

自然言語を用いた構造化介護記録アプリの評価

Evaluation of Structured Caregiver Record App Using Natural Language

○¹ 田中 龍之介, ¹ 井上 創造
○¹ Ryunosuke Tanaka, ¹ Sozo Inoue
¹ 九州工業大学
¹ Kyushu Institute of Technology

Abstract: Using multi-label data for eight symptoms, we created a model to analogize symptoms from text and built an interactive care recording system. Six testers conducted the experiment assuming a real nursing home setting. A comparison between the model creation and the actual experiment was made. There is a need to consider pre-processing with the relationship between the data in mind. Examine the flow of new interactive systems.

1 はじめに

近年、世界的に高齢化が進んでいる中、高齢化社会の中で重要となるサービスの ICT 化が進められている。介護サービスの業務の一環として介護記録がある。介護士が行った業務を事細かく記入する必要がある、介護士の業務内でも大きく時間を割かれている現状である。高齢化が問題視されていると同時に介護業務の効率化が社会問題として注目されている。介護業務の効率化の一端として、介護記録の ICT 化によるペーパーレス化が進められているが全国的な普及には至っていない。介護記録の ICT 化には、注目が集まっており様々な研究が行われている。本稿では、介護記録の ICT 化を軸に音声入力による介護記録入力効率化と新しい介護業務の支援手法を検討する。

2 先行研究・関連研究

本稿の基盤となるシステムに関する研究と自然言語理解タスクとして使用したデータセットに関する研究を紹介する。

まず、本稿の基盤となるシステムに関する研究として同研究室 Tittaya ら [1] が提案している dialogue system care record (DSCR) が挙げられる。Tittaya らが提案する DSCR とは、スマートフォンをベースとして音声入力を使用して効率的に介護記録を入力するシステムである。音声認識を使用することで、従来の電子カルテ入力より文章化速度が向上し、満足度も上昇することが報告されている。DSCR は自然言語理解のためタスク指向型対話システムである。タスク型指向対話システムを構成するためには、タスクに見合ったコーパスが必要になる。Tittaya らが提案する DSCR は英語のコーパスを利用しており、日本ではプライバシーの観点からデータ収集が難しく同等の DSCR を構成することが困難である。

次に、自然言語理解タスクとして利用するデータセットについて紹介する。自然言語理解タスクとして若宮ら [2] が公開している NTCIR-13 MedWeb を利用する。NTCIR-13 MedWeb は 8 つの病気または症状のマルチラベルが

付与された擬似ツイートデータであり、日本語と中国語、英語の三つの言語をカバーするクロス言語・マルチラベルコーパスである。今回の実験では、日本語コーパスのみを使用し DSCR を構成する。

本稿では、タスクとして NTCIR-13 MedWeb を使用し、Tittaya らが提案する DSCR を基盤とした日本語をシステム入力とする DSCR を構築する。日本語でのタスクを構成することで、タスクの評価と日本の介護現場での導入を視野に入れた DSCR の提案を検討する。

3 提案手法

3.1 システム構成

システム構成図を下記に示す。本稿で作成したシステムは Tittaya らが提案している DSCR を基盤としており、TensorFlow[3] で NLU モデルを作成しており Web サーバーとして Flask[4] を使用している。NLU モデルと前処理については次節で詳しく解説する。

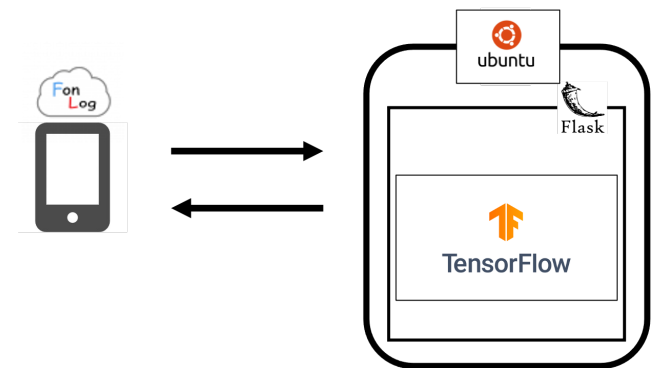


図 1: System Design 図

3.2 NLU モデル

前節でも述べたように NLU のモデル作成には Tensor-Flow を使用した。モデル作成時の前処理としてコーパスに対して以下の前処理を行った。

- 特殊文字 [!,?..etc] などの削除
- 数字の半角化とカタカナの全角化
- 大文字のアルファベットの小文字化
- Janome[5] を使用して形態素解析
- ストップワードの除去

形態素解析に関しては、品詞による分類を行い [名詞・動詞・助動詞・形容詞] のみを抽出し、名詞では固有名詞を除外している。また、形態素解析により活用系なしの状態にする。ストップワードに関しては、関連が薄いと考えられるものを手作業で除外している。前処理に関してはシステム入力の際も同様の処理を行なっている。

作成したモデルの要約を以下に示す。モデルは lstm を使用して作成しており、Embedding として Word2Vec[6] を用いて作成された Wikipedia[7] の学習済みモデルを使用している。

3.3 実験

実験は全て録画し以下の手順で行った。後日、書き起こしと手動によるクラスタリングを行った。

実験方法 8つのタスクから発話者に開示し、ランダムに選択してもらいシステム入力をしてもらう。

実験回数 一人当たり 5 回

実験人数 6 人

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 19)]	0
embedding (Embedding)	(None, 19, 300)	506700
lstm (LSTM)	(None, 128)	219648
dense (Dense)	(None, 8)	1032
Total params: 727,380		
Trainable params: 220,680		
Non-trainable params: 506,700		

図 2: model 図

4 実験結果

データの内訳を表 1 に示す。テキスト内容を表 2 に示す。ラベルごとの f1 値の図を図 3 に示す。モデル作成時の f1 値の図を図 4 に示す。モデル作成時の総合 f1 スコアが 70%であったのに対して、実際の実験では 20%ほど減少している。

表 1: 実験結果内訳

症状	テキスト数
Influenza	3
Diarrhea	5
Hayfever	2
Cough	4
Headache	5
Fever	6
Runnynose	5
Cold	2

表 2: 実験結果テキスト

テキスト
頭が痛い
熱っばい
寒気がする
鼻水が出る
お腹が痛い
洋服の袖がテカテカになって光っている人がいるけど多分鼻水を拭いたんだらうなって思ってる
頭痛が痛い
風邪引いた
牛乳飲んだら下痢になった
風邪引くと鼻水発熱頭痛あと咳が出ますよね
私はインフルエンザになりました
私今日下痢なんですよね
私最近花粉症気味です
最近咳がひどいです
今日は頭痛が痛いです
頭がガンガンします
インフルエンザかもしれません
目がかゆいです
咳が止まりません
熱が高くてしんどいです
今日熱っばい
今日お腹が痛い
今日は鼻水が出ます
今日は喉が痛いです
今日は頭が痛いです
鼻水が出て喉が痛いです
昨日から食べてるものが下痢になってお腹が痛いです
咳っぽくて喉が痛いです
鼻水がジュルジュル出て困ります
インフルエンザじゃないかと思うんですけどあの熱で頭が痛くて喉も痛いです

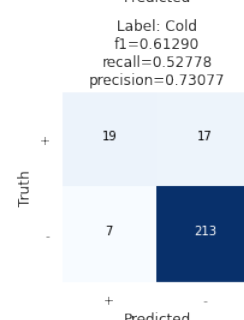
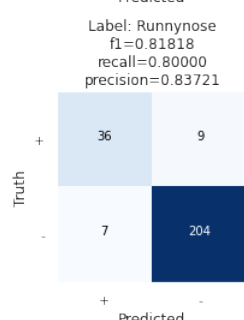
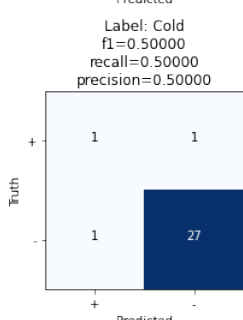
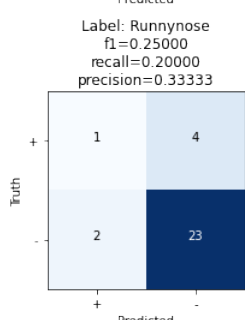
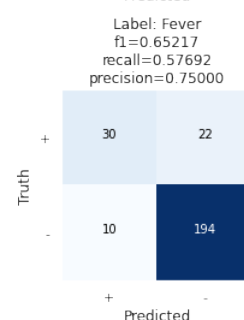
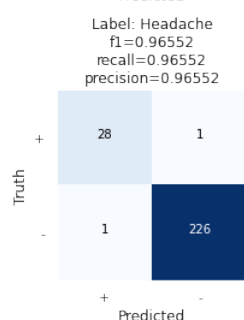
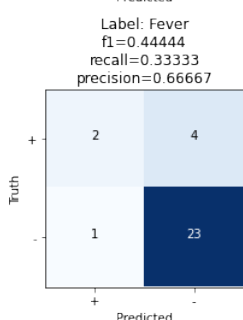
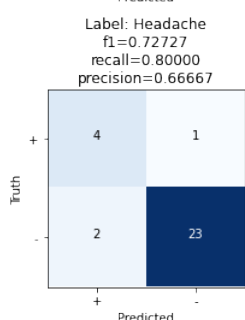
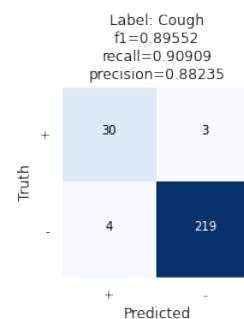
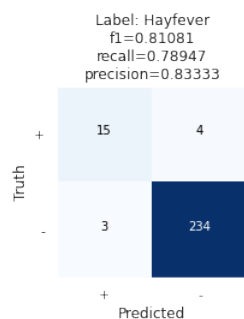
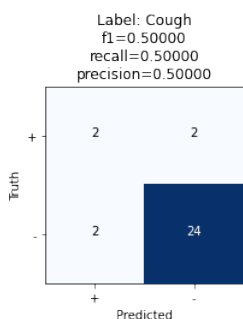
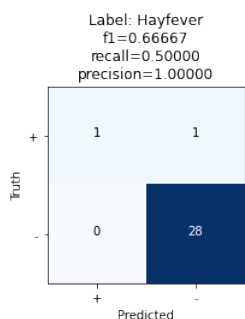
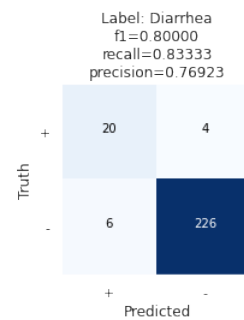
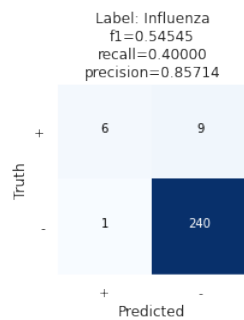
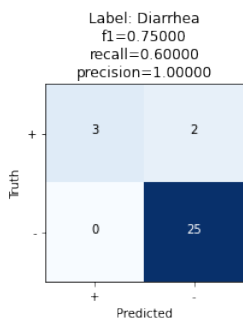
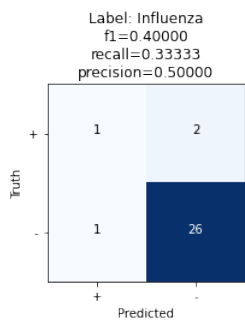


図 3: f1 値 テスト

図 4: f1 値 モデル作成時

5 考察

実験結果より、実験に対しての考察を行う。前提として、実験の思考回数が少ないため、モデルの正確な精度を測定することはできない。表1よりテキスト数が4以上のものに対してのみ評価を行う。まず音声による入力ということを踏まえて、以下の二通りの誤判定パターンが考えられる。

- 音声入力時の誤変換による誤判定
- NLU モデルによる誤判定

今回の実験では、音声入力時の誤変換による誤判定は存在しなかった。そのため発生している誤判定は全て NLU モデルによる誤判定であると考えられる。次に NLU モデルの誤判定のパターンとしては以下の理由の影響が高いと考えられる。

- データ欠落
- コーパス内におけるデータ間の関係

データ欠落について検討する。日本語の表現は多岐にわたる。コーパス内に存在しなかった語群は検討要素として含まれない。欠落した表現が、文に対して重要ではない場合は精度を向上させるが、文に対して重要である場合精度を低下させる。今回のモデル作成では、データ欠落が起きる可能性がある点として、Janome での形態素解析と Embedding、コーパス内に含まれない表現があげられる。

まず、Janome による形態素解析について検討する。今回は、品詞を限定することで文章の前処理を行なった。品詞による分類を行うと「発熱」と「熱っぽい」、「熱」という単語群はどれも熱に関することであるがどちらも別のベクトルとして保持される。テスト結果より「熱っぽい」といった表現を使用した場合、「発熱」のフラグは発生しない。これは、単語間の同一性を無視した結果であると考えられる。

次に Embedding に関しては、Wikipedia の情報を元に作成しているため、発話ベースのテキストであることを考慮していなかったことが原因ではないかと考えられる。コーパスがテキストベースなのに対して、発話は音声ベースである。話し言葉と書き言葉によるデータの違いが見られたのではないかと推測される。

コーパス内に含まれない表現は、モデル内に含まれていないため単語をベクトル表現にした際に欠落してしまう。そのため、コーパス内に含まれていない表現を使用することで、誤判定を起こす可能性がある。

次にコーパス内におけるデータ間の関係について検討する。今回のコーパスはマルチラベルである。データ内では症状間に「インフルエンザ」と「発熱」のように相関があるものと無いものが存在する。マルチラベル内で症状の相関が少ない「頭痛」と「下痢」に関しては、高い精度が出ている。これは、重複しにくい症状のため固有の表現が画一されているためであると考えられる。

6 議論

本項では、日本語の dialogue system care record (DSCR) を作成し、実験と考察を行なった。思考回数が少ないため、全てに対して検討を行うことができなかったが、症状間に相関があるものに対して NLU を行う場合では、前処理において工夫が必要である可能性を含んでいる。症状間の相関関係より、片方の症状からの類推が成り立つ可能性がある。

今後の展望として、片方の症状より類推をすることで追加の対話を作成することができる。また、症状から症状の度合いを質問する対話に切り替えるフローが作成できる。症状の度合いにより、介護士または看護師に対してアクションを行うフローが作成できる。今回のモデルは、NTCIR-13 MedWeb のデータのみを使用してモデルを作成している。今回作成したアプリケーションでは、テキストを送信することでフラグが返答されるシステムになっている。発話者に正しいフラグを入力してもらい、送信してもらうことでアノテーション済みのデータを獲得できるシステム構成に変更できる。

参考文献

- [1] Tittaya Mairittha , Nattaya Mairittha, 井上 創造:
”Evaluating a Spoken Dialogue System for Recording Systems of Nursing Care”, MDPI Sensors, Vol. 19, No. 3736, pp. 12 pages, 2019/08/29.
- [2] Shoko Wakamiya, Mizuki Morita, Yoshinobu Kano, Tomoko Ohkuma and Eiji Aramaki: Overview of the NTCIR-13 MedWeb Task, In Proceedings of the 13th NTCIR Conference on Evaluation of Information Access Technologies (NTCIR-13), pp. 40-49, 2017.
- [3] TensorFlow <https://www.tensorflow.org/overview?hl=ja>
- [4] Flask <https://flask.palletsprojects.com/en/1.1.x/>
- [5] Janome <https://mocabeta.github.io/janome/>
- [6] Word2Vec <https://code.google.com/archive/p/word2vec/>
- [7] Wikipedia <https://ja.wikipedia.org/wiki/>

連絡先

九州工業大学生命体工学研究科 井上創造研究室
E-mail: ryunosuke@sozolab.jp