.338
u3	Anxiety, Depression, Eating disorders, Migraine	Prozac	Nausea	0.217	**0.572**	0.211
			Skin problems	0.356	0.180	**0.461**
			Pain	0.358	0.183	**0.459**
u4	ADHD, Acne, Depression, Insomnia	Adderall, Running	*Fatigue*	0.355	0.183	**0.462**

TABLE VII: Sample check-ins done by different users, the weights of each local expert networks assigned by the gating network. Symptoms in *italics* are deemed unsubstantiated side effects by expert medical knowledge base. Local experts with maximum probability (in bold) are the likely cause for the predicted symptoms.

substantiated side effects vs. unsubstantiated ones. This is intuitive, since the side effects that are known to be associated with a treatment, will be reported by many users of the treatment. In contrast, unsubstantiated side effects will rarely be reported by many users, resulting in $E_{treatment}$ being unable to model it, and it will be assigned a lower weight by the gating network. Interestingly, the opposite phenomenon is observed for E_{user} and E_{cond} (see Figure 3a and 3c). This indicates that users report some symptoms that are not associated with the administered treatments, and are more reliably predicted by user features (E_{user}) or her pre-existing medical conditions (E_{cond}).

Table VII shows a few case studies of the symptoms correctly predicted by MoMEx and the corresponding probabilities of the local experts. We observe that most of the symptoms that are substantiated side effects of one of the treatments correspond to $E_{treatment}$, indicating that the symptoms are due to the treatment. In contrast, the unsubstantiated side effects (in italics) correspond to E_{cond}, suggesting that they are likely to be symptoms of users' pre-existing conditions.

For user $u1$, MoMEx predicted the symptom 'Out of Breath' and assigned the highest weight to $E_{treatment}$. This matches with $u1$'s reported symptom after taking Bupropion in his check-in. Similarly, for user $u2$, MoMEx predicted the symptom 'anxiety' with $E_{treatment}$ having the highest weight. Again, this prediction matches the $u2$'s check-in and indeed, anxiety is a known side effect of Prednisolone. In contrast, user $u4$ suffers from insomnia and has reported experiencing *'fatigue'*. MoMEx is able to correctly predict this symptom and attribute it to the condition 'Insomnia'.

These demonstrate that analyzing the probability distributions of local experts generated from large scale user data is useful in interpreting reported symptoms, and could be of interest to both the web mining and medical communities.

V. Related Work

Pharmaceutical companies often carry out laboratory clinical trials and post-market surveillance to discover side effects of drugs. However they are either limited in number or incur significant time delays to gather enough information [22], [23]. Existing research has focused on augmenting medical knowledge base by detecting ADE mentions from post texts as a supervised [4], [24], [25], [3] or semi-supervised [5], [26] binary classification task. However, they do not consider other possible confounding factors such as user characteristics or underlying medical conditions.

Our work focuses on personalizing the predictions of symptoms for different users and is closer to recommendation systems, where we try to predict the experience (symptoms and their severity) of a user (patient) to an item set (treatments). Collaborative Filtering (CF) based approaches have been widely used for recommendation systems in the past decade [27], [19], [28], [29]. Recently, few neural network based architectures [30], [21] have been proposed to model the non-linear interaction between user-item features in a CF framework. However these models focus on implicit user feedback for item recommendation instead of rating prediction. For explicit rating prediction, [31] has proposed a neural network that uses not only the user-item information but also the review text which are not always available. Instead our proposed model only focuses on user-treatment interactions to predict the rating. Another class of recommendation algorithms such as Tensor Factorization, Factorization Machines [32], [33] incorporate contextual features along with the usual user-item interactions. However, they are primarily designed for cases where the context varies for every interaction [34], whereas medical conditions are dependent on the user and remain constant across all check-ins.

Mixture of experts approach [12], [13] have focused on different expert configurations [16], [35], for an ensembling approach [36] or for enabling conditional computation in a very large network [37]. The mixture of experts architecture used in our model is most similar to an ensembling approach [36], where we combine the outputs of each expert using the probabilities from the gating layer. Since each of our expert is designed to represent one of the three factors causing a symptom. Unlike the usual architecture, inputs to our experts are specific to only a single factor, giving the experts a semantic meaning for their specialization. This has the added benefit of providing insights to the model's decisions and opens up avenues for further exploratory analysis.

VI. Conclusion

We have systematically investigated the characteristics of user reported symptoms in an online platform. We find that users report diverse symptoms, while undergoing the same treatments and a significant percentage of the symptoms could not be substantiated as side effects of the treatment. Further investigation revealed that the reported symptoms are often more correlated with the pre-existing medical conditions of the users than with the treatments. We have proposed a novel neural architecture to predict personalized user responses for different treatments, in terms of symptoms and severity rating. Experimental evaluation on a real-world dataset shows that our approach is able to outperform state-of-the-art models for both tasks. Although we specialize our model for the use of symptom prediction in this paper, we believe our model is general in nature and could be applicable to other scenarios involving users, items and multiple interaction targets.

References

[1] S. Fox and M. Duggan, "Health online 2013," *Washington, DC: Pew Internet & American Life Project*, 2013.

[2] I. I. for Healthcare Informatics, "Engaging patients through social media," *Report*, 2014. [Online]. Available: http://www.theimsinstitute.org/

[3] V. Plachouras, J. L. Leidner, and A. G. Garrow, "Quantifying self-reported adverse drug events on twitter: signal and topic analysis," in *Proc. of the International Conference on Social Media & Society*, 2016.

[4] Z. Zhang, J. Nie, and X. Zhang, "An ensemble method for binary classification of adverse drug reactions from social media," in *Proc. of the Social Media Mining Shared Task Workshop at the Pacific Symposium on Biocomputing*, 2016.

[5] K. Lee, A. Qadir, S. A. Hasan, V. Datla, A. Prakash, J. Liu, and O. Farri, "Adverse drug event detection in tweets with semi-supervised convolutional neural networks," in *WWW*, 2017.

[6] M. Swan, "Crowdsourced health research studies: an important emerging complement to clinical trials in the public health research ecosystem," *JMIR*, vol. 14, no. 2, 2012.

[7] R. W. White, R. Harpaz, N. H. Shah, W. DuMouchel, and E. Horvitz, "Toward enhanced pharmacovigilance using patient-generated data on the internet," *Clinical Pharmacology & Therapeutics*, vol. 96, no. 2, pp. 239–246, 2014.

[8] R. J. Cline and K. M. Haynes, "Consumer health information seeking on the internet: the state of the art," *Health education research*, 2001.

[9] G. Peterson, P. Aslani, and K. A. Williams, "How do consumers search for and appraise information on medicines on the internet? a qualitative study using focus groups," *JMIR*, vol. 5, no. 4, 2003.

[10] S. E. Baumgartner and T. Hartmann, "The role of health anxiety in online health information search," *Cyberpsychology, Behavior, and Social Networking*, vol. 14, no. 10, pp. 613–618, 2011.

[11] G. P. Schoenherr and R. W. White, "Interactions between health searchers and search engines," in *Proc. of SIGIR*. ACM, 2014.

[12] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural computation*, 1991.

[13] M. I. Jordan and R. A. Jacobs, "Hierarchical mixtures of experts and the em algorithm," *Neural computation*, 1994.

[14] M. Iyyer, V. Manjunatha, J. Boyd-Graber, and H. Daumé III, "Deep unordered composition rivals syntactic methods for text classification," in *ACL*, 2015.

[15] J. Liu, W.-C. Chang, Y. Wu, and Y. Yang, "Deep learning for extreme multi-label text classification," in *SIGIR*, 2017.

[16] B. Yao, D. Walther, D. Beck, and L. Fei-Fei, "Hierarchical mixture of classification experts uncovers interactions between brain regions," in *NIPS*, 2009.

[17] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2014.

[18] N. Barbieri, "Regularized gibbs sampling for user profiling with soft constraints," in *ASONAM*, 2011.

[19] Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in *SIGKDD*, 2008.

[20] S. Rendle, "Factorization machines," in *ICDM*, 2010.

[21] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *WWW*, 2017.

[22] H.-J. Dai, M. Touray, J. Jonnagaddala, and S. Syed-Abdul, "Feature engineering for recognizing adverse drug reactions from twitter posts," *Information*, 2016.

[23] M. Rastegar-Mojarad, R. K. Elayavilli, Y. Yu, and H. Liu, "Detecting signals in noisy data-can ensemble classifiers help identify adverse drug reaction in tweets," in *Proc. of the Social Media Mining Shared Task Workshop at the Pacific Symposium on Biocomputing*, 2016.

[24] B. Ofoghi, S. Siddiqui, and K. Verspoor, "Read-biomed-ss: Adverse drug reaction classification of microblogs using emotional and conceptual enrichment," in *Proc. of the Social Media Mining Shared Task Workshop at the Pacific Symposium on Biocomputing*, 2016.

[25] J. Jonnagaddala, T. R. Jue, and H. Dai, "Binary classification of twitter posts for adverse drug reactions," in *Proc. of the Social Media Mining Shared Task Workshop at the Pacific Symposium on Biocomputing*, 2016.

[26] S. Gupta, S. Pawar, N. Ramrakhiyani, G. K. Palshikar, and V. Varma, "Semi-supervised recurrent neural network for adverse drug reaction mention extraction," *CoRR*, 2017.

[27] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *NIPS*, 2001.

[28] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," in *NIPS*, 2008.

[29] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using markov chain monte carlo," in *ICML*, 2008.

[30] Y. Wu, C. DuBois, A. X. Zheng, and M. Ester, "Collaborative denoising auto-encoders for top-n recommender systems," in *ICWSM*, 2016.

[31] P. Li, Z. Wang, Z. Ren, L. Bing, and W. Lam, "Neural rating regression with abstractive tips generation for recommendation," 2017.

[32] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver, "Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering," in *Proc. of RecSys*. ACM, 2010.

[33] S. Rendle, "Factorization machines with libfm," *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2012.

[34] G. Adomavicius and A. Tuzhilin, "Context-aware recommender systems," in *Recommender systems handbook*. Springer, 2011.

[35] C. E. Rasmussen and Z. Ghahramani, "Infinite mixtures of gaussian process experts," in *NIPS*, 2002.

[36] E. Garmash and C. Monz, "Ensemble learning for multi-source neural machine translation." in *COLING*, 2016.

[37] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," *arXiv:1701.06538*, 2017.