

AUEB NLP Group at ImageCLEFmedical Caption 2023

Panagiotis Kaliosis¹, Georgios Moschovis¹, Foivos Charalampakos¹,
John Pavlopoulos¹ and Ion Androutsopoulos¹

¹Department of Informatics, Athens University of Economics and Business, 76, Patission Street, GR-104 34 Athens, Greece

Abstract

This article describes the methods that the AUEB NLP Group experimented with during its participation in the 7th edition of the ImageCLEFmedical Caption sub-tasks, namely Concept Detection and Caption Prediction. The former intends to automatically classify biomedical images into a set of one or more tags based solely on the visual input, while the latter aims to generate a syntactically and semantically accurate diagnostic caption that addresses the medical conditions depicted on a given image. For the Concept Detection sub-task, extending our previous work, we utilized a wide range of Convolutional Neural Network encoders followed by a Feed-Forward Neural Network, both in a single-task and a multi-task fashion, as well as combined with a contrastive learning approach. Our methods concerning the Caption Prediction sub-task are influenced by both our previous work and recent progress in Natural Language Processing (NLP) methods. Our two base systems use CNN-RNN and Transformer-to-Transformer encoder-decoder architectures, respectively. Additionally, we experimented with a Transformer-based denoising component, which was trained to reformulate the generated captions in a more syntactically coherent and medically accurate way. Our group ranked 1st in Concept Detection and 3rd in Caption Prediction.

Keywords

Natural Language Processing, Computer Vision, Biomedical Images, Convolutional Neural Networks, Multi-Label Classification, Caption Generation, Generative Models, Transformers, Deep Learning

1. Introduction


ImageCLEF [1] is a multi-modal machine learning campaign running every year since 2003 as part of the Cross Language Evaluation Forum (CLEF)¹. It fosters research breakthroughs, as well as the development of advanced multimedia processing systems in the areas of computer vision, image analysis, classification and retrieval in multi-modal, cross-language contexts [1]. ImageCLEFmedical is one of the four main tasks of this year's ImageCLEF campaign. It consists of a series of challenges that range from image captioning to synthetic image generation and question-answering. We participated in the ImageCLEFmedical Caption task, which took place for the 7th time [2]. Following the previous years' campaigns, the task consisted of two sub-tasks, namely Concept Detection and Caption Prediction. The goal in Concept Detection is to


CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece

✉ pkaliosis@aueb.gr (P. Kaliosis); geomos@aueb.gr (G. Moschovis); phoebuschar@aueb.gr (F. Charalampakos); annis@aueb.gr (J. Pavlopoulos); ion@aueb.gr (I. Androutsopoulos)

🌐 <https://www.linkedin.com/in/geomos/> (G. Moschovis); <https://ipavlopoulos.github.io/> (J. Pavlopoulos); <https://www2.aueb.gr/users/ion/> (I. Androutsopoulos)

🆔 0000-0003-0547-0581 (G. Moschovis); 0000-0001-9188-742 (J. Pavlopoulos)

 © 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

¹<https://www.clef-campaign.org/>, Last accessed: 2023-07-07

link a biomedical image with one or more medical concepts (categories), whereas in Caption Prediction the goal is to automatically generate a draft diagnostic report that accurately outlines the medical situation, as well as the topology of the body structures and organs shown in the image.

Diagnostic Captioning still constitutes a challenging research problem that aims to assist the diagnostic process for a patient by providing a draft report, rather than replacing the doctors and any human factor involved in the procedure [3]. It may thus be viewed as an assistive tool, capable of providing an initial draft diagnostic report regarding the patient's condition. Such a draft would ideally allow the doctors' attention to focus on important regions of the image [4] and aid them to produce more accurate medical diagnoses with improved accuracy and speed [5]. Experienced clinicians could improve their throughput, by analyzing faster and more efficiently the large volume of medical examinations that they daily handle. Less experienced clinicians could ideally consider the automatically generated captions in order to reduce the probability of clinical errors [6]. Concept Detection may assist Diagnostic Captioning by detecting key concepts that need to be mentioned in the draft report. It can also be used to index medical images by relevant concepts.

1.1. AUEB NLP Group contributions

In this work we present the experiments conducted, as well as the systems submitted as part of AUEB NLP Group's participation in this year's Concept Detection and Caption Prediction tasks. We experimented with several extensions of our previous work [7, 8, 9, 10, 11] in the Diagnostic Captioning task, in addition to a number of new approaches influenced by the expeditious progress of Transformer-based [12] Deep Learning methods in Sequence-to-Sequence (Seq2Seq) architectures [13] and Large Language Models (LLMs) [14].

Our submissions to the Concept Detection sub-task revolve around two main directions. In the first one, we employed a Convolutional Neural Network (CNN) encoder in order to obtain the images' visual representations, followed by a Feed-Forward Neural Network (FFNN) that classifies the images into one or more medical concepts. In the second direction, we employed contrastive learning [15, 16], aiming at bringing the high-dimensional representations of images and their assigned concepts closer in the vector space. Finally, we experimented with various ensembles of our proposed systems, either by performing majority voting based on each system's predictions or by calculating the intersection and the union of their predicted concepts.

For the Caption Prediction sub-task, our work can be divided into three major directions. The first one, following our last year's submissions [11], is a Show and Tell model [17], which more specifically adopts an architecture that includes a CNN and a Recurrent Neural Network (RNN). The CNN-RNN architecture still remains competitive, while it also lays the foundations for further experiments, such as investigating new variants and modified forms [18]. Furthermore, we implemented an encoder-decoder model, where we employed Transformers for both the encoder and decoder components. More specifically, we employed a Vision Transformer (ViT) [19] instance as the image encoder and a GPT-2 [20] decoder in charge of generating the predicted captions. As our third major direction, we implemented a novel pipeline, where we used a denoising sequence-to-sequence model on top of the two aforementioned architectures. We trained our denoising model to rewrite or rephrase the initial draft radiology reports by

providing it with the ground truth captions. Thus, it was able to learn and subsequently correct the common mistakes of our two base models, resulting in a more fluent and consistent generated caption.

Extending our history of successful entries [7, 8, 10, 11] in the ImageCLEFmedical campaign, our submissions ranked 1st among 10 participating groups in the Concept Detection sub-task and 3rd among 13 participating groups in the Caption Prediction sub-task. In Section 2 below, we provide insight into this year’s dataset, followed by a discussion of our methods in Section 3. In Section 4, we present our experimental results for each sub-task. Finally, in Section 5 we summarize our findings and suggest directions for future research.

2. Data

In this year’s edition of the ImageCLEFmedical Caption task, a dataset consisting of 71,355 biomedical images along with their respective medical concepts, in the form of UMLS [21] terms,² and diagnostic captions was provided. The set was originally split by the organizers into training and validation subsets. Following the previous years’ campaigns, the dataset constitutes an updated and extended version of the Radiology Objects in Context (ROCO) dataset [22], which originates from a range of biomedical articles available in the PubMed Central Open Access (PMC OA) subset³.

The dataset, common for both sub-tasks, comprised images of different modalities (i.e., X-Ray, Computed Tomography), although no further insight was provided regarding the different types of images included. Concept Detection is a multi-label classification problem covering a broad range of 2,125 distinct biomedical concepts, originating from the Unified Medical Language System (UMLS) [21], whereas caption prediction aims at open-ended generation of diagnostic texts for the medical images. After merging the provided training and validation data, we split them into three subsets, holding out a development (private test) subset for evaluation purposes. We followed a 75%-10%-15% split, keeping relatively equal data distributions in all three subsets. We confirmed it by comparing the concepts distribution between the subsets. Thus, we considered 53,516 images as our training data, 7,135 images as our validation set, while the remaining 10,704 images constituted our held-out development set. Moreover, an official test set, consisting of 10,473 images was shared. All of our submissions were evaluated based on their performance on the official test set.

2.1. Concept Detection

Regarding the Concept Detection sub-task, a set of one or more medical concepts were originally assigned to each radiology image. The concepts are offered in the form of Concept Unique Identifiers (CUIs) in accordance with the Unified Medical Language System (UMLS) [21]. For example, the biomedical concept “Pericardial Effusion” is associated with the CUI term “C0031039”. Each concept is retrieved from the image’s corresponding diagnostic caption in order to be

²UMLS: <https://www.nlm.nih.gov/research/umls/index.html>, Last accessed: 2023-07-07

³PMC Open Access: <https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/>, Last accessed: 2023-07-07

employed as the training target. Some examples of images and their corresponding ground truth concepts can be found in Figure 1.

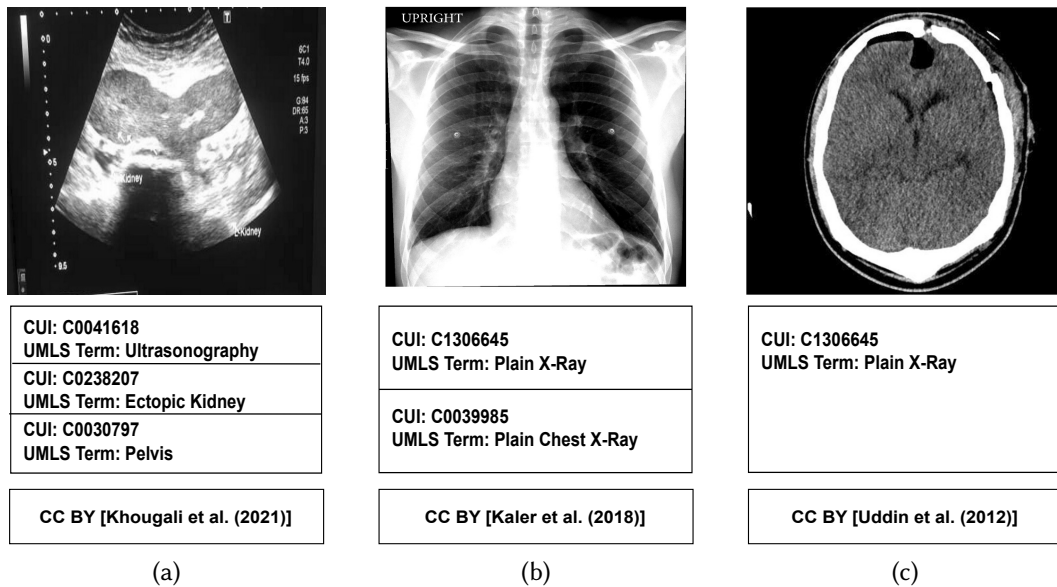


Figure 1: Three example images from the ImageCLEFmedical2023 [2] dataset, along with their corresponding CUIs and UMLS terms [21].

The dataset contains 2,125 distinct biomedical concepts. It is highly imbalanced in terms of concepts, as there are some that appear more than 20,000 times, while others are assigned to only 4 or 5 images. Figure 2 below illustrates the dataset’s long-tail distribution (left plot) by plotting the number of each concept’s appearances in descending order against its index (class index). Furthermore, after performing a thorough exploratory analysis of this year’s dataset, we observed that some concepts were more common, while also representing a greater category of medical examinations, such as X-Ray or Ultrasonography. Besides, we observed that the vast majority of the images is associated with one of these concepts, in addition to the rest, more specific concepts. Based on this observation, we decided to explore the potential of a multi-task classification model based on a shared backbone encoder, which will be described in Section 3.

Table 1 shows the ten most common concepts that occur in the dataset. We were able to partition the data into four main modalities; Computed Tomography Scan (CT), X-Ray, Magnetic Resonance Imaging (MRI) and Ultrasonography, based on the corresponding concept’s appearances. Later on, these modalities became the central focus of our work, mainly in the Concept Detection sub-task. The maximum and minimum number of concepts assigned to a single image are 32 and 1, occurring in 1 and 7,109 images respectively. The average number of assigned tags per image is 3.74. The aforementioned observations are outlined in the histogram in Figure 2b.

Table 1

The ten most common concepts (CUIs) and corresponding UMLS term found in the ImageCLEFmedical2023 dataset, along with the number of images they are associated with.

Most common concepts			
Position	CUI	UMLS Term	Images
1	C0040405	X-Ray Computed Tomography	24695
2	C0024485	Magnetic Resonance Imaging	11554
3	C0041618	Ultrasonography	9949
4	C0817096	Chest	8199
5	C1999039	Anterior-Posterior	7153
6	C0449900	Contrast used	5854
7	C0002978	Angiogram	4707
8	C0037303	Bone structure of cranium	3456
9	C0039985	Plain chest X-ray	3451
10	C0000726	Abdomen	3368

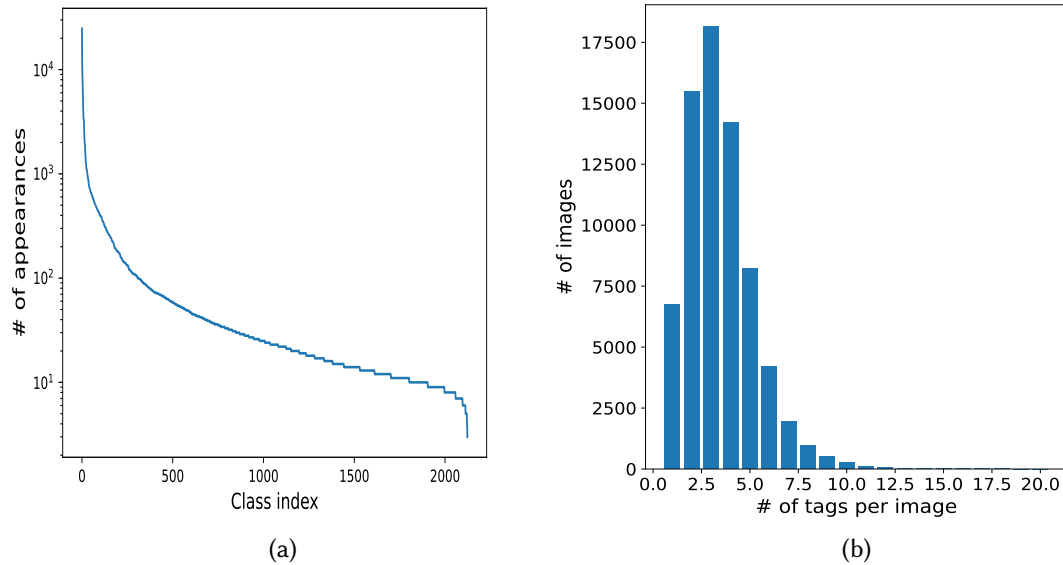


Figure 2: (a) Visualization of the data’s long-tail distribution. The y -axis shows the number of appearances for each tag, and the x -axis the tag’s class index. (b) Histogram with 20 fixed-size bins (horizontal axis) depicting the number of gold tags per image.

2.2. Caption Prediction

In the Caption Prediction sub-task, the images are accompanied by a diagnostic caption that expresses the medical conditions present in the image. There are 71,355 captions across the whole dataset, one for each provided image. Similarly to last year’s campaign, the vast majority of the captions, specifically 99.46% (70,974 out of 71,355 captions) are unique. This is an important differentiation from previous versions of this task, where this percentage was significantly lower [11]. Consequently, typical retrieval methods based on nearest neighbours search [23]

are not so efficient this year, including extended variations with weighting mechanisms relying on the cosine similarities of the retrieved images [24]. Therefore, more elaborate captioning methods are needed.

We found out that the maximum number of words in a single caption is 315 (occured once), while the minimum is 1 (encountered 134 times). The average caption length is 16.04 words. These statistics refer to the dataset as a whole, but we have carefully checked that they remain consistent in all three subsets. The five most common captions, as well as the ten most popular words, after excluding the stopwords, can be found in Tables 2 and 3 respectively. In Figure 3, we provide a histogram, as well as a box-plot, both showing that most of the captions do not exceed 100 tokens.

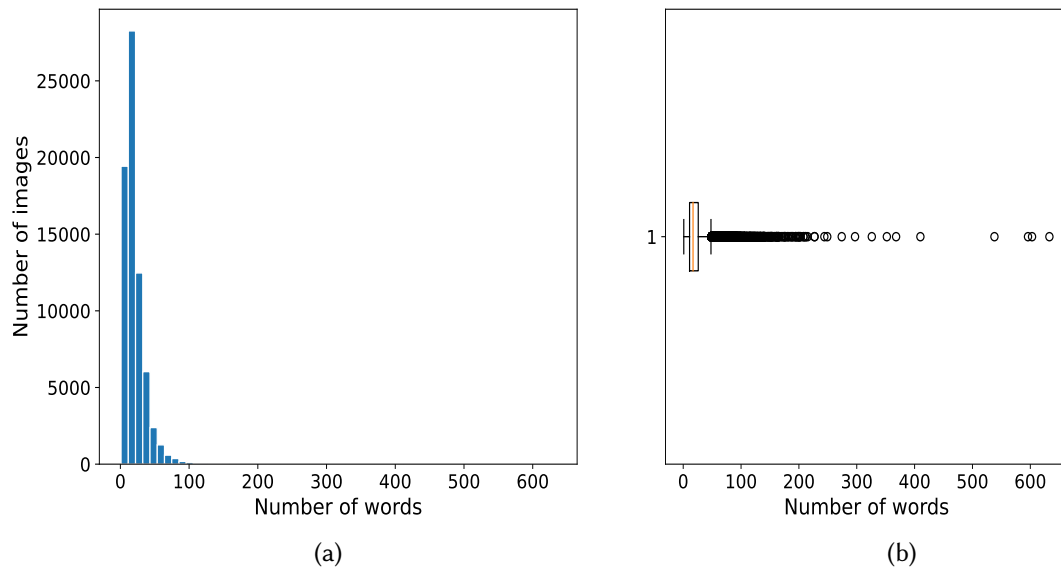


Figure 3: (a) Histogram that visualizes the captions’ length distribution. The y -axis contains the number of images that fall into each bin, while the x -axis contains the number of words included in the caption. (b) Box-plot over the same distribution, which highlights the outliers in the range of 100 to 200 words.

Table 2

The five most common captions found in the ImageCLEFmedical2023 dataset alongside the number of images they are associated with.

Most common captions		
Position	Caption	Occurences
1	Initial panoramic radiograph.	38
2	Final panoramic radiograph.	36
3	Chest X-ray.	18
4	Chest radiograph.	16
5	Preoperative CT scan.	9

According to the organizers [2], each caption is pre-processed before evaluated in the following manner:

- The caption is converted to lower-case.
- Numbers are replaced by words, e.g., number 10 becomes “ten”.
- Punctuation is removed.

Table 3

The ten most common words and their frequencies in the ImageCLEFmedical2023 dataset, after removing stop-words.

Most common words (excluding stop-words)										
Word	showing	right	left	ct	image	chest	scan	computed	tomography	shows
Occurrences	19904	16163	16008	13187	9148	8761	8342	8013	7822	7775

Unlike last year’s campaign [11], we decided to perform experiments while both adopting and ignoring the pre-processing procedure during training, taking into consideration that this year stop-words were not removed during pre-processing by the organizers. Removal of the stop-words could potentially lead to distortion of important words in either the predicted or the ground truth captions.

3. Methods

In this section, we present the methods we used in our submissions for both the Concept Detection and the Caption Prediction sub-tasks.

3.1. Concept Detection

Our submissions in this year’s Concept Detection sub-task are based on three groundwork systems. First, following our previous work [7, 8, 10, 11], we thoroughly experimented with a CNN+FFNN system, as well as a multitask classifier with a more complex, yet similar architecture. Moreover, we implemented a contrastive learning retrieval-based classifier, using it as a standalone system as well as combining it with the best performing CNN+FFNN system. Additionally, we made several submissions using ensembles (MAJORITY, UNION and INTERSECTION-based) of the three aforementioned systems, as they achieved higher performance on the primary evaluation metric of the task in our held-out development set.

3.1.1. CNN+FFNN

Our first system utilizes a CNN encoder backbone, followed by an FFNN classification head that employs one or more hidden layers. The image features that represent the visual input are extracted from the last convolutional layer of the CNN. Then, a global pooling layer is used in order to acquire the final feature vector. We experimented with three global pooling strategies; max, average and Generalized-Mean (GeM) global pooling [25], which all resulted in enhanced

performance compared to no pooling scheme. Max pooling retrieves the maximum value of each feature map, while average pooling computes the respective mean value [26]. In addition, GeM pooling is a generalized version of both the max and average pooling strategies [25].

Specifically, given an input image, the CNN encoder outputs a three-dimensional tensor X of shape $H \times W \times K$. K denotes the number of channels (feature maps), while H and W represent the image’s height and width. Let X_k be a feature map, hence equal to $H_k \times W_k$ for $k \in [1, 2, \dots, K]$, and m , a , g be the max, average and GeM pooling functions respectively. The pooling layer’s output for input X_k is a single value v_k that can be computed based on Equations 1, 2, and 3, hereunder, depending on the pooling strategy employed:

$$v_k^{(m)} = m(X_k) = \max_{x \in X_k} x \quad (1)$$

$$v_k^{(a)} = a(X_k) = \left(\frac{1}{|X_k|} \cdot \sum_{x \in X_k} x \right) \quad (2)$$

$$v_k^{(g)} = g(X_k) = \left(\frac{1}{|X_k|} \cdot \sum_{x \in X_k} x^{p_k} \right)^{\frac{1}{p_k}} \quad (3)$$

GeM pooling is equivalent to max pooling when $p_k \rightarrow \infty$, and equivalent to average pooling when $p_k = 1$ [25]. The hyperparameter p_k can either be trained by integrating it in the network’s training process, or be manually initialized beforehand.

The FFNN component, consisting of multiple hidden and dropout [27] layers, classifies the image into one or more concepts. The network’s output layer consists of $|C|$ neurons, where C is the set of the unique concepts in the dataset and is featured with sigmoid activation gates in order to squash the neurons’ values between 0 and 1, hence transforming them into probabilities. We therefore end up with one probability per label and if it exceeds a specific threshold value t , then the corresponding concept is assigned to the image. The threshold (same value for all concepts) was selected by performing a grid search in the range (0.1, 0.7) on our validation set aiming to optimize the competition’s primary metric, the F_1 score. Our model was trained by minimizing the binary cross-entropy loss, treating each concept as a binary classification problem. Moreover, we used the Adam [28] optimizer, as well as a linear decreasing learning rate strategy and early stopping based on our validation set loss with patience equal to 3. We do not exploit the validation set for our final model since there is no guarantee that the same number of epochs is the best when using all training data and it has been previously observed that "the gain of re-training the model after merging all the splits is almost negligible" [29]. We experimented with various initial learning rates (e.g., $1e - 3$, $1e - 4$) and decreasing factors (e.g., 0.1, 0.05) using random search.

3.1.2. CNN+FFNN-based Multi-task Classifier

Our second system adopts the CNN+FFNN architecture described in the previous section and utilizes it in a multi-task fashion. We observed that some of the medical concepts were more common and represented generic medical terms (see also Section 2). This observation led

us into experimenting with a multi-task classification model composed of a shared encoding backbone and two task-specific classification heads. The first head corresponds to a single-label classification task (*Modality prediction*), while the second one to a multi-label classification problem (*Modality-specific concepts prediction*). An overview of the system’s architecture can be found in Figure 4.

The first head is in charge of classifying the image into one out of five candidate classes; the four main modalities (namely X-Ray, Computed Tomography, Magnetic Resonance Imaging and Ultrasonography) or none of them. Concurrently, the second head performs multi-label classification on the image features excluding the main modality tags, attempting to identify the rest of the concepts present in the image. The intuition behind this method is that, through the aggregated loss, the shared backbone will be driven to learn optimized image representations suitable for both tasks.

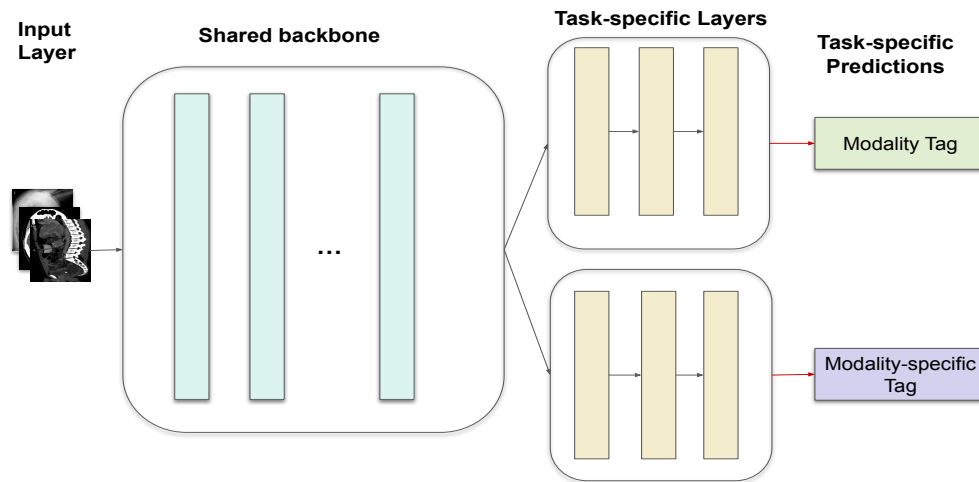


Figure 4: Illustration of our CNN+FFNN-based Multi-task classifier architecture.

Moreover, we stay consistent with our first system and use multiple hidden and dropout [27] layers. The *Modality Prediction* head consists of five neurons on the output layer, one for each modality (including the “None” option), featured with a Softmax activation function. It was trained by attempting to minimize the categorical cross-entropy loss. On the other end, the *Modality-specific classification head’s* output layer consists of $|C| - 4$ neurons (where $|C|$ denotes the overall number of possible concepts) alongside a sigmoid activation gate, and is trained by minimizing the binary cross-entropy loss. Both components’ learning rates were initialized at a relatively low value, swiftly increased to a pre-defined maximum and then slowly decreased until the end of the optimization process. This strategy is shown to preserve training stability and minimize the degree of divergence in the network’s parameters, especially in the deeper layers [30].

The entire network, composed of the shared backbone encoder and the two task-specific classifiers, is trained based on the aggregated loss that derives from the two FFNN components. Specifically, let \mathcal{L} be the network’s loss, CCE_{loss} be the single-label classifier’s loss, and finally

BCE_{loss} be the multi-label classification component’s loss. The total loss is equal to:

$$\mathcal{L} = \gamma \cdot CCE_{\text{loss}}(y_{\text{single}}, \hat{y}_{\text{single}}) + (1 - \gamma) \cdot BCE_{\text{loss}}(y_{\text{multi}_{\text{rest}}}, \hat{y}_{\text{multi}_{\text{rest}}}), \quad 0 < \gamma \leq 1 \quad (4)$$

where γ is initialized to 0.5 and can either stay fixed or be automatically adapted during training. In the case where γ is adaptive, we used the following approach. At the end of each epoch, if the total loss is increased compared to the previous epoch, we proceed to examine the partial task-specific losses. If only the CCE_{loss} increased, then we increase γ by a pre-defined factor (e.g., 10%), aiming to put more emphasis on reducing the CCE_{loss} throughout the next epoch. The same procedure is followed vice versa regarding the BCE_{loss} . In case both losses increased, then we slightly adjust γ either upwards or downwards, depending on which loss increased more. Moreover, even if the total loss decreased, we still attempt to optimize the losses’ weights. To do so, we modify γ ’s value, in accordance with which loss decreased more between the two. We either increase or decrease it aiming to place greater emphasis on the component with the less decreased loss.

3.1.3. Contrastive Learning-based Tagger

This system is based on the idea proposed by CLIP [16], which is a framework based on multi-modal learning. CLIP employs a contrastive learning objective and jointly trains an image encoder and a text encoder to predict the correct pairings of (image, text) examples. The (bidirectional) contrastive objective aims at bringing the representations of true pairings closer in the vector space, while pushing the representations of mismatching pairs far away.

Based on this approach, we formulate a similar training procedure where we utilize the available (image, concepts) pairs instead. We again use an image encoder and a text encoder based on BERT [31]. The encoders are trained to map the image representations and their gold concepts representations to nearby points in a joint representation space (see Figure 5). We compute the embeddings of the 2,125 concepts using the text encoder before training and treat them as trainable variables, which we update instead of updating the text encoder’s parameters. We use a bidirectional temperature-scaled version of the binary cross-entropy function as the training objective with the images-concepts similarity matrix $S \in \mathbb{R}^{n \times |C|}$ (computed via the dot product of the respective embeddings in a batch of n images) as the logits (see Eq. 5). The goal is to maximize the similarity of the image embeddings with the embeddings of the gold truth concepts assigned to each image.

$$\mathcal{L}_{\text{CLIP}}(y_{\text{multi}}, S) = \frac{BCE(y_{\text{multi}}, S/\tau) + BCE(y_{\text{multi}}^T, S^T/\tau)}{2} \quad (5)$$

where τ is the temperature hyper-parameter.

During inference, given an image, we compute the similarities between its embedding and the $|C|$ concepts and assign to it the top- k most similar concepts. We select to learn the k parameter using the following scheme: we create a dataset $\mathcal{D}' = \{\mathbf{s}_i, c_i\}_{i=1}^N$ where $\mathbf{s}_i = [s_{i1}, \dots, s_{i|C|}]$ is the vector that contains the similarities of the embedding of the i -th image with each embedding of the $|C|$ concepts and c_i is the number of concepts assigned to this image. Using \mathcal{D}' , we train a Multi-Layer Perceptron (MLP) regressor in order to predict the

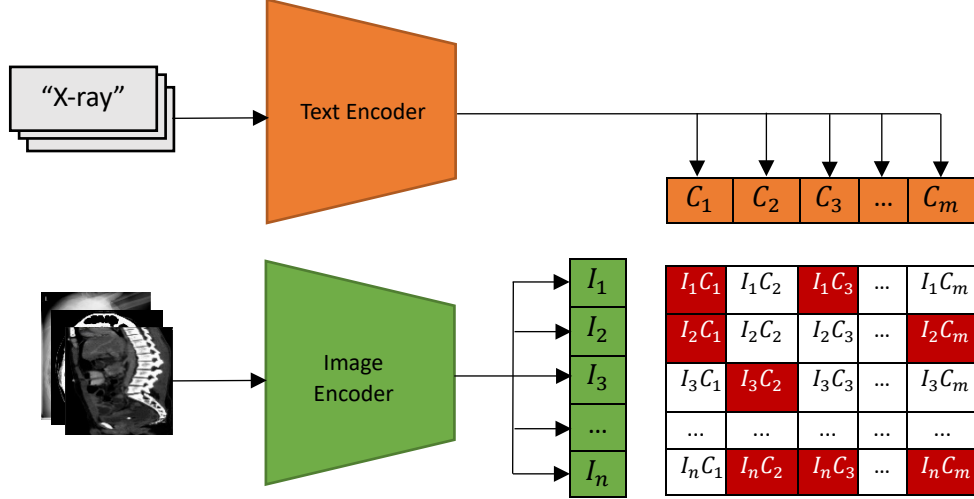


Figure 5: Overview of our CLIP-based approach. For each batch of n images, we compute the embeddings for $m = |C|$ concepts (and the n images) and aim to maximize the red-colored similarity values which correspond to the similarities between each image and its gold truth concepts. Figure adapted from [16].

number of assigned concepts (c_i) of the i -th image based on its similarity vector with the $|C|$ concepts. Thus, we feed this network image-concepts similarities and it outputs a number k as the expected assigned concepts for this image.

We also used this system with an ensemble-like design together with a FFNN. In the system described above, we added a trainable FFNN classifier which was fed the image embeddings from the CNN encoder. These same embeddings were also used in the calculation of the similarity matrix. The final output logits were formed by interpolating the classifier’s logits $L_c \in \mathbb{R}^{n \times |C|}$ and the similarity matrix S : $L_{Sc} = \lambda \cdot \sigma(L_c) + (1 - \lambda) \cdot \sigma(S)$, where λ is a trainable parameter and σ is the sigmoid function. Additionally, the system was trained using both the $\mathcal{L}_{\text{CLIP}}$ and the standard binary cross entropy loss BCE_{loss} :

$$\mathcal{L}_{\text{ensemble}}(y_{\text{multi}}, L_{Sc}) = \frac{\mathcal{L}_{\text{CLIP}}(y_{\text{multi}}, L_{Sc}) + BCE_{\text{loss}}(y_{\text{multi}}, L_{Sc})}{2} \quad (6)$$

3.2. Caption Prediction

Our submissions in the Caption Prediction sub-task again revolve around three main systems, two of which follow an encoder-decoder approach. The first one utilizes a CNN as the encoder

and an RNN as the decoder, while the second one is Transformer-based, employing a ViT [19] as the encoding unit and OpenAI’s GPT-2 [20] as the decoding unit. Furthermore, we implemented a sequence-to-sequence [13] denoising component, which when employed on top of the two aforementioned systems forms a novel captioning pipeline.

3.2.1. CNN-RNN

Our first system is based on the CNN-RNN encoder-decoder [17] method, which employs a CNN encoder and an RNN decoder that generates the caption.

In Figure 6, we present a high-level overview of the system’s architecture. The CNN encoder is responsible for extracting image representations, which are then passed to the decoder. The RNN decoder has been implemented with Gated Recurrent Units (GRU cells) [32] and concatenates the encoded visual features with the hidden states of its encoding cells. At each recurrent step, the previous GRU cell’s state, which contains knowledge about the extracted visual features and the part of the caption that has been generated so far, is passed alongside the previously generated word as an input to the current GRU cell. Afterwards, the GRU output is passed to an MLP component that yields a probability distribution over the model’s vocabulary words and the one with the highest probability is selected as the sentence’s next token. This recurrent process terminates once a special token, denoting the end of the generated sequence, is predicted. The model is trained by attempting to maximize the likelihood of the provided ground truth caption given a visual instance [17].

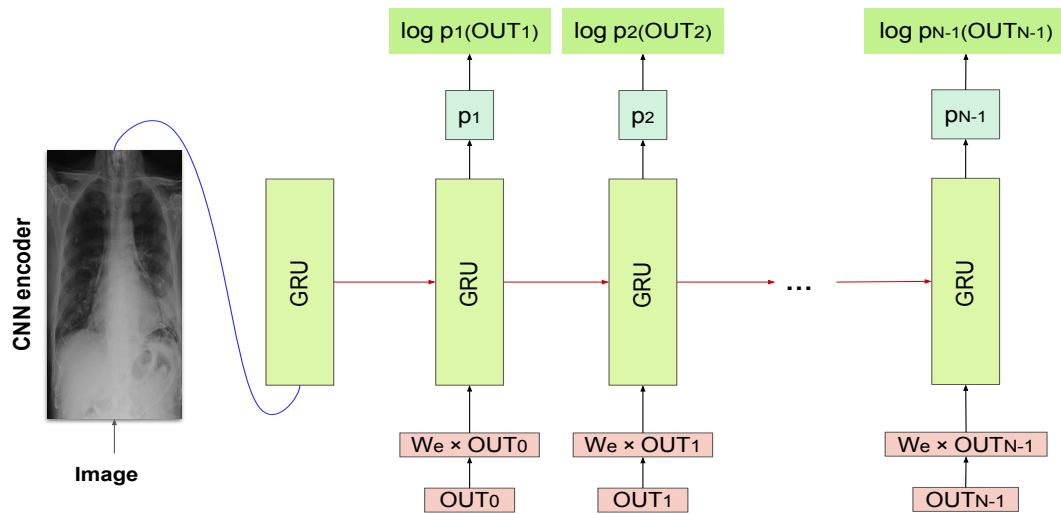


Figure 6: An overview of our CNN-RNN system’s architecture. W_e denotes a word embedding matrix, while OUT_i represents a 1-hot vector of the previously generated word. Figure adapted from [17].

Following the pre-processing steps of Show&Tell [17], we first added two special tokens in each training caption, a *<start of sequence>* and an *<end of sequence>* token. Next, we created the model’s vocabulary by keeping all words that appeared at least 4 times throughout the training set, replacing the out-of-vocabulary (OOV) words with the *<UNK>* special token. We experimented with multiple maximum length values ranging from 40 to 120 tokens, unlike our last year’s submission [11], where we had used a fixed maximum length of 40 tokens based on preliminary experiments.

As far as the decoding method is concerned, we ran experiments using both greedy and beam search decoding [33]. In the former option, we selected the word with the highest probability yielded by the MLP component at each step, while in the latter case we would search for the most probable sequences of tokens, by maintaining and updating a set of the n best candidates at each decoding step. The selection of these candidates is based on the likelihood of each path being the correct choice. It is calculated as the sum of the log probabilities of the so far generated sequence’s tokens. Greedy decoding can be considered as a special case of beam search decoding, when the beam size is equal to one ($n = 1$). We experimented with numerous values for the beam size n , specifically $n \in \{2, 3, 5\}$. Overall, beam search decoding resulted in better performance than following the greedy choice at each step.

3.2.2. ViT-GPT2

Our second system for the Caption Prediction sub-task is also based on the encoder-decoder framework, only that in this case we employed Transformer-based encoders and decoders. Influenced by the expeditious progress in the domain of Large Language Models (LLMs), as well as the impressive performance that these systems are able to achieve in NLP and Speech Recognition tasks, we decided to create a pipeline where Transformers are also utilized for computer vision [34].

The encoding component of our model, which is responsible for extracting the feature representation of a given image, consists of a Vision Transformer (ViT) [19] instance loaded from a pre-trained checkpoint. Regarding the decoding component, we employed GPT-2 [20], an open source, autoregressive LLM that achieves notable results in numerous text generation tasks. We also experimented with its distilled version (distilGPT2) as it is considered to be more time efficient with little to no decrement in performance [35]. However, we preferred to use the GPT-2 base version, as it performed better in preliminary experiments.

GPT-2 [20] is an autoregressive decoder-only model that is composed of a stack of 12 Transformer decoder blocks. Each one of these blocks sequentially processes the visual representation of the image, obtained by the image encoder, and the so far generated tokens. Following the last decoder block, a dense linear layer followed by a Softmax activation function is in charge of yielding a probability distribution over the model’s vocabulary, and thus predict the next generated token. The process described so far forms a single decoding step. A vector containing the word embedding of each step’s output, concatenated with its positional embedding is autoregressively fed to the bottom decoder block. This gradual, step-wise generation procedure is repeated until a special token, which denotes the end of the generated sequence, is predicted. We experimented with multiple decoding strategies; namely greedy decoding, beam search decoding (as described in Section 3.2.1), as well as top- k and nuclear sampling [33, 36]. Both

beam search decoding and the two sampling methods achieved equally competitive performance. In addition, we followed the same pre-processing steps that we have previously described.

3.2.3. 2xE-D: Captioning Model + Seq2Seq denoiser

Our third system is a denoising model, which we employ on top of the two aforementioned caption prediction systems (see Sections 3.2.1 and 3.2.2) resulting in a novel captioning pipeline. The model is trained on the captions output by our two basic systems and the corresponding ground truth captions, in order to improve readability. Both the original generative pipeline and the denoising component feature an encoder and a decoder. Hence, we call this system 2xE-D, where *E-D* denotes the encoder-decoder architecture. For the denoising part, we experimented with two prominent sequence to sequence architectures, BART [37] and T5 [38].

BART is a denoising autoencoder which is trained by reconstructing text that has been distorted by an arbitrary noise function [37]. It constitutes of a bidirectional encoder and a left-to-right autoregressive decoder. The denoising autoencoder is pre-trained on a series of tasks, which have been altered by one or more of the following corruption processes applied stochastically to the input sequences:

- Random token masking.
- Random token deletion.
- k -random tokens masking (employing a single masking token).
- Sentence permutation.
- Document rotation.

In detail, we intended to employ an instance of BART on a task similar to the one that it has been originally pre-trained on. We started off from a pre-trained BART checkpoint and fine-tuned it by providing the intermediate captions as input and the respective ground truth captions as the target text. We utilized the large version of the model, which contains 12 bidirectional encoder blocks and an equal number of decoder blocks. Table 4 shows three captions; the provided ground truth, the CNN-RNN generated one, and its revised version generated by our Seq2Seq denoising model. The denoiser was able to correct part of the initial generated caption, as it successfully revised the existing medical condition from “a mass” to “a lesion” and also accurately re-addressed the point of contention from “a liver lobe” to “a hepatic lobe”. Moreover, it chose to state “Computed Tomography” as its abbreviation (“CT”), which is a common tactic in diagnostic reports [39].

Extending this idea, we decided to fine-tune BART in a larger collection of noisy and denoised caption pairs. Therefore, we implemented a noise-insertion function, in accordance with the aforementioned noise transformations that BART is pre-trained on [37], and applied it to our training ground truth captions. In this way, we created an alternative text-to-text training set, consisting of (noisy - ground truth) caption pairs. We once again fine-tuned a pre-trained BART instance on the newly created dataset in order to build a *ClinicalBART* model, hoping it would acquire extended knowledge of the biomedical domain, and therefore generate more medically fluent text sequences.

Furthermore, we also experimented with T5, another encoder-decoder model pre-trained in a series of both supervised and unsupervised tasks [38], including denoising tasks. Last but

not least, we were granted access to ClinicalT5 through PhysioNet⁴. ClinicalT5 is a biomedical version of T5, pre-trained on the MIMIC-III dataset [40]. We further fine-tuned ClinicalT5 similarly to BART, in order to rephrase the intermediate captions produced by the CNN-RNN model (see Section 3.2.1) to approximate the gold ones.

Table 4

Comparison between the caption generated from our CNN-RNN model, its BART-generated rewritten version (denoted as BART@CNN-RNN) and the ground truth diagnosis.

Generated captions comparison	
Ground Truth	Full-body CT scan showing hepatic lesions.
CNN-RNN	computed tomography scan of the abdomen and pelvis showing a mass in the right lobe of the liver
BART@CNN-RNN	CT scan of the abdomen and pelvis showing a large cystic lesion in the right hepatic lobe

4. Experiments, Submissions and Results

In this section, we provide details and insight into our experiments regarding this year’s campaign [2]. Moreover, we share details about our submissions and the scores achieved in our held-out development set, as well as the official test set of the competition for both sub-tasks.

4.1. Concept Detection

In the Concept Detection sub-task, we submitted our nine best performing models, after evaluating them on our held-out development set. We submitted a single instance of our CNN+FFNN model (see Section 3.1.1) and two instances of our Contrastive learning-based tagger (henceforward *ContrastiveTagger*, see Section 3.1.3). The rest of our submissions were ensemble systems. We investigated the combination of the predictions of two or more instances by calculating the UNION or the INTERSECTION of their predicted concept sets. We also experimented with a majority voting rule. That is, given an ensemble system consisting of n models, a concept is assigned to the image if at least $\frac{n}{2} + 1$ models predicted it. All of our submitted ensemble systems were combinations of our CNN+FFNN and CNN+FFNN-based multi-task classifiers (henceforward *MultiTask-CNN+FFNN*).

This year’s primary evaluation metric for the Concept Detection sub-task was the F_1 -score between the predicted and the ground truth captions. It is calculated as the sum of the F_1 -score for each test image, divided by the total number of test images. Each partial score is calculated between the binary multi-hot candidate vector and the corresponding ground truth vector. Precisely, let F_1 be the overall F_1 -score, and \hat{f}_1 be the individual F_1 -score for every test image. Moreover, let p_t and g_t be the predicted and ground truth concepts for an image t . Finally, let T denote the test set. Then, F_1 is computed as:

⁴<https://www.physionet.org/content/clinical-t5/1.0.0/>, Last accessed: 2023-07-07

$$F_1 = \frac{1}{|T|} \sum_{t \in T} \hat{f}_1(p_t, g_t) \quad (7)$$

Moreover, a secondary evaluation metric was calculated that only included manually validated concepts, such as anatomy, topography and modality [2].

In the case of our first two systems (*CNN+FFNN*, *MultiTask-CNN+FFNN*), and specifically regarding their backbone component, we experimented with a wide range of CNN encoders. Namely, we trained the two networks using state-of-the-art CNN architectures, like EfficientNet [41], DenseNet [42] and ResNet [43]. In addition, we extended the CNN experimental range compared to our previous participations, by utilizing MobileNet [44], InceptionNet [45] and CheXNet [46]. We also experimented with Vision Transformers (ViT) [19], as well as older CNN encoders like VGG [47] and AlexNet [48]. However, they were not included in our submissions as they did not provide competitive results. These were either pre-trained on ImageNet [49] or were trained with uniformly initialized weights.

Table 5

Summary of our individual experiments (no ensembles included) in the ImageCLEFmedical2023 Concept Detection sub-task. The table contains the best scores that our systems achieved in our held-out development set for each method.

Individual Concept Detection Experiments		
ID	Method	Development
cd1	EfficientNetB0@CNN+FFNN	0.5173
cd2	DenseNet121@CNN+FFNN	0.5152
cd3	EfficientNetB0v2@CNN+FFNN	0.5151
cd4	MobileNet@CNN+FFNN	0.5128
cd5	InceptionNetv3@CNN+FFNN	0.5127
cd6	ResNet101@CNN+FFNN	0.5116
cd7	DenseNet169@CNN+FFNN	0.5093
cd8	CheXNet@CNN+FFNN	0.4910
cd9	ViT@CNN+FFNN	0.4776
cd10	EfficientNetB0@MultiTask-CNN+FFNN	0.4802
cd11	DenseNet121@MultiTask-CNN+FFNN	0.4792
cd12	ContrastiveTagger	N/A
cd13	MultitaskContrastiveTagger	0.5080

As expected, the model instances pre-trained on ImageNet [49] performed better than the randomly initialized ones in terms of the corresponding F_1 score. The training loss converged faster, despite the fact that biomedical images like the ones we deal with, come from a different domain compared to ImageNet’s training set. Moreover, CNN backbones outperformed ViT, which is in line with previous observations that they typically outperform other architectures such as ViT and Hybrid-ViT in classification and semantic segmentation for generic images [50], as well as classification of biomedical images [29, 5]. EfficientNetB0 [41] and DenseNet-121 [42] were the two best performing ones for both systems in terms of the primary evaluation metric (Equation 7).

We also experimented with freezing some of the encoder’s layers, in order to speed up the

training process and also prevent their weights from being modified, in an effort to preserve the model’s already acquired knowledge [51] and potentially prevent catastrophic forgetting [52]. However, our experiments showed that training the whole network resulted in higher F_1 score, while the speed up in terms of training time was not large enough in order to trade off the higher performance levels obtained by a fully-trainable network. Moreover, we experimented with data augmentation techniques [53] (i.e, random rotation, random cropping) during the loading of each image, but they did not provide any significant improvement in the system’s performance.

Table 6

Summary of our submissions in the ImageCLEFmedical2023 Concept Detection sub-task. The table contains the scores that our systems achieved in our held-out development set and the official test set, along with rankings among all the systems submitted from 10 participating teams. The annotation @U denotes a UNION ensemble, @Uoi indicates a UNION OF INTERSECTION ensemble, while the * symbol means that the corresponding submission was late, and therefore is not ranked.

AUEB NLP Group -Submission Table						
ID	Run ID	Approach	Primary F1		Secondary F1	Rank
			Dev	Test		
cd15	4	3xCNN+FFNN@U - (cd1, cd2, cd3)	0.5224	0.5222	0.9258	1
cd16	8	2xCNN+FFNN@Uoi - (cd15, cd17)	0.5223	0.5220	0.9276	2
cd17	2	3xCNN+FFNN@U - (cd1, cd2, cd4)	0.5223	0.5218	0.9220	3
cd18	7	2xCNN+FFNN@U - (cd1, cd2)	0.5217	0.5212	0.9277	4
cd19	6	Union Ensemble of cd1, cd2, cd10	0.5214	0.5208	0.9154	5
cd20	3	3xCNN+FFNN@U - (cd2, cd3, cd4)	0.5221	0.5207	0.9234	6
cd21	5	3xCNN+FFNN@U - (cd1, cd2, cd5)	0.5216	0.5188	0.9194	7
cd22	1	EfficientNetB0@CNN+FFNN - (cd1)	0.5216	0.5174	0.9306	8
cd23	10	ContrastiveTagger - (cd12)	N/A	0.4423	0.8112	30
cd24	11	MultiTaskContrastiveTagger - (cd13)*	0.5080	0.5092	0.9112	-

Furthermore, we observed that despite the relatively high performance in terms of the primary evaluation metric, our models were not able to achieve satisfactory results in the prediction of the under-represented concepts. In other words, the high F_1 -score levels were due to the system’s good performance in the common concepts (see Table 1), rather than to an overall classification ability. In an attempt to tackle this behaviour, we experimented with training a different instance of our CNN+FFNN classifier for each one of the four main modalities, in hopes that each classifier would be able to excel at some modality-specific characteristics. The results were mixed; two of the modality classifiers (X-Ray and MRI) were able to achieve almost 30% increase in their performance, while the other two performed even worse compared to the original version of the model. Overall, this approach did not manage to achieve more competitive results.

In Table 5, we list all the methods we experimented with during our participation in this year’s Concept Detection sub-task, along with the best score achieved in our development set for each one of the methods, as we experimented with numerous configurations (i.e. learning rate scheduler, number of hidden layers). To facilitate easier referencing in the rest of this section, we assign a unique ID to each method in the first column of Table 5. Moreover, in Table

6, we present an overview of our nine valid submissions regarding the Concept Detection task. We include each method’s performance on the primary F_1 -score in both the development and test subset, as well as the official results regarding the secondary evaluation metric. The last column contains the rank of our systems across all the task’s submitted runs.

Our team officially ranked 1st among 10 participating research groups in terms of the primary evaluation metric. Our best performing model was a UNION ensemble consisting of three instances of our CNN+FFNN system, where three different encoding backbones were used; EfficientNetB0 [41], EfficientNetB0v2 and DenseNet121 [42]. Furthermore, we ranked 2nd in the secondary evaluation metric by employing a single CNN+FFNN instance, using EfficientNetB0 [41] as the image encoder.

4.2. Caption Prediction

In the Caption Prediction sub-task, we also submitted nine systems, which were selected after evaluating them on our development set. We submitted two instances of our CNN-RNN encoder-decoder (see Section 3.2.1) and two instances of our Transformer-based ViT-GPT2 model (see Section 3.2.2). The difference between each model’s submissions lays in the number of beams used during the beam search decoding. We submitted instances where the beam size was equal to three and five, and therefore denote them as CNN-RNN-BS3, CNN-RNN-BS5, ViT-GPT2-BS3 and ViT-GPT2-BS5. In addition, we submitted five instances of our Seq2Seq denoising system employed on top of the four aforementioned submissions. The denoising models utilized were T5 [38], ClinicalT5, BART [37], as well as ClinicalBART (BART-large further pre-trained in ImageCLEF captions, see Section 3.2.3).

In this year’s campaign, BERTscore [54] was used as the primary evaluation metric, in contrast to last year that used BLEU [55]. ROUGE-1 [56] constitutes the secondary evaluation metric. Unlike BLEU and ROUGE-1, BERTscore [54] offers a more contextual evaluation system, as it leverages BERT’s [31] word embeddings and attempts to compute the semantic affinity between the words of the predicted and ground truth captions based on their cosine similarity.

Table 7

Summary of our submissions in the ImageCLEFmedical2023 Caption Prediction sub-task. The table contains the scores that our systems achieved in our held-out development set and the official test set, along with rankings among all the captioning systems submitted from 13 participating teams.

AUEB NLP Group - Submission Table							
ID	Run ID	Approach	BERTscore		ROUGE-1		Rank
			Dev	Test	Dev	Test	
cp1	2	BART@CNN-RNN-BS3	0.6141	0.6170	0.2111	0.2130	7
cp2	3	ClinicalBART@CNN-RNN-BS3	0.6118	0.6147	0.2123	0.2143	10
cp3	4	ClinicalT5@CNN-RNN-BS3	0.5923	0.6098	0.2146	0.2188	19
cp4	1	CNN-RNN-BS3	0.6046	0.6064	0.2249	0.2273	27
cp5	8	BART@CNN-RNN-BS5	0.6043	0.6058	0.1881	0.1884	29
cp6	9	CNN-RNN-BS5	0.5933	0.5960	0.2146	0.2155	35
cp7	6	BART@ViT-GPT2-BS5	0.5861	0.5879	0.1711	0.1708	38
cp8	7	ViT-GPT2-BS5	0.5703	0.5629	0.1787	0.1682	51
cp9	5	ViT-GPT2-BS3	0.5421	0.5416	0.1697	0.1682	65

Regarding our CNN-RNN model, we relied on our last year’s experiments and only adopted the encoding architectures that performed best; EfficientNetB0 [41] and DenseNet121 [42]. The encoder extracted the image representations, which we stored, before feeding them to the RNN decoding unit. We experimented with retrieving the image features from either a pre-trained CNN instance or the encoding unit of our best performing CNN+FFNN classification model, in hopes that it has learned to generate quality biomedical image representations through the training procedure. An interesting research point would be to try to train the CNN and the RNN encoder concurrently. Overall, the CNN-RNN encoder decoder achieved decent performance in the BERTscore [54] metric and, as in our last year’s participation [11] noteworthy scores in the ROUGE-1 [56] evaluation metric.

Our ViT-GPT2 model did not yield the expected results. We experimented with numerous configurations, like higher or lower learning rate along with scheduling techniques, increased generation penalty, as well as data augmentation. Specifically, we transformed each image on the fly, during the loading process. We first rotated it by an angle of 30 degrees towards a random direction and then resized it to $224 \times 224 \times 3$ pixels, which is the size that we selected to employ for every image. In this way, a slightly different view of the same image was passed to the encoding component in each epoch aiming to increase the data variety, improve the model’s robustness, as well as prevent it from quickly overfitting [53].

Our best submission, which managed to rank 7th out of 70 submitted systems, was the 2xE-D model, which is comprised of one of the aforementioned captioning models and a subsequent denoising component. Specifically, the instance that used BART outperformed the three other denoising models; T5 [38], ClinicalT5 and ClinicalBART (see section 3.2.3). We also experimented with multiple configurations, as well as decoding schemes. In this case, beam search decoding outperformed both nucleus and top- k sampling [33, 36] in multiple preliminary experiments.

Table 8

Summary of our submissions regarding the Caption Prediction sub-task. The table contains each system’s performance on all officially reported measures.

AUEB NLP Group Submissions - Evaluation on All Metrics								
ID	BERTscore	ROUGE-1	BLEURT	BLEU	METEOR	CIDEr	CLIPscore	Rank
cp1	0.6170	0.2130	0.2950	0.1692	0.0719	0.1466	0.8038	7
cp2	0.6147	0.2143	0.2877	0.1522	0.0695	0.1582	0.8059	10
cp3	0.6098	0.2188	0.2991	0.1919	0.0742	0.1447	0.7978	19
cp4	0.6064	0.2273	0.3048	0.2061	0.0789	0.1661	0.8025	27
cp5	0.6058	0.1884	0.2730	0.1222	0.0606	0.1275	0.8010	29
cp6	0.5960	0.2155	0.3050	0.2039	0.0807	0.1360	0.8043	35
cp7	0.5879	0.1708	0.2590	0.1340	0.0539	0.0815	0.7569	38
cp8	0.5629	0.1682	0.2793	0.1514	0.0655	0.0486	0.7602	51
cp9	0.5416	0.1682	0.2780	0.1322	0.0638	0.0388	0.7600	65

In Table 7, we present a summary of our nine submissions, including the method’s identifiers, their performance on the primary and secondary metric for both the development and test set, as well as its official rank across 70 submitted systems. Our group ranked 3rd among 13 teams in the Caption Prediction sub-task based on the primary evaluation metric. Our best model was BART@CNN-RNN-BS3, followed in close distance, by ClinicalBART@CNN-RNN-BS3, the biomedically-wise fine-tuned instance of the same system. In Table 8 we present our submissions’ performance on all the official metrics, as reported by the organizers, in order to provide a more thorough evaluation of their capabilities.

5. Conclusions

Regarding Concept Detection, our best-performing system was a CNN+FFNN pipeline (Section 3.1.1), while our remaining submissions included a CNN+FFNN-based multi-task classifier (Section 3.1.2), a contrastive learning-based system with a CLIP-like objective (Section 3.1.3) and ensembles employing the aforementioned approaches based on majority voting, union, intersection, as well as scaling by a factor λ in the case of our contrastive system. Our ensembles based on the CNN+FFNN pipeline, including its multi-task version, were ranked at positions 1, 2, 3, 4, 5, 6 and 7 among approximately 60 systems in the respective sub-task, which is consistent with their successful performance in previous years [10, 11], while our best-performing individual CNN+FFNN system was ranked at position 8 [2].

In the Caption Prediction sub-task, we ranked 3rd among the participating groups, by both extending our previous work [11] and exploiting the state-of-the-art methods in NLP. Our systems included a typical Show and Tell model [17] with a CNN backbone encoder and a recurrent decoder with GRU cells [32], a Transformer-based pipeline using a ViT encoder [19] and GPT-2 decoder [20], as well as a sequence-to-sequence [13] denoising autoencoder employed on top of the two other systems, in order to rephrase and correct the initial draft radiology reports.

In future work, we plan to expand our research in biomedical LLMs and their reasoning abilities, towards the goal of exploiting the generative capabilities of models like BioGPT [57] or BioMedLM [58] to produce high-quality captions; possibly via instruction tuning and, more generally, alignment with user needs [59]. Furthermore, apart from making use of the knowledge encoded in the weights of the LLMs, we aim to shed light in the use of dense retrieval [60] in biomedical image captioning [5, 29], based on architectures similar to Retrieval Augmented Generation [61]. Such pipelines will allow us to increase the LLMs’ capacity by an additional, non-parametric memory, in the form of a FAISS index [62], towards the goal of improving their reasoning abilities. We would also be interested to discover potential associations between the two sub-tasks. Last but not least, the qualitative differences in the captions generated by the different methods are to be considered, since they highlight their practical usefulness in real-life scenarios [5, 29].

References

- [1] B. Ionescu, H. Müller, A. Drăgulinescu, W. Yim, A. Ben Abacha, N. Snider, G. Adams, M. Yetisgen, J. Rückert, A. Garcia Seco de Herrera, C. M. Friedrich, L. Bloch, R. Brüngel, A. Idrissi-Yaghir, H. Schäfer, S. A. Hicks, M. A. Riegler, V. Thambawita, A. Storås, P. Halvorsen, N. Papachrysos, J. Schöler, D. Jha, A. Andrei, A. Radzhabov, I. Coman, V. Kovalev, A. Stan, G. Ioannidis, H. Manguinhas, L. Ştefan, M. G. Constantin, M. Dogariu, J. Deshayes, A. Popescu, Overview of ImageCLEF 2023: Multimedia retrieval in medical, socialmedia and recommender systems applications, in: *Experimental IR Meets Multilinguality, Multimodality, and Interaction, Proceedings of the 14th International Conference of the CLEF Association (CLEF 2023)*, Springer Lecture Notes in Computer Science LNCS, Thessaloniki, Greece, 2023.
- [2] J. Rückert, A. Ben Abacha, A. G. Seco de Herrera, L. Bloch, R. Brüngel, A. Idrissi-Yaghir, H. Schäfer, H. Müller, C. M. Friedrich, Overview of ImageCLEFmedical 2023 – Caption Prediction and Concept Detection, in: *CLEF2023 Working Notes*, CEUR Workshop Proceedings, CEUR-WS.org, Thessaloniki, Greece, 2023.
- [3] J. Pavlopoulos, V. Kougia, I. Androutsopoulos, D. Papamichail, Diagnostic captioning: a survey, *Knowledge and Information Systems* 64 (2022) 1–32.
- [4] H. C. Shin, K. Roberts, L. Lu, D. Demner-Fushman, J. Yao, R. M. Summers, Learning to Read Chest X-Rays: Recurrent Neural Cascade Model for Automated Image Annotation, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016) 2497–2506.
- [5] G. Moschovis, Medical image captioning based on Deep Architectures, Master’s thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2022. URL: <http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-323528>, Last accessed: 2023-07-07.
- [6] J. Pavlopoulos, V. Kougia, I. Androutsopoulos, A Survey on Biomedical Image Captioning, in: *Proceedings of the Second Workshop on Shortcomings in Vision and Language*, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 26–36.
- [7] V. Kougia, J. Pavlopoulos, I. Androutsopoulos, AUEB NLP group at ImageCLEFmed Caption 2019, in: *Working Notes of CLEF 2019 - Conference and Labs of the Evaluation Forum*, Lugano, Switzerland, September 9-12, volume 2380 of *CEUR Workshop Proceedings*, 2019.
- [8] B. Karatzas, J. Pavlopoulos, V. Kougia, I. Androutsopoulos, AUEB NLP group at ImageCLEFmed Caption 2020, in: *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*, Thessaloniki, Greece, September 22-25, volume 2696 of *CEUR Workshop Proceedings*, 2020.
- [9] V. Kougia, J. Pavlopoulos, I. Androutsopoulos, Medical Image Tagging by Deep Learning and Retrieval, in: *Experimental IR Meets Multilinguality, Multimodality, and Interaction: 11th International Conference of the CLEF Association*, CLEF 2020, Thessaloniki, Greece, September 22–25, 2020, *Proceedings*, 2020, p. 154–166.
- [10] F. Charalampakos, V. Karatzas, V. Kougia, J. Pavlopoulos, I. Androutsopoulos, AUEB NLP group at ImageCLEFmed Caption tasks 2021, in: *Proceedings of the Working Notes of CLEF 2021 - Conference and Labs of the Evaluation Forum*, Bucharest, Romania, September 21-24, volume 2936 of *CEUR Workshop Proceedings*, 2021, pp. 1184–1200.
- [11] F. Charalampakos, G. Zachariadis, J. Pavlopoulos, V. Karatzas, C. Trakas, I. Androutsopoulos, AUEB NLP Group at ImageCLEFmedical Caption 2022, in: *CLEF2022 Working Notes*,

- CEUR Workshop Proceedings, CEUR-WS.org, Bologna, Italy, 2022, pp. 1355–1373.
- [12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is All you Need, in: *Advances in Neural Information Processing Systems*, volume 30, 2017.
 - [13] I. Sutskever, O. Vinyals, Q. V. Le, Sequence to Sequence Learning with Neural Networks, *NIPS'14*, MIT Press, Cambridge, MA, USA, 2014, p. 3104–3112.
 - [14] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, Y. Du, C. Yang, Y. Chen, Z. Chen, J. Jiang, R. Ren, Y. Li, X. Tang, Z. Liu, P. Liu, J. Nie, J. Wen, A survey of large language models, *ArXiv abs/2303.18223* (2023).
 - [15] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A Simple Framework for Contrastive Learning of Visual Representations, in: *Proceedings of the 37th International Conference on Machine Learning, ICML'20*, 2020.
 - [16] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, Learning Transferable Visual Models from Natural Language Supervision, in: *Proceedings of the 38th International Conference on Machine Learning*, volume 139, PMLR, 2021, pp. 8748–8763.
 - [17] O. Vinyals, A. Toshev, S. Bengio, D. Erhan, Show and Tell: A neural image caption generator, *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2014) 3156–3164.
 - [18] M. Stefanini, M. Cornia, L. Baraldi, S. Cascianelli, G. Fiameni, R. Cucchiara, From Show to Tell: A survey on Deep Learning-Based Image Captioning, *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022).
 - [19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, in: *International Conference on Learning Representations*, 2021.
 - [20] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, *OpenAI blog* 1 (2019).
 - [21] O. Bodenreider, The Unified Medical Language System (UMLS): integrating biomedical terminology, *Nucleic Acids Research* 32 (2004).
 - [22] O. Pelka, S. Koitka, J. Rückert, F. Nensa, C. Friedrich, Radiology Objects in Context (ROCO): A Multimodal Image Dataset: 7th Joint International Workshop, CVII-STENT 2018 and Third International Workshop, LABELS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, *Proceedings*, 2018, pp. 180–189.
 - [23] G. Liu, T. H. Hsu, M. B. A. McDermott, W. Boag, W. Weng, P. Szolovits, M. Ghassemi, Clinically Accurate Chest X-Ray Report Generation, in: *Proceedings of the Machine Learning for Healthcare Conference, MLHC 2019*, 9-10 August 2019, Ann Arbor, Michigan, USA, volume 106 of *Proceedings of Machine Learning Research*, PMLR, 2019, pp. 249–269.
 - [24] F. Charalampakos, Exploring Deep Learning Methods for Medical Image Tagging, Master's thesis, Athens University of Economics and Business, Athens, Greece, 2022.
 - [25] F. Radenović, G. Tolias, O. Chum, Fine-tuning CNN image retrieval with no human annotation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41 (2019) 1655–1668.
 - [26] A. Zafar, M. Aamir, N. Nawi, A. Arshad, S. Riaz, A. Alruban, A. Dutta, S. Alaybani, A

Comparison of Pooling Methods for Convolutional Neural Networks, *Applied Sciences* 12 (2022) 8643.

- [27] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent Neural Networks from overfitting, *Journal of Machine Learning Research* 15 (2014) 1929–1958.
- [28] D. Kingma, J. Ba, Adam: A method for stochastic optimization, *International Conference on Learning Representations* (2014).
- [29] G. Moschovis, E. Fransén, Neurdynamicslab at imageclef medical 2022, in: CLEF2022 Working Notes, CEUR Workshop Proceedings, CEUR-WS.org, Bologna, Italy, 2022.
- [30] A. Gotmare, N. S. Keskar, C. Xiong, R. Socher, A Closer Look at Deep Learning Heuristics: Learning rate restarts, Warmup and Distillation, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, 2019.
- [31] J. Devlin, M. W. Chang, K. Lee, K. Toutanova, 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: Human Language Technologies, Volume 1 (Long and Short Papers)*, Association for Computational Linguistics, Minneapolis, Minnesota, USA, 2019, pp. 4171–4186.
- [32] J. Chung, Ç. Gülçehre, K. Cho, Y. Bengio, Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling, *CoRR* (2014).
- [33] S. Zarrieß, H. Voigt, S. Schüz, Decoding Methods in Neural Language Generation: A Survey, *Information* 12 (2021).
- [34] M. Naseer, M. Hayat, S. W. Zamir, F. Khan, M. Shah, Transformers in Vision: A Survey, *ACM Computing Surveys* 54 (2022).
- [35] T. Li, Y. E. Mesbahi, I. Kobyzev, A. Rashid, A. Mahmud, N. Anchuri, H. Hajimolahoseini, Y. Liu, M. Rezagholizadeh, A short study on compressing decoder-based Language Models, *ArXiv abs/2110.08460* (2021).
- [36] G. Wiher, C. Meister, R. Cotterell, On Decoding Strategies for Neural Text Generators, *Transactions of the Association for Computational Linguistics* 10 (2022) 997–1012.
- [37] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, 2020, pp. 7871–7880.
- [38] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of Transfer Learning with a Unified Text-to-Text Transformer, *J. Mach. Learn. Res.* 21 (2020) 140:1–140:67.
- [39] W. Qi, P. Stetson, A study of abbreviations in clinical notes, *AMIA, Annual Symposium proceedings / AMIA Symposium* (2007) 821–5.
- [40] A. Johnson, T. Pollard, L. Shen, L. W. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Celi, R. Mark, MIMIC-III, a freely accessible critical care database, *Scientific Data* 3 (2016) 160035.
- [41] M. Tan, Q. V. Le, Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks, in: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, volume 97 of Proceedings of Machine Learning*

- Research*, 2019, pp. 6105–6114.
- [42] G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely Connected Convolutional Networks, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2261–2269.
 - [43] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.
 - [44] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, *CoRR* (2017).
 - [45] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1–9.
 - [46] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Y. Ding, A. Bagul, C. P. Langlotz, K. S. Shpanskaya, M. P. Lungren, A. Y. Ng, CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning, *CoRR* abs/1711.05225 (2017).
 - [47] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, in: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015.
 - [48] A. Krizhevsky, One weird trick for parallelizing convolutional neural networks, *CoRR* (2014).
 - [49] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, L. Fei-Fei, ImageNet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.
 - [50] I. Athanasiadis, G. Moschovis, A. Tuoma, Weakly-Supervised Semantic Segmentation via Transformer Explainability, in: ML Reproducibility Challenge 2021 (Fall Edition), 2022.
 - [51] I. Kandel, M. Castelli, How Deeply to Fine-Tune a Convolutional Neural Network: A Case Study Using a Histopathology Dataset, *Applied Sciences* 10 (2020).
 - [52] R. M. French, Catastrophic forgetting in connectionist networks, *Trends in Cognitive Sciences* 3 (1999) 128–135.
 - [53] A. Mikołajczyk, M. Grochowski, Data augmentation for improving deep learning in image classification problem, in: 2018 International Interdisciplinary PhD Workshop (IIPhDW), 2018, pp. 117–122.
 - [54] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, BERTScore: Evaluating Text Generation with BERT, in: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020.
 - [55] K. Papineni, S. Roukos, T. Ward, W. J. Zhu, Bleu: a method for automatic evaluation of machine translation, in: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, 2002, pp. 311–318.
 - [56] C. Y. Lin, ROUGE: A package for automatic evaluation of summaries, in: Text Summarization Branches Out, Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 74–81.
 - [57] R. Luo, L. Sun, Y. Xia, T. Qin, S. Zhang, H. Poon, T. Y. Liu, BioGPT: generative pre-trained transformer for biomedical text generation and mining, *Briefings in Bioinformatics* (2022).

- [58] E. Bolton, D. Hall, M. Yasunaga, T. Lee, C. Manning, P. Liang, BioMedLM, 2022.
- [59] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Gray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, R. Lowe, Training language models to follow instructions with human feedback, in: *Advances in Neural Information Processing Systems*, 2022.
- [60] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, W. Yih, Dense Passage Retrieval for Open-Domain Question Answering, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Online, 2020, pp. 6769–6781.
- [61] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, in: *Advances in Neural Information Processing Systems*, volume 33, Curran Associates, Inc., 2020, pp. 9459–9474.
- [62] J. Johnson, M. Douze, H. Jégou, Billion-scale similarity search with GPUs, *IEEE Transactions on Big Data* 7 (2019) 535–547.