Beyond Topics: Discovering Latent Healthcare Objectives from Event Sequences[*]

Adrian Caruana[1], Madhushi Bandara[1],
Daniel Catchpoole[1,2], and Paul J. Kennedy[1]

[1] Australian Artificial Intelligence Institute, Faculty of Engineering and IT
University of Technology Sydney, Sydney Australia
{adrian.caruana,madhushi.bandara,paul.kennedy}@uts.edu.au
[2] Biospecimen Research Services, The Children's Cancer Research Unit
The Children's Hospital at Westmead, Westmead 2145, NSW Australia
daniel.catchpoole@health.nsw.gov.au

Abstract. A meaningful understanding of clinical protocols and patient pathways helps improve healthcare outcomes. Electronic health records (EHR) reflect real-world treatment behaviours that are used to enhance healthcare management but present challenges; protocols and pathways are often loosely defined and with elements frequently not recorded in EHRs, complicating the enhancement. To solve this challenge, healthcare objectives associated with healthcare management activities can be indirectly observed in EHRs as latent topics. Topic models, such as Latent Dirichlet Allocation (LDA), are used to identify latent patterns in EHR data. However, they do not examine the ordered nature of EHR sequences, nor do they appraise individual events in isolation. Our novel approach, the Categorical Sequence Encoder (CaSE) addresses these shortcomings. The sequential nature of EHRs is captured by CaSE's event-level representations, revealing latent healthcare objectives. In synthetic EHR sequences, CaSE outperforms LDA by up to 37% at identifying healthcare objectives. In the real-world MIMIC-III dataset, CaSE identifies meaningful representations that could critically enhance protocol and pathway development.

Keywords: topic modelling · healthcare management · healthcare representation · sequence encoding · electronic health records

1 Introduction

A high-level understanding of healthcare patterns is critical for management optimisation, protocol development, and resource allocation in healthcare. Healthcare patterns can provide a schematic for understanding healthcare processes, protocols, and pathways. Understanding healthcare patterns is especially relevant to population-scale healthcare management as exemplified in the studies from the United States [16], Australia [1], and Canada [6].

Population-scale healthcare management tools are typically developed manually using clinical best practice guidelines. This presents a challenge for developing further management tools for other diseases, and maintaining them to reflect new or updated guidelines. Furthermore, these tools may not accurately reflect how patients are treated in practice, since treatment patterns typically vary with demographic or geographic factors. The development of population-scale healthcare management tools should be informed by electronic health records (EHR) since they reflect actual treatment behaviour across a healthcare system.

EHRs contain sequences of treatment events, such as diagnostic activities, drug prescriptions, or surgical procedures. These events are typically recorded using a categorical coding system. The International Classification of Diseases (ICD) codes are one example of such a system, and its ninth version, ICD-9 [19], contains over 13,000 unique codes. In contrast, the clinical protocols that are used to systematise patient care typically describe a set of guidelines, procedures, or objectives. Healthcare protocols vary by region and organisation, are not standardised, and consequently are not recorded in EHRs.

This paper defines an abstraction layer, referred to as 'healthcare objective', that encapsulates reasoning behind the formation of particular EHR sequences. Healthcare objectives group and abstract individual events in EHR sequences can facilitate analysis of EHR sequences for understanding healthcare patterns. An EHR sequence may consist of many latent healthcare objectives. For example, a 'diagnostic' objective may occur before a 'treatment' objective in the treatment of a broken limb. Healthcare objectives also influence the specific events which are recorded in an EHR sequence. In the same example, the specific ICD codes that are recorded will depend on the location or severity of the injury. Treatment codes may be associated with many distinct healthcare objectives, and a patient may express many latent healthcare objectives during a treatment sequence.

Topic models can identify groups of elements within a sequence that likely occurred due to a latent theme or state. In natural language processing (NLP), topic models determine the topic of a document from the words which it contained. Topic modelling has also been used for clinical pathway analysis in healthcare [10]. Topic models use collective, unordered, macro-scale views of sequences (e.g. entire documents in NLP, or entire hospital visits in EHR). They do not appraise individual elements in isolation, nor do they consider the sequential relationship between elements. In this paper, we transcend topic modelling to consider event-level associations of treatment events in pursuit of rich representations of healthcare objectives.

The contributions of this paper are:

1. Characterisation of healthcare objectives, and prerequisites for identifying them from EHR sequences.
2. Description of a synthetic data model for modelling of healthcare objectives.
3. Introduction of Categorical Sequence Encoding (CaSE), a generalised methodology for generating representations of categorical sequences.
4. Experimental validation of healthcare objective identification in synthetic and authentic EHR data.

This paper is organised as follows: Sec. 2 outlines healthcare objective characteristics and discusses related work, Sec. 3 details our synthetic data model and CaSE, Sec. 4 applies our methodology to synthetic and authentic EHR sequences, and Sec. 5 summarises our work and discusses some limitations and future work.

2   Preliminaries and Related Work

2.1   Prerequisites for Representing Healthcare Objectives

EHR sequences contain rich, yet sometimes loosely defined concepts and information. The relationship between healthcare events and their associated healthcare objective is complex since healthcare events could be associated with multiple healthcare objectives, resulting in out-of-order healthcare events and other relational complexities. Furthermore, EHRs seldom accompany any structured information concerning healthcare objectives, so it is not possible to learn this structure in a supervised manner. Approaches that seek to represent EHR sequences to reveal healthcare objectives must: appraise individual events in EHR sequences, consider the sequential nature of the data, and learn this relationship in a unsupervised manner.

2.2   Topic Modelling in Sequence Data

Natural language is structurally similar to EHR. In each case, data is recorded as a sequence of items (tokens in NLP, and events in EHR), each drawn from a discrete sample space (dictionary in NLP, and ICD codes in EHR). Long sequences may be delineated into smaller groups (paragraphs or documents in NLP, and hospital visit or departmental segregation in EHR).

Topic models are statistical models employed to discover latent topics in documents. Topic models assume that documents are about particular topics; keywords appearing more or less frequently because of the topic being discussed in the document. A significant method for topic modelling is Latent Dirichlet Allocation (LDA) [2], and is part of a larger family of Bayesian approaches to clustering grouped data [24]. A key limitation of LDA is the modelling of topics at a document level. Relationships that occur on a more minute lexical scale (such as a sentence or paragraph) are smaller than can be perceived by the document analysis performed by LDA. Further, the positional relationships between words, sentences, and paragraphs cannot be captured through the LDA.

Clinical pathway (CP) analysis is a healthcare research approach that systemically aims to manage patient care. Bayesian approaches [10, 11] have been employed to analyse EHR in pursuit of CP analysis. Like in NLP, Bayesian modelling of EHR does not directly consider the sequential nature of the data. This limits their capacity to reveal the dependencies between events in a sequence.

Sequence-based learning methods, such as long short-term memory (LSTM) [8], recurrent neural networks (RNN) [23], and most recently transformer networks [25], have shown success in several NLP tasks including topic modelling [5, 9, 17]. Unlike Bayesian approaches, these approaches consider the sequential nature of the data and learn item-level relationships of sequences.

Neural network-based approaches have been applied to learn representations of healthcare events. Choi *et al.* [4] learn visit-level representations in healthcare. In their approach, events in an entire sequence are aggregated into a binary vector, ignoring the sequential information carried by the healthcare sequence. Like topic models, this approach is not capable of determining patterns that occur on a finer scale than a hospital visit. Siamese networks [3] are neural networks that use the same parameters to encode pairs of inputs to the same feature space. They been used for text similarity and sentence embedding in NLP [18,22].

We propose to observe healthcare objectives in EHR sequences. Using a synthetic data model of healthcare objectives, we hypothesise that a sequence-based approach will distinguish treatment events in EHR sequences that are expressed by distinct healthcare objectives. This model can subsequently be applied to authentic EHR sequences to observe similar structures and illustrate other natural characteristics of healthcare objectives.

3   Methodology

3.1   Latent Treatment Groups in Electronic Health Records

Observed treatment events are categorical samples from the discrete set of all possible treatment events $\mathbb{E}$. Let x be a sample in an EHR dataset where $X \in \mathbb{E}$ such that $X \sim P$ with P a discrete probability distribution over the set $\mathbb{E}$.

In practice, P is not uniformly distributed and depends on the healthcare objective being applied. Each healthcare objective will alter the distribution of observed treatment events, resulting in a 'treatment group' g. Accounting for g, the distribution of treatment events is given by $P(X, G)$, and each event is sampled based on which treatment group g is being expressed from a set of possible treatment groups $\mathbb{G}$ where $g \in \mathbb{G}$. Given many treatment event observations, we seek to construct a representation $\hat{P}$ that approximates P. The goal of this methodology is to identify areas of high local density in $\hat{P}$ to infer the existence latent treatment groups $G \in \mathbb{G}$.

Synthetic Electronic Health Records We implement a synthetic data model defining a set of possible treatments $\mathbb{E}$, a set of treatment groups $\mathbb{G}$, and yields observations x drawn from a discrete probability distribution $P(X, G)$ where $X \in \mathbb{E}$ and $G \in \mathbb{G}$. The distribution of P for a particular treatment group g is $P(X \mid G{=}g) \sim \mathrm{Zipf}(\beta_g)$ with β_g indicating the parametrisation of the distribution. Additionally, each g corresponds to a random choice from the automorphism-group $\mathrm{Aut}(\mathbb{E})$ denoting all possible permutations over the set $\mathbb{E}$ as shown in Fig. 1. A patient sequence of $i \in [1, n]$ events is then defined as

$$x_1, \ x_2, \ x_3, \ ..., \ x_i, \ ..., \ x_n, \tag{1}$$

and at each element in the sequence the patient expresses a treatment group

$$g_1, \ g_2, \ g_3, \ ..., \ g_i, \ ..., \ g_n. \tag{2}$$

The 1st treatment group g_1 is determined by a random sample from $\mathbb{G}$ where $P(G) \sim$ Uniform over $\mathbb{G}$. The ith treatment group g_i is determined as either g_{i-1} or as a random sample from $\mathbb{G}$ where

$$P(G = g_i \mid Q = q) = \begin{cases} g \leftarrow P(G) & q < \alpha \\ g_{i-1} & q \geq \alpha \end{cases}, \tag{3}$$

where $\alpha \in [0, 1]$ and Q is a random variable following a continuous uniform distribution over the interval $[0, 1]$. α indicates the likelihood that a treatment group g changes between any two consecutive treatment events. Finally, the ith treatment event is determined by

$$x_i \leftarrow P(X \mid G = g_i). \tag{4}$$

This synthetic data model yields synthetic EHR datasets such that treatment events for patients exhibit a relationship to a latent treatment group (Eq. 4). However, the latent treatment group can at any point change to any other treatment group (Eq. 3), influencing the treatment events observed (Fig. 2). $\mathbb{E}$ is a set of categorical items, encoded as one-hot vectors $x_i \in \{0, 1\}^{|\mathbb{E}|}$.
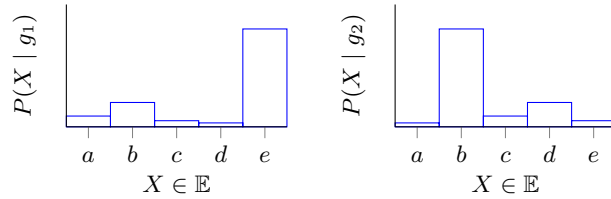

Fig. 1: The distribution of treatment events $X \in \{a, b, c, d, e\}$ given a latent treatment group g_i, with $|\mathbb{G}| = 2$ and $|\mathbb{E}| = 5$. Each group $G \in \mathbb{G}$ randomly permutes $\mathbb{E}$, with the distribution being Zipf.
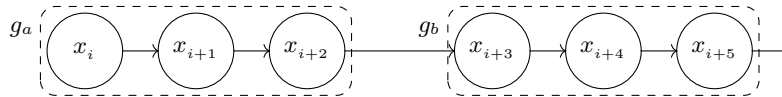

Fig. 2: The diagram depicts a sequence of observed treatment events x (circles), sequence progression (arrows), and the latent treatment groups g (rectangles).

MIMIC-III The MIMIC-III dataset [12] is a large, freely-available database comprising de-identified health-related data associated with over forty thousand patients who stayed in critical care units of the Beth Israel Deaconess Medical

Center between 2001 and 2012. We use sequences of ICD-9 [19] diagnosis codes of events observed by patients during hospital visits. Visits of sixteen or fewer events were removed. The dataset contains 46,520 patients, 58,976 separate hospital admissions, and 267,703 diagnosis events; from which 5,262 unique ICD-9 codes are observed. The codes form the set $\mathbb{E}$ and are encoded as one-hot vectors.

3.2 Treatment Group Representations

We propose Categorical Sequence Encoding (CaSE): a generalised method for representing sequences of categorical items. CaSE consists of a two-stage encoding process: First, a siamese network encodes categorical items. Subsequently, a transformer network generates an encoded representation of the sequence.

The siamese network learns a representation of categorical treatment events such that local neighbourhoods of events emerge. To do this, we employ a multilayer-perceptron (MLP), which we will refer to as **Cat2Vec**, that encodes an input vector $\mathbf{x}$ to a latent space vector $\mathbf{y}$ as $\mathbf{Cat2Vec}(\theta) : \mathbf{x} \to \mathbf{y}$. θ comprises the parameters fully-connected layers ℓ_1 and ℓ_2 from the input of dimension D to a hidden layer of dimension H and H to the encoding dimension N respectively. ℓ_1 may be repeated to consider multivariate categorical event data, in which case each repetition is concatenated before being passed forward to ℓ_2. ℓ_1 and ℓ_2 are activated using ReLU and sigmoid functions respectively.

To optimise the parameters θ, each training step encodes a pair of successive, one-hot encoded events $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$ from a sequence to yield vectors $\mathbf{y}_i$ and $\mathbf{y}_{i+1}$. The parameters θ are optimised using Adam stochastic optimisation [13] to minimise the mean-squared error as in (5). In effect, **Cat2Vec** learns to encode sequential events closely in the latent encoding space, as shown in Fig. 3a.

$$\min_{\theta} \frac{1}{N} \sum (\mathbf{y}_i - \mathbf{y}_{i+1})^2 \tag{5}$$

The transformer architecture from Vaswani *et al.* [25] is uniquely positioned to capture event-level detail due to the attention mechanism. In a self-attention configuration, the mechanism considers the relationship between all pairs of elements from a sequence. Furthermore, the architecture's use of positional encoding is also critical as it carries positional features of the input sequence.

We use the `Transformer` Module from PyTorch [20], which implements the architecture from Vaswani *et al.* [25]. We configure it as follows: Model depth is equivalent to the encoding dimension N from **Cat2Vec**. Other parameters – the number of heads H, sequence length L, feed-forward dimension F, and number of encoder E and decoder D layers – are determined experimentally. Masking of source or target sequences is not relevant to our learning task.

The transformer model is configured in an auto-encoder fashion [7], which we will refer to as **Seq2Seq**. The architecture contains two main sections, an encoder which produces an encoding from the input sequence, followed by a decoder, which can be used to produce a resultant sequence. In the auto-encoder configuration, the model learns to reproduce the input sequence from its internal learned representation of the input sequence ω (Fig. 3b).
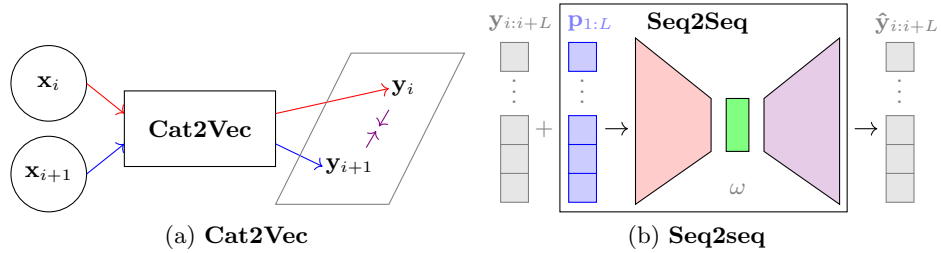
(a) **Cat2Vec**   (b) **Seq2seq**

Fig. 3: Fig. 3a depicts the siamese network, which learns to minimises the distance (violet) between adjacent events (red and blue) encoded to the latent vector space. Fig. 3b depicts the transformer model, which sums an input sequence **y** with a positional encoding vector **p** (blue). The model encodes (red) the sequence to produce an internal, encoded representation of the input sequence ω (green), before decoding (violet) ω to produce an output sequence $\hat{\mathbf{y}}$.

Seq2Seq is trained on sequences of L consecutive events, each encoded by **Cat2Vec**, for a given patient treatment sequence. The input **y** is a sequence of length L with N features for each event. **Seq2Seq** produces an encoded representation of the sequence ω, and decodes ω to generate a resultant sequence $\hat{\mathbf{y}}$. The parameters η of **Seq2Seq** are optimised using Adam stochastic optimisation [13] such that the mean-squared error is minimised (6). Once trained, the encoder stage of **Seq2Seq** produces an encoded representation ω of the sequence, which can then be used for subsequent analytics tasks.

$$\min_{\eta} \frac{1}{N} \sum \left(\mathbf{y}_{i:i+L} - \hat{\mathbf{y}}_{i:i+L} \right)^2 \tag{6}$$

4   Experiments

4.1   Treatment Groups in Synthetic Data

First, we perform a visual experiment to demonstrate CaSE identifying latent treatment groups in synthesised EHR sequences (Sec. 3.1). We configure the synthetic data model with $|\mathbb{G}| = 6$, $|\mathbb{E}| = 100$, $\alpha = 0.03$, and $\{\beta_g = 2 , \forall\, g \in \mathbb{G}\}$. Appendix 6.1 outlines further configuration parameters. Fig. 4 shows a 2D UMAP embedding [15] of **Cat2Vec** and **Seq2Seq** event representations. **Cat2Vec** captures the categorical nature of the $|\mathbb{E}| = 100$ events (Fig. 4a), while **Seq2Seq** groups these events into clusters of the $|\mathbb{G}| = 6$ treatment groups (Fig. 4b).

Next, we evaluate our treatment group identification approach via a clustering task. We vary the number of treatment groups $|\mathbb{G}|$ and the number of treatment events $|\mathbb{E}|$ in the synthetic data model configuration. LDA is used as a baseline. For a set of sequences, LDA yields a distribution over topics for each sequence. However, each patient treatment sequence expresses many topics throughout the
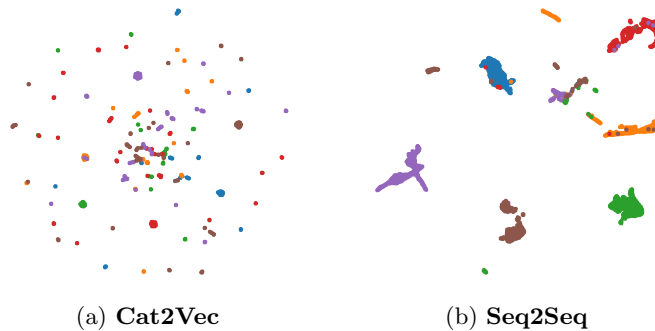
(a) **Cat2Vec**   (b) **Seq2Seq**

Fig. 4: UMAP visualisation of events encoded from treatment sequences in synthetic treatment data. Points represent treatment events, and colour depicts the treatment group expressed by the event.

sequence. A sliding window of 32 events[1] over each patient sequence is used to enable LDA to identify many treatment groups throughout a single sequence. Treatment group identification performance is first evaluated at the window-level for context, and at the event-level to compare against our method.

For CaSE, treatment groups are assigned using the HDBSCAN [14] clustering algorithm clustering with default configuration on the **Seq2Seq** encodings. A post-hoc clustering (PHC) acts on events that are classified as noise by HDBSCAN using a consensus of a local neighbourhood of the 20 nearest events[2] in the encodings. PHC is appropriate in our case as $P(G) \sim$ Uniform, (Sec. 3.1).

Table 1 shows the treatment group identification performance quantified by the Adjusted Mutual Information score [26]. LDA performs well at the window-level as expected, but suffers at the event-level task. In contrast, CaSE with PHC exceeds the event-level performance of LDA in all experiments. The results indicate that treatment group classification suffers as $|\mathbb{E}|$ decreases. This is because the task is more difficult for small values of $|\mathbb{E}|$ due to a phenomenon we refer to as 'cross-talk'. Cross-talk is inversely proportional to $|\mathbb{E}|$, and it describes the tendency for events to occur in more than one treatment group as the sample space of possible events is restricted. Appendix 6.2 provides further details on the CaSE and LDA implementations.

4.2   Group Representations in MIMIC-III

In Fig. 4, we observed CaSE clustering synthetic treatment events into treatment groups without prior knowledge of the treatment groups. We now observe how treatment events behave when applying CaSE to the MIMIC-III dataset (Sec. 3.1). We learn representations using events from individual patient treatment

---

[1] The sliding window length of 32 is the mean length $(1/\alpha)$ of treatment groups.

[2] Because HDBSCAN is nonlinear, PHC works best when the neighbourhood is small.

	$|\mathbb{E}|$	LDA (window)			LDA (event)			CaSE (event)			CaSE + PHC (event)		
		100	1000	10000	100	1000	10000	100	1000	10000	100	1000	10000
	6	0.866	0.894	0.840	0.655	0.656	0.627	0.803	0.962	0.960	**0.878**	**0.995**	**0.996**
$|\mathbb{G}|$	12	0.886	0.891	0.909	0.707	0.704	0.709	0.771	0.887	0.958	**0.821**	**0.976**	**0.990**
	24	0.899	0.908	0.931	0.749	0.750	0.774	0.705	0.845	0.878	**0.788**	**0.947**	**0.966**
	48	0.880	0.925	0.931	0.763	0.795	0.796	0.655	0.775	0.844	**0.783**	**0.878**	**0.953**

Table 1: Adjusted mutual information score of treatment group identification using LDA and our method as $|\mathbb{G}|$ and $|\mathbb{E}|$ vary. LDA works well in a window-level configuration, however this is not sufficient for event-level classification of healthcare objectives. Window-level LDA is included only for context.

sequences, where each event contains an ICD-9 code, and the ontological information associated with the code from the Clinical Classifications Software (CCS). The multivariate event data is used to contextualise the events and is encoded via **Cat2Vec** using the method described in Sec. 3.2. We visualise the representations using a 2D UMAP embedding. Appendix 6.3 contains configuration parameters.
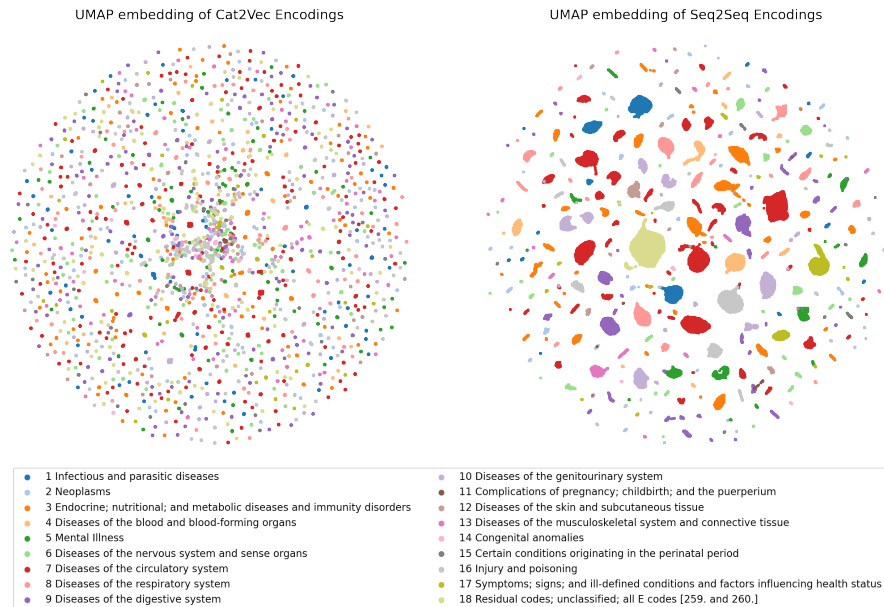
Fig. 5 illustrates three findings: 1. Like the experiment depicted in Fig. 4, **Cat2Vec** captures the categorical nature of treatment events, while the **Seq2Seq** representation captures the sequential context of EHR sequences. 2. When colouring events by their level-1 CCS categorisation, the **Seq2Seq** representation separately clusters different types of treatment events indicating different treatment groups (Fig. 5a). 3. When colouring events by their position in a treatment sequence, clusters of events express a dominant colour indicating inter-treatment group dynamics (Fig. 5b). These findings demonstrate that CaSE captures the features that are characteristic of healthcare objectives as prescribed in Sec. 2.1.
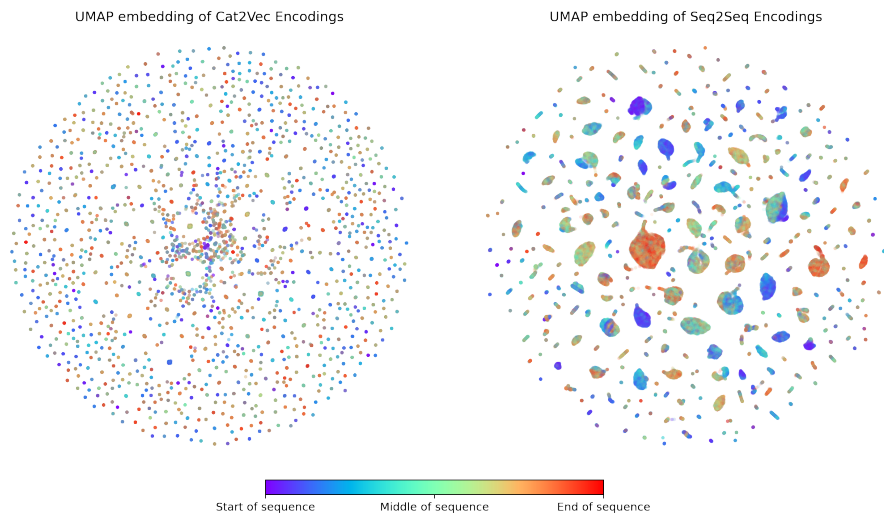
4.3   Implementation Details

Cat2Vec and **Seq2Seq** were each implemented in Python 3.9 using the python package PyTorch [20] V. 1.9.0. V. 0.8.27 of the HDBSCAN [14] python package was used for clustering. V. 0.24.2 of the Scikit-learn [21] python package was used for computing the adjusted mutual information metric and implementing LDA.

5   Conclusion and Future Work

This paper explores the task of using EHR to better inform population-scale healthcare management. Using EHR data to facilitate this understanding is valuable but challenging. We introduce the 'healthcare objective' to bridge between loosely defined healthcare management tools and well defined event-level EHR information. Sec. 3.1 describes the interaction between healthcare objectives and healthcare events in EHR sequences, and Sec. 3.2 outlines why the macro-scale approach of topic modelling can not capture the nuance of this

UMAP embedding of Cat2Vec Encodings   UMAP embedding of Seq2Seq Encodings

1 Infectious and parasitic diseases	10 Diseases of the genitourinary system
2 Neoplasms	11 Complications of pregnancy; childbirth; and the puerperium
3 Endocrine; nutritional; and metabolic diseases and immunity disorders	12 Diseases of the skin and subcutaneous tissue
4 Diseases of the blood and blood-forming organs	13 Diseases of the musculoskeletal system and connective tissue
5 Mental Illness	14 Congenital anomalies
6 Diseases of the nervous system and sense organs	15 Certain conditions originating in the perinatal period
7 Diseases of the circulatory system	16 Injury and poisoning
8 Diseases of the respiratory system	17 Symptoms; signs; and ill-defined conditions and factors influencing health status
9 Diseases of the digestive system	18 Residual codes; unclassified; all E codes [259. and 260.]

(a) MIMIC-III: by CCS

UMAP embedding of Cat2Vec Encodings   UMAP embedding of Seq2Seq Encodings

Start of sequence   Middle of sequence   End of sequence

(b) MIMIC-III: by sequence

Fig. 5: UMAP visualisation of **Cat2Vec** encodings (left) and **Seq2Seq** encodings (right) of events from MIMIC-III treatment sequences. In Fig. 5a, points are events coloured by their Level 1 categorisation in CCS. In Fig. 5b, points are events coloured by their position in their source treatment sequence.

interaction. This interaction results in 'treatment groups', which are groups of healthcare events that are thematically linked to a healthcare objective. Our methodology, Categorical Sequence Encoder (CaSE), considers the sequential nature of EHR and uses treatment groups to capture the event-level relationships and thematic links between categorical items in EHR data. We demonstrate that CaSE outperforms topic models at identifying healthcare objectives in our synthetic data experiment, and we establish the capacity of CaSE to identify temporal characteristics of healthcare objectives in MIMIC-III.

One limitation of our synthetic data model is the sampling of treatment events x and treatment groups g does not depend on their position i in the sequence (Eq. (3,4)). Future work will extend this approach to impose structure on how treatment events and treatment groups are sampled, and perform sensitivity analysis on model parameters. One further limitation of our work is that we were unable to evaluate healthcare objectives identified by CaSE in the MIMIC-III experiment because MIMIC-III does not contain any structured information concerning healthcare objectives. Acquiring meaningful healthcare objective labels aligned to EHR sequences is an ongoing challenge in our research.

References

1. Bergin, R.J., Whitfield, K., White, V., Milne, R.L., Emery, J.D., et al.: Optimal care pathways: A national policy to improve quality of cancer care and address inequalities in cancer outcomes. Journal of Cancer Policy **25**, 100245 (2020), https://doi.org/10.1016%2Fj.jcpo.2020.100245
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. the Journal of machine Learning research **3**, 993–1022 (2003)
3. Chicco, D.: In: Siamese Neural Networks: An Overview, pp. 73–94. Springer US (2020), https://doi.org/10.1007%2F978-1-0716-0826-5_3
4. Choi, E., Bahadori, M.T., Searles, E., Coffey, C., Thompson, M., Bost, J., et al.: Multi-layer representation learning for medical concepts. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM (2016), https://doi.org/10.1145%2F2939672.2939823
5. Dieng, A.B., Wang, C., Gao, J., Paisley, J.W.: Topicrnn: A recurrent neural network with long-range semantic dependency. In: ICLR (Poster) (2016)
6. Forster, K., Tsang, K., Li, S., Ieraci, L., Murray, P., Woltman, K.J., et al.: Can concordance between actual care received and a pathway map be measured on a population level in Ontario? a pilot study. Current Oncology **27**(1), 27–33 (2020), https://doi.org/10.3747%2Fco.27.5349
7. Hinton, G.E.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006), https://doi.org/10.1126%2Fscience.1127647
8. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997), https://doi.org/10.1162%2Fneco.1997.9.8.1735
9. Hoyle, A.M., Goel, P., Resnik, P.: Improving neural topic models using knowledge distillation. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics (2020), https://doi.org/10.18653%2Fv1%2F2020.emnlp-main.137
10. Huang, Z., Dong, W., Bath, P., Ji, L., Duan, H.: On mining latent treatment patterns from electronic medical records. Data Mining and Knowledge Discovery **29**(4), 914–949 (2014), https://doi.org/10.1007%2Fs10618-014-0381-y

11. Huang, Z., Ge, Z., Dong, W., He, K., Duan, H.: Probabilistic modeling personalized treatment pathways using electronic health records. Journal of Biomedical Informatics **86**, 33–48 (2018), https://doi.org/10.1016%2Fj.jbi.2018.08.004
12. Johnson, A., Pollard, T., Mark III, R.: Mimic-iii clinical database (version 1.4). Physio Net **10**, C2XW26 (2016)
13. Kingma, D.P., Ba, J.L.: Adam: A method for stochastic optimization. In: ICLR 2015 : International Conference on Learning Representations 2015 (2015)
14. McInnes, L., Healy, J., Astels, S.: hdbscan: Hierarchical density based clustering. The Journal of Open Source Software **2**(11), 205 (2017), https://doi.org/10.21105%2Fjoss.00205
15. McInnes, L., Healy, J., Melville, J.: Umap: Uniform manifold approximation and projection for dimension reduction. arXiv (2018), http://arxiv.org/abs/1802.03426
16. Mohler, J.L., Antonarakis, E.S., Armstrong, A.J., D'Amico, A.V., Davis, B.J., Dorff, T., et al.: Prostate cancer, version 2.2019, NCCN clinical practice guidelines in oncology. Journal of The National Comprehensive Cancer Network **17**(5), 479–505 (2019), https://doi.org/10.6004%2Fjnccn.2019.0023
17. Mueller, A., Dredze, M.: Fine-tuning encoders for improved monolingual and zero-shot polylingual neural topic modeling. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics (2021), https://doi.org/10.18653%2Fv1%2F2021.naacl-main.243
18. Neculoiu, P., Versteegh, M., Rotaru, M.: Learning text similarity with siamese recurrent networks. In: Proceedings of the 1st Workshop on Representation Learning for NLP. Association for Computational Linguistics (2016), https://doi.org/10.18653%2Fv1%2Fw16-1617
19. Organization, W.H.: International classification of diseases : [9th] ninth revision, basic tabulation list with alphabetic index
20. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al.: Pytorch: An imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019), http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf
21. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
22. Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using siamese BERT-networks. Association for Computational Linguistics (2019), https://doi.org/10.18653%2Fv1%2Fd19-1410
23. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**(6088), 533–536 (1986), https://doi.org/10.1038%2F323533a0
24. Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M.: Hierarchical dirichlet processes. Journal of the American Statistical Association **101**(476), 1566–1581 (2006), https://doi.org/10.1198%2F016214506000000302
25. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., et al.: Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. vol. 30, pp. 5998–6008 (2017)
26. Vinh, N.X., Epps, J., Bailey, J.: Information theoretic measures for clusterings comparison. ACM Press (2009), https://doi.org/10.1145%2F1553374.1553511

6 Appendix

6.1 Visual Representation of Synthetic Treatment Events

The synthetic data model synthesised treatment sequences for 100 patients, with 1,000 treatment events in each sequence. Treatment group representations are learned using the method described in Sec. 3.2, configuring **Cat2Vec** with $D = |\mathbb{E}|$ and $H = N = 8$, and configure **Seq2Seq** with $H = 8$, $L = 64$, $F = 64$, $E = 4$, and $D = 1$. as these parameters were found to produce consistent results during testing. Each model is trained until its loss converges. UMAP was configured with `n_neighbors=15` and `min_dist=0.1`.

6.2 Treatment Group Identification in Synthetic Data

Latent Dirichlet Allocation: Window vs Event LDA requires the number of components or topics (in our case, the number of treatment groups) as a hyper-parameter, and is provided by counting the number of unique treatment groups in the dataset $|\mathbb{G}|$.

The use of a sliding window over treatment sequences yields many sequences for which a minimum number of treatment events is expressed. Each sequence is aggregated into a fixed-length vector of length $|\mathbb{E}|$ with the frequency of each event in the window as values. LDA is used to transform the window into a fixed-length vector of length $|\mathbb{G}|$, and the component with the highest likelihood is taken as the inferred treatment group for the entire window. Event-level treatment group labels are determined as the modal topic of the treatment groups of all windows that a given treatment event appears.

The treatment group identification of LDA is then evaluated both at the event level and the window-level. At the window level, a majority of treatment group labels for each event within the window is taken as the label for the window, whereas at the event level each topic that has been assigned to a window by LDA is inherited by each event within the window. It is critical to note that EHR data seldom includes treatment group data, and so an approximation $|\hat{\mathbb{G}}|$ is required in practice when using LDA as it is not known a priori. When $|\hat{\mathbb{G}}| \neq |\mathbb{G}|$, treatment group identification performance suffers. In contrast, this is not the case for our approach as $|\hat{\mathbb{G}}|$ is approximated quantitatively by a clustering algorithm.

CaSE Configuration for Synthetic Data Treatment group representations are learned using the method described in Sec. 3.2, configuring **Cat2Vec** with $D = |\mathbb{E}|$ and $H = N = 128$, and configure **Seq2Seq** with $H = 128$, $L = 64$, $F = 64$, $E = 4$, and $D = 1$. Each model is trained until its loss converges.

Adjusted Mutual Information The Adjusted Mutual Information score is implemented as

$$AMI(y, \hat{y}) = \frac{MI(y, \hat{y}) - E(MI(y, \hat{y}))}{avg(H(y), H(\hat{y})) - E(MI(y, \hat{y}))} \tag{7}$$

where the clusters y are the treatment groups $\mathbb{G}$ that produced the event, the clusters $\hat{y}$ are the treatment groups identified by the analysis method, MI is the Mutual Information, and H is entropy.

6.3   CaSE Configuration for MIMIC-III

We configure **Cat2Vec** with $D = |\mathbb{E}|$, $H = 64$, and $N = 256$, and configure **Seq2Seq** with $H = 32$, $L = 16$, $F = 64$, $E = 4$, and $D = 1$. **Cat2Vec** is configured with two ℓ_1 layers: one for ICD-9 codes, and another for their CCS designation. Activations from each ℓ_1 are then concatenated before ℓ_2. Each model is trained until its loss converges. UMAP was configured with `n_neighbors=15` and `min_dist=0.1`.