

要介護認定における機械学習技術の活用

Application of machine learning technology for care-need assessment of the Long-term Care Insurance

神谷 達夫, 岡本 悦司, 奥村 貴史

要旨

保健医療福祉行政においては、データの収集・整理・分析に膨大な手作業が発生しており、その解決が望まれる。その解決のためには、近年著しく発展した機械学習等の人工知能(AI)技術の応用による効率化が期待されるが、行政内部の業務は一般に公開されておらず、効率化の検討が進んでいない。

本論文では、介護保険の要介護認定業務における認定審査会による二次判定作業に着目し、ケーススタディとして、認定審査会の二次判定作業に機械学習を活用することで業務効率化の可能性を検討した。

その結果、簡単なテキストマイニングの手法と機械学習アルゴリズムの 1 つであるロジスティック回帰分析のモデルを作成することができ、要介護認定における認定審査会の業務を効率化できる見通しができた。

本研究で用いた処理方法は、現在のコンピューター環境においては、容易に実現できる方法であり、この方法を普及させることにより、AI を用いて介護保険を効率化できると期待される。

キーワード: 機械学習、要介護認定

Keywords: machine learning, care-need assessment, Long-term Care Insurance

1. はじめに

人工知能(artificial intelligence, AI)は人間の経験に基づく判断や意志決定を学習により機械に代替させるものであり、自治体業務のなかで適用可能な業務に適用することによって以下の効果が期待できる。

- 1) 過去の経験知の活用
- 2) 決定の迅速化

3) 事務量と人件費の節約

本論文では、この AI の効果を保健医療福祉業務に適用した場合の効果を考えるためのケーススタディとして、要介護認定の認定審査業務の効率化を検討した。

本研究において着目した介護保険の要介護認定業務における非効率な要素は、以下の通りである。

1) 認定審査会の委員も異動等で頻繁に入れ代わり、その都度、素人からのスタートとなるため、認定にバラツキが生じるおそれがある。

2) 現在の要介護認定では、被保険者からの申請→訪問調査員による訪問調査→コンピューターによる一次判定→かかりつけ医意見書→認定審査会による二次判定と時間がかかる。

3) 認定審査会は多忙な専門職を通常 5 人から構成される合議体であり、毎月の日程調整に市町村担当者は忙殺される。また委員への日当等も支払われる。また認定審査会の権限は、白紙から要介護度を判定するのではなく、コンピューターによる一次判定を変更するか否か、という制約された権限であり、実際には大半の例で、一次判定がそのまま追認されている。この場合、結果的に二次判定により判定が遅延する。

これらの非効率な要素は、AI によってそれぞれ下記のように解決できるものと考えられる。

1) AI 技術の導入により、過去や他市町村の認定審査会に蓄積された夥しい経験知が継続的に生かせるため、異動等による委員交代による判断のバラツキをなくし正確かつ迅速な判断が可能となる。さらに無限に経験を蓄積することにより AI の判断力は向上してゆく。

2) AI 技術の導入により一次二次判定を同時実施できるので申請から認定までの時間を短縮できる。

3) 認定審査会業務に AI 技術を導入することにより事務量と費用を節約できる。

本研究では、上記のような AI 技術の導入の利点を示すために、要介護認定審査判定事例集[1] に示されている 32 例の判定結果から機械学習アルゴリズムの 1 つであるロジスティック回帰分析により、判定のためのモデルを作成した。

現時点では、この方法が最善の方法であると示すことができていない。ただ、本論文で示すような簡単な方法でも十分有用であり、AI の非専門家によっても AI 技術の恩恵を得ることができると考えられる。

2. 要介護認定への AI 技術の応用

2.1 機械学習の導入

本研究では、市町村役場における AI 適用可能な業務を検討した。その結果、AI を適用可能な業務は以下のような条件がある。

1) 定型かつ多量に発生する判断・決定業務

- 2) 基準が定められており、恣意的な決定は避けるべきこと
- 3) 過去の判断事例とその理由の集積がある

以上の条件を満たす業務として、介護保険の要介護認定業務が考えられたため、本研究ではケーススタディとして介護認定審査の効率化への機械学習技術の導入を検討した。

2.2 要介護認定審査の流れ

要介護認定審査とは、介護保険において被保険者がどれだけの介助を要するかを推定し、月単位の支給限度額を決定するために不可欠の業務である。

介護保険の受給を希望する者は、まず市町村に「要介護者であることを認めて欲しい」と申請する。申請を受けた市町村は、訓練された訪問調査員を派遣し、被保険者の現状を観察し、調査票に記入する。調査票（図 1）は客観的な設問で、マークシート方式で集計可能である。また調査票に記入できない情報は「特記事項」として文章として追記される。

認定調査票（基本調査）

1-1 麻痺等の有無について、あてはまる番号すべてに○印をつけてください。（複数回答可）

特記事項 → 1
1. ない 2. 左上肢 3. 右上肢 4. 左下肢 5. 右下肢 6. その他

1-2 関節の動く範囲の制限の有無について、あてはまる番号すべてに○印をつけてください。（複数回答可）

⇒ 1
1. ない 2. 肩関節 3. 肘関節 4. 股関節 5. 膝関節 6. 足関節 7. その他

2-1 登壇について、あてはまる番号に一つだけ○印をつけてください。

⇒ 2
1. つかまらぬ 2. 何かにつかまれば 3. できない

2-2 起き上がりについて、あてはまる番号に一つだけ○印をつけてください。

⇒ 2
1. つかまらぬ 2. 何かにつかまれば 3. できない

2-3 座位保持について、あてはまる番号に一つだけ○印をつけてください。

⇒ 2
1. できる 2. 自分の手で支えれば 3. 支えてもらえれば 4. できない

2-4 両足での立位保持について、あてはまる番号に一つだけ○印をつけてください。

⇒ 2
1. 支えなしで 2. 何か支えがあれば 3. できない

2-5 歩行について、あてはまる番号に一つだけ○印をつけてください。

⇒ 2
1. つかまらぬ 2. 何かにつかまれば 3. できない

2-6 移乗について、あてはまる番号に一つだけ○印をつけてください。

⇒ 2
1. できる 2. 見守り等 3. 一部介助 4. 全介助

2-7 移動について、あてはまる番号に一つだけ○印をつけてください。

⇒ 2
1. できる 2. 見守り等 3. 一部介助 4. 全介助

3-1 立ち上がりについて、あてはまる番号に一つだけ○印をつけてください。

⇒ 3
1. つかまらぬ 2. 何かにつかまれば 3. できない

3-2 片足での立位保持について、あてはまる番号に一つだけ○印をつけてください。

⇒ 3
1. 支えなしで 2. 何か支えがあれば 3. できない

図 1 介護認定の調査票[2]

このマークシートは、市町村においてコンピューター判定される。判定は、決定木と呼ばれる手法によって行われ、要介護時間が推計される。要介護度は推計要介護時間に応じて以下のように判定される(コンピューターによる一次判定)。

しかしながら一次判定がそのまま確定するのではなく、認定審査会にかけられ、一次判定を見直すべきかどうか審査され、場合によって、重度又は軽度に変更されることがある(二次判定)。

二次判定は、あくまで調査票に記載された特記事項と主治医意見書の記載内容のみを基に一次判定を見直すべきかどうかを評価するのであって決して一次判定から審査しなおすのではない。

認定審査委員会は医療福祉の専門職 5 人程度の合議体からなり、事前に事務局(市町村介護保険課)一次判定結果の資料を作成し、それと特記事項と主治医意見書のみを参照して審査する。最終的に判定を変えるか否かを決定するが、その理由を事務局が必ず議事録として作成する。

判定はしかし恣意的であってはならず、委員が適切な判定をくだせるよう事例集が刊行されている。事例集は全国の認定審査会の議事録より精選されたものである。本研究では、この事例集を題材にして要介護認定の機械学習による判定を試みた。

2.3 分析用データの前処理

今回は、AI の専門家でない者にも理解が容易でかつ容易に実現可能とするため、機械学習アルゴリズムの 1 つであるロジスティック回帰分析を用いた。

まず、要介護認定審査判定事例集[1]の内容をテキストファイル化した(図 2)。この時に、全ての情報をテキスト化するのではなく、要介護認定に対してどの項目が影響しているのかを検討し、テキストファイル化する項目を選定した。マークシートによる決定木の項目は要介護認定の判定見直しには影響していないことが分かったため、判定結果の要介護認定時間のみを利用していない。

このマークシートによる決定木の結果が判定結果に影響していないのは、その項目が不要ということではなく、決定木の内容が「要介護基準時間」の項目にすでに反映されているためである。

また、文章で記述された項目は全て認定見直しに利用されているため、全ての文章をテキストファイル化した。他には、主治医による意見書において、「症状としての安定性」と「介護の必要の程度に関する予後の見通し」をテキストファイル化した。また、このファイルは、項目と内容をタブで分離している。さらに、認定審査に影響のあった項目には 3 カラム目にアスタリスク*でマークをつけている(図 2)。図 2 の例では、2-7 の項目が認定審査に影響を与えていたため、2-7 の項目の 3 カラム目に*記号を記入している。

今回は PDF ファイルによって公開されている事例を手動により図 2 のようなテキストファイルに変更した。今回のように、手動での作成が可能であったのは、事例が 32 例と少なかったからである。本来はもっと多い事例を使用すべきであり、その場合は自動的に必要とするデータが取り出せるように元のデータを設定するべきであったと考えられる。

テキストファイル化した後、認定審査に影響のあった文を集め、MeCab[3]により形態素解析した。

形態素解析とは、文章を単語に分け、その単語の品詞等を分類することである。

形態素解析の結果、出現頻度が多く、記号や助詞でなく、「右」や「上」のようにその単語のみで意味が分からない語を除くと、認定審査に影響のある単語は図3のようになった。

この結果を用いて、各単語がそれぞれの事例に幾つ含まれるのか集計した。この集計のためには、専用のプログラムを用いた。このプログラムは、UNIX系OSで用いられるbshのスク립トで作成した。このbshスク립トは、比較的容易にプログラムを作成することができるが、習得がRPAと比較すると容易ではない。しかし、この作業は、事例集[1]をテキストファイル化することと異なり、定形作業であるため、プログラミングの非専門家がこの処理を実行するには、RPAのような簡易なプログラミング環境による自動化が容易である。

●性別・年齢	女性・81歳
●介護環境	居宅
●申請区分	新規
●一次判定結果	要支援
●二次判定結果	非該当（軽度変更）
●有効期間	なし
要介護認定基準時間 26.0	
2-5	腰痛があり、杖を使用することはある。
2-7	近くのデパートであれば、地下鉄を一人で利用。遠出は不安があり、家族が付き添う。 *
5-6	地域の高齢者クラブに参加しているが、書類等の理解は困難なため、

図2 テキストファイルの例

この例は、要介護認定審査判定事例集[1]の事例1を
テキストファイル化した状態を示している。

集計の結果と要介護認定基準時間、主治医の意見書に含まれている「症状の安定性」と「予後の見通し」を合わせ、判定モデルを作成するための元データのファイルを作成した。このファイルは、CSVファイルとし、表計算ソフトウェアで読み込み可能とした。

この先は、表計算ソフトウェア上の作業であり、この処理を可能とする者の数も十分多く、またRPAによる自動化にも適している。

表計算ソフトウェア上では、「症状の安定性」の項目が「不安定」であった場合を1、それ以外を0とし、「予後の見通し」が「悪化」であれば1、それ以外であれば0であるとして、数量化した。さらに、要介護認定の判定が認定審査会によって変更された場合判定結果は1、変更されなかった場合

の判定結果は 0 とした。この判定結果を目的変数とした。

説明変数は、図 3 で示す単語の出現数と要介護基準時間、数量化した「症状の安定性」、「予後の見通し」とした。

18	必要
18	介助
17	自分
17	関節
16	歩行
13	使用
12	低下
12	移動
12	できる
11	介護
11	見守り
10	入院
10	転倒
10	障害
10	現在
9	麻痺
9	痴呆
9	食事
9	下肢
8	可能

図 3 形態素解析の結果

出現数が多く、意味の無い単語や記号を削除した。

2.4 ロジスティック回帰分析

ロジスティック回帰分析は、数量データの説明変数と 2 群のカテゴリデータを目的変数とする回帰分析である。ロジスティック回帰分析においては、目的変数と説明変数を式(1)の関係で記述する。ただし、 y は目的変数、 x_n は説明変数、 n は説明変数の個数、 a_n は係数であり特に a_0 は定数項を示す。

$$y = \frac{1}{1 + e^{-(a_n x_n + \dots + a_1 x_1 + a_0)}} \quad (1)$$

ロジスティック回帰分析では、式(1)の係数 a を決定し、目的変数を得るためのモデルを構築する。ロジスティック回帰分析では、式(1)によって計算された尤度を求め、その尤度の対数である対数尤度が最大となる係数 a の組を求める。

本研究では係数 a の計算のために、表計算ソフトウェアによりロジスティック回帰分析を実行する。対数尤度の最大値を求めるためには、表計算ソフトウェアのソルバーを用いた。

ロジスティック回帰分析の結果得られたモデルによる元のデータの推定 100% 正しく推定できた。さらに、モデル適合検定の結果の p 値は 0.00682 となり、モデルの妥当性確認できた。この結果から、分析結果のモデルが問題なく介護認定結果に適合していることが分かる。

モデル適合検定は、対数尤度の合計を LL 、判定見直しの数を n_1 、判定維持の個数を n_2 とすると、統計的検定量は、式 2 ようになる。

$$k = n_1 \log n_1 + n_2 - n \log n$$

$$\text{統計的検定量} = -2LL - 2k \quad (2)$$

式 2 の統計的検定量で χ^2 乗検定により上側確率を求めることにより、 p 値を求めることができる。この場合の帰無仮説は、「求められたモデルが適合していない」であり、 p 値が十分小さいので、帰無仮説が棄却される。したがって、求められたモデルには、正当性があると考えられる。

表 1 オッズ比の大きいもの

項目	オッズ比
現在	3.007E+10
移動	9.948E+07
下肢	5.771E+05
予後の見通し	5.860E+03
低下	2.681E+03
歩行	8.314E+02
麻痺	1.506E+02
入院	1.039E+02
見守り	9.727E+01
介助	6.864E+01
必要	4.047E+01
要介護認定基準時間	1.074E+00

ロジスティック回帰分析のオッズ比は表 1 のようになった。項目の中で「予後の見通し」と「要介護基準時間」以外はそれぞれの単語を示している。さらに、表 1 は、オッズ比が 1 以上のものを示しており、これ以外の項目は 1 未満である。また、表 1 において、「必要」のオッズ比は 40 を超えている一方、「要介護認定時間」のオッズ比は 1 余りと急激に小さくなっている。このことから、今回のロジスティック回帰分析の結果から、「必要」よりオッズ比の大きい項目が支配的になっていると考

えられる。

表 1 から、「現在」や「予後の見通し」、「低下」のように時間的変化に関わる項目と「移動」や「下肢」、「歩行」のような動作に関わるような項目の影響が大きいということが分かる。したがって、表 1 の項目で示す語が含まれる場合、介護認定が認定審査会において見直される可能性が高いといえる。また、このことは定性的感覚とも大きく異ならないと思われる。

3. 考察

本研究では、保健医療福祉に AI 技術を導入するケーススタディとして、介護保険における要介護認定業務の効率化をとりあげた。前章では、認定審査の事例集[1]から取り出した情報から判定の見直しが発生するかどうか判断するためのモデルを作成した。その結果、十分有効なモデルを得ることができた。この結果から見て、本手法により要介護認定における認定審査会の業務を軽減できる可能性が確認された。実際に実用化するためには、もっと多数のデータによる検証が必要ではあるが、手法としては十分実用化できると考えられる。

このことは、簡単な AI 関連技術でも保健医療福祉業務に適用できる可能性のあることを示している。今回、ケーススタディに用いた手法は、簡単なテキストマイニングと機械学習アルゴリズムの 1 つであるロジスティック回帰分析である。この手法は、比較的容易に計算できる方法であり、非専門家によっても十分実施できる。ただし、現状ではその環境が完全には整っておらず、特に、処理方法のドキュメントの整備は重要である。

4. まとめ

本研究においては、各種 AI 関連技術を調査し、要介護認定審査のために適した方法を検討した。その結果、簡単なテキストマイニングの手法と機械学習アルゴリズムの 1 つであるロジスティック回帰分析のモデルを作成することができた。これにより、要介護認定における認定審査会の業務を軽減できる見通しができた。したがって、AI 関連技術は有望な解決策と考えられる。

また、本研究で用いた処理方法は、現在のコンピューター環境においては、容易に実現できる方法であり、この方法を知らしめることにより、AI を用いて保健医療福祉行政を効率化できると期待される。

《参考文献》

- (1) 要介護認定審査判定事例集(2004 年 11 月): <https://www.jcma.or.jp/news/association/1611.html> (2019.3.28 閲覧)
- (2) 厚生労働省, 障害者自立支援法案における支給決定・サービス利用プロセスについて, <https://www.mhlw.go.jp/topics/2005/04/tp0428-1c/02.html> (2020.1.31 閲覧)
- (3) MeCab: Yet Another Part-of-Speech and Morphological Analyzer <http://taku910.github.io/mecab/> (2019.3.28 閲覧)