Bringing Semantic Structures to User Intent Detection in Online Medical Queries

Chenwei Zhang*¶, Nan Du†, Wei Fan‡∥, Yaliang Li†, Chun-Ta Lu*, Philip S. Yu*§

*Department of Computer Science, University of Illinois at Chicago, Chicago, IL, USA

†Baidu Research Big Data Lab, Sunnyvale, CA, USA

‡Tencent Medical AI Lab, Palo Alto, CA, USA

§Institute for Data Science, Tsinghua University, Beijing, China

Email: *{czhang99,clu29,psyu}@uic.edu, ‡davidwfan@tencent.com, †{nandu, yaliangli}@baidu.com

Abstract—The Internet has revolutionized healthcare by offering medical information ubiquitously to patients via web search. The healthcare status, complex medical information needs of patients are expressed diversely and implicitly in their medical text queries. Aiming to better capture a focused picture of user's medical-related information search and shed insights on their healthcare information access strategies, it is challenging yet rewarding to detect structured user intentions from their diversely expressed medical text queries. We introduce a graph-based formulation to explore structured concept transitions for effective user intent detection in medical queries, where each node represents a medical concept mention and each directed edge indicates a medical concept transition. A deep model based on multi-task learning is introduced to extract structured semantic transitions from user queries, where the model extracts word-level medical concept mentions as well as sentence-level concept transitions collectively. A customized graph-based mutual transfer loss function is designed to impose explicit constraints and further exploit the contribution of mentioning a medical concept word to the implication of a semantic transition. We observe an 8% relative improvement in AUC and 23% relative reduction in coverage error by comparing the proposed model with the best baseline model for the concept transition inference task on real-world medical text queries.

Index Terms—Information Search; Intent Detection; Concept Transition; Neural Network

1. Introduction

The shortages of healthcare professionals are leading to healthcare systems plagued by bottlenecks. According to the World Health Organization, the world will face a shortfall of nearly 13 million healthcare professionals by 2035 [5]. In the meanwhile, an increasing number of medical-related online services emerge on the world wide web to offer ubiquitous medical information to patients via their web

---

¶ *Part of the work was done when the author was an intern at Baidu Research Big Data Lab.*
∥ *Part of the work was done when the author was employed by Baidu Research Big Data Lab.*

search [12]. For example, the Chinese search engine Baidu processes over 6 billion search queries every day, while 60 million of them are healthcare-related text queries[1]. Online medical question answering forums such as xywy.com[2] has 120 million registered users and more than 22 million unique daily visitors.

With the flourishing demand for medical-related services, it is crucial for service providers to infer implicit user intentions from the diversely expressed medical text question: what medical concepts a user mentions and how concept transitions are formulated among these concepts. Generally, medical text queries that users search online or post on medical question-answering websites express various medical-related conditions and indicate different information needs, as shown in Table 1.

○ Medical Text Questions
● Inferred Concept Mentions & Concept Transitions

○ Why do I get dizzy so often?
● *Symptom* → *Cause*

○ My three-year-old child is sick with a temperature of 100 degrees she can't keep anything down including liquids. What kind of medicine should I give my child, and how much?
● Symptom → *Medicine* → *Instruction*

○ Do I have insomnia if I have trouble staying asleep? Any medication is recommended to help me fall asleep easier?
● *Disease* ← *Symptom* → *Medicine*

TABLE 1. MEDICAL QUERIES AND THE EXTRACTED MEDICAL CONCEPT MENTIONS &TRANSITIONS.

Usually, medical semantic transitions are formulated by users during their efforts to express their existing medical conditions as well as their intended medical-related information needs, either explicitly or implicitly. The diversely expressions cover the mention of different types of medical concepts, each represents a set of notions such as symptoms, diseases, medicines etc. In real-world medical text queries, various expressions can be referred to as a concept mention, either explicitly (e.g. "Tylenol", "Ibuprofen" for the *medicine* concept) or implicitly (such as "what", "which drug/medicine"). Even for the same medical concept, differ-

---

ent expressions can be used. For example, "nose plugged", "blocked nose" and "sinus congestion" all belong to the same symptom concept mention but expressed very differently.

The way concept mentions organized in a question naturally forms a structured concept transition graph that reflects users information-seeking intentions. Such ubiquitous observation in medical text questions is rarely studied in previous literatures. A typical formulation for medical intent detection is to model each semantic transition as a single label [43], or as a two-element tuple indicating 1) what a user have described and 2) what information the user is looking for [40]. For example, we can have $(Symptom, Disease)$ and $(Symptom, Medicine)$ for the last query in Table 1. This formulation defines each tuple as an individual label and ignores the correlations among the multiple semantic transitions in a single query. In real-world medical text queries, multiple semantic transitions in a single question may conjugate with each other by mentioning the same medical concept. For example, $(Symptom, Disease)$ and $(Symptom, Medicine)$ share the same concept $Symptom$ by expressing symptoms: "trouble staying asleep" and "fall asleep" in the query. The above formulations fail to consider the semantic interactions among multiple medical concepts in a medical query, which prevent them from satisfactorily detect sophisticated user intentions with complex semantic structures.

Alternatively, we can formulate concept transitions over a directed, highly structured concept graph where concept mentions are nodes and transitions between concepts are directed edges between them. For example, with a graph-based formulation, the concept transition for the second question in Table 1 is formulated as $Symptom \rightarrow Medicine \rightarrow Instruction$ since the user first describes his/her symptoms ("sick", "temperature of 100 degrees") and inquires about information on medicine concepts ("What kind of medicine"), followed by phrases ("and how much") indicating further information seek intentions about instructions on the medicine. Real-world text questions often exhibit a mixture of multiple concept transitions in each question (See Section 2.3 for details), in which shared concept mentions serve as a bridge coupling two or more concept mentions into a structured concept transition. Thus, a graph-based formulation would essentially allow us to jointly model and infer correlations between concept mentions and multiple concept transitions simultaneously, which is one of our key contributions.

Problem Studied: In order to better capture a focused picture of people's medical-related information search and information access strategies, we propose and study the concept transition inference problem for online healthcare questions with a graph-based formulation. Given a question and a concept graph indicating the full spectrum of concept transitions in the medical domain, our goal is to effectively infer concept transitions that are activated by the given medical text question, as shown in Figure 1.

Challenges: A typical solution for the concept transition inference problem evolves hand-engineering features based on expert knowledge in the medical domain, such as constructing a word-concept mapping dictionary [40] or using pre-defined rules [11] or templates [33] for question intent classification . Even if one discounts the tedious effort required for feature engineering, those features are usually designed for a limited number of questions accessible to domain experts and do not generalize to handle various user expressions in real-world medical text questions. People with different knowledge background tends to express the same idea in different ways. For example, a medicine concept can be mentioned by specific drug names such as "Tylenol", "Ibuprofen" or phrases like "what kind of medicine/drug/medication". The decent performance of those approaches usually comes at the cost of acquiring an external knowledge base to handle varying linguistic modalities and diversified expressions. How to minimize feature engineering without compromising the performance for the concept transition inference is still challenging.

Moreover, comparing with general-purpose text questions which people have been posting or searching for online, where users only focus on a single concept (such as "weather", "politics" or "stocks"), concept transitions in medical questions usually involve multiple concepts. It would take strenuous efforts to model correlations among multiple concept transitions without considering the shared concept mentions effectively. What's more, unlike many existing works on medical text analysis such as sentiment classification [15], [2] which consider positive, negative or neutral sentiments in medical texts, it is challenging yet rewarding to consider structured concept transitions which model sophisticated medical semantic transitions in real-world medical text queries.

Proposed Work: To overcome those challenges, we introduce a novel neural network architecture that bring structures to semantic transitions for user intent detection in medical text queries. We observe an appealing property that
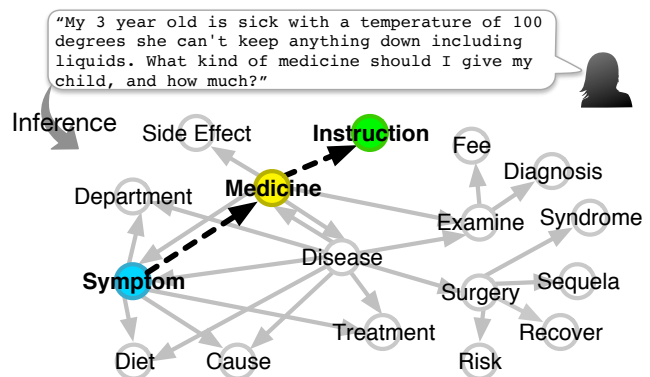

Figure 1. The concept transition inference problem over the full concept graph. Each node in the concept graph represents a concept. Each directed edge between two nodes indicates an information-seeking transition over two concepts. Given a text question and an existing medical concept transition graph, the concept transition inference problem extracts a directed, structured subgraph consisting of a set of concepts being mentioned (shown as colored nodes) as well as a set of concept transitions being encoded (shown as black dashed lines) from the question.

real-world medical text questions exhibit a strong coupling between concept mentions and concept transitions. Consequently, a graph-based formulation is defined to jointly model correlations between concept mentions and multiple concept transitions. The proposed model can effectively infer concept transitions from real-world medical text queries and extract a structured representation of users information-seeking intentions.

Also, we introduce an end-to-end solution to the concept transition inference problem which is well integrated with the graph-based concept transition formulation. The concept transition inference task is formulated with a multi-task learning schema that learns to extract concept mentions as well as to infer concept transitions collectively. A customized graph-based mutual transfer loss function is designed to impose explicit constraints to reduce the conflicts between the concepts being mentioned and the concept transitions being activated.

Furthermore, the proposed model minimizes hand-feature engineering by using only the text information from the medical text query and an existing medical concept graph introduced in [40], without relying on other external knowledge bases. The neural network is trained to automatically discover concept mentions and infer concept transitions from raw text questions, in contrast to relying on fixed dictionaries for word-concept mapping [6], [14], [40] or using pre-defined parsing rules [11] or templates [33] in prior works. The neural network also learns to assign confidence scores to words as an attention mechanism [39], which makes the model self-explaining in indicating the contribution of each medical concept mention to the structured semantic transition.

Moreover, the proposed method learns both semantic and syntax representations for each word and its Part-of-Speech tag respectively. The learned embeddings are fed into two separate recurrent neural networks to build a memory summarizing multiple concept transitions over the input text sequence. This compositionality of input embedding lends the proposed method to handle diversified expressions in user questions.

Experiments are conducted on real-world medical text queries collected from a medical question-answering forum, which is publicly available. We contrast the performance of the proposed model with other alternatives by an 8% improvement in micro-AUC and an 23% reduction in coverage loss for the concept transition inference problem.

Overall, our paper makes the following contributions:

1) We observe and formally define concept transitions in medical text questions and show appealing properties among concept transitions and shared concept mentions.
2) We study the concept transition inference problem with a graph-based formulation, which brings semantic structures to diversely expressed natural language queries.
3) We propose an end-to-end solution with a novel neural network model to the concept transition inference problem without excessive external knowledge requirements.
4) We collect datasets and empirically evaluate the proposed method on real-world medical text queries.

2. Preliminaries

We now formally define the terminologies and describe the concept transition inference problem. Also, we provide observations to show appealing coupling properties of concept mentions and concept transitions in real-world text queries, which motivates a graph-based concept transition formulation.

2.1. Terminologies

Concept Let a concept c be a group or class of objects and/or abstract ideas representing similar fundamental characteristics in a certain domain. $C = \{c_1, c_2, ..., c_M\}$ is list of a full spectrum of M concepts in a specific domain. (e.g. the medical domain contains concepts of diseases, symptoms, medicine and so on). Users can mention concepts in a query by mentioning specific object names as explicit mentions ("Tylenol", "Ibuprofen" or "xxx caplet/capsule/drop/syrup"), as well as implicit mentions by abstract ideas that refer to concept (e.g. "remedy") or phrases indicating this concept (e.g. "which medication/medicine/drug").

Concept Transition Let a concept transition $t_{i \rightarrow j}$ defines a transition of a user information search intent from the concept c_i to concept c_j. A concept transition $t_{i \rightarrow j}$ exists in a query when two concepts $c_i, c_j \in C$ are mentioned (either explicitly or implicitly) with a semantic transition between them. For example, medical queries with concept transitions $t_{Symptom \rightarrow Medicine}$ usually start with patients describing their symptoms and asking for related information about medications that help them alleviate their symptoms.

T contains the full spectrum of N concept transitions in a certain domain, which can be indexed as $T = \{t_1, t_2, ..., t_N\}$ for simplicity instead of $\{t_{i \rightarrow j}\}$. Those two index notations are used interchangeably in this paper. Multiple concept transitions can be activated by a single query and the direction of a concept transition does not necessarily follow the order of concepts being mentioned in a query. Multiple concept transitions in a query may follow a natural chain-like path, such as the path $Symptom \rightarrow Medicine \rightarrow Instruction$.

Concept Graph Let $G = \langle C, T \rangle$ be a concept graph where each node represents a concept $c_m \in C$ and $t_{i,j} \in T$ be a directed edge from node c_i to c_j. A concept graph G is a graph representation of all possible concepts and concept transitions in a certain domain. Note that the domain-specific concept graph can be obtained from domain experts, which we adopted in this paper, or constructed from large text corpora by existing techniques [18], [38], [42].

Active Concept Graph Let an active concept graph $G_Q = \langle C_Q, T_Q \rangle$ be a subgraph of $G = \langle C, T \rangle$, indicating concepts $C_Q \subseteq C$ mentioned by a query Q and concept transitions $T_Q \subseteq T$ activated by the the query Q.

2.2. Problem Statement

The Concept Transition Inference Problem: Given 1) a text query Q which consists of K elements $\{q_1, q_2, ..., q_K\}$, where each element is a word or a phrase and 2) a concept graph $G = \langle C, T \rangle$, where C denotes all possible concepts and T indicates all possible concept transitions, the concept transition inference problem tries to effectively infer an active concept graph $\hat{G}_Q = \langle \hat{C}_Q, \hat{T}_Q \rangle$ given a query Q. Figure 1 illustrates this idea.

2.3. Observations

Based on the terminologies and the problem defined, we would like to observe the existence of active concept graphs given real-world medical text queries. We sample 10,000 medical text queries from an online medical question answering forum and label them with concept transitions being activated. We end up having 17 unique types of concepts and 23 unique types of concept transitions (details in Section 4.1). Table 2 shows the top frequent concepts that

Medical Concept	Symptom	Disease	Cause	Medicine	Treatment
Frequency	7650	7446	5380	3733	2504

TABLE 2. TOP FREQUENT CONCEPTS MENTIONS.

are mentioned either explicitly or implicitly in medical text queries.

We also show 9 popular active concept graphs in medical text queries, shown in Figure 2. By characterizing concept transitions with a graph-based formulation, it maps natural language queries with diversified expressions into a structured form, which show users information needs in a structured way.

More importantly, we found that active concept graphs rarely have disconnected components, from a perspective of the graph theory. This not only implies that users tend to have multiple concept transitions within a single medical text query but also indicates that multiple concept transitions in the same query are expressed and developed together, coupled with some shared concept mentions. The connectivity of active concept graphs implies that by taking advantages of the concepts and concept transitions formulated on a concept graph, we may able to utilize the correlations between nodes and edges for a better inference and therefore, users information search intent or information access strategy for their healthcare conditions can be modeled and inferred more effectively.

3. Medical Concept Transition Inference

We introduce a novel neural network structure which provides an end-to-end solution to the concept transition
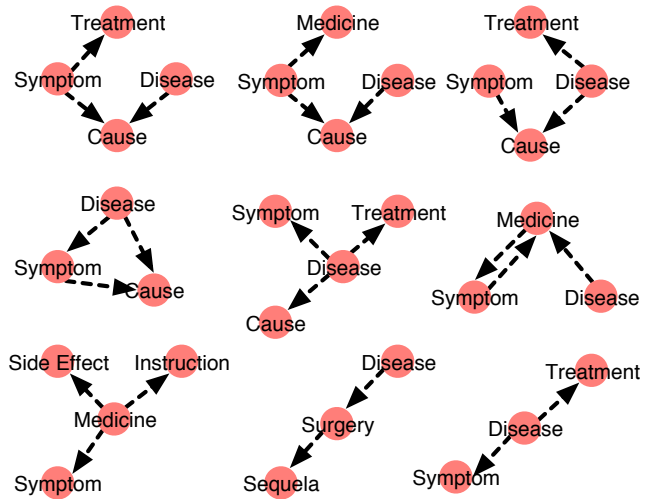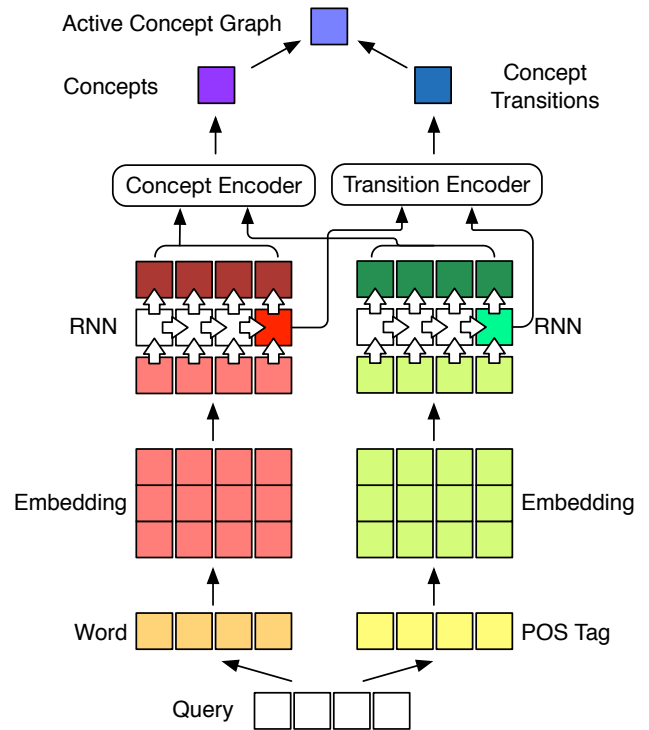

Figure 2. Popular active concept graphs.

Figure 3. The proposed neural network architecture.

inference problem where the input is a text query and the output is an active concept graph inferred by the given query. The model utilizes distributed representations for sequences of words and their POS tags respectively, from which semantic and syntax are embedded on a word-level. After that, two recurrent neural networks are adopted to model the sequential information from distributed representations of word and POS tag sequences in each query respectively. In the graph-based co-inference procedure, concepts and concept transition are inferred collectively and simultaneously.

A concept encoder is proposed to utilize the joint outputs of two RNNs to encode each element into a concept vector. Especially, the concept encoder is able to learn a confidence score which indicates the contribution of each element in encoding concept mentions in a query. While for inferring concept transitions, a transition encoder exploits the last hidden states of two RNNs to construct a transition vector, from which we infer a probability distribution on all possible concept transitions. The loss of the neural network structure not only incorporates prediction errors between the predict concept transitions and the true concept transitions but also exploit a mutual transfer loss indicating the conflicts between the inferred concepts and their corresponding concept transitions. An active concept graph is presented with the inferred concepts and concepts transitions, by collectively minimizing a graph-based mutual transfer loss based on the concept graph. Figure 3 gives an overview of the proposed neural network architecture.

3.1. Semantic-Syntax Representations

Unlike traditional methods which ignore the sequential information of the input text query and treat it as a bag-of-words (BoW), in this work a text query Q is considered as a sequence of elements $\{q_1, q_2, ..., q_K\}$, where each element q_k can be a word or a phrase. K is the length of a text query, which varies from different text queries. For each element q_k in a text query Q, we utilize both the word indicating the semantic information, as well as its corresponding Part-of-Speech (POS) tag as the syntax information.

Part-of-speech (POS) tags bring useful syntax information about general word categories (such as noun, verb, adjective, etc.), which is helpful in dealing with ambiguous words and diversified expressions. For example, "fever" can be either a noun or a verb. The word "fever" with a POS tag "noun" is defined as a disease that causes an increase in body temperature and the fever with a POS tag "verb" can be considered as someone in a fever, as a symptom. In this work, an existing POS tagger[3] is utilized to give general POS tags to each element in the query. The semantic-syntax joint representation consists of words along with POS tags are shown to be effective in modeling both semantic (words) and syntax (POS tags) from the natural language text corpus in various tasks [23], [40]. In this work, each element Q_k of a query Q is represented by words and POS tags as a tuple:

$$q_k = (w_k, p_k) \ \ s.t. \ w_k \in \mathbb{R}^{V_{word}}, p_k \in \mathbb{R}^{V_{pos}}, \quad (1)$$

where w_k is the one-hot representation of the k-th word in the query Q and V_{word} is the number of unique words, namely the vocabulary size. Similarly, p_k is the one-hot representation of the k-th word's POS tag in the query. V_{POS} is the POS vocabulary size.

3.2. Word Embedding

The one-hot representation suffers from the curse of dimensionality since the representation becomes extremely sparse as the vocabulary becomes large. The word embedding is used to transfer one-hot representation of each word w_k and POS tag p_k into a dense representation:

$$w\_embed_k \in \mathbb{R}^{D_{word}}, p\_embed_k \in \mathbb{R}^{D_{pos}}, \quad (2)$$

where V_{word} usually can be large up to millions while D_{word} is reduced to several hundreds. Note that D_{word} and D_{pos} are usually set empirically. In this work, we set $D_{word} = 100$ and $D_{pos} = 20$.

The embedded representation of each w_k and p_k are learned respectively by a linear mapping via a skip-gram model [29]:

$$embed\_w_k = E_{word} \ w_k$$
$$embed\_p_k = E_{pos} \ p_k, \quad (3)$$

where $E_{word} \in \mathbb{R}^{D_{word} \times V_{word}}$ and $E_{pos} \in \mathbb{R}^{D_{pos} \times V_{pos}}$ are weights. The skip-gram learns a distributed representation of each word or POS tag based on its context. In the medical text queries, that means an explicit mention of a concept ("Tylenol") and an implicit mention of a concept ("Which medicine") may have similar representations when they occur in similar context, when trained properly. That helps us solve the diversified expressions in medical text queries.

In this work, the embedding is initialized with word vectors pre-trained from 64 million medical text queries and updates with the model during training. After the word embedding, the k-th element in the text query q_k has a semantic-syntax representation, represented by a tuple:

$$e_k = (embed\_w_k, embed\_p_k). \quad (4)$$

3.3. Recurrent Neural Network

Once we obtained semantic-syntax representations e_k for each element q_k in a query Q, the $embed\_w_k$ sequence and the $embed\_p_k$ sequences are fed into two recurrent neural networks, namely RNN_W and RNN_P, respectively.

In general, a recurrent neural network keeps hidden states over a sequence of elements and update the hidden state h_k by the current input x_k as well as the previous hidden state h_{k-1} where $k > 1$ by a recurrent function:

$$h_k = \text{RNN}(x_k, h_{k-1}) \quad (5)$$

The simplest form of an RNN is as follows:

$$h_k = \alpha(W_{xh}x_k + W_{hh}h_{k-1} + b_h), \quad (6)$$

where $W_{xh} \in \mathbb{R}^{D_h \times D_x}, W_{hh} \in \mathbb{R}^{D_h \times D_h}, b_h \in \mathbb{R}^{D_h}$ are weights and bias that need to be learned as model parameters. $\alpha(\cdot)$ is a non-linear transformation function such as Rectified Linear Unit (ReLU): $\alpha(x) = max(0, x)$. This form of RNN fails to learn long-term dependencies due to gradient vanish or explosion problem [3], [19], which is not

suitable to learn dependencies from a long input sequence in practical.

To address the gradients decay or exploding problem over long sequences, the Gated Recurrent Unit (GRU) [8] is proposed as a variation of the Long Short-term Memory (LSTM) unit [20]. The GRU has been attracting great attentions since it overcomes the vanishing gradient in traditional RNNs and is more efficient than LSTM on certain tasks [9]. The GRU is designed to learn from previous time stamps with long time lags of unknown size between important time stamps. A typical GRU is formulated as:

$$
\begin{aligned}
r_k &= \delta(W_{xr}x_k + R_{hr}h_{k-1} + b_r) \\
z_k &= \delta(W_{xz}x_k + R_{hz}h_{k-1} + b_z) \\
\tilde{h}_k &= tanh(W_{xh}x_k + W_{hh}(r_k \otimes h_{k-1}) + b_h) \\
h_k &= z_k \otimes h_{k-1} + (1 - z_k) \otimes \tilde{h}_k,
\end{aligned}
\tag{7}
$$

where a reset gate r_k is designed to makes the GRU acts whether as if it is reading the first element of an input sequence or not, allowing it to forget the previously computed state. The GRU maintains an update gate z_k to balance between previous activation h_{k-1} and the candidate activation $\tilde{h}_k$. $\delta(\cdot)$ and $tanh(\cdot)$ are the sigmoid and tangent activation function and $\otimes$ denotes the element-wise multiplication operator. An output vector o_k is generated for each hidden state at time stamp k, by the following equation:

$$
o_k = \sigma(W_{ho}h_k)
\tag{8}
$$

, where W_{ho} is the weight and $\sigma(\cdot)$ is a softmax function. The output vector can be considered as the vector representation of each input x_k, taking the hidden state h_k maintained by the RNN into the consideration. Note that, the output vector is not affected by any gates in GRU, which makes the GRU more appealing to our problem setting since we need an output vector o_k without any output gating for each word in a query.

In this work, two separate RNN with GRU cells, namely RNN_W and RNN_P, are adopted to model the sequential information for the sequence of embedded words $embed\_w_k$ and the sequence of embedded POS tags $embed\_p_k$:

$$
\begin{aligned}
h\_w_k, o\_w_k &= \text{RNN}_W(embed\_w_k, h\_w_{k-1}) \\
h\_p_k, o\_p_k &= \text{RNN}_P(embed\_p_k, h\_p_{k-1}),
\end{aligned}
\tag{9}
$$

3.4. Graph-based Co-inference

In order to fully exploit the correlations of concept transitions and corresponding concepts, concepts and concept transitions are inferred collectively over a concept graph for each query. The concept inference is aimed to select a subset of concept $\hat{C}_Q \in C$ that are mentioned in a query Q, which is achieved by the concept encoder. To inference transitions, we also utilize a transition encoder. The concepts $\hat{C}_Q$ and transitions $\hat{T}_Q$ are inferred collectively, by minimizing a mutual transfer loss which indicates the conflicts within the collectively inferred active concept graph $\hat{G}_Q$ on a concept graph G.

3.4.1. Concept Encoder. In concept inference, a concept encoder is proposed to encode all the concept mentions from a sequence of output states of an RNN to concept vectors accordingly. Since some words in a query may contribute more to a concept mention in a query while some other words are less contributive, the concept encoder itself learns to assign a confidence score to each output state. Let o_k be the k-th output vector of an RNN, while in this work we concatenate the output vectors of RNN_W and RNN_P:

$$
o_k = [o\_w_k, o\_p_k], o\_w_k \in \mathbb{R}^{1 \times D_{o_w}}, o\_p_k \in \mathbb{R}^{1 \times D_{o_p}},
\tag{10}
$$

where D_{o_w} and D_{o_p} are the output dimension of output vectors in RNN_W and RNN_P. The concept encoder assigns a score s_k for each o_k indicating the degree of confidence based on the value of o_k:

$$
s_k = CE(o_k, \theta) \quad \text{s.t.} \sum_k s_k = 1, \forall s_k \in [0, 1],
\tag{11}
$$

where θ is the parameter of the concept encoder that we learn along with the whole model. all s_k scores in a query are normalized to sum up to one. The concept encoder CE can be also considered as a mapping from each output vector o_k to a real value $s_k \in [0, 1]$. In this work, the concept encoder is implemented as a single layer neural network with a non-linear activation function ReLU. Thus $\theta = \{W_\theta \in \mathbb{R}^{(D_{ow}+D_{op}) \times 1}, b_\theta \in \mathbb{R}\}$. Note that although weights and biases are applied on each of the o_k, they are shared among all $o_1, o_2, ..., o_k$.

Figure 4 shows the architecture of the concept encoder. The $o_{CE} \in \mathbb{R}^{(D_{ow}+D_{op}) \times K}$ is a representation of encoded
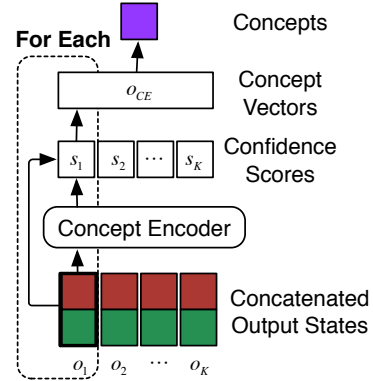

Figure 4. The concept encoder is used to determine confidence scores for each joint output state. This figure shows an example of a score s_1 learned from the concept encoder for o_1.

concepts from the query, which is calculated based on all the output states $\{o_1, o_2, ..., o_K\}$:

$$
O_{CE} = \begin{bmatrix} (CE(o_1, \theta) \cdot o_1)^T & ... & (CE(o_K, \theta) \cdot o_K)^T \end{bmatrix}
\tag{12}
$$

The probability that a concept $c_i \in C$ is activated in a query Q is defined as:

$$
(\hat{C}_Q)_m = P(c_m | c_m \in C, \theta, W_{CE}, b_{CE}) = \frac{1}{1 + e^{-W_{CE}O_{CE}+b_{CE}}},
\tag{13}
$$

where $W_{CE} \in \mathbb{R}^{1 \times (D_{ow} + D_{op})}$, $b_{CE} \in \mathbb{R}$ are weights and biases for such probability inference. We use $\hat{C}_Q \in \mathbb{R}^{1 \times M}$ to quantify the probability distribution on all M concepts for a given query Q.

3.4.2. Transition Encoder. In the field of machine translation, a novel recurrent neural network encoder-decoder has gained attention [35], where the encoder recurrent neural network encodes the global information spanning over the whole input sentence in its last hidden state. The effectiveness of the last hidden states in modeling natural language sequences are also witnessed in application like dialog systems [32]. Inspired by those ideas, we propose a transition encoder which leverages the last hidden state of the neural network for both RNN_W, RNN_P to make inferences on concept transitions, where the transition vector o_{TE} is constructed by

$$O_{TE} = [h\_w_K, h\_p_K], \quad (14)$$

where K is the length of the query. The probability that a transition $t_n \in T$ is activated given a query Q is quantified by:

$$(\hat{T}_Q)_n = P(t_n \in T | W_{TE}, b_{TE}) = \frac{1}{1 + e^{-W_{TE}O_{TE} + b_{TE}}}, \quad (15)$$

where $W_{TE} \in \mathbb{R}^{1 \times (D_{ow} + D_{op})}$, $b_{TE} \in \mathbb{R}$ are weights and biases for the transition encoder. Similarly, $\hat{T}_Q \in \mathbb{R}^{1 \times N}$ denotes the inferred probability distribution on all N concept transitions given a query Q.

3.5. Mutual Transition Loss

The idea of mutual transition loss is to characterize the loss caused by transferring the inferred concept transitions to their corresponding concepts, and the other way around. Since for each concept transition $t_{i \to j} \in T$, two concepts c_i and c_j are evolved in the query. If a concept transition $t_{i \to j}$ is inferred with a high probability while its corresponding concepts c_i, c_j have low probabilities, then that indicates conflicts in the final active concept graph. The mutual transition loss is proposed in the co-inference procedure to minimize the conflicts between the inferred concepts and concept transitions so that the resulting active concept graph can be more reasonable.

The graph-based formulation for concept graph gives an appealing property that transitions and their proximate concepts can be clearly characterized by a transfer matrix $A \in \mathbb{R}^{M \times N}$ over the concept graph $G = \langle C, T \rangle$. Each entry $a_{mn} = 1$ if and only if the concept c_m involves in at least a concept transition $t_{m \to}$ or $t_{\cdot \to m}$.

The mutual transfer loss is defined on $\hat{C}_Q, \hat{T}_Q, \mathcal{T}_Q$ as:

$$\mathcal{L}_{MTL}(\hat{C}_Q, \hat{T}_Q, \mathcal{T}_Q) = H(\mathcal{T}_Q, \hat{T}_Q) + E(\hat{C}_Q, \hat{T}_Q), \quad (16)$$

where $\mathcal{T}_Q$ is a ground truth one-hot indicator for concept transitions given a query Q. $\hat{C}_Q$ and $\hat{T}_Q$ are inferred concepts and concept transitions with the proposed method. $H(\cdot, \cdot)$ calculates the cross entropy [36]. $E(\hat{C}_Q, \hat{T}_Q)$ is

an energy-based function on inferred transitions $\hat{T}_Q$ and inferred concepts $\hat{C}_Q$. Each combination of $\hat{C}_Q$ and $\hat{T}_Q$ corresponds with an energy value, the lower energy level a combination of $\hat{C}_Q$ and $\hat{T}_Q$ has indicates less conflicts among the inferred concepts and transitions. In this work, an energy-based function for $E(\hat{C}_Q, \hat{T}_Q)$ is proposed as:

$$E(\hat{C}_Q, \hat{T}_Q) = \mathcal{L}_R(\hat{C}_Q, \hat{T}_Q A^T) + \mathcal{L}_R(\hat{T}_Q, \hat{C}_Q A), \quad (17)$$

where $\mathcal{L}_R$ is similar with the ranking loss [30]. In this work, $\mathcal{L}_R$ penalizes cases where the inferred concepts/transitions after transformation by matrix A have high probabilities but order below the ranking of the originally inferred concepts/transitions in a query. $\mathcal{L}_R$ has a general form:

$$\mathcal{L}_R(\hat{X}, \hat{Y}) = \frac{1}{\left|\hat{X}\right| (L - \left|\hat{X}\right|)} |\{(p, q) : \hat{Y}_p < \hat{Y}_q, \hat{X}_p \geq \hat{X}_q\}, \quad (18)$$

where $\hat{X} \in \mathbb{R}^{1 \times L}$ is the originally inferred labels and $\hat{Y} \in \mathbb{R}^{1 \times L}$ is the inferred labels from the transformation with A. $|\cdot|$ denotes the number of ground truth labels being assigned. L is the label size, where we have M for concepts and N for concept transitions.

4. Evaluation

4.1. Data Set

We collect medical queries from an online medical question answering forum[4], on which user posted their healthcare related questions and medical professionals give online suggestions or advice. The obtained corpora are in Chinese. Due to the fact that Chinese text queries are not naturally split by spaces, word segmentation is performed using a Chinese word segmentation package [7]. The segmentation results do not simply segment queries by each Chinese character. Instead, it tries to combine strongly correlated consecutive characters into words, thus "word" referred in this work can contain more than one Chinese character. After preprocessing and annotation, a medical text query has the following format: { "text": "宫颈管 慢性 炎症 伴 鳞状 上皮 内 挖 空 细胞 聚集 是 宫颈癌 吗 严重 吗 需要 leep 手术 吗", "pos": "n b n v n n n n n v v n y a y v eng n y", "concept": "fee|disease|surgery|recover|treatment", "concept_transition": "disease → surgery → recover"}, where the POS tagging uses ICTCLAS annotation [41]. Among 10,000 medical text queries, 11,531 unique words and 60 unique POS tags are observed. The average length of question is 13.8, with a standard variation of ± 6.1. The average number of concepts in labeled queries is 3.6020 ± 0.8. The average number of concept transitions is 2.4723 ± 0.7. Word embeddings are pre-trained using a skip-gram model [29] on 64 million unlabeled medical text queries separately. Context window size is set to 8 and we specify a minimum occurrence count of 5. The vocabulary contains 100-dimension vectors on 382216 words. Words not presented

---

4. http://club.xywy.com

in the set of pre-trained words are initialized as random vectors. All word vectors will be updated during training.

4.2. Experiment Settings

4.2.1. Comparison Methods. To show the advantages of the proposed method in addressing the concept transition inference problem, we compare it with the following baseline models.

- LR: a logistic regression model applied with POS tagging features and word representations.
- NNID-JM [40]: the neural network intention detection model with joint modeling. Both words and POS tags are used to characterize the question . Domain-specific POS tags, such as "noun_medicine", are used in NNID-JM instead of "noun" for word "Tylenol". The NNID-JM doesn't explicitly exploit label correlations on the output level.
- CI: the concept inference model which only infers mention of concepts from queries with the concept encoder. $H(\mathcal{C}_Q, \hat{C}_Q)$ is used as the loss function for the CI task.
- CTI: the concept transition inference model without co-inference. Only concept transitions are inferred from queries without considering concepts. The last output states of two RNNs are concatenated to predict the concept transitions. $H(\mathcal{T}_Q, \hat{T}_Q)$ is used as the loss function.
- coCTI: the concept transition inference model with co-inference. $H(\mathcal{T}_Q, \hat{T}_Q) + H(\mathcal{C}_Q, \hat{C}_Q)$ is used as the loss function. This variation can be seen as a multi-task learning model for concept and concept transitions, where both tasks share the neural network structure for word representation.
- coCTI-MTL: the proposed model with co-inference and a mutual transfer loss $\mathcal{L}_{MTL}$, where the CI task and CTI task not only share the neural network structure, but also guided by the mutual transfer loss.

4.2.2. Evaluation Metrics. Each edge in the concept graph is considered as an individual label and we evaluate inferred concept transitions as a multi-class, multi-label classification problem. Receiver operating characteristic (ROC) [17], the micro/macro-average area under the curve (micro-AUC, macro-AUC) [10], coverage error [36] and label ranking average precision (LRAP) [28] are used to evaluate the effectiveness of the proposed model in inferring concept transition in medical text queries. The ROC and AUCs focus on the quality of prediction, while the coverage error and LRAP are introduced to evaluate the completeness/ranking of the prediction. ROC is the curve created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. Micro-AUC computes the averaged area under the ROC curve over all the labels. Coverage error computes the average number of labels that we need to have in the final prediction in order to predict

all true labels. LRAP score favors better rank to the labels associated to each sample, which is used in multi-label ranking problems.

4.2.3. Experiment Settings. The embeddings for word and POS tagging have a dimension of 100 and 20, respectively. The hidden layer and the output layer of the GRU unit have a dimension of 100. For training the proposed neural network structure, 70% of the labeled data are used for training and 10% data serve as a validation set to tune for the best parameter set. The remaining data are used for testing. Cross-validation is used and we combine test data in each fold to report the test performance. The optimization is performed in a mini-batch fashion with a batch size of 32. The Adam Optimizer [21] is applied to train the neural network and the initial learning rate is set to 10^{-4}. Weight variables are initialized with the Xavier initializer [13] and bias variables are initialized as zeros. The proposed model is implemented in Tensorflow [1].
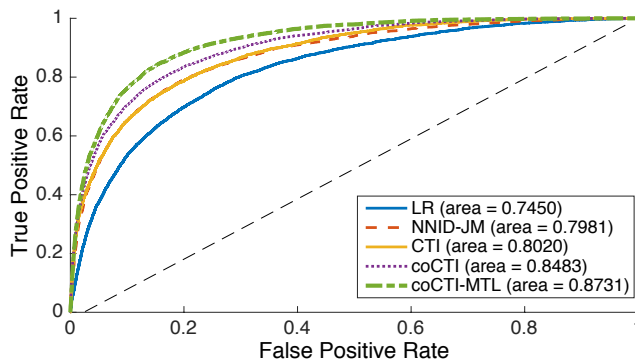
4.3. Evaluation Results

Figure 5. micro-AUC scores and ROC curves.

Figure 5 shows the effectiveness of the proposed model by micro-AUC and ROC curves. Generally, neural network based models (NNID-JM, CTI, coCTI, coCTI-MTL) outperform traditional logistic regression model (LR) consistently. For NNID-JM, in order to make a fair comparison, domain specific POS tags (such as noun_disease, noun_medicine, noun_symptom) are maintained as an external knowledge base. Those POS tags are used by the POS tagger in NNID-JM as its default setting. When compared with NNID-JM, the proposed CTI model achieves similar performance on micro-AUC, while it doesn't rely on any other external knowledge like domain-specific POS tags in NNID-JM. In practical, utilizing a concept transition graph is usually more feasible than tagging words and building dictionaries to maintain words for each domain-specific concept.

From Figure 5 we can further observe that CTI-MTL achieves the best performance (0.8731 in micro-AUC) among all the comparison methods in inferring concept transitions in medical queries. The CTI-MTL model has a nearly 2.5% improvement on micro-AUC when compared

with coCTI and a nearly 7.5% improvement with CTI. This demonstrates that the mutual transfer loss which penalizes conflicts between the inferred concepts and inferred concept transitions can improve the inference quality.
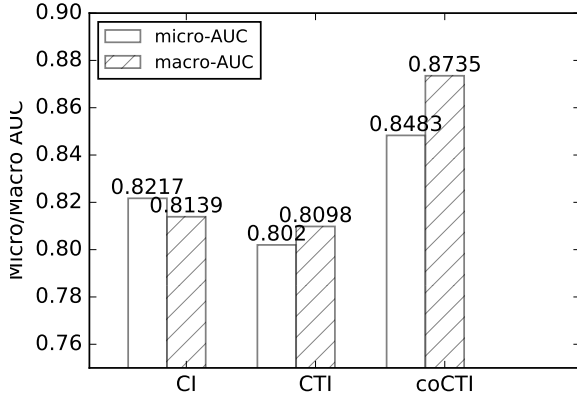

Figure 6. Micro/Macro-AUC scores for collective inference (coCTI) VS. concept inference(CI) and concept transition inference (CTI) separately.

Concept Transition	LR	NNID-JM	CTI	coCTI	coCTI-MTL
Symptom→Diet	0.6544 (5)	0.7755 (4)	0.7669 (3)	0.7959 (2)	0.8495 (1)
Symptom→Medicine	0.7022 (5)	0.7893 (4)	0.8242 (3)	0.8571 (2)	0.8624 (1)
Symptom→Cause	0.7600 (5)	0.8549 (4)	0.8786 (3)	0.8911 (1)	0.8880 (2)
Disease→Diet	0.7818 (5)	0.8670 (4)	0.8681 (3)	0.9059 (2)	0.9458 (1)
Disease→Treatment	0.7181 (5)	0.7787 (3)	0.7482 (4)	0.8456 (2)	0.8836 (1)
Disease→Examine	0.6397 (5)	0.6707 (4)	0.7838 (3)	0.8221 (2)	0.8480 (1)
Disease→Medicine	0.7623 (5)	0.8726 (4)	0.8749 (3)	0.8873 (2)	0.9015 (1)
Surgery→Recover	0.8117 (5)	0.9126 (3)	0.9012 (4)	0.9239 (2)	0.9396 (1)
Surgery→Sequela	0.7385 (5)	0.8031 (4)	0.8214 (3)	0.8417 (2)	0.8972 (1)
Surgery→Syndrome	0.7896 (5)	0.7994 (4)	0.8634 (2)	0.8619 (3)	0.9172 (1)
Surgery→Risk	0.6613 (5)	0.8063 (4)	0.8688 (3)	0.8715 (2)	0.9099 (1)
Medicine→Symptom	0.6861 (5)	0.8275 (3)	0.7553 (4)	0.8294 (2)	0.8598 (1)
Medicine→Side Effect	0.6652 (5)	0.8162 (3)	0.7771 (4)	0.8135 (2)	0.8814 (1)
Medicine→Disease	0.6806 (4)	0.6514 (5)	0.8081 (3)	0.8126 (2)	0.8678 (1)
Medicine→Instruction	0.7090 (5)	0.7761 (3)	0.7603 (4)	0.8170 (2)	0.8820 (1)
Examine→Fee	0.7576 (5)	0.9049 (3)	0.8981 (4)	0.9425 (2)	0.9482 (1)
Examine→Diagnosis	0.6832 (5)	0.7956 (3)	0.7445 (4)	0.8383 (2)	0.8822 (1)
Symptom→Treatment	0.6817 (5)	0.7640 (4)	0.7313 (4)	0.8130 (2)	0.8531 (1)
Symptom→Department	0.5978 (5)	0.6460 (3)	0.6013 (4)	0.6738 (2)	0.8080 (1)
Disease→Cause	0.7306 (5)	0.8206 (4)	0.8515 (3)	0.8608 (2)	0.8634 (1)
Disease→Symptom	0.6936 (4)	0.7552 (5)	0.6845 (5)	0.7554 (2)	0.8372 (1)
Disease→Department	0.6931 (5)	0.7387 (4)	0.7431 (3)	0.7652 (2)	0.8290 (1)
Disease→Surgery	0.7801 (5)	0.8795 (4)	0.9029 (3)	0.9236 (2)	0.9380 (1)

TABLE 3. FINE-GRAINED AUC SCORES FOR CONCEPT TRANSITION INFERENCE FOR EACH CONCEPT TRANSITION (EACH EDGE IN THE CONCEPT GRAPH).

Figure 6 shows the effectiveness of the co-inference procedure by comparing the performance of CTI with coCTI. The CI infers concept mentions so we can't simply compare its performance with CTI/coCTI where concept transitions are inferred. However, for CTI and coCTI, the improved performance on both micro-AUC and macro-AUC validate the effectiveness of inferring concept transitions and concept mentions collectively than inferred separately. The coCTI model can be considered as a multi-task learning model where the question representation is learned jointly and shared between two inference tasks.

Furthermore, the fine-grained AUC scores on all concept transitions without micro/macro-averaging are shown in Table 3. A general observation we can draw from the results

is that the coCTI-MTL model is able to outperform other baselines in almost all types of concept transitions.
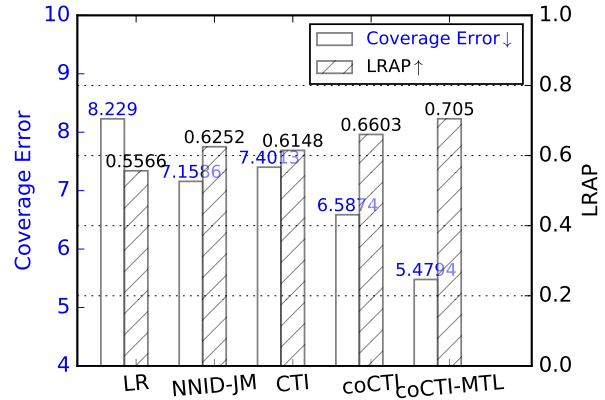

Figure 7. Coverage Loss and Label Ranking Average Precision (LRAP).

Figure 7 shows the coverage loss and LRAP over proposed methods and other baselines, where the coCTO-MTL model is able to achieve the lowest coverage error and the highest label ranking average precision score.

Figure 8. Confidence scores assigned by the concept encoder on sample queries. A darker color indicates a higher score.

Five case studies are presented in Figure 8 to show scores assigned by the concept encoder on real-world medical text queries. Some stop-words are removed for clarity. We can see that the concept encoder is properly trained as it is able to assign important words or words refer to concepts higher confidence scores, while common words are less likely to receive such high scores. This observation indicates the effectiveness of the proposed concept encoder in encoding concept mentions without relying on domain-specific external knowledge bases.

5. Related Works

5.1. Medical Query Analysis

As a growing number of people are posting medical related questions or searching with medical text queries online, researchers have been focusing on new problems and applications based on medical queries or search queries that users generated. [25] analyzes the conceptual relationship in medical records for a better medical search. [34] studies the circumlocution problem in diagnostic medical queries, where users are not able to express their ideas effectively.

[40] tries to model user intentions as a classification task for medical text queries. [27] proposes a technique to detect whether users express patient experiences in their medical text queries. In [26], authors introduce a neural network model to understand users healthcare related questions and try to generate answers appropriately. Being able to infer medical concept transitions from noisy, user-generated healthcare questions may further facilitate various medical applications such as healthcare question-answering, medical dialog systems or recommendation. For example, once we extracted the concept transition $Symptom \rightarrow Medicine$ from a question *Any medication is recommended to help me fall asleep easier?*, we may follow up by recommending the user to the nearest pharmacy for further medical consultations on corresponding OTC medicines on Insomnia.

5.2. Text Classification

Recently, lots of neural network models are developed for classifying natural language texts into different categories [37], [44], [22], [16], [39]. Those methods achieve decent performance on general text classification tasks. The proposed concept transition problem can be cast as a multi-class multi-label classification problem. Unlike traditional text classification tasks like news classification where the existence of some topic words may easily dominate the label for a news title, users tend to mention multiple medical concepts in a single medical text query. It is crucial to extract user medical concept transitions among multiple medical concepts, besides just concept mentions individually.

Also, the aforementioned methods consider the textual information only. With a graph-based formation in this paper, our model is able to seamlessly incorporates an existing concept graph with the medical text query. Moreover, we propose to predict concept mentions as nodes and transitions as links on an abstract level collectively, while most existing works have been focusing on predicting links among concrete entities, e.g. among users in social networks [24], or predicting links among entities on a knowledge graph [31], [4].

6. Conclusions

People nowadays are posting or searching with medical text queries extensively on the world wide web. Various medical information needs are expressed diversely in users medical text queries. In this work, we bring semantic structures to user intention detection in real-world online medical queries by mapping diversely expressed medical queries to a concept graph where each node on a concept graph represents a concept mention and concept transitions are represented as directed edges. A novel neural network structure based on multi-task learning is introduced to extract concept mentions as well as medical concept transitions that users encoded in online healthcare questions collectively. Evaluation results on real-world medical questions address the effectiveness of the proposed model.

References

[1] Martın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous systems. 1, 2015.

[2] Tanveer Ali, David Schramm, Marina Sokolova, and Diana Inkpen. Can i hear you? sentiment analysis on medical forums. In *IJCNLP*, pages 667–673, 2013.

[3] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

[4] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795, 2013.

[5] J Campbell, G Dussault, J Buchan, F Pozo-Martin, M Guerra Arias, C Leone, A Siyam, and G Cometto. A universal truth: no health without a workforce. *Geneva: World Health Organization*, 2013.

[6] Fei Chiang, Periklis Andritsos, Erkang Zhu, and Renée J Miller. Autodict: Automated dictionary discovery. In *ICDE*, pages 1277–1280, 2012.

[7] Jieba chinese word segmentation package. https://github.com/fxsjy/jieba.

[8] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[9] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[10] Corinna Cortes and Mehryar Mohri. Auc optimization vs. error rate minimization. *NIPS*, 16(16):313–320, 2004.

[11] Arijit De and Sunil Kumar Kopparapu. A rule-based short query intent identification system. In *ICSIP*, pages 212–216. IEEE, 2010.

[12] S Fox and M Duggan. One in three american adults have gone online to figure out a medical condition. *Pew Internet & American Life Project*, 2013.

[13] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, volume 9, pages 249–256, 2010.

[14] Shantanu Godbole, Indrajit Bhattacharya, Ajay Gupta, and Ashish Verma. Building re-usable dictionary repositories for real-world text mining. In *CIKM*, pages 1189–1198. ACM, 2010.

[15] Felix Greaves, Daniel Ramirez-Cano, Christopher Millett, Ara Darzi, and Liam Donaldson. Use of sentiment analysis for capturing patient experience from free-text comments posted online. *Journal of medical Internet research*, 15(11), 2013.

[16] Edward Grefenstette and Phil Blunsom. A Convolutional Neural Network for Modelling Sentences. *ACL*, 2014.

[17] James A Hanley and Barbara J McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

[18] Takaaki Hasegawa, Satoshi Sekine, and Ralph Grishman. Discovering relations among named entities from large corpora. In *ACL*, page 415. Association for Computational Linguistics, 2004.

[19] Sepp Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6(02):107–116, 1998.

[20] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[21] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[22] Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. Recurrent Convolutional Neural Networks for Text Classification. *AAAI*, pages 2267–2273, 2015.

[23] Joël Legrand and Ronan Collobert. Joint rnn-based greedy parsing and word composition. In *ICLR*, 2015.

[24] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[25] Nut Limsopatham, Craig Macdonald, and Iadh Ounis. Inferring conceptual relationships to improve medical records search. In *OAIR*, 2013.

[26] Chaochun Liu, Huan Sun, Nan Du, Shulong Tan, Hongliang Fei, Wei Fan, Tao Yang, Hao Wu, Yaliang Li, and Chenwei Zhang. An augmented lstm framework to construct medical self-diagnosis android. In *ICDM*, 2016.

[27] Yunzhong Liu, Yi Chen, Jiliang Tang, and Huan Liu. Context-aware experience extraction from online health forums. In *ICHI*, 2016.

[28] Gjorgji Madjarov, Dragi Kocev, Dejan Gjorgjevikj, and Sašo Dzeroski. An extensive experimental comparison of methods for multi-label learning. *Pattern Recognition*, 45(9):3084–3104, 2012.

[29] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119, 2013.

[30] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[31] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2016.

[32] Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, 2016.

[33] Amanda Spink, Yin Yang, Jim Jansen, Pirrko Nykanen, Daniel P Lorence, Seda Ozmutlu, and H Cenk Ozmutlu. A study of medical and health queries to web search engines. *Health Information & Libraries Journal*, 21(1):44–51, 2004.

[34] Isabelle Stanton, Samuel Ieong, and Nina Mishra. Circumlocution in diagnostic medical queries. In *SIGIR*. ACM, 2014.

[35] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.

[36] Grigorios Tsoumakas, Ioannis Katakis, and Ioannis Vlahavas. Mining multi-label data. In *Data mining and knowledge discovery handbook*, pages 667–685. Springer, 2009.

[37] Puyang Xu and Ruhi Sarikaya. Contextual Domain Classification in Spoken Language Understanding Systems Using Recurrent Neural Network. *ICASSP*, (Lm):3–7, 2014.

[38] Yulan Yan, Naoaki Okazaki, Yutaka Matsuo, Zhenglu Yang, and Mitsuru Ishizuka. Unsupervised relation extraction by mining wikipedia texts using information from the web. In *ACL*, pages 1021–1029. ACL, 2009.

[39] Shuangfei Zhai, Keng-hao Chang, Ruofei Zhang, and Zhongfei Mark Zhang. Deepintent: Learning attentions for online advertising with recurrent neural networks. In *KDD*, 2016.

[40] Chenwei Zhang, Wei Fan, Nan Du, and Philip S Yu. Mining user intentions from medical queries: A neural network based heterogeneous jointly modeling approach. In *WWW*, 2016.

[41] Hua-Ping Zhang, Hong-Kui Yu, De-Yi Xiong, and Qun Liu. Hhmm-based chinese lexical analyzer ictclas. In *SIGHAN*, pages 184–187. Association for Computational Linguistics, 2003.

[42] Jingyuan Zhang, Chun-Ta Lu, Mianwei Zhou, Sihong Xie, Yi Chang, and Philip S Yu. Heer: Heterogeneous graph embedding for emerging relation detection from news. In *Big Data (Big Data), 2016 IEEE International Conference on*, pages 803–812. IEEE, 2016.

[43] Mi Zhang and Christopher C Yang. Classification of online health discussions with text and health features sets. In *Proceedings of AAAI International Workshop on the World Wide Web and Public Health Intelligence 2014 (W3PHI 2014)*, 2014.

[44] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *NIPS*, 2015.