TabText: a Systematic Approach to Aggregate Knowledge Across Tabular Data Structures

Abstract—Processing and analyzing tabular and time series data productively and efficiently is essential for building successful machine learning applications in fields such as healthcare. However, the lack of a unified framework for representing and standardizing tabular information poses a significant challenge to researchers and professionals alike. In this work, we present TabText, a methodology that leverages the unstructured data format of language to encode efficiently and accurately tabular data from different table structures and time periods. Using two healthcare datasets and four predictions tasks, we show that features extracted via TabText outperform those extracted with traditional processing methods by 2-5%. Furthermore, we analyze the sensitivity of our framework against different choices for sentence representations of missing values, meta information, and language descriptiveness and provide insights into winning strategies that improve performance.

Index Terms—BERT, Tabular Data, Healthcare, Time Series, Feature Extraction

I. INTRODUCTION

Tabular data is arguably one of the most used and available data formats across different data science domains. Especially in healthcare systems, tabular data plays a crucial role in recording information for each patient, such as vitals, co-morbidities, diagnoses, and treatments. However, finding an approach that can systematically encode information from different tabular data structures and across hospital systems remains challenging. As noted by [1]–[3], due to the lack of standardization and harmonization in current healthcare data recording systems, significant amounts of expert knowledge as well as manual labor are dedicated to selection, encoding, and imputation of the data.

The source of the problem is that, since most of the data in healthcare systems is recorded in tabular format, machine learning models for healthcare applications often stick to the tabular structure even though a lot of data sources are not tabular in nature. For instance, data attributes involving language, such as diagnosis, can have thousands of different values commonly processed as categorical features with an extremely large number of categories. Another difficulty regarding the tabular approaches is that patients with different diagnoses require specific tests and treatments. Therefore, many attributes are recorded only for a very specific group of patients. Predictive models either exclude such attributes, potentially ignoring valuable data, or impute the many missing values with the very few recorded instances.

In contrast to tabular approaches, language is a very flexible data modality that can easily represent information about different patients without imposing any structural similarity between them. Furthermore, there exist very successful off-the-shelf models for biomedical language representation that can be readily applied. In particular, deep learning methods such as BERT [4] have received much attention in recent years due to their superior performance in tasks ranging across different areas of expertise, including but not limited to question answering and sentence completion.

Several previous works have shown the potential of using natural language processing (NLP) models to systematically...
and efficiently process tabular data in the form of language
[5]–[8]. However, these works have relied mainly on training
fixed BERT-based models that are not flexible to changes in
tabular structures. Thus, in practice, they cannot be used once
multiple healthcare institutions enter the picture. These works
have mostly assumed that encoding data using advanced
deep learning models leads to better performance compared
to traditional data processing methods. However, concrete
evidence has not been provided. In addition, language models
are considered sensitive to their input representations [9],
and most previous works do not thoroughly investigate how
the choice of language affects their results. Besides, they do
not explore different language representations for missing
values. Lastly, previous approaches do not consider how to
cohesively combine static information from tabular data with
time-series components, in which several observations may
correspond to the same patient at different time periods.

In this paper, we build and evaluate a new data processing
methodology and address the aforementioned questions.

The main contributions of this work are as follows:

- We develop a BERT-based method for unified data
  processing that uses language to process systematically
  1) tabular data from different table structures, 2) time-
  series features, and 3) additional non-tabular data such
  as meta information.

- We analyze how the exact choice of language utilized by
  our model for data representation affects its performance.

- We show that our processing method is compatible with
  standard machine learning models and outperforms tra-
  ditional processing approaches by 2-5% AUC across two
  different datasets.

II. TabText

In standard data processing methods, categorical and
binary features are generally handled using label-encoding
or one-hot encoding, time-series data is represented by
including features with statistical summaries of the data
(e.g., mean, min, max, variance, average change, etc.), and
the remaining numerical features are processed with their
original values. In addition, missing values are often imputed
with 0, although more sophisticated methods can also be used.

This section describes our method TabText — a new data
processing framework for machine learning applications in
healthcare. We explain how TabText can process data from
different tabular structures without imposing any structural
constraints and how it handles time-series features and missing
data.

A. From Tabular Data to Language

When interpreting tabular data, humans read each row by
scanning the specific values in it and the corresponding column
headers that provide context for each cell. In addition, they
take into account the meta information of the table, like the
title or any other available descriptions. Inspired by these
facts, with TabText, we process tabular data by creating a
sentence for each row that contains the column attribute with
its corresponding value and any available meta-information
(see Figure 1). We then pass this sentence into a pre-trained,
publicly available BERT model to generate embeddings that
can later be used to conduct downstream tasks with any
standard machine learning model (e.g., neural networks, trees).
Since we focus on healthcare applications, we use a pretrained
Bio+ClinicalBERT [10], [11] model from HuggingFace that
generates embeddings for biomedical language. These em-
beddings have a fixed dimension (768) and can be produced
for any text with less than 510 characters. Whenever the
sentence for a particular row exceeds this limit, we separate it
into smaller sentences that satisfy the length requirement; we
generate embeddings for each sub-sentence and average them
all to obtain the final embedding.

B. Time Series Evaluation

![Figure 1. General Framework of the TabText Processing](image)

![Figure 2. Weighted Average of Time-Series Data](image)
Time-series data is an essential component for healthcare applications since medical decisions take into account the entire medical record of the patients. However, it remains a challenge to systematically and holistically combine non-time-series with time-series data into a cohesive predictive framework. To process rows that correspond to the same patient at different points in time, TabText generates embeddings for each one of those rows separately and then combines them using a weighted sum, where the weight corresponding to a particular row is the respective time-stamp. This approach allows TabText to give more importance to recent observations while still obtaining a fixed-size embedding that represents the medical history of each patient. We demonstrate this approach in Figure 2.

C. Sentence Representation
We consider several ways of representing our data as language that could potentially impact the results of the performance. Specifically, we investigate the following:

1) **Missing Values**: Biomedical data usually has substantial amounts of missing values, since specific medical records are tracked for specific patients. Machine learning models then rely on preprocessing methods such as imputation [12] to fill missing entries with synthetically generated values computed to approximate the real ones. However, this approach has important limitations. For example, missingness itself could be a representation of particular biases in the original dataset, which missing value imputation could artificially exacerbate. Furthermore, traditional methods often convert non-numerical features to numerical values before imputation, which further induces another layer of error propagation and information loss. We thus investigate four possible ways of directly encoding missing values into our TabText framework:

- **Exclusion**: we simply do not include this information at all in our generated language.
- **Encode Missing**: we encode the information by adding in the sentence that the column “is missing”.
- **Zero Padding**: we encode the information as “is 0”.
- **Maintain Original Information**: we encode the information with the original missing value, be it an empty string or “NaN”.

2) **Inclusion of Meta Information**: Meta information such as high-level descriptions of the table content is often available and can be helpful for downstream models to distinguish features generated from different data sources. Thus, for each embedding, we test the impact of including or excluding this information.

3) **Descriptiveness of Language Representation**: Creating sentences that follow the natural language form could help the BERT models better understand the model’s content. We thus consider two possibilities: (i) either directly concatenate column names with column values into the sentence, or (ii) manually write descriptive language for encoding the information.

4) **Process Languages from Different Sources**: BERT models may benefit from texts whose sentences share the same context. We then consider the impact of maintaining separate embeddings for each tabular source via concatenation versus combining the languages from different sources into a single paragraph for BERT embedding.

---

**Figure 3** shows an example of each representation choice.

| Age | Sex | BMI | Zip Code | State |
|-----|-----|-----|----------|-------|
| 39  | Female | 26.4 | 02139 | MA |

| Pulse | Blood Pressure | Weight |
|-------|----------------|--------|
| 70    | 110            | 170    |

**Missing Values**
- “age is missing”
- “age is 0”
- “age is NaN”

**Descriptiveness of the Columns**
- “age:39, Sex: Female, BMI: 26.4, Zip Code: 02139, State: MA”
- “This person has an age of 39, Sex is Female, BMI is 26.4, Zip Code is 02139, State is MA”

**Inclusion of Table Description**
- “The following is the demographics information of this patient, which describes information such as name, date of birth and address, along with insurance information. age:39, Sex: Female, BMI: 26.4, Zip Code: 02139, State: MA”

**Languages from Different Sources**
- “age:39, Sex: Female, BMI: 26.4, Zip Code: 02139, State: MA”
- “Pulse: 70, Blood Pressure: 110, Weight: 170”

---

**III. RESULTS**

A. **TabText for Classification Problems**
We study the effect of TabText on classification problems using two confidential hospital datasets. The first dataset (A) concerns the prediction of patient mortality after the dispatch of a rapid response team, in which event an imminent clinical deterioration summons immediate provider attention. The second dataset (B) concerns the prediction of length of stay and mortality at end-of-stay for each inpatient in the hospital.
Dataset A for rapid response mortality prediction uses 1590 samples with 121 positive labels and includes information on demographics, encounters (non-time-series), medication, problems, signs, and socials (time-series). Dataset B contains medical records of 47345 patients, with 30 different attributes on demographics, vitals, treatment, and diagnosis. We use Dataset B for three different binary classification tasks: end-of-stay mortality (whether patients die or not at the end of their stay), discharge in the next 24 hours (whether patients are discharged or not within the next 24 hours), and similarly discharge in the next 48 hours.

We first split the original data into 80/20 training and testing, then train a gradient boosted tree model using only the training set and select the best hyperparameter combination that gives the best validation performance using grid search. We evaluate our performance using the area under the receiver operating curve (AUROC) on the test set. Below we report the best performing AUROC on the test set achieved by any TabText sentence representation, as well as the best performing AUROC achieved by traditional processing through either of the time-series approaches.

### Table I

| Dataset | Task of Interest                 | Traditional AUROC | TabText AUROC |
|---------|--------------------------------|-------------------|---------------|
| A       | Rapid Response Mortality Prediction | 0.691             | 0.741         |
| B       | End of Stay Mortality Prediction    | 0.791             | 0.830         |
| B       | Discharge in the next 24h Prediction | 0.730             | 0.744         |
| B       | Discharge in the next 48h Prediction | 0.726             | 0.743         |

We see that, on average, across the 4 binary classification tasks, TabText improves on traditional tabular processing by 2-5% on AUROC.

### B. Sentence Representation Sensitivity

In Table II, we use the rapid response example to demonstrate that different sentence representations can substantially change the outcome of the prediction tasks. We note that in comparison to the baseline model, the best performing combination outperforms by 5%, whereas the worst-performing combination underperforms by 6%.

#### Table II

| Missing Handling | Meta Info | Descriptiveness | Test AUROC |
|------------------|-----------|-----------------|------------|
| Is missing       | Include   | Yes             | 0.741      |
| Is missing       | Does not Include | Yes       | 0.721      |
| Original         | Include   | No              | 0.715      |
| Original         | Does not Include | No        | 0.713      |
| Is 0             | Include   | No              | 0.712      |
| Is 0             | Does not Include | Yes       | 0.704      |
| Original         | Include   | Yes             | 0.684      |
| Original         | Does not Include | No        | 0.679      |
| Is Missing       | Does not Include | Yes       | 0.653      |
| Is Missing       | Include   | No              | 0.651      |
| Is Missing       | Does not Include | No        | 0.650      |
| Original         | Does not Include | No        | 0.659      |
| Original         | Does not Include | Yes        | 0.635      |

Below, we also report the aggregated performance from the above table averaged by each type of representation that we are interested in.

1) **Descriptiveness of Language Representation:** We see that, on average, writing sentences in a more formal and human-readable manner does not outperform a naive machine-generated sentence that includes the same content.

#### Descriptiveness

| Descriptiveness | Test AUROC |
|-----------------|------------|
| Yes             | 0.687      |
| No              | 0.684      |

2) **Missing Values:** Overall, we see that exclusion of all missing information or encoding the missingness of the information directly into the language representation improves upon keeping the original column information by around 2.5%. However, the exact representation method for missing values does not play an essential role in improving performance.

#### Missing Handling

| Missing Handling | Test AUROC |
|------------------|------------|
| Exclusion        | 0.691      |
| Is Missing       | 0.695      |
| Is 0             | 0.690      |
| Original         | 0.667      |

3) **Inclusion of Meta-Information:** The inclusion of meta-information improves the AUROC by 1%, implying that giving the model high-level conceptual information helps its performance.

#### Inclusion of Meta-Information

| Inclusion of Meta-Information | Test AUROC |
|-------------------------------|------------|
| Include                       | 0.691      |
| Does not include              | 0.680      |

### C. Aggregate Knowledge Across Tabular Data Structures

In comparison to traditional methods of manual processing, a crucial practical advantage our framework provides is its
ability to easily integrate tabular data of different sources and structures in a standardized and harmonized format. By leveraging the unstructured data format of language, our method provides unparalleled flexibility to encode all types of information holistically. For example, traditional data processing includes:

Data cleaning:
- Anything that is not of standardized format needs to be processed, for example, height of 5’3” cannot be used directly.
- Information with no standardization possible, for example, a column of dosage unit may have both mL and capsule, without specifying the exact definitions and conversion rules between these possible values.

Categorical variables:
- Requires us to convert to either ordered numerical levels (consecutive integers) or binary categories using one-hot encoding.
- If the number of categories is too large, we need to restrict to a smaller subgroup of categories and discard the rest of the categories.

Missing values:
- Filter population samples by discarding samples into complete-case study.
- Compute missing entries with existing imputation methods.

However, TabText can efficiently resolve all of the above issues by adding this information as part of the generated language paragraph. Doing so first provides immense data cleaning and processing speedups as it requires minimal human labor as opposed to traditional methods. Secondly, it does not sacrifice any information to construct them into a standardized format and maintains all information in their most native form.

IV. DISCUSSIONS AND LIMITATIONS
We introduce a new framework for processing tabular data via natural language. We use BERT-based methods to generate embedding that can be used to make downstream predictions. We demonstrate that our method offers both substantial performance improvements and practical advantages for industry deployment. We show that the different choices of representing missing values, meta-information, and descriptiveness of the generated language form are crucial to TabText’s success. Lastly, we provide approaches to combine information from non-time-series data and time-series data cohesively.

We note that the current approach should be tested on more benchmark datasets and other machine learning tasks such as regression and clustering. Furthermore, we wish to investigate possible approaches such as BioNumQA-BERT that improve on previous methods for encoding numerical values [13]. Lastly, further research is needed for settings in which the final feature space is very high dimensional, which can affect predictive power and model interpretability.

REFERENCES

[1] Lester Darryl Geneviève et al. “Factors influencing harmonized health data collection, sharing and linkage in Denmark and Switzerland: A systematic review”, In: PLOS ONE 14.12 (Dec. 2019). Ed. by Danuta Wasserman, e0226015. DOI: 10.1371/journal.pone.0226015. URL: https://doi.org/10.1371/journal.pone.0226015.

[2] Yang Nan et al. “Data harmonisation for information fusion in digital healthcare: A state-of-the-art systematic review, meta-analysis and future research directions”. In: Information Fusion 82 (June 2022), pp. 99–122. DOI: 10.1016/j.inffus.2022.01.001. URL: https://doi.org/10.1016/j.inffus.2022.01.001.

[3] Evan Sweeney. Experts say IBM Watson’s flaws are rooted in data collection and interoperability. Sept. 2017. URL: https://www.fiercehealthcare.com/analytics/ ibm-watson-s-flaws-traceback-to-data-collection-interoperability.

[4] Jacob Devlin et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2018. DOI: 10.48550/ARXIV.1810.04805. URL: https://arxiv.org/abs/1810.04805.

[5] Jonathan Herzig et al. “TaPas: Weakly Supervised Table Parsing via Pre-training”. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, July 2020, pp. 4320–4333. DOI: 10.18653/v1/2020.acl-main.398. URL: https://aclanthology.org/2020.acl-main.398.

[6] Pengcheng Yin et al. “TaBERT: Pretraining for Joint Understanding of Textual and Tabular Data”. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online: Association for Computational Linguistics, July 2020, pp. 8413–8426. DOI: 10.18653/v1/2020.acl-main.745. URL: https://aclanthology.org/2020.acl-main.745.

[7] Inkit Padhi et al. “Tabular Transformers for Modeling Multivariate Time Series”. In: ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2021, pp. 3565–3569. DOI: 10.1109/ICASSP39728.2021.9414142.

[8] Gowthami Somepalli et al. SAINT: Improved Neural Networks for Tabular Data via Row Attention and Contrastive Pre-training. 2021. DOI: 10.48550/ARXIV.2106.01342. URL: https://arxiv.org/abs/2106.01342.

[9] Aamir Miyajiwala et al. On Sensitivity of Deep Learning Based Text Classification Algorithms to Practical Input Perturbations. 2022. eprint: https://arxiv.org/abs/2101.00318.

[10] Jinhuyuk Lee et al. “BioBERT: a pre-trained biomedical language representation model for biomedical text min- ing”. In: (2019). DOI: 10.48550/ARXIV.1901.08746. URL: https://arxiv.org/abs/1901.08746.

[11] Emily Alsentzer et al. Publicly Available Clinical BERT Embeddings. 2019. DOI: 10.48550/ARXIV.1904.03323. URL: https://arxiv.org/abs/1904.03323.

[12] Stef Van Buuren and Karin Ou Shoom. Flexible multivariate imputation by MICE. Leiden: TNO, 1999.

[13] Ye Wu et al. “BioNumQA-BERT: Answering Biomedical Questions Using Numerical Facts with a Deep Language Representation Model”. In: Proceedings of the 12th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics. BCB ’21, Gainesville, Florida: Association for Computing Machinery, 2021. ISBN: 9781450384506. DOI: 10.1145/ 3459930.3469557. URL: https://doi.org/10.1145/ 3459930.3469557.