Detect and Classify – Joint Span Detection and Classification for Health Outcomes

Micheal Abaho[1]   **Danushka Bollegala**[1,2*]   **Paula Williamson**[1]   **Susanna Dodd**[1]

[1]University of Liverpool, United Kingdom
[2]Amazon
{m.abaho,danushka,prw,shinds}@liverpool.ac.uk

Abstract

A health outcome is a measurement or an observation used to capture and assess the effect of a treatment. Automatic detection of health outcomes from text would undoubtedly speed up access to evidence necessary in healthcare decision making. Prior work on outcome detection has modelled this task as either (a) a *sequence labelling task*, where the goal is to detect which text spans describe health outcomes, or (b) a *classification task*, where the goal is to classify a text into a pre-defined set of categories depending on an outcome that is mentioned somewhere in that text. However, this decoupling of span detection and classification is problematic from a modelling perspective and ignores global structural correspondences between sentence-level and word-level information present in a given text. To address this, we propose a method that uses both word-level and sentence-level information to *simultaneously* perform outcome span detection and outcome type classification. In addition to injecting contextual information to hidden vectors, we use label attention to appropriately weight both word and sentence level information. Experimental results on several benchmark datasets for health outcome detection show that our proposed method consistently outperforms decoupled methods, reporting competitive results.

1 Introduction

Access to the best available evidence in context of patient's individual conditions enables healthcare professionals to administer optimal patient care (Demner-Fushman et al., 2006). Healthcare professionals identify outcomes as a fundamental part of the evidence they require to make decisions (van Aken et al., 2021). Williamson et al.

sentence	There were no significance between-group differences in the incidence of **wheezing** or **shortness of breath**
OSD	Outcomes: **wheezing**, **shortness of Breath**
OC	Outcome type: Physiological
Joint OSD & OC	Outcomes-Outcome type **wheezing**-Physiological **Shortness of Breath**-Physiological
sentence	Cumulative incidence and relative risks with 95% confidence intervals for **death from any cause**, **death from prostate cancer**, and metastasis were estimated in intention-to-treat and per-protocol analyses.
OSD	Outcomes: **death from any cause**, **death from prostate cancer**
OC	Outcome type: Mortality
Joint OSD & OC	Outcomes-Outcome type **death from any cause**-Mortality **death from prostate cancer**-Mortality

Table 1: Comparing the output of the three separate HOD tasks given two sample sentences. OSD retrieves the outcome spans, OC classifies the text span into a set of outcome types, and Joint OSD & OC retrieves outcomes and classifies them into outcome types.

(2017) define an outcome as a measurement or an observation used to capture and assess the effect of treatment such as assessment of side effects (risk) or effectiveness (benefits). With the rapid growth of literature that reports outcomes, researchers have acknowledged and addressed the need to automate the extraction of outcomes from systematic reviews (Jonnalagadda et al., 2015; Nye et al., 2018) and answering clinical questions (Demner-Fushman and Lin, 2007). Jin and Szolovits (2018) mention that automated Health Outcomes Detection (HOD)

could speed up the process of analysing and assessing the effectiveness of clinical interventions in Evidence Based Medicine (EBM; Sackett et al., 1996).

HOD has been conducted in the past as either an *Outcome Span Detection* (OSD) task, where we must detect a continuous span of tokens indicating a health outcome (Nye et al., 2018; Brockmeier et al., 2019) or as an *Outcome Classification* (OC) task, where the goal is to classify the text spans into a pre-defined set of categories (Wallace et al., 2016; Jin and Szolovits, 2018; Kiritchenko et al., 2010). However, the two tasks are highly correlated and local token-level information enables us to make accurate global sentence-level outcome predictions, and vice versa. An outcome type predicted for a text span in a sentence must be consistent with the other outcome spans detected from the same sentence, while the outcome spans detected from a sentence must be compatible with their outcome types. These mutual compatibility constraints between outcome spans and their classes will be lost in a decoupled approach, resulting in poor performance for both OSD and OC tasks.

Two illustrative examples in Table 1 show the distinction between the OSD, OC and Joint OSD & OC tasks. Specifically, in the first sentence, OSD extracts all outcomes i.e. *wheezing* and *shortness of breath*, OC classifies the text into an outcome type, Physiological, and then Joint OSD & OC extracts an outcome span and classifies it concurrently i.e. it extracts *wheezing* and also classifies it as a Physiological outcome. Motivated by the recent success in joint modelling of tasks such as aspect extraction (AE) and aspect sentiment classification (ASC), which together make a customer sentiment analysis task called Aspect Based Sentiment Analysis (ABSA; Xu et al., 2019), we model HOD as a joint task involving both OSD and OC. HOD can be formally defined as follows:

Health Outcome Detection (HOD): Given a sentence $s = w_1, \ldots, w_M$ extracted from a clinical trial abstract, the goal of HOD is to identify an outcome span $o_d = b_i, \ldots, b_N$ (i.e OSD), and subsequently predict a plausible outcome type $t(o_d) \in \mathcal{Y}$ for o_d (i.e. OC), where $1 \leq i \leq N \leq M$, and $\mathcal{Y}$ is a predefined set of outcome types.

We propose Label Context-aware Attention Model (LCAM), a sequence-to-sequence-to-set (SEQ2SEQ2SET) model, which uses a single encoder to represent an input sentence and two decoders – one for predicting the label for each word in OSD and another for predicting the outcome type in OC. LCAM is designed to jointly learn contextualised label attention-based distributions at word- and sentence-levels in order to capture which label/s a word or a sentence is more semantically related to. We call them contextualised because they are enriched by global contextual representations of the abstracts to which the sentences belongs. Label attention incorporates label sparsity information and hence semantic correlation between documents and labels.

A baseline BiLSTM and or clinically informed BERT$_{base}$ (Devlin et al., 2019) models are used at the encoding stage of our model and later for decoding with sigmoid prediction layers. We also use a multi-label prediction (MLP) layer for the two tasks (i.e. OSD and OC), with a relaxed constraint at token-level that ensures only the top (most relevant) prediction is retained, whereas all predicted (relevant) outcome types are retained at the sentence-level during OC. We use an MLP layer because some annotated outcomes belong to multiple outcome types. For example, *depression* belongs to both *"Physiological"* and *"Life-Impact"* outcome types.

HOD remains a challenging task due to the lack of a consensus on how outcomes should be reported and classified (Kahan et al., 2017). Dodd et al. (2018) recently built a taxonomy to standardise outcome classifications in clinical records, which has been used to annotate the EBM-COMET (Abaho et al., 2020) dataset. Following these recent developments, we use EBM-COMET to align outcome annotations in the evaluation dataset we use in our experiments (Dodd et al., 2018). Our main contributions in this work are summarised as follows[1]:

1. We propose the Label Context-aware Attention Model to simultaneously learn label-attention weighted representations at word- and sentence-level. These representations are then evaluated on a biomedical text mining task that extracts and classifies health outcomes (HOD).

2. We introduce a flexible, re-usable unsupervised text alignment approach that extracts parallel annotations from comparable datasets.

---

[1]Our Code and datasets are located at https://github.com/MichealAbaho/Label-Context-Aware-Attention-Model.git

We use this alignment for data augmentation in a low-resource setting.

3. We investigate the document-level contributions by a piece of text (e.g. an abstract) for predictions made at the token-level.

2 Related work

Joint training to achieve a dichotomy of tasks has previously been attempted, particularly for sequence labelling and sentence classification. Targeting Named Entity Recognition (NER) and Relation Extraction (RE), Chen et al. (2020) transfer BERT representations via a joint learning strategy to extract clinically relevant entities and their syntactic relationships. In their work, the joint learning models exhibit dramatic performance improvements over disjoint (standalone) models for the RE task. Our work differs from (Chen et al., 2020) in that we use attention layers prior to the first and second classification layers. Ma et al. (2017) train a sparse attention-based LSTM to learn context features extracted from a convolution neural network (CNN). The resulting hidden representations are used for label prediction at each time step for sequence labelling, and subsequently aggregated via average pooling to obtain a representation for sentence classification. The sparse constraint is strategically biased during weights assignment (i.e. important words are assigned larger weights compared to less important words).

Karimi et al. (2020) perform ABSA (Xu et al., 2019) by feeding a BERT architecture with a sentence $s = ([CLS], x_{1:j}, [SEP], x_{j+1:n}, [SEP])$, where $x_{1:j}$ is a sentence containing an aspect of a product, $x_{j+1:n}$ is a customer review sentence directed to the aspect and $[CLS]$ is a token not only indicating the beginning of a sequence, but also a sentiment polarity in the customer review about the aspect. They fine-tune a BERT model to conduct both aspect extraction and aspect sentiment classification. The above mentioned works tend to generate attention-based sentence-level representations that encapsulate the contribution each word would make in predicting sentence categories. We however generate label-inclined attention representations at word-level that can be used to effectively deduce word categories/labels. To the best of our knowledge, we are the first to perform a joint learning task that achieves MLP at two classification stages, token- and sentence-levels, while using only the top predictions at token level.

	EBM-COMET	EBM-NLP	EBM-COMET + EBM-NLP
# of Abstracts	300	5000	5300
# of sentences	5193	40092	45285
# of outcome labels	5	6	5
avg sentence length	21.0	26.0	25.0
# of Training sentences	4155	32074	36229
# of Testing sentences	1038	8018	9056

Table 2: Datasets statistics rounded off to zero decimal

3 Data

The absence of a standardised outcome classification systems prompted Nye et al. (2018) to annotate outcomes with an arbitrary selection of outcome type labels aligned to Medical Subject Headings (MeSH) vocabulary.[2] Moreover their outcome annotations have been discovered with flaws in recent work (Abaho et al., 2019), such as *statistical metrics* and *measurement tools* annotated as part of clinical outcomes e.g. *"mean arterial blood pressure"* instead of *"arterial blood pressure"*, *"Quality of life Questionnaire"* instead of *"Quality of life"*, *"Work-related stress scores"* instead of *"Work-related stress"*.

Motivated by the taxonomy proposed by Dodd et al. (2018) to standardise outcome classifications in electronic databases and inspired the annotation of EBM-COMET dataset (Abaho et al., 2020), we attempt to align EBM-NLP's arbitrary outcome classifications to standard outcome classifications that are proposed by Dodd et al. (2018). These standard classifications were found (after extensive analysis and testing) to provide sufficient granularity and scope of trial outcomes. We propose an unsupervised label alignment method to identify and align parallel annotations across the EBM-NLP and EBM-COMET. Additionally, we use the discovered semantic similarity between the two datasets and merge them in order to create a larger dataset for evaluating our joint learning approach. The merged dataset contains labels that follow the taxonomy proposed by Dodd et al. (2018). All three datasets are used during evaluation, with each one being randomly split into two, where 80% is retained for training and 20% for testing as shown in Table 2. We hypothesise that the merged dataset would improve performance we obtain on the original independent datasets.

---

[2]https://www.nlm.nih.gov/mesh

	Physiological	Mortality	Life-Impact									Resource-use			Adverse-effects
	P 0	P 1	P 25	P 26	P 27	P 28	P 29	P 30	P 31	P 32	P 33	P 34	P 35	P 36	P 38
Adverse-effects	0.0615	0.1532	0.1226	0.1893	0.2001	0.1348	0.1169	0.2555	0.2320	0.0897	0.1936	0.2561	0.1768	0.1043	**0.0562**
Mental	0.0387	0.1829	**0.0444**	**0.0928**	**0.1529**	**0.0623**	**0.0419**	**0.2214**	**0.1624**	**0.0624**	**0.1063**	**0.2537**	0.1955	0.1041	0.1904
Mortality	01330	**0.0187**	0.1722	0.2562	0.2563	0.2171	0.1821	0.2594	0.2956	0.1559	0.2349	0.2855	0.1976	0.1905	0.2082
Pain	0.0947	0.2310	0.1266	0.2181	0.1906	0.1316	0.1634	0.2662	0.2089	0.1290	0.2209	0.2770	0.2269	0.1422	0.2096
Physical	**0.0114**	0.1582	0.0698	0.1494	0.1878	0.1126	0.0788	0.2363	0.2059	0.0639	0.1461	0.2539	**0.1758**	**0.0761**	0.1803

Table 3: Cosine distance between representations of EBM-NLP labels (first column) and EBM-COMET labels (top and second row). EBM-COMET outcome type labels were drawn from the outcome domains defined in (Dodd et al., 2018) taxonomy. Due to space limitations, we denote these domains as P X such as P 0, P 1 etc. The taxonomy hierarchically categorised them into 5 outcome types which are accordingly included in the top row. Outcome domains definitions are, P 0-Physiological/clinical, P 1-Mortality/survival, P 25-Physical functioning, P 26-Social functioning, P 27-Role functioning, P 28-Emotional functioning/wellbeing, P 29-Cognitive functioning, P 30-Global quality of life, P 31-Perceived health status, P 32-Delivery of care, P 33-Personal circumstances, P 34-Economic, P 35-Hospital, P 36-Need for further intervention, P 37-Societal/carer burden, P 38-Adverse events/effects

3.1 Label alignment (LA) for Comparable Datasets

Given two datasets $\mathcal{S}$ and $\mathcal{T}$ with comparable content, with $\mathcal{S}$ containing x labels such that $L_s = \{l_s^1, \ldots, l_s^x\}$ and $\mathcal{T}$ containing y labels $L_t = \{l_t^1, \ldots, l_t^y\}$, we design LA to measure the similarity between each pair of labels (l_s, l_t).

For this purpose, we first create an embedding for each label l_s in a sentence $s(\in \mathcal{S})$ by applying mean pooling over the span of embeddings (extracted using pre-trained BioBERT (Lee et al., 2020)) for the tokens corresponding to an outcome annotated with l_s as shown in (1). Next, we average the embeddings of all outcome spans that are annotated with l_s in all sentences in $\mathcal{S}$ to generate an outcome type label embedding l_s. Likewise, we create an outcome type label embedding, l_t for each outcome type in the target dataset $\mathcal{T}$. After generating label embeddings for all outcome types in both $\mathcal{S}$ and $\mathcal{T}$, we compute the cosine similarity between each pair of l_s and l_t as the alignment score between each pair of labels l_s and l_t respectively.

$$\boldsymbol{O}_{l_s} = \frac{1}{d} \sum_{i}^{i+(d-1)} \text{Biobert}(w_i) \quad (1)$$

where O_{l_s}, is an outcome span annotated with outcome type label l_s, i and $i+(d-1)$ are the locations of the first and last words of the outcome span.

$$\boldsymbol{l}_s = \frac{1}{|l_s|} \sum_{1}^{|l_s|} \boldsymbol{O}_{l_s} \quad (2)$$

where $|l_s|$ is the number of outcome spans annotated with label l_s and $\boldsymbol{l}_s$ is label l_s embedding.

Table 3 shows the similarity scores for label pairs (l_s, l_t) across $\mathcal{S}$ (EBM-COMET) and $\mathcal{T}$ (EBM-NLP) respectively. For each label (which is an outcome domain) in EBM-COMET, we identify the EBM-NLP label which is most similar to it by searching for the least cosine distance across the entire column. After identifying those pairs that are most similar, we automatically replace outcome type labels in EBM-NLP with EBM-COMET outcome type labels as informed by the similarity measure.

Results show that Physiological outcomes (containing domain P 0) are similar to Physical outcomes and therefore the latter outcomes are labelled Physiological, Life-Impact outcomes are similar to Mental outcomes and therefore the latter outcomes are labelled Life-Impact. Mortality and Adverse-effects outcomes both remain unchanged because both categories exists in source and target datasets, and their respective outcomes are discovered to be similar. We evaluate the LCAM architecture on the resulting merged dataset, and additionally, evaluate the alignment approach by comparing the performances before and after merging.

4 Label Context-aware Attention Model

Figure 1 illustrates an end-to-end SEQ2SEQ2SET architecture of the LCAM model. It depicts a two-phased process to achieve classification at token and sentence level. In phase 1, input tokens are encoded into representations which are sent to a decoder (i.e. a sigmoid layer) to predict a label for each word, hence OSD. Subsequently, in phase 2, the token-level representations are used to generate individual outcome span representations, which are sent to another decoder (sigmoid layer) that is
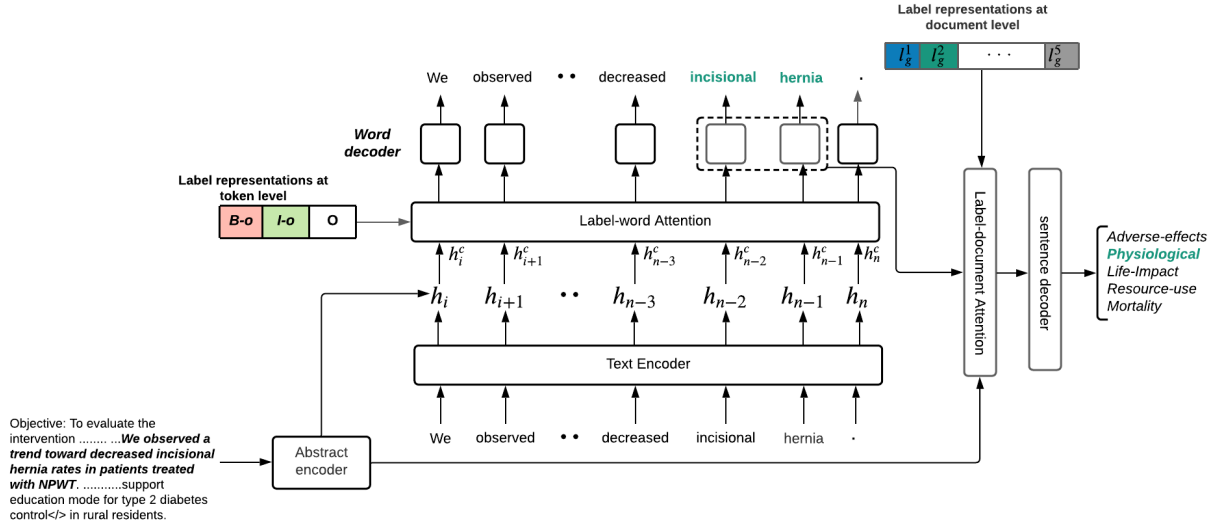
Figure 1: Illustration of the LCAM Architecture. It encodes a sequence of tokens of a sentence within an abstract, generates contextualised representations by adding a global representation of the abstract at word- and sentence-level. Two attention layers are used to aid generation of label-aware representations used to decode labels at word-level for OSD and sentence-level for OC.

used to predict the label/s for each outcome span, hence OC. We use MLP for the OC task because some outcomes are annotated with multiple outcome types. The pseudo code for LCAM is shown in the Supplementary.

4.1 Outcome Span Detection (OSD)

Given a set of sentences $\mathcal{S} = \{s_i\}_{i=1}^{|\mathcal{S}|}$ within an abstract a, each s_i having N words, $s_i = w_1, \ldots, w_N$, with each word tagged to a label l_w and use BIO tagging scheme (Sang and Veenstra, 1999). OSD aims to extract one or more outcome spans within s_i. For example, in Figure 1, OSD extracts the outcome span *"incisional hernia"* given the input sentence.

Encoder: In our OSD task setting, we initially implement a baseline LCAM using a BiLSTM to encode input tokens (that are represented by d-dimensional word embeddings we obtain using GloVe (Pennington et al., 2014[3]) into hidden representations for every word within an input sentence. We then consider generating each input words hidden representation using a pre-trained clinically informed BERT$_{\text{base}}$ model called BioBERT (Lee et al., 2020). The LCAM model learns (3),

$$\begin{aligned} \boldsymbol{h}_n &= \text{BiLSTM}(w_n), \\ \boldsymbol{h}_n &= \text{BioBERT}(w_n) \end{aligned} \quad (3)$$

where $w_n \in s_i$, $\boldsymbol{h}_n \in \mathbb{R}^{k \times 1}$ and k is the dimensionality of the hidden state. The upper equation under 3 is used for a BiLSTM Text encoder and the lower for a BioBERT one.

4.2 Abstract Hidden State Context

To make the hidden state representation context-aware, we add a compound representation of the abstract in which the sentence containing w_n belongs.

$$\boldsymbol{h}_n^c = \boldsymbol{h}_n + f(\text{AbsEncoder}(a)) \quad (4)$$

where f is a function computing the average pooled representation of the encoded abstract, $\text{AbsEncoder} \in \{\text{BiLSTM}, \text{BioBERT}\}$, $\text{AbsEncoder}(a) \in \mathbb{R}^{k \times |a|}$, $|a|$ is the length of the abstract (measured by the number of tokens contained in it) and $f(\text{AbsEncoder}(a)) \in \mathbb{R}^{k \times 1}$.

4.3 Label-word attention

We compute two different attention scores, the first is to enable the model pay appropriate attention to each word when generating the overall outcome span representation. Then the second attention score, is to allow the words interact with the labels in order to capture the semantic relation between them, hence making the representations more label-aware. To obtain the first attention vector $\boldsymbol{A}^{(1)}$, we use a self-attention mechanism (Al-Sabahi et al., 2018; Lin et al., 2017) that uses two weight parameters and a hyper parameter b that can be set

---
[3] https://github.com/stanfordnlp/GloVe

arbitrary,

$$A_n^{(1)} = \text{softmax}(\mathbf{W}\tanh(\mathbf{V}h_n^c)) \qquad (5)$$

where $\mathbf{W} \in \mathbb{R}^{|l_w| \times b}$, $\mathbf{V} \in \mathbb{R}^{b \times k}$ and $A^{(1)} \in \mathbb{R}^{|l_w| \times 1}$. $|l_w|$ is the number of token-level labels. Furthermore, we obtain a label-word attention vector $A^{(2)}$ using a trainable matrix $\mathbf{U} \in \mathbb{R}^{|l_w| \times k}$. Similar to the interaction function Du et al. (2019) use, this attention is computed in (6) as the dot product between the h_n^c and $\mathbf{U}$,

$$A_n^{(2)} = \mathbf{U}h_n^c \qquad (6)$$

where $A_n^{(2)} \in \mathbb{R}^{|l_w| \times 1}$.

Label-word representation The overall representation used by the decoder for classification of each token is obtained by merging the two attention distributions from the previous paragraphs as shown by (7),

$$E_n^{t_l} = A_n^{(1)}h_n^{c\top} + A_n^{(2)}h_n^{c\top} \qquad (7)$$

where $E_n^{t_l} \in \mathbb{R}^{|l_w| \times k}$, denotes the token-level (t_l) representation. The training objective is to maximise the probability of a singular ground truth label and minimise a cross-entropy loss,

$$L_{osd} = -\sum_{n=1}^{N}\sum_{i=1}^{|l_w|} y_{n,i}\log(\hat{y}_{n,i}). \qquad (8)$$

where N is number of tokens in a sentence, l_w is the number of labels.

4.4 Outcome Classification (OC)

OC predicts outcome types for the outcome spans extracted during OSD. Similar to what is done at token-level, we add an abstract representation (which is a mean pool of its token's representations) to add context to each tokens representation. An outcome span is represented by concatenating the vectors of its constituent words,

$$O_s = \bigoplus_{i=1}^{m}(E_i^{t_l} + f(\text{AbsEncoder}(a))) \qquad (9)$$

where m is the number of tokens contained in outcome span O_s. We adopt the aforementioned self-attention and label-word attention methods at sentence-level to aid extraction of an attention

based sentence-level representation of an outcome as follows:

$$E_s^{s_l} = A^{(1)}O_s + A^{(2)}O_s \qquad (10)$$

where $[A^{(1)}, A^{(2)}] \in \mathbb{R}^{|l_s| \times m}$, $O_s \in \mathbb{R}^{m \times k}$ and $s \geq 0$. Given an outcome span representation E^{s_l}, the training objective at sentence-level (s_l) is to maximize the probability of the set of terms,

$$\underset{\theta}{\text{argmax}}\,P(y = (l_s^1, l_s^2, ..., l_s^6) \in l_s|E^{s_l}; \theta) \quad (11)$$

$$L_{oc} = -\sum_{i=1}^{|l_s|} y_i\log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i) \qquad (12)$$

where $y_i \in \{0, 1\}$, $\hat{y}_i \in [0, 1]$ $l_s \in \{$Physiological, Mortality, Life-Impact, Resource-use, Adverse-effects$\}$. The overall joint model loss is:

$$L = L_{osd} + L_{oc} \qquad (13)$$

5 Experiments

The joint learning LCAM framework is evaluated on the three datasets discussed in section 3: the expertly annotated EBM-COMET, the EBM-NLP (Nye et al., 2018) and the merged dataset created by aligning (covered in section section 3) parallel annotations between EBM-NLP and EBM-COMET.

5.1 Implementation

For pre-processing the data, we first label each word in the sentences contained in an abstract with either one of $\{B, I, O\}$. Subsequently, to the end of each sentence, we include a list of outcome types corresponding to the outcome spans in the sentence. However, it is important to note that, not all sentences within an abstract had outcome spans. For example, the annotated sentence below contains outcome span "Incisional hernia" whose outcome label (Physiological) is placed at the end of the sentence.

"We/[O] observed/[O] a/[O] trend/[O] toward/[O] decreased/[O] incisional/[B-outcome] hernia/[I-outcome] rates/[O] in/[O] patients/[O] treated/[O] with/[O] NPWT/[O] ./[O]". [[Physiological]]

We tuned hyper-parameters using 20% of the training data of the merged dataset (EBM-NLP+EBM-COMET) as a development set. The optimal settings included, a batchsize of 64,

8714

Dataset	Model	Task setup	OSD			OC		
			P	R	F	P	R	F
EBM-COMET	Baseline	Joint	63.0	55.0	59.0	78.0	73.0	74.0
	BioBERT	Standalone	74.0	74.3	**74.2**	76.7	78.4	77.5
	SCIBERT	Standalone	72.3	72.9	72.6	76.3	78.1	77.2
	LCAM-BioBERT	Joint	73.0	64.0	68.0	83.0	76.0	**83.0**
EBM-NLP	Baseline	Joint	49.0	40.0	44.0	65.0	59.0	61.0
	BioBERT	Standalone	48.2	51.5	49.8	65.7	74.6	**69.9**
	SCIBERT	Standalone	48.5	49.7	49.1	64.2	66.5	65.3
	LCAM-BioBERT	Joint	57.0	49.0	**51.0**	67.0	65.0	66.0
EBM-COMET+EBM-NLP	Baseline	Joint	62.0	54.0	58.0	68.0	64.0	65.0
	BioBERT	Standalone	58.6	61.4	60.0	81.4	83.0	**82.2**
	SCIBERT	Standalone	56.2	62.3	59.1	73.4	75.7	74.5
	LCAM-BioBERT	Joint	61.0	61.0	**61.0**	78.0	72.0	75.0

Table 4: Outcome span detection (OSD) and Outcome classification (OC) results in terms of F1 on the three datasets. Baseline, is a LCAM architecture with a BiLSTM sequence encoder.

dropout of 0.1, 10 epochs, hidden state dimension for the BiLSTM and BioBERT encoders was set to 300 and 768 respectively. For the BioBERT model, we used features from BioBERT's ultimate layer, a practice that has been endorsed in the past (Naseem et al., 2020; Yoon et al., 2019; Hao et al., 2020). We use the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.001. Experiments were performed using a Titan RTX 24GB GPU.

5.2 Setup

The Joint setup is concurrent sequence labelling (OSD) and sequence classification (OC) whereas the standalone setup, is OSD and OC performed separately. The former is achieved using (a) a Baseline model, LCAM-BiLSTM (using a BiLSTM encoder) (b) LCAM-BioBERT (using BioBERT encoder), whereas the latter is achieved by fine-tuning the original (c) BioBERT and (d) SciBERT (Beltagy et al., 2019) models. Our datasets are novel in the sense that the outcome type labels of the outcomes are drawn from Dodd et al. (2018) taxonomy, which is not the basis of prior outcome annotations such as the EBM-NLP dataset. The models were evaluated on the tasks by reporting the macro-averaged F1. For the standalone models, we use token-classification and text-classification fine-tuning scripts provided by Huggingface (Wolf et al., 2020) for OSD and OC respectively. Inaddition to the macro-F1, we visualise ranking metrics pertaining to MLP, in order to compare our model to related work for MLP. The metrics of focus include precision at top n P@n (fraction of the top n

predictions that is present in the ground truth) and Normalized Discounted Cumulated Gain at top n (nDCG@n).

5.3 Results

The first set of results we report in Table 4 are based on the independent test sets (Table 2) for each of the datasets. The joint LCAM-BioBERT and standalone BioBERT models are not only competitive but they consistently outperform the baseline model for both OSD and OC tasks. We observe the LCAM-BioBERT model outperform the other models in the OSD experiments for the last two datasets in Table 4. On the other hand, the standalone BioBERT model achieves higher F1 scores for the last two datasets in the OC task.

5.3.1 Impact of the abstract context injection and Label attention

As shown in Table 6, the performance deteriorates (with respect to the results reported in Table 4) without the attention layers ("- Attention") by averagely 10% for OSD and 11.3% for OC. Similarly, exclusion of the abstract representation ("- Abstract") leads to an average performance decline of 4.3% for OSD and 2.7% for OC. As observed the decline resulting from "- Abstract" is less significant than that resulting from "- Attention" for both OSD and OC tasks.

This decline explains the significant impact of both (1) the semantic relational information between both tokens and labels as well as outcome spans and labels gathered by the attention mechanism, (2) information from the text surrounding

	OSD			OC		
LCAM-BioBERT	P	R	F	P	R	F
EBM-COMET	73.0/83.0	64.0/64.0	68.0/71.0	83.0/90.0	76.0/80.0	83.0/84.0
EBM-NLP	57.0/60.0	49.0/47.0	51.0/53.0	65.0/76.0	65.0/72.0	64.0/74.0

Table 5: Effect of dataset merging via label alignment. For each dataset, we report the performance on its test split obtained by LCAM-BioBERT trained on the corresponding train split (shown on the left side of /) vs. on the merger of the train splits of EBM-COMET and EBM-NLP (shown on the right side of /).
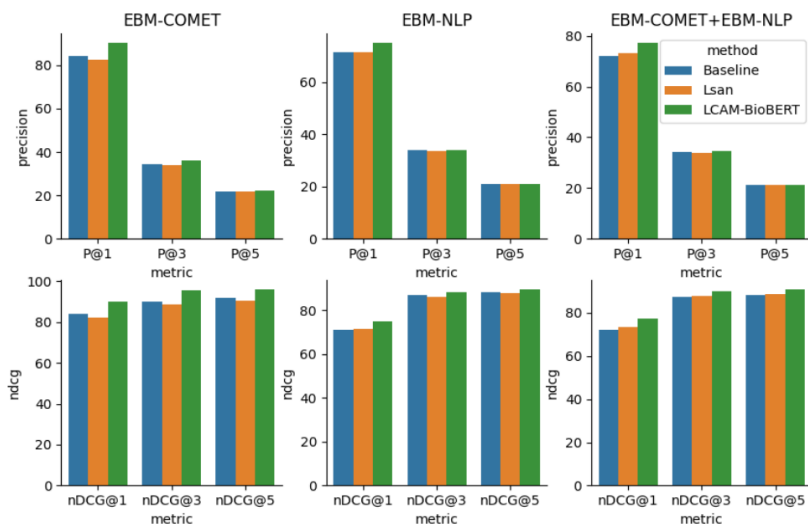

Figure 2: P@n and nDCG@n for three datasets

	LCAM	OSD(F)	OC(F)
EBM-COMET	- Attention	-10.0	-12.0
	- Abstract	-3.0	-5.0
EBM-NLP	- Attention	-9.0	-7.0
	- Abstract	-7.0	-2.0
EBM-COMET	- Attention	-11.0	-15.0
+EBM-NLP	- Abstract	-3.0	-1.0

Table 6: OSD and OC performance percentage decline when either the attention mechanism or the abstract representation are eliminated from the joint learning model (LCAM-BioBERT).

a token or an outcome span embedded into an abstract representation. This therefore justifies inclusion of both these components.

To evaluate the proposed label alignment method (subsection 3.1), we train a model using the aligned dataset (EBM-COMET+EBM-NLP) and evaluate it on the test sets of the original datasets in Table 5. We see significant improvements in F-scores for OSD in both EBM-COMET and EBM-NLP. Additionally, for OC, we see a significant improvement in F-score on EBM-NLP dataset and a slight im-

provement in F-score on the EBM-COMET dataset. Overall, this result shows that the proposed label alignment method enables us to improve performance for both OSD and OC tasks.

To further evaluate the LCAM-BioBERT model, we focus on the OC task results alone where the classifier returns the outcome types given an outcome span, and compare MLP performance to the baseline and another related MLP model, label-specific attention network (LSAN) (Xiao et al., 2019), that learns biLSTM representations for multi-label classification of sentences. For comparison, we compute P@n and nDCG@n using formulas similar to (Xiao et al., 2019). As illustrated in Figure 2, the LCAM model outperforms its counterparts for all datasets, and most notably for P@1. Our joint BiLSTM baseline model performs comparably with LSAN, and indeed outperforms it on the EBM-COMET dataset for P@1, nDCG@1 and nDCG@3. We attribute LCAMs superior performance to (1) Using a domain-specific (biomedical) language representation model (BioBERT) at its encoding layer, (2) Applying label-specific attention prior to classifying a token as well as before classifying the mean pooled representation of an

	Example Input sentence	Predicted labels P@1	Predicted labels P@2
Ground truth	The primary outcomes were **hospitalised death**[1], **severe disability**[2] at 15 months of age, **neonatal behavioural neurological**[3] assessment (nbna) score at 28 days of age, and Bayley scales of **infant development**[4] (BSID) score (including **mental development**[5] index (mdi) score and **psychomotor development**[6] index (pdi) score) at 15 months of age at follow-up.	1. Mortality 2. Life-Impact 3. Life-Impact 4. Life-Impact 5. Life-Impact 6. Life-Impact	
LCAM Output	The primary outcomes were hospitalised **death**[1], **severe**[2] **disability**[3] at 15 months of age, neonatal behavioural neurological assessment (nbna) score at 28 days of age, and Bayley scales of infant development (BSID) score (including mental **development**[4] index (mdi) score and **psychomotor development**[5] index (pdi) score) at 15 months of age at follow-up.	1. Mortality 2. Physiological 3. Life-Impact 4. Life-Impact 5. Life-Impact	
Ground truth	These results confirm retrospective studies and add that histopathology subtype is a strong determinant of **disease-free survival (DFS)**[1], in resected MAGE-A3-positive MSCLC.	1. Physiological	1. Mortality
LCAM Output	These results confirm retrospective studies and add that histopathology subtype is a strong determinant of **disease-free survival**[1] (DFS), in resected MAGE-A3-positive MSCLC.	1. Physiological	1. Mortality
Ground truth	The duration of total **hospital stay**[1], and **postoperative hospital stay**[2] in the ag (10.86 +/- 5.64, 5.69 +/- 4.55) d were significantly shorter than that in the cg (.10.86 +/- 5.64, 5.09 +/- 4.55) d (p=0.01, p=0.01))	1. Resource-use 2. Resource-use	
LCAM Output	The duration of total **hospital**[1] **stay**[2], and **postoperative**[3] **hospital stay**[4] in the ag (10.86 +/- 5.64, 5.69 +/- 4.55) d were significantly shorter than that in the cg (.10.86 +/- 5.64, 5.09 +/- 4.55) d (p=0.01, p=0.01))	1. Resource-use 2. Physiological 3. Physiological 4. Resource-use	

Table 7: Sample error predictions made by the joint learning model, with coloured words representing the outcome phrase (both in ground truth and output) and the colours representing different outcome types which are output. For multi-label predictions, we include P@1 and P@2 to indicate the top most predictions for the outcome phrase in question such as in example 2.

outcome span and finally (3) injecting global contextual knowledge from the abstract into the token and document (outcome-span) representations.

5.3.2 Error Analysis

We review a few sample instances that exhibit the mistakes the joint LCAM model makes in the OSD and OC tasks in Table 7.

OSD errors: We observe the model partially detecting outcome phrases e.g. In Example 1, it detects death instead of hospitalised death, development instead of mental development, and in Example 2, it does not detect "(DFS)" as apart of the outcome phrase. Additionally, it completely misses some outcomes such as infant development in Example 1.

OC errors: Incorrect token-level predictions will most likely result into incorrect outcome classification. In Example 1, Instead of severe disability, the model detects "severe" as an outcome and "disability" as a separate outcome and classifies them as Physiological and Life-Impact respectively. Similarly, in Example 3, both outcomes are misclassified because at token level multiple outcomes are detected rather than one, hospital and stay rather than hospital stay, postoperative and hospital stay

rather than postoperative hospital stay.

6 Conclusion

We proposed a method to jointly detect outcome spans and types using a label attention approach. Moreover, we proposed a method to align multiple comparable datasets to train a reliable outcome classifier. Given real-world scenarios where it is often impractical or computationally demanding to build a model for each and every single task, our experimental results demonstrate the effectiveness of an approach that simultaneously (jointly) achieves two different task without compromising the performance of the individual tasks when decoupled.

7 Ethical Considerations

Joint learning can have multiple applications, where multiple tasks are simultaneously achieved whilst preserving (or even improving) standalone performance when tasks are separately conducted. In this particular work, we are motivated by the need to jointly model a pair of tasks (Outcome span detection and Outcome classification) in order to enhance outcome information retrieval. Recent developments in the domain such as emergence of an outcome classification system that is aimed at

standardising outcome reporting and classification motivated us to re-construct the datasets we use in order to align them with this classification. The datasets contain text from abstracts of clinical trials published on PubMed. We cannot ascertain that all these abstracts are unbiased assessments of effects of interventions, especially with recurring articles citing several biases including *selection bias* (trial clinicians favour certain participating patients because of personal reasons), *reporting/publishing bias* (only reporting statistically significant results) and many more. Nevertheless, we provide more details and reference these datasets both within the article and the supplementary material.

References

Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. 2019. Correcting crowdsourced annotations to improve detection of outcome types in evidence based medicine. In *CEUR Workshop Proceedings*, volume 2429, pages 1–5.

Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. 2020. Assessment of contextualised representations in detecting outcome phrases in clinical trials. *Manuscript submitted for publication*.

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. Flair: An easy-to-use framework for state-of-the-art nlp. In *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59.

Betty van Aken, Jens-Michalis Papaioannou, Manuel Mayrdorfer, Klemens Budde, Felix Gers, and Alexander Loeser. 2021. Clinical outcome prediction from admission notes using self-supervised knowledge integration. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 881–893, Online. Association for Computational Linguistics.

Kamal Al-Sabahi, Zhang Zuping, and Mohammed Nadher. 2018. A hierarchical structured self-attentive model for extractive document summarization (hssas). *IEEE Access*, 6:24205–24212.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620. Association for Computational Linguistics.

Austin J Brockmeier, Meizhi Ju, Piotr Przybyła, and Sophia Ananiadou. 2019. Improving reference prioritisation with pico recognition. *BMC medical informatics and decision making*, 19(1):1–14.

Miao Chen, Ganhui Lan, Fang Du, and Victor Lobanov. 2020. Joint learning with pre-trained transformer on named entity recognition and relation extraction tasks for clinical analytics. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 234–242.

Dina Demner-Fushman, Barbara Few, Susan E Hauser, and George Thoma. 2006. Automatically identifying health outcome information in medline records. *Journal of the American Medical Informatics Association*, 13(1):52–60.

Dina Demner-Fushman and Jimmy Lin. 2007. Answering clinical questions with knowledge-based and statistical techniques. *Computational Linguistics*, 33(1):63–103.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics:Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Susanna Dodd, Mike Clarke, Lorne Becker, Chris Mavergames, Rebecca Fish, and Paula R. Williamson. 2018. A taxonomy has been developed for outcomes in medical research to help improve knowledge discovery. *Journal of Clinical Epidemiology*, 96:84–92.

Cunxiao Du, Zhaozheng Chen, Fuli Feng, Lei Zhu, Tian Gan, and Liqiang Nie. 2019. Explicit interaction model towards text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6359–6366.

Yaru Hao, Li Dong, Furu Wei, and Ke Xu. 2020. Investigating learning dynamics of bert fine-tuning. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 87–92.

Xiaoli Huang, Jimmy Lin, and Dina Demner-Fushman. 2006. Evaluation of PICO as a knowledge representation for clinical questions. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, pages 359–63.

Di Jin and Peter Szolovits. 2018. Advancing pico element detection in medical text via deep neural networks. *CoRR*.

Siddhartha R Jonnalagadda, Pawan Goyal, and Mark D Huffman. 2015. Automating data extraction in systematic reviews: a systematic review. *Systematic reviews*, 4(1):1–16.

Brennan C Kahan, Brian Feagan, and Vipul Jairath. 2017. A comparison of approaches for adjudicating outcomes in clinical trials. *Trials*, 18(1):1–14.

Akbar Karimi, Leonardo Rossi, Andrea Prati, and Katharina Full. 2020. Adversarial training for aspect-based sentiment analysis with bert. *arXiv preprint arXiv:2001.11316*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Svetlana Kiritchenko, Berry De Bruijn, Simona Carini, Joel Martin, and Ida Sim. 2010. Exact: automatic extraction of clinical trial characteristics from journal publications. *BMC medical informatics and decision making*, 10(1):1–17.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*.

Mingbo Ma, Kai Zhao, Liang Huang, Bing Xiang, and Bowen Zhou. 2017. Jointly trained sequential labeling and classification by sparse attention neural networks. *arXiv preprint arXiv:1709.10191*.

Usman Naseem, Katarzyna Musial, Peter Eklund, and Mukesh Prasad. 2020. Biomedical named-entity recognition by hierarchically fusing biobert representations and deep contextual-level word-embedding. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Benjamin Nye, Junyi Jessy Li, Roma Patel, Yinfei Yang, Iain J. Marshall, Ani Nenkova, and Byron C. Wallace. 2018. A Corpus with Multi-Level Annotations of Patients, Interventions and Outcomes to Support Language Processing for Medical Literature. In *Proc. of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 197–207.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

David L Sackett, William M C Rosenberg, J A Muir Gray, R Brian Haynes, and W Scott Richardson. 1996. Evidence based medicine: what it is and what it isn't. *BMJ*, 312(7023):71–72.

Erik F Sang and Jorn Veenstra. 1999. Representing text chunks. *arXiv preprint cs/9907006*.

Byron C Wallace, Joël Kuiper, Aakash Sharma, Mingxi Zhu, and Iain J Marshall. 2016. Extracting pico sentences from clinical trial reports using supervised distant supervision. *The Journal of Machine Learning Research*, 17(1):4572–4596.

Paula R. Williamson, Douglas G. Altman, Heather Bagley, Karen L. Barnes, Jane M. Blazeby, Sara T. Brookes, Mike Clarke, Elizabeth Gargon, Sarah Gorst, Nicola Harman, Jamie J. Kirkham, Angus McNair, Cecilia A.C. Prinsen, Jochen Schmitt, Caroline B. Terwee, and Bridget Young. 2017. The COMET Handbook: Version 1.0.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Lin Xiao, Xin Huang, Boli Chen, and Liping Jing. 2019. Label-specific document representation for multi-label text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 466–475.

Hu Xu, Bing Liu, Lei Shu, and Philip Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 2324–2335, Minneapolis, Minnesota. Association for Computational Linguistics.

Wonjin Yoon, Jinhyuk Lee, Donghyeon Kim, Minbyul Jeong, and Jaewoo Kang. 2019. Pre-trained language model for biomedical question answering. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 727–740. Springer.

Appendices

A Joint learning using LCAM

To demonstrate the flow of our joint learning training, we use the pseudo code in algorithm 1 to show how we arrive at the joint model loss. For each token's hidden state (line 8), we compute a context aware hidden state by adding to it an encoded abstract representation line 9 and then compute two attention scores (line 10 - 14) that both capture the contribution the token makes to each label.

These are then used to generate a label-word representation (line 16), all label-word representations forming a sentence (line 17) are used to compute an outcome extraction(OE) loss using eqn 9 (line 19). Once again we add context to the newly generated toke-level representations (line 20). For every outcome, we repeat steps in lines 10-14 to obtain label attention scores., i.e. depicting the contribution the particular outcome phrase makes to each label and these are used to obtain a label-document representation for the outcome (line 30). This representation is then used to compute the outcome classification loss (line 32). The loss we minimise in the joint learning is computed as shown by line 33.

B Hyperparameters and Run time

We perform a grid search through multiple combinations of hyperparameters included in Table 8 below. Using 20% of EBM-COMET+EBM-NLP dataset as a dev set, we obtain the best F1 values. Table 8 shows the range of values (including the lower and upper bound) for which the LCAM-BioBert is tuned to obtain optimal configurations. Using a shared TITAN RTX 24GB GPU, the baseline joint model i.e. LCAM-BiLSTM runs for approximately 45 minutes when evaluating on the EBM-COMET dataset, 190 minutes when evaluating on the EBM-NLP dataset and at-least 320 minutes on the merged dataset EBM-COMET+EBM-NLP. For the LCAM-BioBERT model, the experiments last at-least 14 hours on the EBM-COMET dataset, 30 hours on the EBM-NLP and 42 hours on the merged EBM-COMET+EBM-NLP.

Table 9 includes the tuned ranges for the Standalone models (BioBERT and SciBERT) which we fine-tune for the outcome extraction (OE) and outcome classification task. Similar to the joint model, the best values are chosen based on the EBM-COMET test set F1 values. Training and evaluation on the EBM-COMET, EBM-NLP and EBM-COMET+EBM-NLP consume 7, 34, and 45 GPU hours respectively.

C Datasets

C.1 EBM-NLP

EBM-NLP corpus (Nye et al., 2018) is a crowd sourced dataset in which ca.5,000 clinical trial abstracts were annotated with elements in the health literature searching PICO framework (Huang et al., 2006). PICO stands for Participants, Interventions,

---

Algorithm 1 LCAM Training

1: **Input**: train data, **Output**: model weights
2: **for** abstract a in train data **do**
3:     Obtain $Abs = \text{AbsEncoder}(a)$
4:     **for** sent s in a **do**
5:         Obtain $\mathbf{H} = Encoder(s)$
6:         where $\mathbf{H} \in \mathbb{R}^{k \times n}$
7:         **Initialise:** an empty tensor S
8:         **for** h_n in $\mathbf{H}$ **do**
9:             $h_n^c = h_n + f(Abs)$
10:             Obtain $A^{(1)} = \text{softmax}(\mathbf{W} \tanh \mathbf{V} h_n^c)$
11:             where $\boldsymbol{V} \in \mathbb{R}^{b \times k}$, $\mathbf{W} \in \mathbb{R}^{|l_w| \times b}$,
12:             and $\boldsymbol{A}^{(1)} \in \mathbb{R}^{|l_w| \times 1}$
13:             Obtain $\boldsymbol{A}^{(2)} = \mathbf{U} h_n^c$
14:             where $\mathbf{U} \in \mathbb{R}^{|l_w| \times k}$, $\boldsymbol{A} \in \mathbb{R}^{|l_w| \times 1}$
15:             **label-word representation**:
16:             $\boldsymbol{E}^{t_l} = \boldsymbol{A}^{(1)} h_n^{c^\top} + \boldsymbol{A}^{(2)} h_n^{c^\top}$
17:             $\text{S} = \text{S} \oplus \boldsymbol{E}^{t_l}$
18:         **end for**
19:         Compute Loss eqn 9 - L_{osd}
20:         $\forall \boldsymbol{E}^{t_l} \in \text{S} : \boldsymbol{E}^{t_l} = \boldsymbol{E}^{t_l} + f(Abs)$
21:         $\forall O_x \in \text{S}$, where $x \geq 0$ & $O_x \in \mathbb{R}^{m \times k}$
22:         i.e. outcome $\boldsymbol{O}_x$ has m tokens
23:         **for** outcome $\boldsymbol{O}$ in S **do**
24:             Obtain $A^{(1)} = \text{softmax}(\mathbf{W} \tanh(\boldsymbol{V} \boldsymbol{O}^\top))$
25:             where $\boldsymbol{V} \in \mathbb{R}^{b \times k}$, $\mathbf{W} \in \mathbb{R}^{|l_s| \times b}$
26:             and $\boldsymbol{A} \in \mathbb{R}^{|l_s| \times m}$
27:             Obtain $\boldsymbol{A}^{(2)} = \mathbf{U} \boldsymbol{O}^\top$
28:             where $\mathbf{U} \in \mathbb{R}^{|l_s| \times k}$, $\boldsymbol{A} \in \mathbb{R}^{|l_s| \times m}$
29:             **label-document representation of an outcome**:
30:             $\boldsymbol{E}^{s_l} = \boldsymbol{A}^{(1)} \boldsymbol{O} + \boldsymbol{A}^{(2)} \boldsymbol{O}$
31:         **end for**
32:         Compute Loss L_{oc} eqn 13
33:         minimise model loss $L = L_{osd} + L_{oc}$
34:     **end for**
35: **end for**

Comparators and Outcomes. The dataset has supported clinicalNLP research tasks (Beltagy et al., 2019; Brockmeier et al., 2019). The corpus has two versions, (1) the "**starting spans**" in which text spans are annotated with the literal "PIO" labels (I and C merged into I) and (2) the "**hierarchical labels**" in which the annotated outcome "PIO" spans were annotated with more specific labels aligned to the concepts codified by the Medical Subject Headings (MeSH) [4], for instance the Outcomes (O) spans are annotated with more granular (spe-

---

[4] https://www.nlm.nih.gov/mesh

Parameter	Tuned-range	Optimal
Batch size	[16,32,64]	64
Drop out	[0.1,0.2,0.3,0.4,0.5]	0.1
Embedding dim		
-Baseline	–	300
-BERT models	–	768
b	[150, 200, 250]	
Optimizer	[Adam, SGD]	Adam
Epochs	[5,10,15]	10
Learning rate	[5e-4, 1e-4, 5e-3, 1e-3, 5e-2, 1e-2]	1e-3

Table 8: Parameter settings for the joint models

Parameter	Tuned-range	Optimal
Train Batch size	[8,16,32]	16,32
Eval Batch size	[8,16,32]	8
Embedding dim	–	768
Optimizer	[Adam, SGD]	Adam
Epochs	[5,10,15]	10
Learning rate	[5e-5, 1e-4, 5e-3, 1e-3]	5e-5

Table 9: Parameter settings for the Standalone models

cific) labels which include Physical, Pain, Mental, Mortality and Adverse effects. For the clinical recognition task we attempt, we use the hierarchical version of the dataset. The dataset has however been discovered to have flawed outcome annotations (Abaho et al., 2019) such as (1) statistical metrics and measurement tools annotated as part of clinical outcomes e.g.“*mean arterial blood pressure*” instead of “*arterial blood-pressure*”,“*Quality of life Questionnaire*” instead of “*Quality of life*” and (2) Multiple outcomes annotated as a single outcome “*Systolic and Diastolic blood- pressure*” instead of “*Systolic blood-pressure*” and “*Diastolic blood-pressure*”.

C.2 EBM-COMET

A biomedical corpus containing 300 PubMed “Randomised controlled Trial” abstracts manually annotated with outcome classifications drawn from the taxonomy proposed by (Dodd et al., 2018). The abstracts were annotated by two experts with extensive experience in annotating outcomes in systematic reviews of clinical trials (Abaho et al., 2020). Dodd et al. (2018)'s taxonomy hierarchically categorised 38 outcome domains into 5 outcome core areas and applied this classification system to 299 published core outcome sets (COS) in the Core Outcomes Measures in Effectiveness (COMET)

database.

C.3 EBM-COMET+EBM-NLP

We merge the two datasets above for two main purposes, (1) to align the annotations of the EBM-NLP to a standard classification system (Dodd et al., 2018) for outcomes and (2) create a larger dataset to use in evaluating our joint learning approach.

C.4 Pre-processing

We create one single vocabulary using the merged dataset and use it for all three datasets. Whilst generating the vocabulary, we simultaneously split the abstracts into sentences using Stanford tokeniser. This vocabulary is then used in creating tensors representing sentences, where each tensor contains id's of the token/words in the sentence. The same procedure is followed to create tensors containing id's of the labels (“BIO”) corresponding to the words in the sentences. Additionally, we create tensors with id's of outcome classification labels, so for each sentence tensor, their is a corresponding token-level label tensor and a sentence-level label (outcome label) tensor. For the baseline where we use a BiLSTM to learn GloVe representations, we follow instructions to extract GloVe[5] specific vectors for words, token-level labels and sentence labels in the dataset. All the files with d-dimensional vectors are stored as .npy files. For the joint BERT-based models, we use flair (Akbik et al., 2019) to extract TransformerWord Embeddings from pretrained BioBERT for the tokens.

---

[5] https://github.com/stanfordnlp/GloVe