Noname manuscript No.
(will be inserted by the editor)

Multi-task Character-level Attentional Networks for Medical Concept Normalization

Jinghao Niu[1,2] · **Yehui Yang**[1,2] · **Siheng Zhang**[1,2] · **Zhengya Sun**[1,2] · **Wensheng Zhang**[1,2]

Received: date / Accepted: date

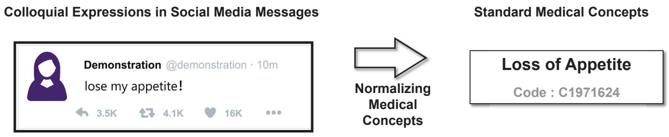
Abstract Recognizing standard medical concepts in the colloquial text is significant for kinds of applications such as the medical question answering system. Recently, word-level neural network methods, which can learn complex informal expression features, achieved remarkable performance on this task. However, they have two main limitations: (1) Existing word-level methods cannot learn character structure features inside words and suffer from "Out-of-vocabulary" (OOV) words, which are common in noisy colloquial text. (2) Since these methods handle the normalization task as a classification issue, concept phrases are represented by category labels. Hence the word morphological information inside the concept is lost. In this work, we present a multi-task character-level attentional network model for medical concept normalization. Specifically, the character-level encoding scheme of our model can alleviate the OOV word problem. The attention mechanism can effectively exploit the word morphological information through multi-task training. It generates higher attention weights on domain-related positions in the text sequence, helping the downstream convolution focus on the characters that are related to medical concepts. To test our model, we first introduce a labeled Chinese dataset (overall 314991 records) for this task. Other two real-world English datasets are also used. Our model outperforms state-of-the-art methods on all three datasets. Besides, by adding four types noises to the datasets, we validate the robustness of our model against common noises in the colloquial text.

Keywords Convolutional neural network · Multi-task learning · Medical concept normalization

✉ Wensheng Zhang
Tel.: +010-82544673
E-mail: zhangwenshengia@hotmail.com

1 Institute of Automation, Chinese Academy of Sciences, 95 Zhongguancun East Road, 100190, Beijing, China

2 University of Chinese Academy of Sciences, Beijing, China



Demonstration @demonstration · 10m

lose my appetite!

Colloquial Expressions in Social Media Messages
3.5K
4.1K
16K
Normalizing Medical Concepts
Standard Medical Concepts
Loss of Appetite
Code : C1971624

Fig. 1 An instance of normalizing medical concepts in social media messages.

1 Introduction

Normalizing medical concepts, which means mapping informal expressions written in layman's language to refined medical concepts, contributes to many applications. For example, HealthTap[1], a community-based healthcare question answering website, accumulates large numbers of personalized medical question-answer pairs generated by healthy seekers and medical experts. By bridging the lexical gap between informal expressions and standard concepts, the normalization operation makes it possible to reuse cross-system knowledge and accomplish automatic disease inference [25]. Some studies have shown that semantic analysis on data from the social media like Twitter[2], can have broad implications for both public health and drug reactions research [26, 31].

However, the normalization task is challenging because the difference between informal expressions and standard medical concepts is significant. Sometimes the text to be processed (e.g., social media messages) even shares no common words with target medical concepts. For instance, "moon face and 30 lbs in 6 weeks" to the medical concept "Weight Gain", or map "head spinning a little" to "Dizziness" [18]. In these cases, normalization methods that depend on simply text similarity calculation perform poorly. Besides, dealing with the noisy text usually suffers from the OOV issue, which could be caused by misspellings easily. For example, the character combination "upppp" in the phrase "It got me upppp!" is a personal expression, which is likely to be an OOV word.

Recently, Limsopatham et al. [18] introduce a convolutional neural network model that achieves state-of-the-art performance on this task. It conspicuously outperforms any other previous method on real-world datasets constructed with social media data. The method [18] takes advantage of the word-based convolution structure [12] and pre-trained word embeddings [19], but it can merely exploit word-level features. Different from word-based neural networks, character-level neural network methods [28, 40, 7, 39], which encode sentences with character-level representations, can exploit local structure features inside words. Methods with character-level inputs are particularly suitable for the medical concept normalization task because they can avoid the OOV issue. However, since most of the real-world knowledge is stored in word-level (e.g., entities in knowledge bases), the character-level methods might have limitations in utilizing structured knowledge. Recently, several studies have reported that hybrid word-character language models outperformed single word-level or character-level models. For example, Miyamoto et al. [20] introduced a gated language model that could adaptively

[1] https://www.healthtap.com/

[2] https://twitter.com/

find the optimal mixture of the character-level and word-level inputs. Yang et al. [35] presented a fine-grained gating mechanism to dynamically combine word-level and character-level representations based on several properties (e.g., named entity tags and part-of-speech tags) of input words. These combination operations belong to downstream fusion processes, which means that the final combined representation is a hybrid of two level separately generated representations. Because different OOV words share the same word-level representation, final representations of OOV words actually depend on their character-level representations. Besides, no matter using the word-level or character-level representation, once the method handles normalization task as a classification issue, the target concept will be represented by its category label (e.g., C1971624 to represent "Loss of Appetite", see Fig. 1). The specific words in the concept (i.e., "Loss", "of" and "Appetite") cannot be used to supervise training. In other words, the word morphological information inside the concept is lost.

In this work, we propose a multi-task character-level attentional network to normalize standard medical concepts in the colloquial text. The character-level encoding scheme of our network can capture character-level features even in OOV words. Besides, the word-level morphological information in the medical concept is effectively exploited to supervise the training of an auxiliary network. This network aims at generating particular character-level attention weights on domain-related words (or character combinations), which is treated as an auxiliary task during the training process of the main task (concept normalization). These attention weights are then fed into the main network, helping our model particularly focus on the domain-related positions in the text sequence. We estimate our model on two English real-world datasets constructed from the collection of social media data. Furthermore, we introduce a new **Ch**inese **m**edical **c**oncept **n**ormalization dataset (*ChMCN*) generated from an online healthcare question answering website. Experimental results show that our method outperforms state-of-the-art methods on both English and Chinese datasets, no matter in the aspect of the normalization accuracy or the robustness against common text noises.

The main contributions of this work are as follows:

- To our knowledge, it is the first work that exploits character-level convolutional neural networks to handle the medical concept normalization task. The character-level network structure effectively alleviates the OOV word issue, which is a significant challenge in the normalization task.
- We propose a multi-task learning framework to exploit the morphological information in concept words. It generates character-level attention weights on domain-related positions in the text sequence, which are validated to help improve the concept normalization accuracy in experiments.
- We construct four types of new datasets by adding common noises to original datasets. We use these noisy datasets to evaluate the model robustness against noises in the colloquial text.

2 Related Work

The normalization operation on standard medical documents such as medical records and literature is a well-studied area [16, 1]. Nevertheless, normalizing medical concepts in the colloquial text is a relatively open problem. Recent studies such as Metke-Jimenez and Karimi [11], OConnor et al [26] and Limsopatham [17] used weighting techniques to handle colloquial messages, such as *TF-IDF*, *BM25* [27] and *Word2Vec* embeddings [19]. These methods can map colloquial texts to standard medical concepts. The state-of-the-art method introduced by Nut Limsopatham et al. [18] is based on the neural network mechanism, which is reported to outperform previous methods obviously. Besides, some recent studies try to map the colloquial texts to other kinds of healthcare-related targets such as the disease category [23, 4], the multi-faced health answer [22], and the mention of adverse drug reaction [34].

Charcter-level Convolution: Recent methods, no matter using weighting techniques [11, 26, 17] or neural network approaches [18], encode the original input text at word level. Character-level convolutional neural network methods have been reported to perform better than word-level models in many classification tasks. For example, Santos et al. [28] proposed a convolutional neural network that could use information from character-level to sentence level to perform sentiment analysis. Zhang et al. [40] offered an empirical exploration on several large-scale text classification datasets, showing the competitive ability of a nine layers character-level CNN model. Most lately, Conneau et al. [7] proposed a deeper convolutional neural network (up to 29 layers) in character-level, reporting significant improvements over several text classification tasks.

Attention Mechanism: An early application of the attention mechanism was a Boltzmann machine for the image classification task, which was introduced by Larochelle and Hinton [15]. Nowadays, the attention mechanism has been successfully used in kinds of Natural Language Processing (NLP) tasks. For example, Bahdanau et al. [8] added the attention structure to improve the performance of neural machine translation model. Shin et al. [30] applied the attention mechanism in a Convolutional Neural Networks (CNN) model for Sentiment Analysis. Golub and He [9] exploited character-level Long Short-Term Memory (LSTM) networks to generate attention weights about entities and predicates to handle the question answering. Shen and Huang [29] proposed an attention-based convolutional neural network architecture for semantic relation extraction. Recently, the attention mechanism became particularly compelling for medical-related tasks, because it can improve the performance of model while remaining interpretable. For instance, Choi et al. [6] propose a reverse time attention model to handle electronic health records data, which focuses on specific clinical information like interpretable key risk factors. Zhang et al. [41] introduce *MDNet* to generate diagnostic reports from medical images. It contains an attention mechanism to learn the mapping from sentence words to image pixels.

Multi-task Learning: Multi-Task Learning (MTL) can improve the generalization performance of single or several tasks through jointly training (sharing some weights) [2]. The application of pre-trained *Word2Vec* embeddings [19] could be treated as an example of multi-task learning. Besides, there are many other successful attention applications in NLP domain. Yang et al. [36] exploited multi-task and cross-lingual joint training to improve the sequence tagging performance. Yu et al. [37] proposed a sentiment classifier model training with two related auxiliary tasks. Sgaard and Goldberg [32] reported that an additional auxiliary task of POS-tagging supervised at lower layers could improve both syntactic chunking and CCG supertagging task. Another interesting result in this study [32] was that only auxiliary tasks that are highly related to the main task could bring about improvements. The effectiveness of MTL has also been established in the medical domain. For example, MTL models [38] [42] considering the relatedness among multiple tasks show better performance in modeling the disease progression. Then, Nie et al. [24] further propose a multi-modal MTL model to handle the disease progression exploiting multimedia and multi-modal observations. Besides, MTL helps to improve the performance of many other related applications such as medical image segmentation [5, 21], clinical time series analysis [33], and mental health condition monitoring [3].

3 Preliminaries

3.1 Text Representation:

The raw data to handle in this work are colloquial text records. We define the k-th record in a dataset as $Q^k = [word_1, word_2, ..., word_i, ..., word_n]$. The main difference between character-level methods and word-level methods (e.g. Limsopatham et al. [18]) is the way to represent Q^k. In the word-level models, every word in Q^k is represented by a vector $\mathbf{w}_i \in \mathbb{R}^d$ (d denotes the embedding dimension) from a word embedding lookup table $\mathbf{W} \in \mathbb{R}^{d\times|V|}$. Vocabulary V can be constructed with the training dataset, or the pre-trained embeddings generated from additional corpus data. Thus the input sentence will be represented as an embedding matrix $\mathbf{S}_w^k \in \mathbb{R}^{d\times n}$.

$$\mathbf{S}_w^k = \begin{bmatrix} | & | & & | & & | \\ \mathbf{w}_1 & \mathbf{w}_2 & ... & \mathbf{w}_i & ... & \mathbf{w}_n \\ | & | & & | & & | \end{bmatrix} \tag{1}$$

Number n is the max record length in the training dataset. One sentence will be padded with a special padding tag for the same length of n if necessary.

For character-level methods, every sentence is regarded as a sequence of characters: $Q^k = [char_1, char_2, ..., char_i, ..., char_m]$, where m is the largest character number of sentences in the training dataset. In our model, a character embedding lookup table $\tilde{\mathbf{W}} \in \mathbb{R}^{d\times|V_c|}$ is used to encode the input sequence into $\mathbf{S}_c^k \in \mathbb{R}^{d\times m}$.

$$\mathbf{S}_c^k = \begin{bmatrix} | & | & & | & & | \\ \mathbf{c}_1 & \mathbf{c}_2 & ... & \mathbf{c}_i & ... & \mathbf{c}_m \\ | & | & & | & & | \end{bmatrix} \tag{2}$$

Initial experimental results on the development set showed that the vocabulary ignoring the difference between upper-case and lower-case letters achieved the better performance, which is consistent with the result reported by Zhang et al. [40]. Thus, the character vocabulary V_c of our model consists of all ignoring case English letters and some special characters including the space symbol:

“˜!@#&$%ˆ*(){}[],.:;“”’? + − _ =<> |\/

abcdefghijklmnopqrstuvwxyz0123456789”

3.2 Convolution and Pooling Layers:

The 1-D convolution operation is used to extract high-level features in the character embedding matrix $\mathbf{S}_c^k$, which is based on the filter $\mathbf{F} \in \mathbb{R}^{d \times l}$ with the embedding dimension d and the filter length of l. Specifically, the convolution operation with the filter $\mathbf{F}$ on every character embeddings window $\mathbf{c}_{i:i+l-1}$ generates the feature:

$$z_i = h(\mathbf{F} \cdot \mathbf{c}_{i:i+l-1} + b) \tag{3}$$

where h is the activation function and b is the bias. Then the sentence matrix $\mathbf{S}_c^k \in \mathbb{R}^{d \times m}$ is transformed into a feature map vector $\mathbf{z}$. As shown in Fig. 2, to extract various high-level features, filters with different sizes (length of {2,3,4,5,6}) are applied in the convolution layer in the main task. In the auxiliary task, we only use the filter of length 1, whose main contribution is to generate a non-linear transformation of original features.

The pooling operation can reduce representation and improve the robustness of convolutional features. Max-pooling takes the maximum of z_i as the final output of the corresponding filter, which is widely used to find out significant features and makes it possible to train deeper neural networks [40]. We apply the max-pooling operation after the character-level convolution. Outputs of the pooling process are flattened to dense vectors and concatenated as the input of the softmax layer.

4 Method

In this section, we introduce the detailed architecture of our proposed **M**ulti-**T**ask **A**ttentional **Char**acter-level **C**onvolution **N**eural **N**etwork (*MTA-CharCNN*). The notations in the main task and the auxiliary task are given. Then we explain the learning of multi-tasks and the role of attention mechanism. Two reduced versions of *MTA-CharCNN* are also introduced. We use a real-world instance to explain our model architecture (see Fig. 2). The input phrase is first translated into a character embedding matrix. The auxiliary task network takes it as the input and generates attention weights for this instance. The attention weights are added to corresponding positions of the character embedding matrix. The main task network takes this weighted matrix as its input.

4.1 The Main Task

The input of the main task (medical concept normalization) is a text sequence. The output of the task is the corresponding target concept category, which should be chosen from a predefined concepts dictionary $C = \{concept_1, ..., concept_j, ..., concept_m\}$. For example, the phrase “lose my appetite” is the original input text sequence. It should be mapped to UMLS[3] concept C1971624 (Loss of Appetite) for the main task. The output of softmax layer in the main task is a probability vector $\mathbf{p}^k = [p_1^k, ..., p_j^k, ..., p_m^k]$, where every element $p_j^k = p(y_j^k = 1 | Q^k, \boldsymbol{\theta}_{main})$ denotes the probability that the given k-th text sequence should be mapped to $concept_j$. We use $\boldsymbol{\theta}_{main}$ to represent all of the model parameters in the main task. The k-th

[3] https://www.nlm.nih.gov/pubs/factsheets/umlsmeta.html

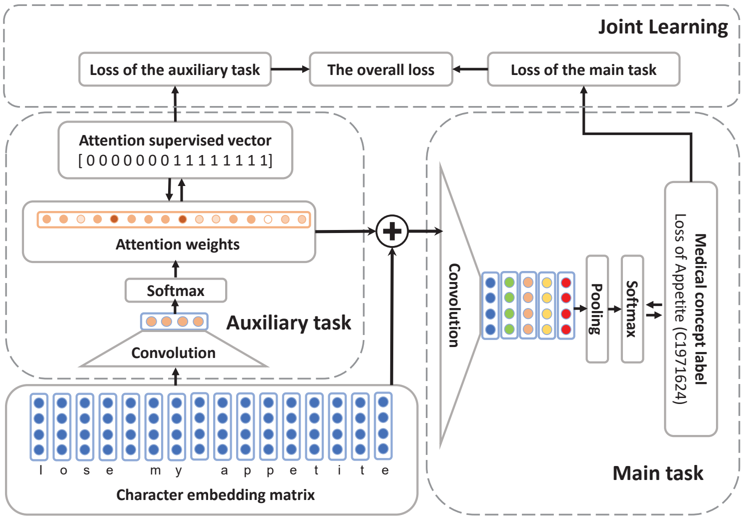


Fig. 2 Architecture of our proposed *MTA-CharCNN* (**M**ulti-**T**ask **A**ttentional **Char**acter-level **C**onvolution **N**eural **N**etwork) model. We noted specific network parts for the main task and the auxiliary task. The character embedding matrix contains the shared parameters for the two tasks. The auxiliary task network takes it as the input and generates attention weights on the character embedding matrix. The weighted embedding matrix then becomes the input of the main task network.

ground truth concept supervised label is $\mathbf{y}^k = [y_1^k, ..., y_j^k, ..., y_m^k]$, which is a one-hot encoded vector. We can learn parameters by minimizing the cross-entropy cost of the main task as:

$$Loss_{main}(\boldsymbol{\theta}_{main}) = -\sum\nolimits_{\mathrm{k}} \sum\nolimits_{j}^{m} [y_j^k \log p_j^k + (1 - y_j^k) log(1 - p_j^k)] \tag{4}$$

4.2 The Auxiliary Task

The auxiliary task is learning to generate character-level domain-related importance weights of the input text sequence. Given a character sequence Q^k, the output of the auxiliary task is an attention weights vector $\mathbf{a}^k = [a_1^k, ..., a_j^k, ..., a_m^k]$, which is of the same length as the input character sequence. Every element a_j^k denotes the confidence that the character in the corresponding position is related to medical domain. This task is instinctively helpful because the model could exploit the additional word morphological information in a multi-task manner. We generated supervised labels by defining the domain-related words set. We collected all constituting words of target medical concepts in two English datasets, yet stop words are not included. Once a word in input text is found to be in this set, positions of characters in this word will be marked with label "1". Then, we will get an attention supervised vector $\mathbf{att}^k = [att_1^k, ..., att_j^k, ..., att_m^k]$, where $att_j^k \in \{0, 1\}$. For the instance in Fig. 2, "appetite" is the only domain-related word in the phrase, thus the attention supervised vector for "lose my appetite" is

[0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1]. These labels may help our model to recognize words that are related to standard concepts.

For English data, because one single character is hard to convey a detailed meaning, attention weights in $\mathbf{a}^k$ are usually locally successive. However, the Chinese character is ideogram, which means one character may be enough to represent lots of information, which makes the potentially related character combinations very complex. Thus, we firstly collected words that make up the medical concepts in training set to build a basic dictionary. Then we asked medical experts who have Bachelor of Medicine for help to enlarge the basic dictionary. Correspondingly we define the training loss function of the auxiliary task as:

$$Loss_{auxiliary}(\boldsymbol{\theta}_{aux}) = -\sum\nolimits_{\mathrm{k}} \sum\nolimits_{j}^{m} [att_j^k \log a_j^k + (1 - att_j^k) log(1 - a_j^k)] \qquad (5)$$

4.3 Joint Learning of Tasks

We use $\boldsymbol{\theta}_{aux}$ to represent all of model parameters in the auxiliary task. Parameters $\boldsymbol{\theta}_{mian}$ and $\boldsymbol{\theta}_{aux}$ share parts of weights (character embedding weights). All of parameters are learned jointly by minimizing the overall loss function. respectively, using a loss weighting parameter $\lambda \in [0, 1]$, we define the overall loss function as:

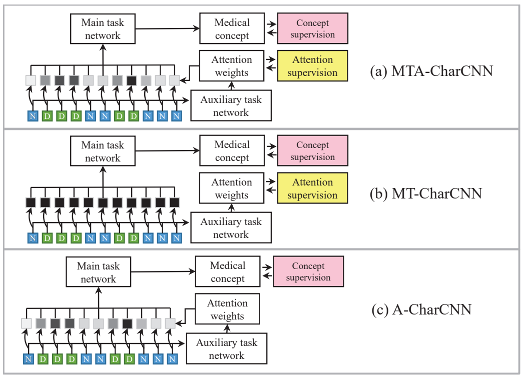
$$Loss_{overall} = (1 - \lambda) Loss_{main}(\boldsymbol{\theta}_{main}) + \lambda Loss_{auxiliary}(\boldsymbol{\theta}_{aux}) \qquad (6)$$

We did grid search of the parameter λ on three datasets. Recommended λ value for our neural network model is 0.3, which was chosen from {0.01, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9}. The recommend λ value performed well in most of initial experiments, however choosing highly targeted λ for different datasets may bring about a slight extra improvement. For training, the mini-batch size is 100 and we use *Adam* [14] as the optimizer. For regularization, *Dropout* [10] probability is 0.5 for embedding-to-convolution layers.

4.4 Attention Mechanism and Reduced Models

The word morphological information is used in the training of the auxiliary task network, which also affects the parameters of the character embedding matrix. However, it cannot benefit other parts of the main task network. Here we add an attention mechanism to our model, which feeds the output of the auxiliary task network to the main task network. After adequate training, the output of the auxiliary task network will be an interpretable attention weights vector. Every weight value in the vector represents the confidence that the corresponding position character is domain-related. It is then transformed into an attention weights matrix $\mathbf{A}^k \in \mathbb{R}^{d \times m}$. Every column of $\mathbf{A}^k$ is a vector containing d copies of a_j^k. $\mathbf{A}^k$ is added to the original character embedding matrix. This mechanism helps downstream convolutions in main task network focus on significant parts of the text sequence.

We introduce two reduced version of *MTA-CharCNN* to estimate the respective contributions of the attention mechanism and the auxiliary task supervision (see Fig. 3). Specifically, *MT-CharCNN* denotes the model that does not add attention weights to the character embeddings. The main task only benefits from auxiliary



(a) MTA-CharCNN

(b) MT-CharCNN

(c) A-CharCNN

Fig. 3 Architectures of our proposed **M**ulti-**T**ask **A**ttentional **Char**acter-level **C**onvolution **N**eural **N**etwork (*MTA-CharCNN*) and its reduced versions (*MT-CharCNN* and *A-CharCNN*). Specifically, *MT-CharCNN* does not feed the attention weights into the main task network. *A-CharCNN* does not exploit the word morphological information to supervise the generating of attention. Domain-related characters are colored with green; other characters are colored with blue. Darker color represents the higher attention weight value.

Table 1 Parameters of our proposed models.

	MTA-CharCNN	MT-CharCNN	A-CharCNN
attention supervision	with	with	without
loss weighting parameter	0.3	0.3	0
embedding dimension	300		
filter length (main task)	2, 3, 4, 5, 6		
filter length (auxiliary task)	1		
number of filters	100		
softmax layer nodes	number of output classes		
activation function	ReLU		

task's regularization effects on shared weights in this reduced model. *A-CharCNN* denotes the model that sets the loss weight parameter λ to be 0, all parameters in our model will not be affected by the attention supervision. Other detailed parameter settings are shown in Table 1.

5 Experiments

5.1 Datasets

To evaluate the performance of our model on the medical concept normalization task, two real-world datasets (i.e. *TwADR-L*, *AskAPatient*)[4] introduced by Limsopatham et al. [18] are used. *TwADR-L* was constructed from the collection of tweets, which contains 1,436 Twitter phrases and the corresponding labeled medical concept chosen from 2,200 standard terms in SIDER 4 database[5]. *AskAPatient* dataset is also a collection of social media messages and their gold-standard mapping medical concepts, which are extracted from the ADR annotation from Karimi et al. [11]. Specific for, it contains 8,662 phrases[6] with their target terms selected from 1,036 medical concepts in SNOMED-CT[7] and AMT (the Australian

[4] http://dx.doi.org/10.5281/zenodo.55013
[5] http://sideeffects.embl.de/
[6] http://www.askapatient.com
[7] http://www.ihtsdo.org/snomed-ct

Medicines Terminology). These datasets are divided into ten equal folds like the settings in Limsopatham et al. [18], which is for the purpose to compare the average accuracy performance of other methods credibly.

Table 2 An example chosen from datasets added with four types of noises.

	Original Records	Records with noises
Type 1	**sleeping** standing up	**#sleeping** standing up
Type 2	**sleeping** standing up	**sleping** standing up
Type 3	**sleeping** standing up	**sleeeping** standing up
Type 4	**sleeping** standing up	**sleeipng** standing up

In order to evaluate methods' robustness against common noises in social media messages, we further generated four types of challenging noisy datasets based on original two datasets. Every record in test sets was added noises: randomly choosing one word and then changing the word into a kind of non-standard format (see Table 2). These changing operations are designed to simulate typical noises in social media messages:

- *Type 1*: adding "#" to the head of the word.
- *Type 2*: deleting one character randomly.
- *Type 3*: doubling one character randomly.
- *Type 4*: randomly choosing one character and changing its position with one of its adjacent letters.

The colloquial messages in *TwADR-L* and *AskAPatient* are mostly at phrase level. To evaluate the performance of our model on longer text, we constructed a new Chinese Medical Concepts Normalization dataset (*ChMCN*). Sentences in *ChMCN* (overall 314991 records) are sampled from a collection of healthcare questions on *KuaiSuWenYiSheng*[8], which is a Chinese online healthcare questions answering website. Each question is labeled with one of the predefined medical concept (overall 300 classes) by medical experts. *ChMCN* is randomly divided into ten equal folds as *TwADR-L* and *AskAPatient*. On every dataset, we evaluate our model and other models based on the average normalization accuracy over ten folds.

5.2 Alternative Methods

As reported by Limsopatham et al. [18], the word-level neural network methods they proposed distinctly outperformed traditional approaches such as *LogisticRegression*, *DNorm* [16] and BM25 [27]. Our initial experiments showed the similar results on *ChMCN*. Thus we do not report the detailed performance of these traditional approaches in this paper. We choose state-of-the-art word-level neural network methods as very competitive baselines. Besides, two character-level convolutional neural network models are also evaluated on three datasets.

- **Rand-CNN** (Limsopatham et al. 2016) [18]: a word-level convolutional neural network model with randomly generated embedding weights.

[8] https://www.120ask.com/

- **Emb-CNN** (Limsopatham et al. 2016) [18]: a word-level convolutional neural network model with general pre-trained word embedding weights. The English pre-trained embedding vectors for *TwADR-L* and *AskAPatient* are generated from *Google News* [19] and the Chinese word embedding vectors are trained with a corpus collected from history records of *KuaiSuWenYiSheng* website.
- **Rand-RNN** (Limsopatham et al. 2016) [18]: a word-level recurrent neural network model with randomly generated embedding weights.
- **Emb-RNN** (Limsopatham et al. 2016) [18]: a word-level recurrent neural network model with the same pre-trained word embeddings as Emb-CNN.
- **CharConvNets** (Zhang et al. 2015) [40]: a character-level 9-layers convolutional neural network with one kind of filter length in convolution layers. We choose the selectable model *Small Feature*, which performed better on datasets in this study according to initial experiments.
- **CharCNN-Small** (Kim et al. 2016) [13]: a character-level convolutional neural network with several kinds of filter lengths in convolution layers and a highway network over characters. The long short-term memory network output layer is replaced by a softmax layer to predict the concept class. There are two kinds of network parameter settings (*CharCNN-Large* and *CharCNN-Small*) for this model. We choose *CharCNN-Small* to compare with our model according to its better performance in initial experiments.

6 Results and Discussion

In this section, we first validate whether our proposed mechanism can generate effective attention weights for the real-world text example. Then we compare the concept normalization accuracy performance of our proposed models versus alternative methods. To analyze the respective contributions of the attention mechanism and the auxiliary task supervision, we also compare *MTA-CharCNN* to reduced models. Finally, we evaluate the robustness of our model against noises.

6.1 Case Study about Attention Mechanisms

To understand the detailed effect of attention mechanisms on the concept normalization task, we first visualize attention weights over specific input character sequences (Fig. 4). In the first two rows of subfigures, the x-axis is the character sequence, and the y-axis is the attention weight value in the corresponding character position. The third-row subfigures show the performance of the present model (the best-saved model after n epochs) on the test set. We choose two real-world records in *ChMCN* to observe their attention weights transformation. The distribution of *A-CharCNN* attention weights (dashed red line) does not change distinctly during the training process. In addition, different input sequences generate similar attention weights after adequate training. Without the supervised information, the end-to-end attention structure of *A-CharCNN* could not visibly recognize the relatively important parts of different sequences. Conversely, attention weights of *MTA-CharCNN* (solid blue line) are obviously relevant to the specific sequence content. We choose two representative instances in the test set to visualize their attention weights. The first instance is a sequence that contains

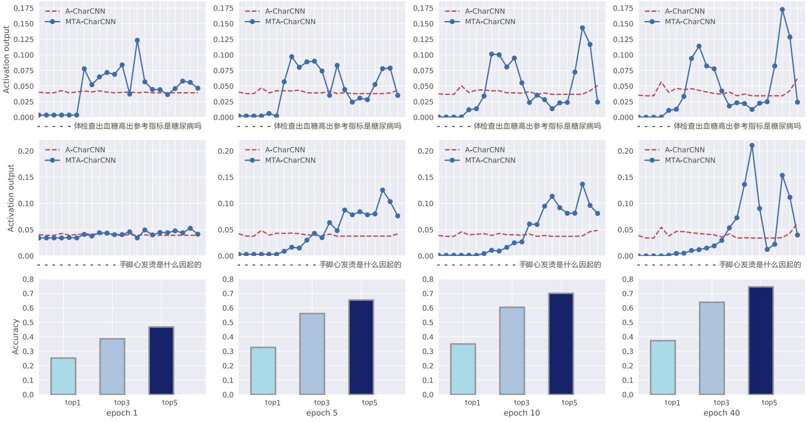


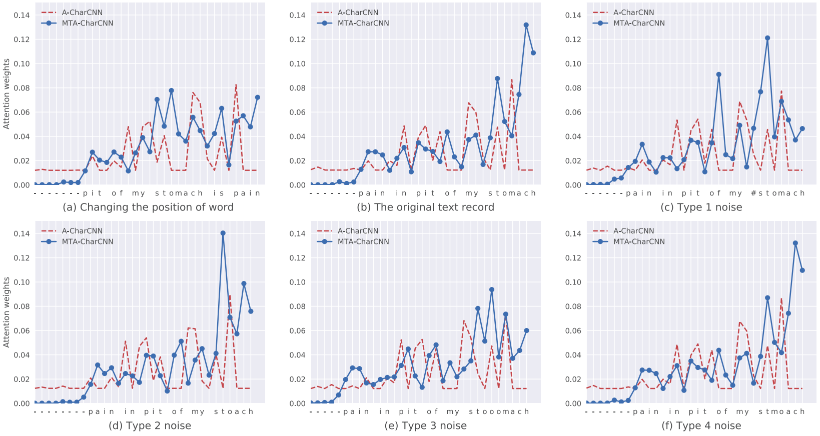
Fig. 4 The transformation of attention weights along with the training process. The first two rows of subfigures show the attention weight distribution on corresponding character sequences. The third-row subfigures show the present top-n (if the right answer appears in your top-n predictions) normalization accuracy performance on the tests set.

identifiable domain-related words (marked with ∗). The second instance contains no identifiable domain-related words, which suggests that it is more colloquial than the first instance. Chinese does not use the word divider (like the space in English). To help to understand the specific meaning of given cases, we show possible word segmentation results (use / as the word divider) and the corresponding approximated English sense of every word as follows.

- 体检/查出/血糖 */高出/参考指标/是/糖尿病 */吗
- physical examination/ checkup to find out/ **blood glucose level***/ higher than/ referenced criteria/ is/ **diabetes***/ auxiliary word
- 手脚心/发烫/是/什么/因起的
- hands palms and feet soles/ feel hot/ is/ what/ reason

As the first row subfigures show, attention weights on domain-related words 血糖 (**blood glucose level**) and 糖尿病 (**diabetes**) finally become relatively high after epochs of training. That is in accordance with the supervised label information. Besides, the word 查出 (checkup to find out) also gets higher weights, which is not in the predefined domain-related word set. It should be noted that 查出 is both semantically and morphologically similar to 检查 (checkup), which is defined to be a domain-related word. This result suggests that *MTA-CharCNN* is not only able to focus on words in the defined set, but also generate high weights on character combinations that is similar to target words. The second instance may help further explain this phenomenon.

The second sequence contains no words in the domain-related words set. However, 手脚心 (hands palms and feet soles) and 发烫 (feel hot) which are colloquial expressions about the patient's symptom, get higher attention weights. 因起 is a common typing error in Chinese. The original intent of the patient should be typing a domain-related word 引起 (*give rise to*, a word in domain-related words set) actually. We observe another attention distribution peak at this wrongly writ-



(a) Changing the position of word

(b) The original text record

(c) Type 1 noise

(d) Type 2 noise

(e) Type 3 noise

(f) Type 4 noise

Fig. 5 Attention weights distribution of text with types of noises. Subfigures (c)-(f) show the performance on four types noises: *Type 1* (adding "#" to the head of the word *stomach*), *Type 2* (deleting one character in the word *stomach*), *Type 3* (doubling one character in the word *stomach*), *Type 4* (changing one character's position with one of its adjacent letters) in the word *stomach*.

ten word, which indicates that our model attention can tolerate some OOV words caused by misspellings.

Detailed investigation of the robustness against attention generating is shown in Fig. 5. We chose an instance from the test set of *AskAPatient* and made several adjustments:

- (a) pit of my **stomach** is pain. (changing the position of the word *stomach*)
- (b) pain in pit of my **stomach**. (the original record)
- (c) pain in pit of my **#stomach**. (adding # to the head of the word *stomach*, Type 1 noise)
- (d) pain in pit of my **stoach**. (deleting a character in word *stomach*, Type 2 noise)
- (e) pain in pit of my **stooomach**. (doubling a character in the word *stomach*, Type 3 noise)
- (f) pain in pit of my **stmoach**. (exchanging the positions of two adjacent letters in the word *stomach*, Type 4 noise)

We visualize the attention weights for given text after enough epochs of training (until model convergence). For the curve of *A-CharCNN*, the result is similar to Fig. 4. We mainly focus on results of *MTA-CharCNN* here. Overall, attention weights are relatively low but not zero at word dividers (space). A possible reason is that they provide useful words' boundary information at least. Although the sequence padding and the word divider share the same supervised label in training, attention weights for paddings in the test set are almost all zero.

The original text *pain in pit of my stomach* should be mapped to the standard concept *Stomach ache*, where a key word in this phrase should be *stomach*. We observe a prominent peak of attention weights at the word *stomach* in Fig. 5-b. If we change the position of the word *stomach* (Fig. 5-a), the attention weights distribution still peaks at the word *stomach*. That indicates our model could generate particular focus on domain-related words among the input text sequence,

Table 3 The concept normalization accuracy performance of our proposed models (i.e., A-CharCNN, MT-CharCNN, and MTA-CharCNN) versus alternative methods. The accuracy performance reported in this table is the average over ten folds of every dataset.

Accuracy Datasets Models	TwADR-L	AskAPatient	ChMCN
Rand-RNN	0.3229	0.3791	0.3618
Emb-RNN	0.3529	0.3882	0.3203
Rand-CNN	0.4267	0.8013	0.3611
Emb-CNN	**0.4478**	0.8141	**0.3681**
CharConvNets	0.3847	0.7901	0.3434
CharCNN-Small	0.4386	**0.8264**	0.3618
A-CharCNN	0.4611	0.8422	0.3737
MT-CharCNN	0.4625	0.8417	0.3735
MTA-CharCNN	**0.4646**	**0.8465**	**0.3742**

regardless their position. Then we add character-level noises to the word *stomach*. As Fig. 5-c, 5-d, 5-e and 5-f show, though these noises change the original word into kinds of OOV words, *MTA-CharCNN* still generates higher character-level attention weights in the original position. Being robust to common OOV words is highly likely to be one of the reasons that *MTA-CharCNN* performs well in the concept normalization task.

In this subsection, we have shown that the auxiliary network with the attention supervision can generate the content-related and robust attention. We will test whether feeding these attention weights to the main task network can improve the performance of concept normalization in the next subsection.

6.2 Concept Normalization Accuracy

We compare our proposed model with alternative methods in the aspect of the concept normalization accuracy. As illustrated in Table 3, *Emb-CNN* and *CharCNN-Small* are relatively the best of the first two among alternative methods. *Emb-CNN* takes advantage of the pre-trained word embeddings, which is the additional information besides the training data. Noticed that finding a suitable and large embedding training corpus might be challenging when the task vocabulary contains large numbers of technical terms and low-frequency words, medical concept normalization is an example. *CharCNN-Small* is a character-level convolutional neural network model. The main difference between our model and it is *CharCNN-Small* does not have an attention mechanism. Comparatively, it holds a highway network structure to learn local features inside words better. *CharConvNets* performs poorly on three concept normalization datasets, though it also has the character-level convolution structure. A possible reason is that *CharConvNets* is initially designed to handle the task of relatively long text classification, whose convolution layers only have one kind of filter length. Besides, its network is deeper than *CharCNN-Small* and our models. However, the input text in the medical concept normalization task is shorter, and the total number of categories is much larger.

Our models *MTA-CharCNN*, *MT-CharCNN*, and *A-CharCNN* outperform all alternative methods on three datasets. Specifically, *MTA-CharCNN* achieves the

Table 4 The robustness evaluation of our proposed models (i.e., A-CharCNN, MT-CharCNN, and MTA-CharCNN) versus other representative word-level and character-level neural network models. Four types of noises are added to original datasets to simulate common text noises: *Type 1* (adding “#” to the head), *Type 2* (deleting one character), *Type 3*(doubling one character), *Type 4* (changing one character's position with one of its adjacent letters).

Accuracy / Models / Datasets	Rand-CNN	Emb-CNN	CharCNN-Small	A-CharCNN	MT-CharCNN	MTA-CharCNN
TwADR-L	0.4267	0.4478	0.4386	0.4611	0.4624	**0.4646**
TwADR-L *Type 1*	0.2123	0.1941	0.4337	**0.4400**	0.4288	0.4330
TwADR-L *Type 2*	0.2187	0.2018	0.3728	0.4169	0.4190	**0.4331**
TwADR-L *Type 3*	0.2404	0.1716	0.4253	0.4457	0.4510	**0.4513**
TwADR-L *Type 4*	0.2221	0.1997	0.3699	**0.4155**	0.4043	0.4113
Mean Accuracy	0.2233	0.1918	0.4005	0.4295	0.4258	**0.4322**
Mean Accuracy Drop	-0.2034	-0.2560	-0.0381	-0.0316	-0.0366	-0.0324
AskAPatient	0.8013	0.8141	0.8264	0.8422	0.8417	**0.8465**
AskAPatient *Type 1*	0.3642	0.3043	0.8218	**0.8394**	0.8332	0.8381
AskAPatient *Type 2*	0.3790	0.3045	0.7363	0.7818	**0.7825**	0.7815
AskAPatient *Type 3*	0.3678	0.2989	0.8154	0.8321	0.8264	**0.8336**
AskAPatient *Type 4*	0.3820	0.3172	0.7347	0.7689	0.7736	**0.7808**
Mean Accuracy	0.3732	0.3062	0.7771	0.8055	0.8039	**0.8085**
Mean Accuracy Drop	-0.4281	-0.5079	-0.0493	-0.0367	-0.0378	-0.0380

highest concept normalization accuracy of 46.46%, 84.65% and 37.42% on *TwADR-L*, *AskAPatient* and *ChMCN*. The performances of reduced models (i.e., *MT-CharCNN* and *A-CharCNN*) are similar. Through comparing *MTA-CharCNN* with the reduced models, controlled experiments validate the effectivenesses of the attention mechanism and the auxiliary task supervision. This result also suggests that the contributions of the two schemes could be incorporated.

6.3 Robustness against Noises

Informal expressions such like misspellings in social media messages could easily result in the appearance of Out-of-vocabulary words. We artificially added four types of noises to test sets of two English datasets for a robustness evaluation (see Table 4). It should be noted that the density of added noises is obviously higher than the real-world situation. Thus it is an extreme test. We did not construct the noisy dataset of *ChMCN* because Chinese character is ideogram, which makes it hard to define the character-level noise properly. The best performing alternative methods (i.e., *CharCNN-Small* and *Emb-CNN*) and another additional word-level model *Rand-CNN* are chosen to compare with our models.

Character-level models (i.e., *CharCNN-Small*, *MTA-CharCNN*, *MT-CharCNN*, and *A-CharCNN*) show clearly better robustness (the ability of tolerating four types noises) than word-level models (*Emb-CNN* and *Rand-CNN*). According to results in Table 4, the average accuracy drop (the difference between average accuracy over four noisy datasets and the original normalization accuracy) of the word-level model even reaches ten times of the character-level model's result. It is instinctive to some extent because the locally character-level change (four types of noises) easily bring about the OOV word, which is an absolute information loss for the word-level model. However, character-level models can still exploit the remaining character structure information. Besides, among character-level model-s, our models outperform *CharCNN-Small* on every noisy dataset. The attention mechanism and the auxiliary task supervision likely improve models' robustness against character-level noises as well.

7 Conclusion

In this paper, we proposed a multi-task character-level attentional neural network for the medical concept normalization task. Our model exploits the character-level encoding and the attention mechanism to alleviate the OOV word problem. The word morphological information inside the medical concept is effectively used as the auxiliary task supervision for generating attention weights. The attention weights are fed into the main task network, helping the downstream convolution focus on the domain-related characters. Experimental results show that our proposed model outperforms state-of-the-art methods on three real-world datasets. Besides, we use two reduced models to validate the effectivenesses of both the attention mechanism and the auxiliary task supervision. We further constructed four types of datasets added with common noises to validate the robustness of our model. In future work, we will investigate the way to generate the properly supervised label of character-level attention.

Acknowledgements This work was supported by the National Natural Science Foundation of China under Grants 61472423, 61432008, U1636220, and Beijing Municipal Natural Science Foundation under Grant No. 4172063.

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Batool R, Khattak AM, Kim Ts, Lee S (2013) Automatic extraction and mapping of discharge summary ' s concepts into SNOMED CT. In: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp 4195–4198
2. Baxter J (2000) A Model of Inductive Bias Learning. Journal of Artificial Intelligence Research 12:149–198
3. Benton A, Mitchell M, Hovy D (2017) Multitask Learning for Mental Health Conditions with Limited Social Media Data. In: Proceedings of Conference of the European Chapter of the Association for Computational Linguistics, vol 1, pp 152–162
4. Chawda VL, Mahalle VS (2017) Learning to recommend descriptive tags for health seekers using deep learning. In: 2017 International Conference on Inventive Systems and Control, pp 1–7
5. Chen H, Qi X, Yu L, Heng PA (2016) DCAN: Deep Contour-Aware Networks for Accurate Gland Segmentation. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp 2487–2496
6. Choi E, Bahadori MT, Kulas JA, Schuetz A, Stewart WF, Sun J (2016) RETAIN: An Interpretable Predictive Model for Healthcare using Reverse Time Attention Mechanism. In: Proceedings of the Neural Information Processing Systems Conference, pp 3504–3512
7. Conneau A, Schwenk H, Barrault L, Lecun Y (2016) Very deep convolutional networks for natural language processing. arXiv preprint arXiv:160601781
8. Dzmitry Bahdana, Bahdanau D, Cho K, Bengio Y (2014) Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:14090473

9. Golub D, He X (2016) Character-Level Question Answering with Attention. arXiv preprint arXiv:160400727
10. Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR (2012) Improving neural networks by preventing co-adaptation of feature detectors. In: arXiv preprint arXiv:1207.0580
11. Karimi S, Metke-Jimenez A, Kemp M, Wang C (2015) Cadec: a corpus of adverse drug event annotations. Biomedical Informatics 55:73–81
12. Kim Y (2014) Convolutional neural networks for sentence classification. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp 1746–1751
13. Kim Y, Jernite Y, Sontag D, Rush AM (2016) Character-Aware Neural Language Models. Proceedings of the AAAI Conference on Artificial Intelligence pp 2741–2749
14. Kingma DP, Ba JL (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:14126980
15. Larochelle H, Hinton G (2010) Learning to combine foveal glimpses with a third-order Boltzmann machine. In: Proceedings of the Neural Information Processing Systems Conference, pp 1243–1251
16. Leaman R, Doan RI, Lu Z (2013) DNorm: disease name normalization with pairwise learning to rank. Bioinformatics 29(22):2909–2917
17. Limsopatham N, Collier N (2015) Adapting phrase-based machine translation to normalise medical terms in social media messages. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp 1675–1680
18. Limsopatham N, Collier N (2016) Normalising medical concepts in social media texts by learning semantic representation. In: Proceedings of ACL, pp 1014–1023
19. Mikolov T, Chen K, Corrado G, Dean J (2013) Distributed representations of words and phrases and their compositionality. In: Proceedings of the Neural Information Processing Systems Conference, pp 3111–3119
20. Miyamoto Y, Cho K (2016) Gated Word-Character Recurrent Language Model. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp 1992–1997
21. Moeskops P, Wolterink JM, van der Velden BHM, Gilhuijs KGA, Leiner T, Viergever MA, Išgum I (2016) Deep Learning for Multi-Task Medical Image Segmentation in Multiple Modalities. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, Cham, pp 478–486
22. Nie L, Li T, Akbari M, Shen J, Chua TS (2014) WenZher: Comprehensive Vertical Search for Healthcare Domain. In: Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, pp 1245–1246
23. Nie L, Wang M, Zhang L, Yan S, Zhang B, Chua TS (2015) Disease Inference from Health-Related Questions via Sparse Deep Learning. IEEE Transactions on Knowledge and Data Engineering 27(8):2107–2119
24. Nie L, Zhang L, Yang Y, Wang M, Hong R, Chua TS (2015) Beyond Doctors: Future Health Prediction from Multimedia and Multimodal Observations. In: Proceedings of the 23rd ACM International Conference on Multimedia, pp 591–600

25. Nie L, Zhao YL, Akbari M, Shen J, Chua TS (2015) Bridging the vocabulary gap between health seekers and healthcare knowledge. IEEE Transactions on Knowledge and Data Engineering 27(2):396–409
26. O'Connor K, Pimpalkhute P, Nikfarjam A, Ginn R, Smith KL, Gonzalez G (2014) Pharmacovigilance on twitter? Mining tweets for adverse drug reactions. In: AMIA Annual Symposium Proceedings, pp 924–33
27. Robertson S (2009) The probabilistic relevance framework: BM25 and beyond. Foundations and Trends in Information Retrieval 3(4):333–389
28. dos Santos CN, Gatti M (2014) Deep convolutional neural networks for sentiment analysis of short texts. In: Proceedings of the 27th International Conference on Computational Linguistics, pp 69–78
29. Shen Y, Huang X (2016) Attention-Based Convolutional Neural Network for Semantic Relation Extraction. In: Proceedings of the International Conference on Computational Linguistics, pp 2526–2536
30. Shin B, Lee T, Choi JD (2016) Lexicon Integrated CNN Models with Attention for Sentiment Analysis. arXiv preprint arXiv:161006272
31. Sidana S, Mishra S, Amer-Yahia S, Clausel M, Amini MR (2016) Health monitoring on social media over time. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp 849–852
32. Søgaard A, Goldberg Y (2016) Deep multi-task learning with low level tasks supervised at lower layers. Proceedings of the Annual Meeting of the Association for Computational Linguistics pp 231–235
33. Song H, Rajan D, Thiagarajan JJ, Spanias A (2017) Attend and Diagnose: Clinical Time Series Analysis using Attention Models. arXiv preprint arXiv:171103905
34. Stanovsky G, Gruhl D, Mendes PN (2017) Recognizing Mentions of Adverse Drug Reaction in Social Media Using Knowledge-Infused Recurrent Models. In: Proceedings of Conference of the European Chapter of the Association for Computational Linguistics, vol 1, pp 142–151
35. Yang Z, Dhingra B, Yuan Y, Hu J, Cohen WW, Salakhutdinov R (2016) Words or Characters? Fine-grained Gating for Reading Comprehension. arXiv preprint arXiv:161101724
36. Yang Z, Salakhutdinov R, Cohen W (2016) Multi-Task Cross-Lingual Sequence Tagging from Scratch. arXiv preprint arXiv:160306270
37. Yu J (2016) Learning Sentence Embeddings with Auxiliary Tasks for Cross-Domain Sentiment Classification. Proceedings of the Conference on Empirical Methods in Natural Language Processing pp 236–246
38. Zhang D, Shen D (2012) Multi-modal multi-task learning for joint prediction of multiple regression and classification variables in Alzheimer's disease. NeuroImage 59(2):895–907
39. Zhang H, Li J, Ji Y, Yue H (2017) Understanding Subtitles by Character-Level Sequence-to-Sequence Learning. IEEE Transactions on Industrial Informatics 13(2):616–624
40. Zhang X, Zhao J, LeCun Y (2015) Character-level convolutional networks for text classification. In: Proceedings of the Neural Information Processing Systems Conference, pp 649–657
41. Zhang Z, Xie Y, Xing F, McGough M, Yang L (2017) MDNet: A Semantically and Visually Interpretable Medical Image Diagnosis Network. In: Proceedings

of the IEEE Conference on Computer Vision and Pattern Recognition, pp 6428–6436

42. Zhou J, Yuan L, Liu J, Ye J (2011) A multi-task learning formulation for predicting disease progression. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press, New York, New York, USA, pp 814–822