Deep Learning for Automatic Classification of Multi-Modal Information Corresponding to Chest Radiology Reports

Nelson Nunes\textsuperscript{a}, Bruno Martins\textsuperscript{a}, Nuno André da Silva\textsuperscript{b}, Francisca Leite\textsuperscript{b}, Mário J. Silva\textsuperscript{a}

\textsuperscript{a}INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal
\textsuperscript{b}Hospital da Luz Learning Health, Luz Saúde, Portugal

Abstract

Hospitals frequently use chest X-ray images, which can provide a detailed view of a patient’s thorax, as a radiology examination technique for the diagnosis of cardiac and pulmonary diseases. However, the interpretation of these images, which often results in a free-text radiology report and/or a classification, requires specialized medical professionals, leading to high labor costs and waiting lists. Automatic inference of thoracic diseases from the results of chest radiography exams is still a challenging task, although several recent studies have proposed machine learning methods for this particular task. Deep neural network architectures can contribute to a more efficient indexing of radiology exams (e.g., associating the data to diagnostic codes), providing inexpensive, accurate, and interpretable classification results that can guide the domain experts. This article advances a novel approach for classifying chest X-ray reports, based on a multi-modal end-to-end neural network that combines representations for the images and the associated textual descriptions. Experimental results show interesting patterns, e.g. validating the high performance of the individual components and showing promising results for the multi-modal processing of radiology examination data, particularly when pre-training the components of the model with large pre-existing datasets (i.e., a 14% increase in terms of the average value for the areas under the receiver operating characteristic curves).

Keywords: Classification of Chest Radiology Exams, Learning from Multi-Modal Data, Deep Learning, Natural Language Processing, Artificial Intelligence in Medicine

1. Introduction

Chest radiography is globally the most common medical imaging examination. The interpretation of chest X-rays, which typically involves specialized medical professionals, is critical for the screening, diagnosis, and management of many diseases.

Accurate models for data classification (e.g., for assigning images and/or text \cite{Laserson2018, Duarte2018} to codes within standard clinical taxonomies) can significantly speed-up the process of coding clinical data, decreasing labor costs, increasing coding consistency, and standardizing the coding of legacy data. The picture archiving and communication systems (PACSs) and the radiology information systems (RISs) of many clinics/hospitals have large amounts of chest radiology images, stored together with free-text radiology reports summarizing their interpretation. Thus, we argue that the combination of state-of-the-art methods for image analysis \cite{Khan2018} and natural language processing \cite{Goldberg2017} can be particularly interesting in the context of coding legacy data, directly considering multi-modal contents (i.e., images together with existing text reports) as a way to improve the classification results. Although many practical concerns can arise with the use of machine learning methods for directly producing diagnosis from the image data, the application of these methods for coding (or assisting the coding) legacy data is much more straightforward, offering many benefits to the access and retrieval of data from large legacy databases.

In this paper, we introduce a novel multi-modal approach to classify chest radiology exams, combining state-of-the-art pre-trained CNN architectures and clinical pre-trained word embeddings. For processing image data, the following CNN architectures were experimented: Dual Path Networks (DPNs) \cite{Chen2017}, Residual Neural Networks (ResNets) \cite{He2016}, Densely Connected Convolutional Networks (DenseNets) \cite{Huang2017}, and EfficientNet \cite{Tan2019}. For modeling free-text data, three different biomedical pre-trained word embeddings were tested, namely BioWordVec \cite{Chen2018}, BioELMo \cite{Jin2019}, and BioBERT \cite{Lee2019}, together with Bidirectional Multiplicative Long Short-Term Memory (bi-mLSTM) units \cite{Krause2017} for creating intermediate representations, which are finally combined through a multi-head attention mechanism \cite{Vaswani2017}. Both parts of the model can be used in isolation, and they can be pre-trained using...
large existing single modality datasets (i.e., the MIMIC-CXR (Johnson et al., 2019) or CheXpert (Irvin et al., 2019) datasets of chest radiology images, or the MIMIC-III (Edward William Johnson et al., 2016) dataset of clinical texts). The complete model concatenates the representations from the convolutional and recurrent parts, and it can be trained end-to-end (or fine-tuned, in case we leverage pre-trained components) from data combining both modalities (e.g. using the standard back-propagation algorithm together with an optimization method such as Adam (Kingma and Ba, 2015)). Early results of this research were presented on a conference paper (Nunes et al., 2019), and we now report on a more extensive set of tests, comparing different modeling choices.

We report on experiments with four distinct datasets where instances are assigned to multiple labels (i.e., frontal chest radiography images from the aforementioned MIMIC-CXR and CheXpert datasets, a subset of MIMIC-III corresponding to chest radiology free-text reports, and multi-modal instances from the OpenI dataset (Demner-Fushman et al., 2016), in all cases considering 14 possible labels for common diseases, and using stratified sampling (Szymański and Kajdanowicz, 2017; Schidis et al., 2011a; Szymański and Kajdanowicz, 2017) to split the data into training (64%), validation (16%) and testing (20%) sub-sets. Our tests contrasted single versus multi-modal approaches, and they assessed the impact of model pre-training, evaluating results in terms of a variety of metrics for multi-label prediction (i.e., classification accuracy, coverage error, label ranking average precision, and micro-and macro-averaged values for precision, recall, F1, and Areas Under Receiver Operating Characteristic Curves (AUROCs)).

When using image contents alone, the best convolutional model (i.e. a EfficientNet-95 network) achieved 0.8693 in micro-AUROC, a result that is in line with other studies on similar tasks (Pooch et al., 2019). On the OpenI dataset and leveraging pre-training, the complete model achieved 0.987 in micro-AUROC, 0.979 in accuracy, 1.665 in coverage error, and 0.945 in terms of label ranking average precision. These results confirm the usefulness of using multi-modal data. By leveraging existing datasets for model pre-training, very accurate results can be attained, opening the way to the development of practical tools for automating and/or assisting clinical coding.

The next section describes related work, while Section 3 presents the proposed method. Section 4 details the evaluation methodology and the obtained results, also describing how visualizations based on pixel- and word-level attention can help to interpret the model predictions. Finally, Section 5 summarizes our conclusions and presents directions for future work.

2. Related Work

Large labeled datasets have recently driven deep learning methods to achieve expert-level performance on a variety of medical imaging interpretation tasks (Irvin et al., 2019; Wang et al., 2017, 2018; Rajpurkar et al., 2017). Recent examples of publicly available datasets, supporting this type of developments, include MIMIC-CXR (Johnson et al., 2019) and CheXpert (Irvin et al., 2019) for the classification of chest X-ray images. Similarly, deep learning methods have also been used on tasks related to interpreting medical text, including the assignment of death certificates, discharge summaries, or other types of clinical text, to nodes in standardized taxonomies such as the International Classification of Diseases (ICD) or the Medical Subject Headings (MeSH) thesaurus (Duarte et al., 2018; Edward William Johnson et al., 2016). Again, these studies have leveraged the availability of large labeled datasets, with a recent example being the MIMIC-III (Edward William Johnson et al., 2016) database on patients admitted to critical care units, which includes free-text contents (e.g., radiology reports) associated with ICD codes.

A recent example for the use of deep learning on the task of coding medical text is the work reported by Duarte et al. (2018), in which the authors evaluated the performance of a hierarchical attention network on the task of classifying death certificates according to ICD-10, using the free-text descriptions in death certificates and associated autopsy reports. The proposed model combined recurrent units and attention mechanisms in a hierarchical arrangement (i.e., bi-directional Gated Recurrent Units (bi-GRUs) were used together with a neural attention layer to generate a representation for each sentence, and a separate combination of bi-GRUs with an attention layer was used to model the sequence of sentences associated with each certificate), generating meaningful intermediate representations for the textual contents (Yang et al., 2016). On a large dataset of death certificates collected from the mortality surveillance system of the Portuguese Ministry of Health, the authors obtained a high classification accuracy (e.g. 76.112%, 81.349%, and 89.320%, respectively for the classification of death certificates according to ICD full-codes, blocks, and chapters).

Karimi et al. (2017) reported on the use of a Convolutional Neural Network (CNN), inspired on a previous method for general text classification (Kim, 2014), in the task of coding free-text radiology reports according to ICD codes. The authors used a grid-search procedure to optimize different parameters of the proposed model (e.g., type and dimensionality of word embeddings, activation functions on both the convolutional and fully-connected layers, and values for convolution filter size, depth, and stride), and also the training hyper-parameters (e.g., learning rate, batch size, number of epochs, and drop-out rates). The best configuration achieved an accuracy of 83.84% over radiology reports from ICD9 dataset (Pestian et al., 2007), with results suggesting that the use of pre-trained word embeddings over in-domain data can improve the overall classification accuracy.

The previous work that is perhaps most similar to ours
of pre-trained language representations (i.e. pre-trained word embeddings and contextual word embeddings), which have improved the performance of deep learning models in a variety of different biomedical application domains (Peng et al., 2019b; Chiu et al., 2016; Peng et al., 2019a). Thus, for representing text, we explore the use of three pre-trained sets of word embeddings, namely BioWordVec (Chen et al., 2018), BioBERT (Lee et al., 2019), and BioELMo (Jin et al., 2019).

BioWordVec are FastText (Joulin et al., 2017) word embeddings pre-trained over large bio-medical corpora, specifically PubMed abstracts and clinical notes from MIMIC-III (Chen et al., 2018). The skip-gram model aims at predicting the surrounding words given a center word. For a sequence of words $w_1, w_2, \ldots, w_N$, representing a textual corpus $N$ words, and a $c$ context window size, the method attempts to maximize the following average log probability:

$$\frac{1}{N} \sum_{n=1}^{N} \left( \sum_{-c \leq i \leq c, i \neq 0} \log \left( p(w_{n+i} \mid w_n) \right) \right)$$  \hfill (1)

where $w_n$ represents the word at position $n$. The standard definition for $p(w_{n+i} \mid w_n)$ applies a softmax function, as described formally in the following equation:

$$p(w_1 \mid w_O) = \frac{\exp(V(w_1)^\top V(w_O))}{\sum_{w=1}^{W} \exp(V(w)^\top V(w_O))}$$  \hfill (2)

where $W$ represents the vocabulary size, and $V(w_1)$ and $V(w_O)$ are vector representations of an input center word and of an output context word, respectively. FastText word embedding model extends the Skip-gram model by representing each word as an $n$-gram of characters. The FastText approach involves training embeddings for individual word tokens and character $n$-grams, allowing us to compute embeddings for out-of-vocabulary words, by averaging the embeddings for the corresponding $n$-grams. This aspect is particularly interesting for clinical text, where orthographic variations and typographical errors are often present.

BioELMo is an extension of the Embeddings from Language Model (ELMo) model introduced by Peters et al. (2018), providing contextual word embeddings pre-trained on PubMed abstracts. The original ELMo approach uses a multi-layer bi-directional recurrent architecture to produce the contextualized word embeddings. BioELMo vocabulary consists 1M most frequent tokens from the pre-training corpus. In particular, the RNN architecture uses to stacked layers of Bi-directional RNNs leveraging Long Short-Term Memory (LSTM) cells, i.e. a particular type of RNN that will be explained latter, together with multiplicative LSTMs (mLSTMs). The model was pre-trained using an unsupervised task named Bidirectional Language Model (BiLM), concerned with predicting if a sequence of words is an authentic sentence. BiLM involves a forward
and a backward Language Model (LM). The forward LM takes as input a sequence of tokens and applies a softmax layer to the last LSTM layer output, in order to predict the next token. Similarly, the backward LM receives a reversed sequence and tries to predict the previous token given the future context. Note that the parameters of the token representations and the softmax layer are connected in both directions, while the the parameters of the LSTMs are not connected in both directions, becoming independent. The word representations are then calculated as a linear weighted combination of the two hidden layers of Bi-LSTMs, together with the context-independent token representation. The context-independent token representation is produced by a 2048 channel char-ngram CNN followed by two highway layers (i.e., an extension of the residual connections idea, modulating the quantity of input signal to be added to the output, by passing the input through fully connected layers with sigmoid and ReLU as non-linearity activation functions) and a linear projection down to 512 dimensions (Kim et al., 2016). Due to computational limitations, we follow ELMo’s feature-based approach, opposed to the jointly fine-tuned on a downstream task approach. Thus, the word representations are extracted and calculated using the learnt downstream task layer weights to average the following layers: one token embedding, and two Bi-LSTM layers.

BioBERT exploits the pre-trained Bidirectional Encoder Representations from Transformers (BERT) model introduced by Devlin et al. (2018), fine-tuning a model pre-trained on general texts on PubMed abstracts and PubMed central full-text articles. Instead of RNNs for modeling sequences of words, BERT leverages a different neural architecture named the Transformer (Vaswani et al., 2017), a stack of encoder and decoder blocks. The encoder block has one layer of a multi-head attention (i.e., layer that will be explained latter) followed by a feed forward layer. The decoder block has this same structure with an extra masked multi-head attention. Contrasting with directional models that process a text sentence sequentially (i.e., left-to-right or right-to-left), BERT is considered a bidirectional model, as the Transformer encoder is fed with the entire sequence of words simultaneously, allowing the model to learn the context of a word based on all of its surroundings. BERT’s multi-layer bidirectional Transformer architecture was originally pre-trained on Wikipedia and books corpora using two unsupervised tasks, namely the masked language model (MLM) and the next sentence prediction (NSP) tasks. MLM consists in predicting a randomly masked word based on the sentence context words, while NSP consist in predicting, for a given pair of sentences, if one follows the other in the original corpora. BERT segments sentences using the WordPiece model with a 30,000 token vocabulary, introduced by Wu et al. (2016). For a given corpora and a pre-defined maximum number of tokens, the WordPiece model is trained to select a set of word pieces that minimizes the total number of word pieces generated while segmenting the entire the corpus.

The segmented sentence starts with the [CLS] token, followed by the WordPiece tokens wrapped with two [SEP] tokens representing the division of the two sentence pair. The input embeddings are the sum of the token embeddings, the segmentation embeddings, and the position embeddings involved in the Transformer architecture. BioBert uses the original BERT vocabulary. Due to computational limitations, we follow BERT’s feature-based approach, opposed to the jointly fine-tuned on a downstream task approach. Thus, the word representations are extracted and combined using the hidden layer representations generated from last 4 Transformer encoders. Leveraging word representations produced by methods such as one of the three models outlined before, typical approaches for representing textual data involve Long Short-Term Memory (LSTM) units for performing the analysis of sequences of embeddings, i.e. neural networks that model sequential data by having a recurrent hidden state regulated by gates. At time step t (i.e., at position t for a given input sequence of word embeddings), a gate \( i_t \) decides which part of the memory cell will be forgotten, a gate \( f_t \) controls how much input information is added to the memory cell, and a gate \( o_t \) filters the cell memory, deciding how much to use for computing the output. The
values for these gates are calculated based on linear combinations of the current input \( x_t \) and the previous state \( h_{t−1} \), passed through an activation function (e.g., the logistic sigmoid \( \sigma() \)). LSTM units can be formally defined as follows, where \( \odot \) corresponds to an element-wise multiplication, and \( y_t \) corresponds to the result from the LSTM unit (i.e., the results produced from the hidden states) for each position \( t \).

\[
\begin{align*}
    h_t &= \tanh (c_t) \odot o_t \\
    c_t &= c_{t−1} \odot f_t + g_t \odot i_t \\
    i_t &= \sigma (x_t \cdot W^{xi} + h_{t−1} \cdot W^{hi}) \\
    f_t &= \sigma (x_t \cdot W^{xf} + h_{t−1} \cdot W^{hf}) \\
    o_t &= \sigma (x_t \cdot W^{xo} + h_{t−1} \cdot W^{ho}) \\
    g_t &= \tanh (x_t \cdot W^{xg} + h_{t−1} \cdot W^{hg}) \\
    y_t &= h_t
\end{align*}
\]  

(3)

Bi-directional LSTM units (bi-LSTMs) can be used to process a sequence of words both in a forward (\( h^+_t \)) and in a backward direction (\( h^-_t \)). By concatenating the states from two independent LSTM units (i.e., \( h_t = [h^+_t, h^-_t] \)), bi-LSTMs can provide a more wide-ranging summary of the information at each position \( t \) in the aforementioned ELMo approach for generating word embeddings uses two Bi-LSTMs stacked together.

In our model, instead of standard bi-LSTMs for processing the representations of word embeddings, we used the recently proposed Multiplicative LSTM units (mLSTMs), also within a bi-directional arrangement. These units combine factorized hidden-to-hidden transitions (i.e., an idea taken from multiplicative recurrent neural networks) with the gating logic from LSTMs (Krause et al., 2017). For a given input, mLSTMs can have different recurrent transition functions, and this approach can be formally defined as follows, where \( m_t \) represents an intermediate state.

\[
\begin{align*}
    h_t &= \tanh (c_t) \odot o_t \\
    c_t &= c_{t−1} \odot f_t + g_t \odot i_t \\
    m_t &= (x_t \cdot W^{xm}) \odot (h_{t−1} \cdot W^{hm}) \\
    i_t &= \sigma (x_t \cdot W^{xi} + m_t \cdot W^{mi}) \\
    f_t &= \sigma (x_t \cdot W^{xf} + m_t \cdot W^{mf}) \\
    o_t &= \sigma (x_t \cdot W^{xo} + m_t \cdot W^{mo}) \\
    g_t &= \tanh (x_t \cdot W^{xg} + m_t \cdot W^{mg}) \\
    y_t &= h_t
\end{align*}
\]  

(4)

Moreover, instead of the standard sigmoid or hyperbolic tangent activation functions, we used the penalized hyperbolic tangent (Xu et al., 2016) as the activation function within the recurrent units, penalizing negative inputs to create an additional non-linearity that facilitates model training, as shown in the next equation:

\[
f(x) = \begin{cases} 
    \tanh(x) & \text{if } x > 0 \\
    0.25 \times \tanh(x) & \text{otherwise}
\end{cases}
\]  

(5)

Previous experiments showed improved results when combining the penalized hyperbolic tangent with standard LSTM units, in a variety of NLP tasks (Eger et al., 2018).

The results from the bi-directional mLSTM units (i.e., the hidden states) can be combined through attention mechanisms, i.e., functions that map a context vector and a set of hidden state vectors into a single output vector (Vaswani et al., 2017; Yang et al., 2016). The output vector is the result of a weighted sum of the hidden states, where the weights are computed by a score function that takes as input the context vector and each hidden state vector. For defining the weight that should be given to the representation at each time step \( t \) (i.e., for weighting the values at each position \( t \) for a given input sequence), a commonly used attention mechanism can be formally defined as follows:

\[
s = \sum_t \left( \frac{\exp (\tanh(W \cdot y_t + b_t) \cdot u)}{\sum_t \exp (\tanh(W \cdot y_t + b_t) \cdot u)} \right) \cdot y_t
\]

(6)

In the previous equation, \( s \) corresponds to the text representation resulting from the attention weighting scheme, while \( W \), \( u \) and \( b \) are learned parameters, respectively a matrix of weights, the context vector, and a bias term. Multiplying the softmax outputs (i.e., the part in brackets) with each hidden state keeps the values of the words that are more significant for the task, and reduces the values for less important words. In our model, we used multiple attention heads, thus considering multiple representations built from the input sequence (i.e., different weights, learned independently in each attention head, associated with the different word positions). The output that is produced from the multi-head attention mechanism corresponds to the concatenation of the weighted vectors produced by each head.

3.2. Representing the Visual Inputs

Deep learning architectures for processing visual inputs often involve stacking convolutional and pooling layers in a network for producing an intermediate representation, followed by fully-connected layers that produce a final classification. As Convolutional Neural Networks (CNNs) become deeper, the number of parameters continues to grow, thereby making it challenging to train.

For instance, He et al. (2016) introduced Residual Networks (ResNets), i.e. CNNs that aim to avoid problems such as vanishing gradients during the training of deep models, by introducing direct identity connections that skip one or more convolutional layers, forming a residual block. These skip connections consist in adding the output of previous convolutional layers together with the output of the current layer, creating paths propagating information from earlier to later layers. The blocks used in ResNet architectures also apply a bottleneck operation, which consists in a 1 × 1 convolution before each 3 × 3 convolution to reduce the number of input feature-maps, and thus to improve computational efficiency. Using the aforementioned block, He et al. also proposed complete CNN architectures for image classification. For instance ResNet-50 consists of a BN-Relu-Conv block, followed by
a max-pooling layer, and 4 stages of bottleneck residual blocks, before a final average pooling layer.

In turn, Huang et al. (2017) presented Dense Convolutional Networks (DenseNets), i.e. CNNs that use additional connections to alleviate vanishing gradient problems by strengthening feature propagation and reuse. In their approach, each convolutional layer input consists in the concatenation of all the output feature maps of all the preceding convolutional layers. This way, the output feature maps of a given layer will be used as part of the inputs of all the subsequent layers. This shorter links between initial and later layers enable CNNs that can be significantly deeper, more accurate, and efficient to train. The DenseNet-201 architecture which consists of a BN-ReLU-Conv block, followed by a max-pooling layer, and 4 stages of dense and transition blocks (i.e.), before a final average pooling layer. A transition block is an BN-ReLU-Conv block followed by an average pooling layer.

More recently, Chen et al. (2017) proposed Dual Path Networks (DPNs), i.e. CNNs that combine ideas from both the ResNet and DenseNet architectures. The feature maps from the DenseNet path are mostly reused and refined, while the feature maps from the ResNet path keep stacking new features. The blocks used in DPN architectures combine both approaches, by extending blocks used in ResNet architectures with a slice and concatenation operation. Tan and Le (2019) introduced the Efficient Network (EfficientNet) neural architecture, combining a particular block with a novel CNN architecture scaling method named compound scaling. The original main building block combines squeeze-and-excitation operations, as described by Hu et al. (2018), together with a mobile inverted bottleneck convolution (MBConv) block, originally described by Tan et al. (2019) and Sandler et al. (2018) and which uses separable depthwise convolutions. Opposing the original residual block approach, where a high number of channels is compressed and then expanded with an $1 \times 1$ convolution for better computational efficiency, the inverted residual block expands and then compresses a low number of channels with an $1 \times 1$ convolution, since the Depthwise separable convolution already significantly reduces the number of parameters. The depthwise convolution used differs from standard convolutions by splitting both the input feature maps and the kernel filters into $n$ channels. For each channel, convolves the corresponding input and filter split, stacking all the split output tensors. Depthwise separable convolution, extends the depthwise convolution by adding an extra step: a $1 \times 1$ convolution across channels. The Squeeze-and-Excitation mechanism opposes the standard mechanism that equal weights each feature map channel when creating the final output feature maps. Using average pooling, followed by a nonlinear fully connected layer and a sigmoid activation, this procedure provides each channel a smooth gating mechanism similar to a content aware mechanism that weights each feature map channel individually.

The multi-objective neural architecture search optimizes both FLOPS and accuracy, while scaling the architecture depth, width, and resolution with a constant ratio $\phi$ (e.g. EfficientNet-B5 scaled from EfficientNet-B0 with $\phi$ equal to 5). This approach can be formally defined as follows, where $\alpha$, $\beta$, $\gamma$ are constants that were determined by a small grid search on the original small EfficientNet-B0 model, with value 1.2, 1.1 and 1.15, respectively.

$$d = \alpha \phi$$
$$w = \beta \phi$$
$$r = \gamma \phi$$
$$s.t. \; \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$$
$$\alpha \geq 1, \beta \geq 1, \gamma \geq 1$$

EfficientNet-B5 results from scaling the EfficientNet-B0 model using 1.6 and 2.2 as the width and depth coefficients.

### 3.3. The Complete Network for processing Multi-Modal Inputs

As shown in Figure 1, the complete neural architecture has two separate branches that extract meaningful representations from the distinct information modalities, namely from X-ray images and from the associated radiology reports. The text classification branch leverages the pre-trained BioWordVec model to produce sequences of biomedical word embeddings, further analysed by bidirectional mLSTM units and combined through an multi-head attention mechanism. The image classification branch leverages the EfficientNet-B5 neural architecture pre-trained in the ImageNet dataset. The complete neural architecture concatenates the representations from both branches, using the resulting features to inform the 14 nodes prediction layer. Due to the multi-label nature of the task of classifying radiology exams, the sigmoid activation function was chosen to convert each node’s score between 0 and 1, independently of the other node’s scores. If a score, for a specific class is bigger than 0.5, the model classifies that class as positive, therefore multiple classes can be classified as positive for the same instance. Although this fusion method is much simpler than the one used in the TieNet model (Wang et al., 2018), it allows us easily to pre-train the two branches of the network.

The complete models were implemented through the keras deep learning library, and the source code supporting our experiments was also made available on GitHub.

### 4. Experimental Evaluation

This section describes the experimental evaluation of the proposed approach, first presenting the datasets and the experimental methodology, and then the obtained results.


Table 1: Statistical characterization of the datasets used in the experiments.

<table>
<thead>
<tr>
<th>Label</th>
<th>Images</th>
<th>Text</th>
<th>Multi-Modal</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIMIC-CXR</td>
<td>CheXpert</td>
<td>MIMIC-III</td>
<td>OpenI</td>
</tr>
<tr>
<td>No Finding</td>
<td>83,336</td>
<td>19,765</td>
<td>133,563</td>
</tr>
<tr>
<td>Cardiomegaly</td>
<td>56,012</td>
<td>30,158</td>
<td>351</td>
</tr>
<tr>
<td>Lung Opacity</td>
<td>60,196</td>
<td>98,759</td>
<td>62</td>
</tr>
<tr>
<td>Lung Lesion</td>
<td>5,047</td>
<td>11,47</td>
<td>9,741</td>
</tr>
<tr>
<td>Edema</td>
<td>43,812</td>
<td>61,555</td>
<td>4,635</td>
</tr>
<tr>
<td>Consolidation</td>
<td>16,614</td>
<td>37,306</td>
<td>43,006</td>
</tr>
<tr>
<td>Pneumonia</td>
<td>38,262</td>
<td>20,664</td>
<td>82,526</td>
</tr>
<tr>
<td>Pneumothorax</td>
<td>12,953</td>
<td>20,808</td>
<td>16,241</td>
</tr>
<tr>
<td>Pleural Effusion</td>
<td>65,449</td>
<td>86,541</td>
<td>29,979</td>
</tr>
<tr>
<td>Pleural Other</td>
<td>3,009</td>
<td>4,318</td>
<td>896</td>
</tr>
<tr>
<td>Fracture</td>
<td>5,675</td>
<td>7,935</td>
<td>16,862</td>
</tr>
<tr>
<td>Support Devices</td>
<td>74,970</td>
<td>108,184</td>
<td>10,504</td>
</tr>
<tr>
<td>Total Instances</td>
<td>250,044</td>
<td>191,229</td>
<td>261,091</td>
</tr>
</tbody>
</table>

4.1. Datasets and Experimental Methodology

Our experiments relied on four different datasets, including two datasets containing frontal chest X-ray images labeled according to 14 observation classes (i.e., CHEXpert [Irvin et al. 2019] and the first version of the MIMIC-CXR dataset), one dataset of full-text radiology reports (i.e., a subset of the data from MIMIC-III [Edward William Johnson et al. 2016] with the ICD labels converted to the same 14 observation classes), and the OpenI multi-modal radiography dataset, collected from multiple institutes by Indiana University and containing frontal chest X-ray images together with full-text reports, also labeled according to the same 14 observation classes. Table 1 provides general data characterization statistics.

In more detail, the MIMIC-CXR dataset includes 371,920 chest X-rays, associated with 227,943 studies involving patients admitted to the Beth Israel Deaconess Medical Center, between 2011 and 2016. In turn, CHEXpert contains 224,316 chest X-rays from 65,240 patients, collected from Stanford Hospital between October 2002 and July 2017. In both cases, we considered only frontal view X-ray images (i.e., 250,044 instances from MIMIC-CXR, and 191,229 instances from CHEXpert), merging both datasets for the experiments involving image data (i.e., when evaluating the different CNN architectures alone, or for pre-training the complete model in the case of multi-modal tests).

MIMIC-III is a freely accessible critical care database containing, among other elements, radiology reports associated with ICD diagnostic codes (taken from patient discharge notes). We filtered the radiology reports according to the occurrence of key-phrases such as *chest, lungs, or thorax*, and using the ICD codes that result from matching the 14 labels of the MIMIC-CXR dataset with the correspondent set of ICD codes. This resulted in a set of 261,091 textual documents.

Finally, the complete OpenI dataset includes 3,851 radiology reports and the associated 7,784 chest X-ray images. We filtered the dataset to consider only 3,689 instances containing full-text reports and frontal X-ray images, labeled according to the 14 labels from MIMIC-CXR/ CHEXpert.

Taking into account the multi-label nature of the datasets, we considered stratification when dividing the data (i.e., the merging of MIMIC-CXR and CHEXpert, or the subsets from MIMIC-III and OpenI) into training (64%), validation (16%), and testing (20%) splits. The considered multi-label stratification method balances the assignment of instances into splits, according to the distribution of single classes and multi-class pairs [Sechidis et al. 2011] [Szymański and Kajdanowicz 2017].

Model training when leveraging the text datasets was made with batches of 64 instances, while models leveraging image and multi-modal inputs were trained with batches of 16 instances, in both cases leveraging back-propagation together with the Adam optimization method [Kingma and Ba 2015]. In order to use pre-trained ImageNet models, the input images were re-scaled to 256 × 256 matrices. The learning rate was initially set to 10⁻³, and then refined through a cyclic pattern [Smith 2017]. The number of epochs was also defined through a criteria based on a validation loss, stopping when the variation between consecutive epochs was less than 10⁻⁶.

To assess the quality of the predictions, we used the following metrics: accuracy (i.e., average number of correct labels per instance), coverage error (CE) (i.e., how many labels, ranked according to prediction scores, need to be checked to cover all the true labels), label ranking average precision (LRAP), micro and macro-averaged scores for multi-label precision, recall, F1 scores, and areas under ROC curves (AUROCs).

4.2. Experimental Results

Our tests considered four different settings: (i) use the DPN-92, ResNet-50, DenseNet-101, and EfficientNet-B5 architectures to classify X-ray grayscale images trained with the combined MIMIC-CXR/CHEXpert data; (ii) use the ResNet-50, DenseNet-101, and EfficientNet-B5 models pre-trained with ImageNet to classify X-ray RGB images trained with the combined MIMIC-CXR/CHEXpert data; (iii) use BioBERT, BioELMo, and BioWordVec embeddings, together with bi-directional mLSTMs and multi-head attention, to classify chest radiology reports from MIMIC-III; and (iv) use the complete model combining EfficientNet-B5 together with BioWordVec to classify the multi-modal instances from the OpenI dataset with and without pre-training of settings (i) and (ii), and evaluate the individual performance of EfficientNet-B5 and BioWordVec models with pre-training of settings (i) and (ii) on single-modal instances from the OpenI dataset. Table 2 presents the results obtained over the testing data splits. The first four rows correspond to Setting (i), the following three rows correspond to Setting (ii), the subsequent three rows correspond to Setting (iii), and the last four rows to Setting (iv).

In the first experimental setting’s results, EfficientNet-B5 achieved consistent better results in terms of almost all the different metrics. The second experimental setting’s
results demonstrate that leverage public available ImageNet pre-trained weights, instead of randomly initialized weights, to classify X-ray RGB images instead of grayscale images, can improve the overall performance of these models. For instance, EfficientNet-B5 achieved consistent better results in terms of almost all the different metrics. The third experimental set of results show that the model with BioWordVec embeddings achieved better results then those achieved with BioELMo and BioBert contextual embeddings. Although these models have shown superior performance in other tasks, it is worth mentioning that methods such as ELMo take character information to construct word representations, therefor attending the problem of out-of-vocabulary words to some extent. However, the individual character information is an insufficient and unnatural linguistic unit for word representation, when compared with FastText embeddings character n-grams to model out-of-vocabulary tokens. The better performance of the BioELMo model, when compared with BioBERT in this classification task, goes in line with the conclusions of Jin et al. (2019), which showed that although fine-tuned BioBERT is better than BioELMo in biomedical named entity recognition and natural language inference tasks, as a fixed feature extractor BioELMo outperforms BioBERT in their probing tasks (i.e., tasks that probe token representations for linguistic properties).

The complete multi-modal model with pre-trained weights and fine-tuned with the OpenI training split, achieved a very high performance in terms of all the different metrics. Pre-training, in particular, contributed significantly to the overall performance of the complete model (e.g., 2% improvement in accuracy, and 14% improvement in macro-AUROC).

### 4.3. Visualization Mechanisms to Help Interpreting the Predictions

In addition to near real-time surveillance of specific lung and heart diseases, the proposed architecture can also be useful for assisting human coders, by providing coding suggestions that are interpretable. In connection to the classification results, the proposed model also allows us to explore two distinct interpretability views. Using the gradient-weighted class activation mapping (Grad-CAM), i.e., a technique that leverages the gradients of a chosen target class, flowing into the final convolutional layer to construct a localization map emphasizing the important regions in the image for that specific class classification, it is possible to visualize which areas of the X-ray were more important to the classification decision (Selvaraju et al., 2017). Simultaneously, using the weights from the multi-head attention mechanism, it is possible to visualize which words in the full-text report were more important to the classification.

Figure 2 presents two example instances from the OpenI test split, correctly assigned to the classes Cardiomegaly and Fracture, which visualization mechanisms to help interpreting the predictions. The original X-ray images were overlayed with the correspondent Grad-CAM heatmap, emphasizing the areas that were more important to support each classification decision. The heatmaps adopt a thorax shape and highlight the areas around the lungs, heart and ribs, regions relevant to the corresponding predicted classes. The colored boxes over the text, denote word attention weights according to two of the attention heads (i.e., blue and green colours). The free-text reports contain words that can be highly indicative of the disease, and that the method assigned higher weights. For instance, for the Cardiomegaly instance, the model gave higher weights to words describing the cardiomegaly condition, such as heart size is mildly enlarged, and the concept itself: cardiomegaly. For the Fracture instance, the model gave higher weights to words describing the fracture and the correspondent location, such as rib fractures.

## 5. Conclusions and Future Work

This article presented a novel deep learning method for the multi-label classification of data from radiology exams according to 14 observation classes, combining both X-ray images and full-text reports. The complete model, with pre-trained weights exploiting large existing datasets and fine-tuned on instances from the OpenI dataset, achieved a
high performance in terms of multiple evaluation metrics for multi-label classification. The obtained results confirm that the proposed deep learning approach combining multi-modal data can indeed achieve a high accuracy in this domain, opening the way to the development of tools that can effectively assist medical professionals (e.g., clinical coders).

Despite the interesting results, there are also several ideas for future work. For example, as an alternative of using BioBERT and BioELMo as fixed feature extractors, fine-tune both while training on the downstream task of classifying chest radiology reports. For instance, instead of mLSTM units, the recently proposed Mogrified LSTM units from Melis et al. (2019) can perhaps be used instead. This particular approach has achieved results in language modeling tasks similar to state-of-the-art Transformer models, having the benefit of having considerably fewer parameters. In order to increase the performance of recurrent models such as mLSTMs, Gu et al. (2019) proposed gating mechanisms that robustly increase the performance of LSTMs on tasks requiring long temporal dependencies. The recent release of MIMIC-CXR Johnson et al. (2019) dataset, which incorporates X-ray images together with the full-text reports, can be used to further evaluate the proposed model and enables new experiments using multimodal bitransformers Kiela et al. (2019), which have achieved promising results on classification of images and text, and Unicoder-VL Li et al. (2019), a universal encoder that aims to learn joint representations of vision and language in a pre-training manner.

Acknowledgements

The authors from INESC-ID were partially supported through Fundação para a Ciência e Tecnologia (FCT), specifically through the INESC-ID multi-annual funding from the PIDDAC programme with reference number UID/CEC/50021/2019. We also gratefully acknowledge the support of NVIDIA Corporation, with the donation of the Titan Xp GPU used in the experiments.

References


