Automated Integration of Genomic Metadata with Sequence-to-Sequence Models

Giuseppe Cannizzaro, Michele Leone, Anna Bernasconi, Arif Canakoglu, and Mark J. Carman (✉)

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Via Ponzio 34/5, 20133, Milano, Italy
giuseppe.cannizzaro@mail.polimi.it,{first.last}@polimi.it

Abstract. While exponential growth in public genomic data can afford great insights into biological processes underlying diseases, a lack of structured metadata often impedes its timely discovery for analysis. In the Gene Expression Omnibus, for example, descriptions of genomic samples lack structure, with different terminology (such as "breast cancer", "breast tumor", and "malignant neoplasm of breast") used to express the same concept. To remedy this, we learn models to extract salient information from this textual metadata. Rather than treating the problem as classification or named entity recognition, we model it as machine translation, leveraging state-of-the-art sequence-to-sequence (seq2seq) models to directly map unstructured input into a structured text format. The application of such models greatly simplifies training and allows for imputation of output fields that are implied but never explicitly mentioned in the input text.
We experiment with two types of seq2seq models: an LSTM with attention and a transformer (in particular GPT-2), noting that the latter outperforms both the former and also a multi-label classification approach based on a similar transformer architecture (RoBERTa). The GPT-2 model showed a surprising ability to predict attributes with a large set of possible values, often inferring the correct value for unmentioned attributes. The models were evaluated in both homogeneous and heterogenous training/testing environments, indicating the efficacy of the transformer-based seq2seq approach for real data integration applications.

Keywords: genomics · high-throughput sequencing · metadata integration · deep learning · translation models · natural language processing

1 Introduction

Technologies for DNA sequencing have made incredible steps in the last decade, producing rapidly expanding quantities of various types of genomic data with ever lower costs[1] and faster production times. Biologists and bioinformaticians

[1] Companies currently offer complete genome sequencing for under 600USD (e.g. https://www.veritasgenetics.com/myGenome) with costs expected to fall.

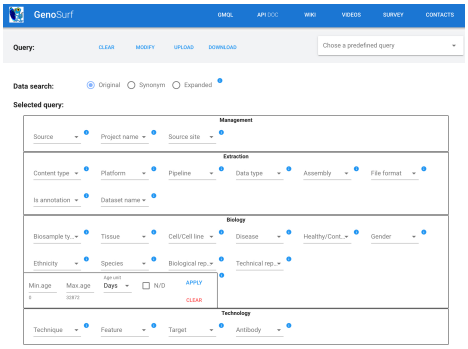

Fig. 1. GenoSurf, a metadata driven search interface for genomic datasets

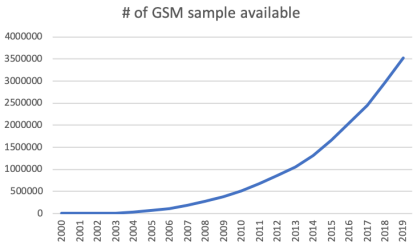

Fig. 2. Growth over time of # samples available in the GEO database

need access to such datasets for their everyday work, and open data is available through various platforms. Unfortunately, each platform enforces its own data model and formats, and this heterogeneity can hinder data analysis. There is need to integrate resources [4] to prevent scientists from missing relevant data or wasting time on data preparation. Metadata-driven search engines such as GenoSurf [7] attempt to do this by allowing users to search for genomic samples with given characteristics using a structured interface (see Fig. 1). GenoSurf integrates metadata schemas from important genomic sources (ENCODE [8], TCGA [31], 1000 Genomes [27], and Roadmap Epigenomics [18]), but misses samples from the largest public genomic repository, the Gene Expression Omnibus (GEO) [2].

The data in GEO is of fundamental importance to the scientific community for understanding various biological processes, including species divergence, protein evolution and complex disease. The number of samples in the database is growing exponentially (see Fig. 2), and while tools for retrieving information from GEO datasets exist[2], large-scale analysis is complicated due to heterogeneity in the data processing across studies and most importantly in the metadata describing each experiment. When submitting data to the GEO repository, scientists enter experiment descriptions in a spreadsheet (see Fig. 3) where they can provide unstructured information and create arbitrary fields that need not adhere to any predefined dictionary. The validity of the metadata is not checked at any point during the upload process[3], thus the metadata associated with gene expression data, usually does not match with standard class/relation identifiers from specialized biomedical ontologies. The resulting free-text experiment descriptions suffer from redundancy, inconsistency, and incompleteness [32].

[2] NCBI E-utilities [17] provide a federated search engine supporting information on experimental protocols, but lack functionality regarding characteristics of the sample, such as species of origin, age, gender, tissue, mutations, disease state, etc.

[3] Information regarding the submission of high-throughput sequences is provided at `https://www.ncbi.nlm.nih.gov/geo/info/seq.html`.

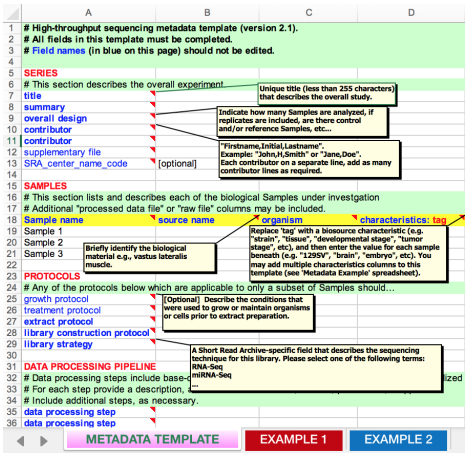

Fig. 3. Spreadsheet for describing data when submitting dataset to GEO

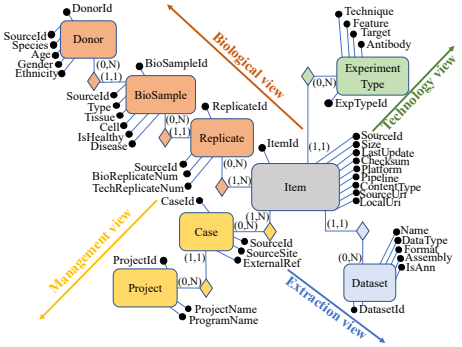

Fig. 4. Genomic Conceptual Model

In this paper, we develop automated machine learning methods for extracting structured information from the heterogeneous GEO metadata. Our aim is to populate a structured database with attributes according to the Genomic Conceptual Model (GCM) [5], which recognizes a limited set of concepts[4] (shown in Fig. 4), that are supported by most genomic data sources. GCM provides a common language for genomic dataset integration pipelines (see META-BASE framework [3]), fuelling user search-interfaces such as GenoSurf.

The main contributions of this paper are the following:

1) We provide a novel formulation of the metadata integration problem as a machine translation (MT) problem, which has a number of benefits over both a named-entity recognition (NER) based approach (since there is no requirement for annotating input training sequences), and over a multi-label classification based approach (since the same model architecture can be used regardless of the target attributes to be extracted).
2) We provide experimental evidence demonstrating the effectiveness of the transformer-based translation models over simpler attention based seq2seq models and over the classification based approach using a similar transformer architecture. Experiments are performed in both homogeneous and heterogeneous training/testing environments, indicating the ability of the seq2seq model to impute values often unobserved in the input, and the efficacy of the approach for real data integration applications.

In the next section we discuss related work on integration for genomics resources. In Section 3, we overview the multi-label classification and the translation-based approaches studied for the proposed problem. In Section 4, we describe

[4] The data model centers on the *item* of experimental data, with views describing *biological* elements, *technology* used, *management* aspects, and *extraction* parameters.

the experiments, including the used datasets, the setup configuration and results. Section 5 concludes the paper.

2 Related Work

There is a compelling need to structure information in large biological datasets so that metadata describing experiments is available in a standard format and is ready for use in large-scale analysis [16]. In recent years, several strategies for annotating and curating GEO database metadata have been developed (see Wang et al. [29] for a survey). We group the approaches into five non-exclusive categories: 1) manual curation, 2) regular expressions, 3) text classification, 4) named-entity recognition, and 5) imputation from gene expression.

Manual curation: Structured methods for authoring and curating metadata have been promoted by numerous authors [14, 24, 20]. Moreover, a number of biological metadata repositories (e.g. RNASeqMetaDB [13], SFMetaDB [19] and CREEDS [30]) manually annotate their datasets, guaranteeing high accuracy. This option is however, highy time-consuming and hardly practicable as the volume and diversity of biological data grows.

Regular expressions: The use of regular expressions for extracting structured metadata fields from unstructured text is common [12]. This simple technique is limited, however, to matching patterns that are: *expressible*, yet identifiers for biological entities often do not follow any particular pattern (e.g., IMR90, HeLa-S3, GM19130); *explicit*, i.e. matching the cell line K562 cannot produce the implied sex, age, or disease information[5]; and *unique*, i.e. it cannot discern between multiple string matches in the same document.

Text classification: Machine learning techniques can be used to predict the value of metadata fields based on unstructured input text. Posch et al. [25] proposed a framework for predicting structured metadata from unstructured text using tf-idf and topics modeling based features. The limitations of the classification approach include that a separate model needs to be trained for each attribute to extract and that values of the attribute need to be known in advance.

Named-entity recognition: NER models are often used to extract knowledge from free text data including medical literature. They work by identifying spans within an input text that mention named entities, and then assigning these spans into predefined categories (such as *Cell Line*, *Tissue*, *Sex*, etc.). By learning their parameters and making use of the entire input sentence as context, these systems overcome the limitations of simple regular expression based approaches. In particular, certain works [11, 16] have employed NER to to map free textual description into existing concepts from curated specialized ontologies that are well-accepted by the biomedical community [6] to improve the integrated use of heterogeneous datasets. In practice, training NER models can be difficult, since the training sequences must be labelled on an individual word level. This is especially time consuming in the genomics domain, where biomedical fields require

[5] K562 is a widely known cell line originally extracted from tissue belonging to a 53 year-old woman affected by myeloid leukemia.

specific and technical labels. Moreover, there is no way to make use of publicly available curated datasets that overlap with GEO to synthesize training data, since the information they contain applies to the entire GEO sample as a whole. A further drawback of NER methods, is that they can produce false positives with high frequency, due to misleading information in samples' descriptions (e.g., presence of pathologies in the family of an healthy patient).

Imputation from gene expression: The automated label extraction (ALE) platform [12], trains ML models based on high-quality annotated gene expression profiles (leveraging on text-extraction approaches based on regular expression and string matching). However, the information is limited to a small set of patient's characteristics (i.e., gender, age, tissues). Authors in [10] also predict sample labels using gene expression data; a model is built and evaluated for both biological phenotypes (sex, tissue, sample source) and experimental conditions (sequencing strategy). The approach is applied on repositories alternative to GEO (i.e., training from TCGA samples, testing on GTEx [22] and SRA).

Each of the aforementioned approaches to genomic metadata extraction have their limitations. As we will discuss in the next section, many (or all) of these limitations can be overcome by making use of a translation (a.k.a. sequence-to-sequence) modeling approach. To the best of our knowledge, no previous work has applied this approach to the problem of automating the integration of experiment metadata before.

3 Approaches

We now discuss two different approaches that we applied to the metadata extraction problem. Both leverage recent advances in Deep Learning for text analysis. The first approach builds a multi-label classifier to predict metadata attribute values using a deep embedding, and will serve as our baseline for later experiments. The second makes use of a novel translation-based approach where powerful sequence-to-sequence models are leveraged to solve the metadata extraction problem in a more elegant and extensible fashion.

3.1 Multi-label Classification Approach

To model the metadata extraction problem using a classification approach, we can simply turn the attribute-value prediction problem into a *multi-label classification problem*, by treating each possible value for each attribute as a separate class to be predicted. An alternative would be to model the task as a *multi-task multi-class classification problem*, where each attribute is associated with its own softmax function (thereby constraining that each attribute must appear in the output and must take on a single – possibly *unknown* – value). For simplicity and extensibility purposes we choose instead to model the task as a single multi-label classification problem where each attribute-value is associated with its own sigmoid function. We then use a post-processing to select the most likely attribute value for each attribute. We note that the classification approach requires that

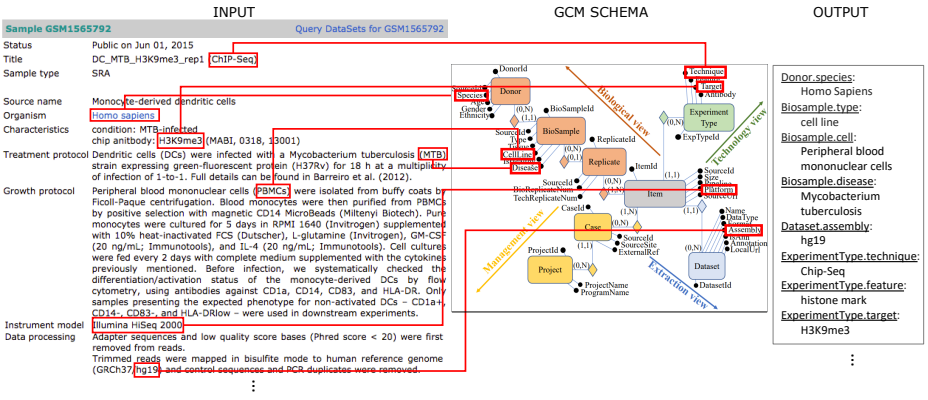

Fig. 5. Example mapping task: from GEO sample GSM1565792 input text, into GCM attributes, to finally produce output key-value pairs

each attribute have a finite set of values and each value must be known at training time. It is most suitable for extracting attributes with a relatively small number of possible values, and does not accommodate the situation where an attribute needs to take on multiple values at once (e.g. because a single GEO sample contains data for multiple cell lines).

RoBERTa: To build an embedding from the input text that can provide the feature space for the classifier, we make use of RoBERTa [21], a variation on the BERT [9] language model. These self-attention based transformer models [28] have recently shown state-of-the-art performance for all kinds of text classification tasks owing to the pre-training of the language model in an unsupervised fashion on large text corpora. To build the multi-label classifier, a dense feed-forward layer is place on top of the transformer stack. The last layer presents a number of neurons equal to the total number of attribute-values, (the target attributes are one-hot-encoded for the multi-label model).

3.2 Translation-based Approach

We treat the problem of extracting metadata from unstructured text as a *translation task*, where instead of translating input text into another language, we translate it into a well structured list of extracted attributes. By approaching the problem in this fashion, many strengths of translation models can be exploited: i) translation models do not expect translated text to follow string patterns; ii) translation models do not use a lookup approach (i.e., they can disambiguate correctly input words whenever the text contains multiple possible choices for a certain concept); iii) translation models do not expect a fixed number of output values; iv) translation models can extract hidden information from the text context.

Input and output formats: Each training sample is composed of *input-output* pairs, where *input* corresponds to the textual description of a biological

Table 1. Size of Encoder and Decoder networks

Network	Layer	Size
Encoder	Embedding	256
	LSTM	512
Decoder	Embedding	256
	LSTM	512
	Tan_h	512
	Dense	Vocab_size

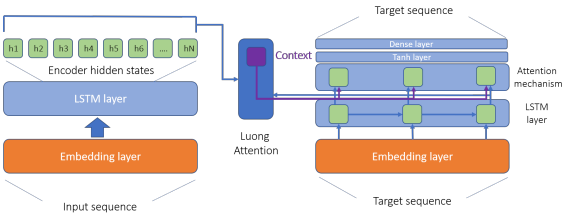

Fig. 6. Encoder-Decoder structure with attention mechanism

sample and *output* is a list of attribute-value pairs. Fig. 5 shows an example translation task: on the left, a metadata record from GEO repository describing a human biological sample, in the middle the target schema, and on the right the resulting output pairs. The text output produced by the translation model should be human and machine readable, so we used a dash-separated list of "key: value" pairs, `Cell Line: HeLa-s3 - Cell Type: Epithelium - Tissue: Cervix`.

We now discuss the Encoder-Decoder LSTM and OpenAI GPT-2 architectures employed as translation models in our experiments.

Encoder-Decoder LSTM: This model is composed of two LSTM networks, an encoder and a decoder, and exploits a Luong attention [23] mechanism. The *encoder* is composed of an embedding layer plus an LSTM layer, which provides hidden states to feed the attention mechanism for the decoding phase. The *decoder* is composed of an embedding layer, an LSTM layer and 2 dense layers as shown in Fig. 6. We report the number of neurons of each layer of the encoder and decoder network in Table 1. The size of the dense layer depends on the vocabulary size (and is thus determined by the tokenizer). The two dense layers are needed for the attention mechanism: The output of the LSTM layer is concatenated with the context vector, thus doubling the size of the vectors coming from the LSTM layer. The first dense layer re-shapes the LSTM output to the same size as the LSTM, while the second maps the output of the first dense layer to the size of the vocabulary. The vocabulary token with the highest probability is then predicted for each time step.

The embedding layer of the Encoder is fed with a tokenized version of the input text and is executed once for each sample (batch of items). The decoding phase takes place iteratively, thus the output is generated token-by-token. At each i time step (corresponds to a single token), the embedding layer of the decoder is fed with a tokenized version of the output text starting from the *start* token (<*start*>) reaching the i-th token. The decoder is trained to generate the $i+1$-th token until the entire sequence has been generated, producing a termination token (<*end*>). The decoder exploits the attention mechanism, as shown in Fig. 6.

The training of LSTM model is performed by learning conditioned probabilities of the next token over the entire vocabulary, given the (embedding for the) current input token, the previous state and the sequence up to that point

(exploiting the attention mechanism). Each token is a tensor that represents a one-hot-encoding over the entire vocabulary.

To evaluate LSTM performance, we generated the output sequences for the input strings in the test set as follows: the model encodes each input sequence, the decoder generates the predicted probabilities over the entire vocabulary given the input and the $< start >$ token. The most probable output token is then selected and concatenated to the input sequence after the $< start >$ token. After that point, the decoder generates a prediction given the input and the generated sequence; the procedure goes on iteratively until the termination token ($< end >$) is generated. In the unlikely case of a generation process that does not end (because the termination character is never generated), production is stopped when the output sequence reaches the maximum output length in the training set.

OpenAI GPT-2 is a more powerful sequence-to-sequence pre-trained language model [26], whose structure is based on Transformer Decoders [28]. Text generation is done in a similar fashion as encoder-decoder, i.e., a generation token-by-token. Differently from LSTM models, the text generation phase is not preceded by an encoding phase. This means that the model is not trained on input-output pairs; instead, it is trained on single sequences. Thus, we prepared sentences composed of both input and output, separated with the "=" character and terminated by '$' (e.g., `[Input sentence] = Cell line: HeLa-S3 - Cell Type: Epithelium - Tissue Type: Cervix - Factor: DNase $`).

GPT-2 training is performed by learning conditioned probabilities of the next token over the entire vocabulary, given only the sequence of previous tokens. As with the LSTM, each token is a tensor that represents a one-hot-encoding over the entire vocabulary. To evaluate GPT-2 performance, we generated the output sequences for the input strings in the test set employing a similar approach as with the Encoder-Decoder LSTM model. GPT-2 outputs the probabilities over the entire vocabulary for a given input sequence which terminates with "=". The output token with the highest probability is then concatenated to the input sequence. The model then outputs the probabilities given the new input sequence (which is composed of the input sequence used at previous step concatenated to the generated token); the process goes on until the termination token ($) is generated.

In both described translation models, after the entire sequence is generated, the tokenized output sequence is decoded back to text.

4 Experiments

Our experiments aim to evaluate and compare results of two seq2seq models: a simple Encoder-Decoder model using a Long Short-Term Memory (LSTM) layer with Luong attention [23], and the OpenAI Generative Pretrained Transformer 2 (GPT-2) Language Model [26], which makes use of transformer decoder cells and has been proved to perform very well in NLP tasks, in particular in those regarding text generation. As our baseline, we used the RoBERTa multi-label

Table 2. Cistrome attributes: percentage of 'None' and count of distinct values

Attributes	%'None'	#distinct values
Cell line	52	519
Cell type	19	152
Tissue type	29	82
Factor	0	1252

Table 3. ENCODE attributes: percentage of 'None' and count of distinct values

Attributes	%'None'	#distinct values
Age	1	169
Age units	32	6
Assay Name	0	26
Assay type	0	9
Biosample term name	0	9
Classification	1	6
Ethnicity	74	15
Genome assembly	16	11
Health status	53	65
Investigated as	48	22
Life stage	1	17
Organism	1	5
Project	0	3
Sex	1	10
Target of assay	48	344

classification model [21]. In the following, we first describe the data that we use in the experiments, we then detail the experiment setup and finally report on the obtained results. The code used in the experiments is publicly available[6].

4.1 Datasets: GEO, Cistrome and ENCODE

We make use of data from GEO, Cistrome and ENCODE for our experiments.

GEO: Input text descriptions are taken from the GEOmetadb database [34]. We extracted the *Title*, *Characteristics_ch1*, and *Description* fields, which include information about the biological sample from the `gsm` table. We format the input by alternating a field name with its content and separating each pair with the dash "-" character, e.g., `Title: [...] - Characteristics: [...] - Description: [...]`. In this way, we allow the model to learn possible information patterns, for example, information regarding "Cell Line" is often included in the "Title" section. We pre-processed the input text by replacing special characters (i.e., !@#$&̂*[]?\—'˜_+") with spaces and by removing "\n" and "\t".

Cistrome: The Cistrome Data Browser [33] provides a collection of publicly available data derived from the GEO Database. More specifically, it contains ChIP-seq and chromatin accessibility experiments, two techniques used to analyze protein interactions with DNA and physically accessible DNA areas, respectively. Importantly, the samples in Cistrome have been manually curated and annotated with the *cell line*, *cell type*, *tissue type*, and *factor name*. We downloaded in total 44,843 metadata entries from Cistrome Data Browser[7] with the four mentioned attributes. As indicated in Table 2, three of the fields contain many "None" values, but these should not be interpreted as missing, since they actually indicate that the specific sample does not carry that kind of information.

ENCODE: The Encyclopedia of DNA Elements [8] is a public genomic repository of datasets related to functional DNA sequences and to the regula-

[6] `https://github.com/DEIB-GECO/GEO-metadata-translator`

[7] `http://cistrome.org/db/\#/bdown`

Table 4. Setup of the three different models for each experiment (BPE = Byte Pair Encoding; LR = learning rate)

Model	Batch size	Loss function	Tokenizer	Optimizer	LR	beta_1	beta_2	epsilon
RoBERTa	10	Cross Entropy	BPE	Adam	2e-4	0.9	0.999	1e-6
LSTM	64	Sparse Cross Entropy	keras	Adam	1e-3	0.9	0.999	1e-7
GPT-2	5	Cross Entropy	BPE	Adam	1e-3	0.9	0.999	1e-6

tory elements that control gene expression. The ENCODE Consortium exploited manual curation to collect and organize metadata for the DNA sequences [15], making the repository one of the most complete and accurate genomic archives from the point of view of data description. We downloaded 16,732 metadata entries from ENCODE web portal[8], by requesting the fields listed in Table 3 for each experiment sample. The free text input related to each sample, was retrieved by either: (i) exploiting a reference to the GEO GSM (only available for 6,233 entries) or (ii) by concatenating the additional ENCODE fields *Summary*, *Description* and *Biosample Description*.

4.2 Experimental Setup

We designed three experiments to validate our proposal. Experiment 1 and 2 allow to compare performances of the three analyzed models on two different datasets: Cistrome (with input from GEO) and ENCODE (with input both from GEO and ENCODE itself). Experiment 3, instead, tested the performance of the best proposed model on randomly chosen instances from GEO.

The Transformer library from HuggingFace[9] was used for the GPT-2 model, while the SimpleTransformers library[10] was used for the RoBERTa model. The LSTM encoder-decoder was built with Tensorflow [1] version 2.1 using the Keras API. For the LSTM model, we performed the tokenization process using the default Keras tokenizer, setting the API parameters to convert all characters into lower case, using empty space as a word separator, and disabling character-level tokenization. We added a space before and after the following characters: opening/closing parenthesis, dashes, and underscores[11]. We also removed equal signs. For the LSTM models, the resulting vocabulary had a size of 36,107 for Experiment 1 and 17,880 for Experiment 2.

RoBERTa and GPT-2 were trained using a Tesla P100-PCIE-16GB GPU, while the LSTM model was trained on Google Colaboratory[12] with GPU accelerator. Table 4 lists the configurations for the systems. All models were subject to early stopping method to avoid over-fitting.

[8] `https://www.encodeproject.org/`

[9] `https://github.com/huggingface/transformers`

[10] `https://github.com/ThilinaRajapakse/simpletransformers`

[11] Pre-processing was motivated by the fact that important character ngrams often appear in sequences separated by special characters, e.g., "RH_RRE2_14028".

[12] `https://colab.research.google.com/`

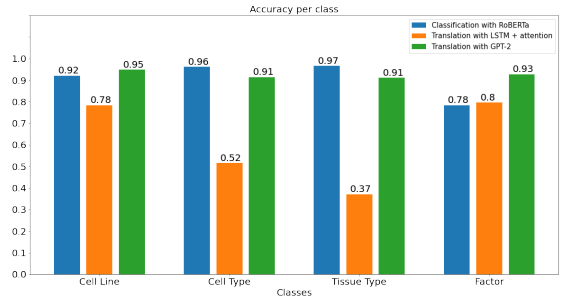

Fig. 7. Experiment 1: per-class accuracy for the three models on Cistrome data.

Table 5. Experiment 1: overall accuracy, precision, and recall. Precision and recall are weighted by the number of occurrences of each attribute value.

Model	# Epochs	Accuracy	Precision	Recall
RoBERTa	69	0.90	0.89	0.91
LSTM + Attention	15	0.62	0.65	0.62
GPT-2	47	0.93	0.93	0.93

For Experiment 1 and 2, data was split into training set (80%), validation set (10%) and test set (10%). Some text cleaning and padding processes were adopted: Encoder-Decoder requires input-output pairs that are padded to the maximum length of concatenation of input and output; GPT-2 requires single sentences that are padded to a maximum length of 500 characters. We excluded sentences exceeding the maximum length.

4.3 Experiments 1 and 2

We evaluated the performances of LSTM (with attention mechanism) and GPT-2 seq2seq models against RoBERTa, using samples from Cistrome (Experiment 1) and samples from ENCODE (Experiment 2).

In both experiments, overall GPT-2 outperforms both Encoder-Decoder LSTM and RoBERTa, as it can be observed in Tables 5-6. Results divided by class are shown in in Fig. 7 for Experiment 1 and in Fig. 8 for Experiment 2.

Experiment 1 considerations. From Fig. 7, RoBERTa seems to perform better for classes that contain a low number of distinct values, i.e. *cell type* and *tissue type* (which contain 380 and 249 possible values). Instead, for *cell line* and *factor* (both with more than a thousand possible values) GPT-2 outperforms RoBERTa. The number of "None" values is taken into consideration (Table 2), the classes *cell line*, *cell type* and *tissue type* present a relevant percentage of "None", the weighted precision and recall analysis, however, shows high scores, despite the unbalance of values count; this implies that the models were able to correctly classify samples which lack of labels for certain classes.

Experiment 2 considerations. From Fig. 8, we appreciate a similar behaviour as in Experiment 1, i.e., translation models perform better for attributes

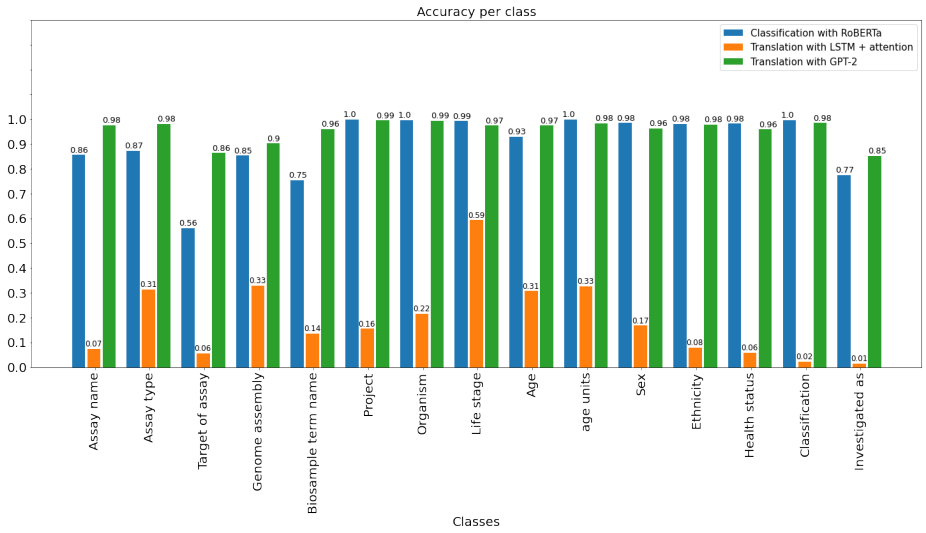

Fig. 8. Experiment 2: per-class accuracy of the three models on ENCODE data.

Table 6. Experiment 2: overall accuracy, precision, and recall. Precision and recall are weighted by the number of occurrences of each attribute value.

Model	# Epochs	Accuracy	Precision	Recall
RoBERTa	71	0.90	0.89	0.90
LSTM + Attention	22	0.19	0.19	0.19
GPT-2	48	0.96	0.96	0.96

with larger amount of distinct values. The attributes *target of assay* and *biosample term name* present the highest number of distinct values and GPT-2 far exceeded RoBERTa in terms of accuracy. Instead, this experiment highlights how the LSTM model with attention does not perform well for a larger amount of target attributes, at least with the tested model size. The labels *health status* and *ethnicity* presented several "None" values (74% and 53%), but both RoBERTa and GPT-2 were able to predict correctly almost the totality of samples, producing results with high weighted precision and weighted recall.

Previous works aimed to extract a restricted set of labels (such as *age* and *sex*) with unsatisfactory results; they often limited the target *age unit* to "years" or "months" and the target *sex* to only "Male" and "Female". A lot of different scenarios for the input text made it impossible – for previous work – to extract correctly the target attributes (for example cases for which the information needs to be inferred, or when the experiment presents multiple cells, consequently multiple ages and multiple sex). This experiment shows that a our proposed translation approach can outperform state-of-the-art approaches, additionally handling a different number of non-standard cases.

Table 7. Experiment 3: Results of prediction of 200 manually labelled samples for ENCODE class *biosample term name*.

Condition	Accuracy	Precision	Recall
Label **present in** the input	0.83	0.70	0.68
Label **absent from** the input	0.062	0.038	0.038

4.4 Experiment 3: Randomly Chosen GEO Instances

In this experiment we study the behaviour of GPT-2 on a realistic scenario involving randomly chosen samples from GEO. These samples were not selected based on presence in a database (Cistrome or ENCODE) and thus provide a realistic test scenario for the proposed use-case of the system. No reference labels are available for the randomly selected set of 200 input descriptions, so each instance was manually checked to provide ground-truth labels. The system was trained using both the Cistrome and ENCODE datasets (10 epochs of training on the former followed by 17 epochs of training on the latter). We note the heterogeneity between the training and test examples for this experiment.

Table 7 reports performance for one exemplar class, the *biosample term name*, which defines the tissue or cell line analyzed in the experimental sample. In order to understand the ability of the system to impute values even when the desired output label is not explicitly present in the input, the evaluation metrics (accuracy, precision, and recall) are computed under two different conditions: the true label is *present in* and *absent from* the input. *Biosample term name* class contains a large number of heterogeneous values: some are only represented by acronyms, e.g., "HeLaS3" that is a cell line; others are more verbose, e.g., "Peripheral blood mononuclear cells"; others indicate tissues of provenance, e.g., "synovial membrane". Accuracy, precision and recall are very promising when the model is able to exploit the information contained in the input text (i.e., condition "label **present in** the input"), while they are low in the opposite case; the model still correctly predicts information that is not contained in the input, in some few cases.

In addition to performance on the *biosample term name* attribute, we noted interesting findings regarding the other classes (examples of input and output shown in Table 8). In particular we found cases in which the output contains a label that is:

1. *unseen* in training data, e.g., no sample contained *target of assay*: *MYC-1.*
2. *absent* from input description, e.g., for the input *"HNRNPK ChIP-seq in K562 K562 HNRNPK ChIP-seq in K562"* the output correctly contained: *Organism: Homo sapiens - Age: 53 - Age units: year - Sex: female - Health status: chronic myelogenous leukemia (CML)*, etc.
3. *multi-valued*: e.g., a particular GEO record contained samples from both male and female donors[13], and the output correctly noted both genders: *"Sex: male,female ..."*.

[13] The input in this case was: *microRNA profile of case NPC362656 survival status (1-death,0-survival): 0; gender (1-male,2-female): 1; age (years): 56; ...*

Table 8. Examples of GPT-2 translations of randomly selected GEO descriptions

Input Sentence	**Output Sentence**
DNase-seq on 91 day old male fetal human kidney tissue left kidney female embryo (59 days) and male embryo (91 day)	Assay name: DNase-seq - Assay type: DNA accessibility - Target of assay: None - Genome assembly: GRCh38 - Biosample term name: left kidney - Project: ENCODE - Organism: Homo sapiens - Life stage: embryonic - Age: 91,59 - Age units: day - Sex: male,female - Ethnicity: None - Health status: None - Classification: tissue - Investigated as: None
RNA Bind-n-Seq (RBNS) pulldown experiment against PUM1	Assay name: RNA Bind-n-Seq - Assay type: RNA binding - Target of assay: PUM1 - Genome assembly: hg19 - Biosample term name: cell-free sample - Project: ENCODE - Organism: None - Life stage: None - Age: None - Age units: None - Sex: None - Ethnicity: None - Health status: None - Classification: None - Investigated as: RNA binding protein
ChIP-seq against eGFP-efl-1 in L1 larva worms. strain YL418 (unc-119(ed3)III; vrIs65 [pGES-1::EFL-1::GFP FLAG:EFL-1 3'UTR, unc-119 (+)]) whole organism L1 larva genetically modified using bombardment	Assay name: ChIP-seq - Assay type: DNA binding - Target of assay: efl-1 - Genome assembly: ce11,ce10 - Biosample term name: whole organism - Project: modENCODE - Organism: Caenorhabditis elegans - Life stage: L1 larva - Age: unknown - Age units: None - Sex: mixed - Ethnicity: None - Health status: None - Classification: whole organisms - Investigated as: cofactor,transcription factor
datatype: DnaseSeq; datatype description: DNaseI HS Sequencing; labversion: baseAlignCounts.pl v 1,fseq v 1.84, iff_generic_male,p-value cutoff: 0.05; replicate: 1,2; cell: Colo829; cell organism: human; cell description: malignant melanoma; cell karyotype: cancer; cell sex: M; labversion: baseAlignCounts.pl v 1 Duke human Colo829 DNase-seq COLO829 malignant melanoma	Assay name: DNase-seq - Assay type: DNA accessibility - Target of assay: None - Genome assembly: GRCh38,hg19 - Biosample term name: COLO829 - Project: ENCODE - Organism: Homo sapiens - Life stage: adult - Age: 45 - Age units: year - Sex: male - Ethnicity: Caucasian - Health status: metastatic melanoma - Classification: cell line - Investigated as: None

4. *reordered* with respect to the input, e.g., an input containing *"Tfh2_3 cell type: Tfh2 CD4+ T cell; ..."* correctly produced the output "*Biosample term name: CD4-positive Tfh2"*.

5 Conclusions and Future Work

In this paper we targeted the problem of extracting useful metadata from free-text descriptions of genomic data samples. Rather than treating the problem as classification or named entity recognition, we model it as machine translation, leveraging state-of-the-art sequence-to-sequence (seq2seq) models to directly map unstructured input into a structured text format. The application of such models greatly simplifies training and allows for imputation of output fields that are implied but never explicitly mentioned in the input text.

We experimented with two types of seq2seq models: an LSTM with attention and GPT-2 (a transformer based language model). We compared the seq2seq models with a multi-label classification based approach using the RoBERTa transformer-based embedding. The GPT-2 model outperforms both the LSTM and the classifier. It demonstrated the ability to predict high-arity attributes and to infer the correct value even for attributes that were not explicitly mentioned in (but were implied by0 the input text. The models were evaluated in

both homogeneous and heterogenous training/testing environments, indicating the efficacy of the transformer-based seq2seq approach for real data integration applications.

A goal for future work is to apply the technique to other genomic and biomedical databases, and to develop a crowdsourcing-based online training framework that can allow us to scale up performance for a production system.

Acknowledgments

This research is funded by the ERC Advanced Grant 693174 GeCo.

References

1. Abadi, M., Agarwal, A., Barham, P., et al.: Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467 (2016)
2. Barrett, T., Wilhite, S.E., Ledoux, P., et al.: NCBI GEO: archive for functional genomics data sets–update. Nucleic acids research **41**(D1), D991–D995 (2012)
3. Bernasconi, A., Canakoglu, A., Masseroli, M., et al.: META-BASE: a novel architecture for large-scale genomic metadata integration. IEEE/ACM Transactions on Computational Biology and Bioinformatics
4. Bernasconi, A., Canakoglu, A., Masseroli, M., et al.: The road towards data integration in human genomics: players, steps and interactions. Briefings in Bioinformatics
5. Bernasconi, A., Ceri, S., Campi, A., et al.: Conceptual modeling for genomics: Building an integrated repository of open data. In: Mayr, H.C., Guizzardi, G., Ma, H., et al. (eds.) Conceptual Modeling. pp. 325–339. Springer International Publishing, Cham (2017)
6. Bodenreider, O.: Biomedical ontologies in action: role in knowledge management, data integration and decision support. Yearbook of Medical Informatics p. 67 (2008)
7. Canakoglu, A., Bernasconi, A., Colombo, A., et al.: GenoSurf: metadata driven semantic search system for integrated genomic datasets. Database **2019** (2019)
8. Davis, C.A., Hitz, B.C., Sloan, C.A., et al.: The encyclopedia of DNA elements (ENCODE): data portal update. Nucleic acids research **46**(D1), D794–D801 (2017)
9. Devlin, J., Chang, M.W., Lee, K., et al.: 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). pp. 4171–4186 (2019)
10. Ellis, S.E., Collado-Torres, L., Jaffe, A., et al.: Improving the value of public RNA-seq expression data by phenotype prediction. Nucleic acids research **46**(9), e54–e54 (2018)
11. Galeota, E., Kishore, K., Pelizzola, M.: Ontology-driven integrative analysis of omics data through onassis. Scientific Reports **10**(1), 1–9 (2020)
12. Giles, C.B., Brown, C.A., Ripperger, M., et al.: ALE: Automated Label Extraction from GEO metadata. BMC Bioinformatics **18**(14), 509 (2017)
13. Guo, Z., Tzvetkova, B., Bassik, J.M., et al.: RNASeqMetaDB: a database and web server for navigating metadata of publicly available mouse RNA-Seq datasets. Bioinformatics **31**(24), 4038–4040 (2015)

14. Hadley, D., Pan, J., El-Sayed, O., et al.: Precision annotation of digital samples in NCBI's Gene Expression Omnibus. Scientific data **4**, 170125 (2017)
15. Hong, E.L., Sloan, C.A., Chan, E.T., et al.: Principles of metadata organization at the ENCODE data coordination center. Database **2016** (2016)
16. Huang, C.C., Lu, Z.: Community challenges in biomedical text mining over 10 years: success, failure and the future. Briefings in Bioinformatics **17**(1), 132–144 (2016)
17. Kans, J.: Entrez direct: E-utilities on the unix command line. In: Entrez Programming Utilities Help [Internet]. National Center for Biotechnology Information (US) (2020)
18. Kundaje, A., Meuleman, W., Ernst, J., et al.: Integrative analysis of 111 reference human epigenomes. Nature **518**(7539), 317 (2015)
19. Li, J., Tseng, C.S., Federico, A., et al.: SFMetaDB: a comprehensive annotation of mouse RNA splicing factor RNA-Seq datasets. Database **2017** (2017)
20. Li, Z., Li, J., Yu, P.: GEOMetaCuration: a web-based application for accurate manual curation of Gene Expression Omnibus metadata. Database: The Journal of Biological Databases and Curation **2018** (2018)
21. Liu, Y., Ott, M., Goyal, N., et al.: RoBERTa: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
22. Lonsdale, J., Thomas, J., Salvatore, M., et al.: The genotype-tissue expression (GTEx) project. Nature genetics **45**(6), 580 (2013)
23. Luong, T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 1412–1421 (2015)
24. Musen, M.A., Sansone, S.A., Cheung, K.H., et al.: CEDAR: Semantic web technology to support open science. In: Companion Proceedings of the The Web Conference 2018. pp. 427–428. International World Wide Web Conferences Steering Committee (2018)
25. Posch, L., Panahiazar, M., Dumontier, M., et al.: Predicting structured metadata from unstructured metadata. Database **2016** (2016)
26. Radford, A., Wu, J., Child, R., et al.: Language models are unsupervised multitask learners. OpenAI Blog **1**(8), 9 (2019)
27. 1000 Genomes Project Consortium: A global reference for human genetic variation. Nature **526**(7571), 68 (2015)
28. Vaswani, A., Shazeer, N., Parmar, N., et al.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
29. Wang, Z., Lachmann, A., MaâĂŹayan, A.: Mining data and metadata from the Gene Expression Omnibus. Biophysical reviews **11**(1), 103–110 (2019)
30. Wang, Z., Monteiro, C.D., Jagodnik, K.M., et al.: Extraction and analysis of signatures from the Gene Expression Omnibus by the crowd. Nature communications **7**(1), 1–11 (2016)
31. Weinstein, J.N., Collisson, E.A., Mills, G.B., et al.: The cancer genome atlas pan-cancer analysis project. Nature genetics **45**(10), 1113 (2013)
32. Zaveri, A., Hu, W., Dumontier, M.: MetaCrowd: crowdsourcing biomedical metadata quality assessment. Human Computation **6**(1), 98–112 (2019)
33. Zheng, R., Wan, C., Mei, S., et al.: Cistrome Data Browser: expanded datasets and new tools for gene regulatory analysis. Nucleic acids research **47**(D1), D729–D735 (2018)
34. Zhu, Y., Davis, S., Stephens, R., et al.: GEOmetadb: powerful alternative search engine for the Gene Expression Omnibus. Bioinformatics **24**(23), 2798–2800 (2008)