LYX_DMIIP_FDU At BioASQ 2025: Utilizing BERT Embeddings For Biomedical Text Mining*

Notebook for the ELCardioCC, BioNNE-L and GutBrainIE task of the BioASQ Lab at CLEF 2025

Yuxuan Liu[1], Shanfeng Zhu[1,2,3,4,*]

[1]*Institute of Science and Technology for Brain-Inspired Intelligence and MOE Frontiers Center for Brain Science, Fudan University, Shanghai, China*

[2]*Key Laboratory of Computational Neuroscience and Brain-Inspired Intelligence (Fudan University), Ministry of Education, Shanghai, China.*

[3]*Shanghai Key Lab of Intelligent Information Processing and Shanghai Institute of Artificial Intelligence Algorithm, Fudan University, Shanghai, China.*

[4]*Zhangjiang Fudan International Innovation Center, Shanghai, China*

Abstract

Due to the increasing volume of biomedical documents, biomedical text mining becomes increasingly important to gather knowledge automatically from large amount of biomedical text. In this paper, we present our method for three information extraction tasks in BioASQ Lab 2025: ELCardioCC, BioNNE-L and GutBrainIE. In these tasks, we utilized different BERT models to generate embeddings for concept representation, which are then used for biomedical named entity recognition, entity linking and relation extraction task. Our methods are simple yet efficient, ranking the first place on ELCardioCC and the multilingual track of BioNNE-L task. On GutBrainIE, our method surpassed the baseline method on Named Entity Recognition and Ternary Mention-Based Relation Extraction subtask. Our results demonstrated that many biomedical information extraction task can be efficiently solved by utilizing BERT embeddings.

Keywords

Biomedical text mining, Named entity recognition, Named entity linking, Relation extraction, Transformers, BERT

1. Introduction

Over the past few years, the amount of biomedical text grows quickly. These texts includes biomedical research papers, clinical notes and patient health records. Biomedical text mining methods are required to efficiently extract useful information from biomedical text. For example, Named Entity Recognition (NER) recognizes different types of entities in biomedical text, Named Entity Linking (NEN) maps the entities found by NER to a certain entry in a given database, and Relation Extraction (RE) finds the relationships between each pair of entities.

Deep learning using Bidirectional Encoder Representations from Transformers (BERT) [1] is currently the most prevalent method for biomedical text mining, due to its great performance and efficiency. Many BERT models are pretrained on large biomedical corpus, and can be easily fine-tuned for downstream tasks. A few models to name are `PubmedBERT` [2], `BiolinkBERT` [3] and `xlm-roberta-large-english-clinical` [4].

We utilized BERT embeddings to participate in three biomedical text mining tasks in the BioASQ lab of CLEF 2025 [5]: ELCardioCC[6], BioNNE-L[7] and GutBrainIE[8]. Overall, our methods achieved good results.

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

*Corresponding author.

yuxliu21@m.fudan.edu.cn (Y. Liu); zhusf@m.fudan.edu.cn (S. Zhu)

0009-0000-2255-3245 (Y. Liu); 0000-0002-6067-5312 (S. Zhu)

2. Related works

BERT has long been a useful tool for many natural language processing (NLP) tasks [9]. After intensive pretraining, BERT embeddings can effectively capture the meanings of a certain token or a sentence along with its current context, and are often fed into a classifier for classification tasks or used for calculating similarity for entity linking task. In the past few years, many works have utilized BERT embeddings for biomedical text mining.

For biomedical NER, sequence labeling is a common approach, where the BERT embedding of each token is first calculated, then each token are classified as either begin of entity(B), continue of entity(I), or not inside entity(O) for a certain entity type. Based on this method, BERN2 [10] developed a multi-task model that treats recognition of different types of entity as different tasks by sharing the same BERT model across all entity types and using a separate classifier for each entity type. As a result, the model can efficiently recognize all given entity types with a single BERT forward pass. Subsequent works include AIONER [11] and Hunflair2 [12], which combines multiple biomedical NER datasets for training and achieve good result on all datasets.

For biomedical NEN, a database is first selected, then a preprocessing step is performed to calculate the BERT embeddings of all entities within the database. After that, the BERT embedding of the target entity is also calculated and matched to the database embeddings to find the entity that has the most similar embedding to the target entity in the database. This process is often accelerated with certain python package like faiss [13]. However, the default BERT model may not generate an embedding that is most suitable for NEN task, thus SapBERT [14] performed an additional pretraining using contrastive learning specifically on NEN task. More recent models like GeBERT [15] and BERGAMOT[16] used graph neural networks to capture the relation between entities, and used carefully designed training objective for a better entity representation.

For biomedical RE, ATLOP [17] combines the BERT embeddings of the two target entities to extract relation from and used them for a 0/1 classification for each relation type. They also proposed a localized context pooling technique that find the information that is important to both entities by utilizing attention weights in BERT layers. BioREX [18] carefully combined multiple biomedical RE datasets with different labeling standards and trained on them for better model performance. A more recent work [19] trained the BERT model to predict not only the relation type but also the relation direction and whether the relation is novel.

3. ELCardioCC

3.1. Task description

The ELCardioCC of BioASQ 2025 requires participants to extract entities from discharge letters written in Greek that record patients' conditions, treatments and outcomes. There are five types of entities: chief complaint, diagnosis, prior medical history, drugs and cardiac echo.

The task consists of three subtasks: NER subtask requires accurate prediction of the span of each entity, EL subtask requires not only the entity span but also its associated ICD-10 code, MLC-X subtask requires identification of all the ICD-10 codes that are present in a letter, while the corresponding entity spans are not required. This subtask also contains a further step that requires participants to identify the mentions from ICD-10 codes using explainable AI techniques. All the subtasks are evaluated using precision, recall and F1-score.

The dataset of this task, containing 1500 discharge letters in Greek (1000 for training, 500 for testing), was collected from the cardiology department of a tertiary hospital in Greece. Sensitive personal information has been removed from the dataset. Then a professional team labeled all the entities and their corresponding ICD-10 codes in the dataset. There are 10168 labeled entities in the dataset in total, and the average length of the entities is 14.312 chars. The organizers also provided a supplementary file that contains all the 324 ICD-10 codes used in the labeling process.

The challenge of this task lies in the fact that Greek is a language of scarce biomedical data, thus the pretrained BERT model might not capture Greek language structures very well. Also, the entity types are not given in the training dataset, so either an additional step should be performed to generate the entity types or the model has to view all types as an unified entity type, which may degrade model performance.

3.2. Method

3.2.1. NER subtask

For the NER subtask, we applied a standard sequence labeling scheme. For the BERT model, we used `bert-base-greek-uncased-v1` [20] from HuggingFace, which is a Greek BERT model pretrained exclusively on Greek data, because our initial experiment shows that Greek only BERT model performs better than the multilingual models. We calculated the final layer embedding of each token using the Greek BERT model and fed them into a simple classifier with two MLP layers to output probability for BIO classes.

During training, we fine-tuned all BERT layers and the classifier simultaneously using a standard cross-entropy loss for classification tasks:

$$L = \sum_{i=1}^{N} -log \frac{exp(l_{GT_i})}{\sum_{j=1}^{C} exp(l_j)} \tag{1}$$

Where N is the number of data points in total, C is the number of classes, l_{GT_i} is the model output logit of the ground truth class of data point i.

The context length was set to 512. We used AdamW [21] as optimizer, with a learning rate of 3e-5 for both BERT and MLP layers and a weight decay of 0.1 for regularization, other parameters are as default. Since the dataset is small, we trained the model for 50 epochs. The model achieved a NER F1-score of 0.7671 on the development dataset.

During prediction, since a single document can be longer than the context length of the model, we applied a sliding window technique that divides the document into overlapping windows of the context length of the model and do inference on them separately. For the overlapping regions, we divide them into two equal halfs. The tokens in the left half are predicted together with the left window, and the tokens in the right half are predicted together with the right window.

3.2.2. EL subtask

For the EL subtask, we selected SapBERT, a model pretrained on UMLS database [22] with an entity linking objective as our BERT model. Since SapBERT works on English only, we used a simple translation model Libretranslate deployed locally to translate the entities recognized by NER into English. Since the label set is small (only 324 different labels), we treated the problem as a classification task, using the translated entity names as BERT input and used the CLS token representation of the last layer as the representation of the entity. We then used two MLP layers to classify each entity into one of the 324 classes. As in the NER task, we fine-tuned SapBERT and classifier simultaneously using cross-entropy loss as described above.

To enrich the dataset, we collected two versions of the ICD-10 database online, and added the entities as well as their corresponding ICD-10 codes into the training dataset. The training parameters of the EL task are similar to NER task, except the learning rate is increased to 3e-4 for MLP layers and 1e-4 for SapBERT. The model achieved a F1-score of 0.7096 on development dataset after training.

3.2.3. MLC-X subtask

For the MLC-X subtask, we didn't design a specific method, and simply used the EL prediction of all entities in a document and removed duplicates to obtain the document-wise labels. We didn't participate in the explainable AI subtask.

Table 1
Official results for the NER subtask of ELCardioCC

Team name	Precision	Recall	F1-score
droidlyx (our team)	**0.7618**	**0.7059**	**0.7328**
ELCardioCC_baseline	0.7460	0.6959	0.7201
svassileva	0.7328	0.7012	0.7167
bhuang	0.5205	0.6448	0.5761
pjmathematician	0.2586	0.2484	0.2534

Table 2
Official results for the EL subtask of ELCardioCC

Team name	Precision	Recall	F1-score
droidlyx (our team)	**0.7046**	**0.6529**	**0.6778**
ELCardioCC_baseline	0.6942	0.6476	0.6701
svassileva	0.6844	0.6548	0.6693
bhuang	0.4852	0.5927	0.5336
pjmathematician	0.0642	0.0616	0.0629

Table 3
Official results for the MLC-X subtask of ELCardioCC

Team name	Precision	Recall	F1-score
droidlyx (our team)	**0.8569**	**0.8377**	**0.8472**
ELCardioCC_baseline	0.9339	0.7422	0.8271
bhuang	0.6947	0.8250	0.7543
pjmathematician	0.5860	0.2656	0.3655
kbogas	0.2115	0.3421	0.2614

3.3. Results and Discussion

The official test results for all three subtasks are shown in Table 1, Table 2 and Table 3, respectively. For simplicity, we only showed the results of the best run of each team as well as the baseline results. We didn't show the results of the explainable AI subtask because only one team submitted this subtask and their results were not satisfactory.

Overall, we achieved good results for this task, surpassing the baseline and ranking the first place in all subtasks. The results shows that a Greek BERT model can capture the language of Greek data well. Using an extra model to predict the entity classes in the training dataset and training the NER model to predict each entity type separately may further improve the results.

For the EL subtask, translating the entity names to English may not be necessary, since one can directly use GreekBERT on the entity names and predict the ICD-10 codes. The svassileva team seems to have applied this method and also achieved good results. This method didn't utilize the ICD-10 database (which is in English). Another method (other than translation) is to use a multilingual model that takes both Greek and English entities as input, but the performance of the multilingual model may not be as good as unilingual model. Nevertheless, our performance improvement over the baseline is slight, which suggests that the addition of the database entries into the training dataset doesn't bring significant performance boost.

Table 4
Dataset statistics of BioNNE-L

Data Split	Num Documents	Total Entities	Avg entities per doc	Avg entity length
English_train	54	2690	49.815	11.726
English_development	50	2494	49.880	12.154
Russian_train	716	24255	33.876	13.238
Russian_development	50	2334	46.680	13.522

4. BioNNE-L

4.1. Task description

The BioNNE-L shared task is based on the BioNNE[23, 24] task of BioASQ 2024 . Different from BioNNE, BioNNE-L requires participants to link the entities recognized by NER to the Unified Medical Language System (UMLS) database. Some of the entities are nested, which means they can be a substring of each other. There are three tracks in this task: English track, Russian track and Bilingual track. These tracks require the model to tackle text written in different languages.

The provided dataset consists of English and Russian scientific abstracts in the biomedical domain. Three types of entities are used for this task: ANATOMY, CHEM and DISO. For each entity in the text, the document ID, entity type, entity span and the UMLS identifier are given. The dataset statistics is shown in Table 4. A vocab file in both English and Russian is also provided, which contains all UMLS identifiers used in this task. However, the Russian version of the vocab is incomplete, so some of the Russian entities has to be linked to English version of UMLS items. There are 1510431 different identifiers in total, with 3902187 English entities and 145803 Russian entities.

For evaluation, the participants may submit up to 5 predictions for each entity, as well as their relative ranking. The submitted results are evaluated based on their Top-1 accuracy, Top-5 accuracy and Mean Reciprocal Rank (MRR) that averages the inverse ranking of the correct answer across all entities. A baseline method as well as its results on development dataset has also been provided. This baseline method directly maps to BERT embeddings of the target entity to the embeddings of vocab entities and selects the most similar entities as prediction.

There are a few challenges when participating in this task: Firstly, the bilingual data requires the model to understand both English and Russian well. Secondly, the nested nature of the entities means that their names can be quite similar to each other, but their meanings may differ, which requires the model to clearly differentiate between them. Thirdly, the provided vocab file is very large, consisting of millions of entities, and many entities in the vocab are quite similar though having different identifiers. For example, the 'Depressed state' entity has the identifier of C0011570, the 'Depressive Disorders' entity has the identifier of C0011581, and the 'Depression' entity has the identifier of C1999266. Fourthly, the UMLS identifiers are not organized with a tree structure. Instead, they only reflects the order they were added into the database, and thus they don't carry any meanings by themselves.

4.2. Method

For the BERT model, we chose the same models the baseline used: `BERGAMOT-multilingual-GAT` [15] and `gebert_eng_gat` [16]. These models utilize graph representations of the relationships between entities to obtain a better representation of each entity.

Inspired by SapBERT [14], we fine-tuned the BERT model using contrastive learning to enhance the BERT representations for entity linking. We fixed the representation of the vocab entities to the embeddings generated by the original model during the entire fine-tuning process. Since the text data was also provided, we used the following input format: [CLS] text_before [SEP] entity [SEP] text_after. Both text_before and text_after are clipped to about 250 chars. This format enables the output embedding of the [CLS] token to be the context-aware representation of the entity.

The detailed training process is as follows: We first used a warm-up training phase to make the model more familiar with the new input format, where the output embeddings of the [CLS] token when the model is provided with context should be similar to when only the entity name is provided. We applied a simple mean squared loss between the two embeddings for this phase:

$$L = \sum_{i=1}^{N} \frac{\sum_{j=1}^{D} (EO_j - EC_j)^2}{D} \tag{2}$$

Where N is the number of data entities, D is the embedding dimension, EO is the embedding without context, and EC is the embedding with context.

Then comes the contrastive learning phase, where for each entity, we select the entities in the vocab that are most similar to the target entities for training. Entities that have the same identifier as the label are used as positive examples, and entities that have different identifiers are used as negative examples. For the loss function, we used InfoNCE which is quite commonly used for contrastive learning:

$$L = \sum_{i=1}^{N} -log \frac{exp(\frac{q \cdot k+}{\tau})}{\sum_{j=1}^{M} exp(\frac{q \cdot kj}{\tau})} \tag{3}$$

Where N is the number of target entities, M is the number of negative samples, q is the output embedding of target entity, $k+$ is the output embedding of the positive entity, k_j is the output embedding of the jth negative entity, and τ is a hyperparameter.

We used AdamW optimizer for both phases, with a learning rate of 3e-5 and weight decay of 0.1. We trained the model for 1 epoch at the warm-up training phase and 4 epochs at the contrastive learning phase. During contrastive learning, we used faiss [13] to select the top 20 most similar entities to the target entity in the vocab, among with the last positive entity and the top 10 negative entities are being used for loss calculation. If all 20 selected entities are negative, we randomly select one entity from the vocab that has the correct identifier as positive entity. If there are less than 10 negative entities, we fill the rest with random entities from the vocab.

During inference, we used faiss to match the output embeddings of the model to the fixed vocab embeddings, and used the associated identifiers of the corresponding vocab entity as output. Among all identifiers, we selected top 5 different identifiers as the final prediction.

We observed that some of the entity identifiers in the training dataset doesn't appear in the original vocab, so before training, we added the entities together with their identifiers into the training and development dataset into the vocab to enrich it (removing duplicates with the same entity name and identifier). We then trained our final version model on both training and development dataset.

We applied our method to all three tracks. Since the `BERGAMOT-multilingual-GAT` model is multilingual, it can be used for all three tracks. For the Russian track, a multilingual model is required since some entities has to be linked to English vocab. For the English track. we attempted to train the English only `gebert` model with only English data and vocab, but the result is not as good as the multilingual model, so we just submitted the prediction of the multilingual model as our final prediction.

4.3. Results and Discussion

The official test results for all three tracks are shown in Table 5, Table 6 and Table 7, respectively. Our method achieved promising results, ranking the first or second place in all three tracks. Although we used the same BERT model as the baseline, our fine-tuned model surpassed the baseline model by a large margin (the top 1 accuracy of the baseline model on the development dataset is 0.53), suggesting that fitting the dataset bias is important in machine learning tasks.

The model result on Russian data is better than on English data. This may be due to more Russian data in the training dataset. Also because of this, we observed that our fine-tuned bilingual model linked almost all entities to Russian vocab entities (If available). This could also be the reason why our English only `gebert` model didn't perform as well.

Table 5
Official results for the English track of BioNNE-L

Team name	Top1 accuracy	Top5 accuracy	MRR
verbanexialab	**0.70**	0.80	0.74
droidlyx86 (our team)	0.66	**0.84**	**0.74**
BlancaPlanca	0.64	0.83	0.72
EeyoreLee	0.64	0.82	0.71
Andoree	0.57	0.78	0.66
Antoinel	0.51	0.79	0.62

Table 6
Official results for the Russian track of BioNNE-L

Team name	Top1 accuracy	Top5 accuracy	MRR
BlancaPlanca	**0.72**	0.83	0.76
droidlyx86 (our team)	0.71	**0.84**	**0.76**
dstepakov	0.70	0.76	0.72
EeyoreLee	0.65	0.74	0.69
Antoinel	0.62	0.72	0.67
Andoree	0.52	0.59	0.55

Table 7
Official results for the Bilingual track of BioNNE-L

Team name	Top1 accuracy	Top5 accuracy	MRR
droidlyx86 (our team)	**0.68**	**0.84**	**0.75**
BlancaPlanca	0.67	0.81	0.73
EeyoreLee	0.63	0.76	0.69
dstepakov	0.63	0.71	0.66
Antoinel	0.58	0.76	0.66
Andoree	0.53	0.70	0.60
angelina_ku	0.41	0.58	0.48

As shown in the results, our model tends to have high top 5 accuracy, but relatively low top 1 accuracy. This may be closely linked to our contrastive training process: we selected the last positive entity as our positive example instead of the first one, thus encouraging the model to form a more robust representation. However, this may also reduce top 1 accuracy.

Adding the entities of the training dataset into the vocab improved our Top 1 accuracy by about 0.02, because some identifiers that were not in the original vocab can now be mapped to. Also, some entities in the training dataset may re-appear in the test dataset, so the prediction might be easier for these entities. A drawback of this trick is that the same entity may have different UMLS identifiers based on different context, so directly adding the entity and its annotated identifier to the vocab may not be exactly accurate and may sometimes cause wrong mapping.

5. GutBrainIE

5.1. Task description

The GutBrainIE task of BioASQ 2025 focuses on extracting structured information from biomedical abstracts related to the gut microbiota and its connections with Parkinson's disease and mental health. It consists of two subtasks: NER subtask and RE subtask. The RE subtask is further split into three tasks: Binary Relation Extraction (BT-RE), Ternary Tag-Based Relation Extraction (TT-RE) and Ternary

Table 8
Dataset statistics of the GutBrainIE task

Collection	Documents	Entities	Avg Entities per Doc	Relations	Avg Relations per Doc
Train Platinum	111	3638	32.77	1455	13.11
Train Gold	208	5192	24.96	1994	9.59
Train Silver	499	15275	30.61	10616	21.27
Train Bronze	749	21357	28.51	8165	11.90
Development Set	40	1117	27.93	623	15.58

Table 9
Performance of different models on the development dataset of the GutBrainIE NER subtask

Model Name	Micro P	Micro R	Micro F1	Macro P	Macro R	Macro F1
BioLinkBERT	**0.8009**	0.8102	0.8055	**0.7209**	0.7548	0.7098
BiomedBERT	0.7953	0.8138	0.8044	0.7127	0.7434	0.7067
xlm-roberta-large	0.7800	0.8317	0.8050	0.7071	0.7660	0.7087
Ensemble of three BERTs	0.7749	**0.8415**	**0.8069**	0.7038	**0.7831**	**0.7121**

Mention-Based Relation Extraction (TM-RE). The first task only requires participants to find all entity class pairs that have relation between them in a document, the second task also requires the exact relation type, and the third task additionally requires the extraction of entities that are involved in each relation. All tasks are evaluated using Micro and Macro Precision, Recall and F1-score.

Four dataset collections are provided in the train data of this task: The Platinum collection is manually annotated by 7 experts and further validated by experts from another university; the Gold collection is also manually annotated by experts but has not been validated; the Silver collection is manually annotated by about 40 students trained by experts; while the Bronze collection is automatically generated by baseline algorithms. There is also a development set annotated by experts. The dataset statistics are shown in Table 8. Each dataset provides the entity spans as well as all relations between them in a document. There are totally 13 entity types in the dataset, and 25 relation types between them are defined.

This task is challenging for a few reasons: For the NER subtask, there are lots of entity types, and the number of annotations of each type is not balanced. For example, In the Platinum training dataset, there are 1232 DDF entities, but only 20 food entities. This imbalance makes recognition of food difficult. For the RE subtask, the two entities that are in relation with each other are not necessarily in the same sentence but can be very far away, which makes RE difficult because the relation may not be explicitly stated in these cases. Also, the RE subtask is based on the recognized entities of the NER subtask, and a wrong NER prediction will result in wrong RE prediction as well.

5.2. Method

5.2.1. NER subtask

For the NER subtask, we used similar sequence labeling scheme as we used for ELCardioCC. The main difference is that we applied a multi-task scheme to recognize all entity types using a single BERT model and used a separate classifier for each entity type. We trained all classifiers and the BERT model together using cross-entropy loss (Equation is given in section 3.2.1). To deal with the imbalance between different entity types, we applied a simple re-weighting to increase the loss of entity types that doesn't have enough training data:

$$L = \sum_{i=1}^{C} \frac{N}{CN_i} L_i \tag{4}$$

Where C is the number of classes, N is the total number of entities, N_i is the number of entities of class i, and L_i is the total loss of entities of class i.

We used AdamW optimizer with a learning rate of 3e-5 and weight decay of 0.1 for training, and we trained on the silver, gold and platinum training set for 25 epochs. For the final submitted version of the model, we also added the development set into the training data.

We noticed that model ensemble by simply merging the outputs of different BERT models by voting can slightly boost model performance. Specifically, after separate inference using each model, we used average predicted probability of each token as the probability of each entity span, and filtered the predicted entity spans based on the total probability across all models. We selected three BERT models for ensemble: `BioLinkBERT_base` [3], `BiomedBERT_base_uncased_abstract_fulltext` [2] and `xlm_roberta_large_english_clinical` [4]. The NER results for each separate model and the ensembled model on the development dataset are shown in Table 9.

5.2.2. RE subtask

For the RE subtask, we applied a simple classification scheme. We first checked all entity pairs between the entities recognized by our NER model and selected those pairs that possibly have one of the 25 relations defined in the competition, then used BERT to do a 0/1 classification for each entity pair and relation type. We also noticed that more than 80% of relations in the training dataset are between entities that are not far from each other (with distance of less than 200 chars). So we only selected close entity pairs as relation candidates. This will reduce the recall of the model, but makes the subsequent classification much easier.

For the classification step, we used `BioLinkBERT` as our backbone model. We used the following input format: [CLS] (EntityA) (Relation) (EntityB) [SEP] (text) [SEP] (EntityA) [SEP] (text) [SEP] (EntityB) [SEP] (text) [SEP], which provides the BERT model with the relation type as well as the entities along with their context, where the context from the sentence with the first entity to the sentence with the second entity is selected. Since we only kept close entity pairs, the entire input can be fit into the context length (512 tokens) of `BioLinkBERT`. We then used a classifier with two MLP layers to map the final layer embedding of the [CLS] token into the probability that the relation is valid.

During training, we used AdamW with a learning rate of 2e-5 for BERT and 5e-5 for the classifier, and weight decay of 0.1. We trained the model for 5 epochs using cross-entropy loss (Equation is given in section 3.2.1). During inference, we first used the fine-tuned `BioLinkBERT` model on the selected entity pairs to obtain Ternary Mention-Based RE results, then combined all relations in each article to obtain Binary RE and Ternary Tag-Based RE results.

5.3. Results and Discussion

The official results for GutBrainIE task are shown in Table 10, Table 11, Table 12, and Table 13, where the performance of the best run of each team are reported. Our team didn't achieve the best results, yet our NER and Ternary Mention-Based RE Results surpassed the baseline which is quite strong given that the baseline method is also trained on the training datasets. Our results on the other two subtasks are not satisfactory, because we didn't optimize our methods for these two subtasks.

Overall, model ensembling seems to be an effective method, and many top performing teams applied this method to improve the model performance. Careful selection of the BERT models in the ensemble may further enhance the results. Also, instead of combining the final entity span, one can take the embeddings generated by each model as input to the classifier, if all model have the same tokenizer. This method may lead to more effective ensemble of BERT models.

In the RE subtask, our method was simple and effective, though it still have much room for improvement. Ignoring entity pairs that are far away degraded our model performance. However, accurately identifying relations that are far away requires effective method to select related information from the context and ignore distracting information, like in the ATLOP baseline. Future work may explore more effective ways of RE in long documents.

Table 10
Official Results of the GutBrainIE NER subtask

Team ID	Macro-P	Macro-R	Macro-F1	Micro-P	Micro-R	Micro-F1
GutUZH	0.7950	0.7736	**0.7613**	**0.8384**	0.8432	**0.8408**
Gut-Instincts	0.7619	0.7813	0.7591	0.8286	0.8480	0.8382
NLPatVCU	0.8139	0.7161	0.7169	0.8255	0.8488	0.8370
ICUE	**0.8216**	0.7451	0.7546	0.8369	0.8294	0.8331
LYX-DMIIP-FDU (our team)	0.7605	**0.7910**	0.7347	0.8020	**0.8513**	0.8259
ata2425ds	0.7199	0.7546	0.7217	0.7914	0.8432	0.8164
greenday	0.7368	0.7682	0.7471	0.7956	0.8278	0.8114
Graphswise-1	0.7691	0.7398	0.7185	0.8066	0.7955	0.8010
BASELINE	0.6883	0.7690	0.7047	0.7639	0.8238	0.7927
ataupd2425-gainer	0.5808	0.5322	0.5281	0.8333	0.7397	0.7837
DS@GT-bioasq-task6	0.6342	0.7849	0.6872	0.7337	0.8197	0.7743
DS@GT-BioNER	0.6731	0.6497	0.6469	0.7783	0.7437	0.7606
ataupd2425-pam	0.6400	0.7435	0.6763	0.6809	0.7745	0.7247
Schemalink	0.4813	0.5038	0.4650	0.5547	0.5659	0.5602
BIU-ONLP	0.4393	0.3585	0.3711	0.4916	0.4721	0.4816
lasigeBioTM	0.2206	0.1034	0.0863	0.3471	0.1964	0.2509

Table 11
Official Results of the GutBrainIE BT-RE subtask

Team ID	Macro-P	Macro-R	Macro-F1	Micro-P	Micro-R	Micro-F1
Gut-Instincts	**0.5166**	0.6315	**0.5386**	0.6304	0.7532	**0.6864**
ONTUG	0.4185	0.4074	0.4057	0.7121	0.6104	0.6573
Graphswise-104	0.4043	0.3748	0.3832	0.7418	0.5844	0.6538
ataupd2425-pam	0.4807	0.6091	0.4993	0.5671	0.7316	0.6389
BIU-ONLP	0.4632	0.3379	0.3713	0.7453	0.5195	0.6122
BASELINE	0.4650	0.3564	0.3864	**0.7584**	0.4892	0.5947
LYX-DMIIP-FDU (our team)	0.3637	0.4269	0.3688	0.6168	0.5714	0.5933
NLPatVCU	0.3975	0.8419	0.5082	0.4381	0.8571	0.5798
Schemalink	0.3758	0.6573	0.4421	0.4531	0.7533	0.5659
ataupd2425-gainer	0.3171	0.3254	0.2969	0.6150	0.4979	0.5504
ICUE	0.3559	**0.8790**	0.4751	0.3894	**0.9221**	0.5476
ToGS	0.2211	0.1304	0.1451	0.5701	0.2641	0.3610

6. Conclusion

In this paper, we describe our methods and results for three tasks in BioASQ lab for CLEF 2025: ELCardioCC, BioNNE-L and GutBrainIE. We utilized pretrained BERT models and designed a specific technique for each tasks. Our methods achieved good results, ranking the first place in ELCardioCC and the multilingual subtask of BioNNE-L, and also surpassing the baseline on the NER and TM-RE subtask of GutBrainIE. These results further show the effectiveness of BERT embedding for biomedical text mining tasks.

Acknowledgments

We want to express thanks to the organizers from Aristotle University of the Thessaloniki for hosting the ELCardioCC task, the organizers from HSE University, Lomonosov Moscow State University and Kazan Federal University for hosting the BioNNE-L task, and the organizers from University of Padua for hosting the GutBrainIE task. This work has been supported by the National Natural Science Foundation of China (Grant No. 62272105).

Table 12
Official Results of the GutBrainIE TT-RE subtask

Team ID	Macro-P	Macro-R	Macro-F1	Micro-P	Micro-R	Micro-F1
Gut-Instincts	0.4663	0.6445	**0.5184**	0.6280	0.7572	**0.6866**
ataupd2425-pam	0.4409	0.5704	0.4694	0.5853	0.7202	0.6458
ONTUG	0.4254	0.4025	0.4058	0.7059	0.5926	0.6443
Graphswise-1	0.4120	0.3709	0.3840	0.7326	0.5638	0.6372
ICUE	0.4011	0.7123	0.4880	0.4974	0.7860	0.6093
BIU-ONLP	0.4725	0.3288	0.3630	0.7362	0.4939	0.5911
BASELINE	**0.4729**	0.3421	0.3745	**0.7533**	0.4650	0.5751
NLPatVCU	0.3810	**0.8005**	0.4868	0.4362	**0.8436**	0.5750
LYX-DMIIP-FDU (our team)	0.3625	0.4171	0.3549	0.5973	0.5432	0.5690
Schemalink	0.3756	0.6592	0.4437	0.4523	0.7613	0.5675
ataupd2425-gainer	0.3167	0.2315	0.2528	0.7405	0.3992	0.5187
ToGS	0.2261	0.1267	0.1414	0.5556	0.2469	0.3419
lasigeBioTM	0.0797	0.0622	0.0646	0.3929	0.0453	0.0812

Table 13
Official Results of the GutBrainIE TM-RE subtask

Team ID	Macro-P	Macro-R	Macro-F1	Micro-P	Micro-R	Micro-F1
Gut-Instincts	0.3310	0.4303	**0.3497**	0.4215	0.5147	**0.4635**
Graphswise-1	0.3323	0.2369	0.2603	0.4686	0.3097	0.3729
ICUE	0.2509	0.4239	0.2825	0.2858	0.5054	0.3651
LYX-DMIIP-FDU (our team)	0.2106	0.2418	0.1990	0.3682	0.3257	0.3457
ONTUG	0.2589	0.2293	0.2266	0.3529	0.3231	0.3373
BASELINE	**0.3514**	0.1829	0.2123	**0.4986**	0.2453	0.3288
Schemalink	0.2265	0.4088	0.2546	0.1948	0.4665	0.2749
ataupd2425-pam	0.1940	0.2764	0.1982	0.2278	0.3432	0.2738
ataupd2425-gainer	0.2203	0.1384	0.1538	0.4272	0.1810	0.2542
NLPatVCU	0.1522	**0.5041**	0.2163	0.1423	**0.6005**	0.2300
BIU-ONLP	0.1171	0.0854	0.0879	0.2339	0.1461	0.1799
ToGS	0.0249	0.0180	0.0203	0.1702	0.0536	0.0815
lasigeBioTM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] 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), 2019, pp. 4171–4186.

[2] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, ACM Transactions on Computing for Healthcare (HEALTH) 3 (2021) 1–23.

[3] M. Yasunaga, J. Leskovec, P. Liang, Linkbert: Pretraining language models with document links, arXiv preprint arXiv:2203.15827 (2022).

[4] L. Lange, H. Adel, J. Strötgen, D. Klakow, Clin-x: pre-trained language models and a study

on cross-task transfer for concept extraction in the clinical domain, Bioinformatics 38 (2022) 3267–3274.
[5] A. Nentidis, G. Katsimpras, A. Krithara, M. Krallinger, M. Rodríguez-Ortega, E. Rodriguez-López, N. Loukachevitch, A. Sakhovskiy, E. Tutubalina, D. Dimitriadis, G. Tsoumakas, G. Giannakoulas, A. Bekiaridou, A. Samaras, G. M. Di Nunzio, N. Ferro, S. Marchesin, M. Martinelli, G. Silvello, G. Paliouras, Overview of BioASQ 2025: The thirteenth BioASQ challenge on large-scale biomedical semantic indexing and question answering, volume TBA of *Lecture Notes in Computer Science*, Springer, 2025, p. TBA.
[6] D. Dimitriadis, V. Patsiou, E. Stoikopoulou, A. Toumpas, A. Kipouros, D. Papadopoulos, A. Bekiaridou, K. Barmpagiannos, A. Vasilopoulou, A. Barmpagiannos, A. Samaras, G. Giannakoulas, G. Tsoumakas, Overview of ElCardioCC Task on Clinical Coding in Cardiology at BioASQ 2025, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), CLEF 2025 Working Notes, 2025.
[7] A. Sakhovskiy, N. Loukachevitch, E. Tutubalina, Overview of the BioASQ BioNNE-L Task on Biomedical Nested Entity Linking in CLEF 2025, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), CLEF 2025 Working Notes, 2025.
[8] M. Martinelli, G. Silvello, V. Bonato, G. M. Di Nunzio, N. Ferro, O. Irrera, S. Marchesin, L. Menotti, F. Vezzani, Overview of GutBrainIE@CLEF 2025: Gut-Brain Interplay Information Extraction, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), CLEF 2025 Working Notes, 2025.
[9] M. V. Koroteev, Bert: a review of applications in natural language processing and understanding, arXiv preprint arXiv:2103.11943 (2021).
[10] M. Sung, M. Jeong, Y. Choi, D. Kim, J. Lee, J. Kang, Bern2: an advanced neural biomedical named entity recognition and normalization tool, Bioinformatics 38 (2022) 4837–4839.
[11] L. Luo, C.-H. Wei, P.-T. Lai, R. Leaman, Q. Chen, Z. Lu, Aioner: all-in-one scheme-based biomedical named entity recognition using deep learning, Bioinformatics 39 (2023) btad310.
[12] M. Sänger, S. Garda, X. D. Wang, L. Weber-Genzel, P. Droop, B. Fuchs, A. Akbik, U. Leser, Hunflair2 in a cross-corpus evaluation of biomedical named entity recognition and normalization tools, Bioinformatics 40 (2024) btae564.
[13] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazaré, M. Lomeli, L. Hosseini, H. Jégou, The faiss library, arXiv preprint arXiv:2401.08281 (2024).
[14] F. Liu, E. Shareghi, Z. Meng, M. Basaldella, N. Collier, Self-alignment pretraining for biomedical entity representations, arXiv preprint arXiv:2010.11784 (2020).
[15] A. Sakhovskiy, N. Semenova, A. Kadurin, E. Tutubalina, Graph-enriched biomedical entity representation transformer, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2023, pp. 109–120.
[16] A. Sakhovskiy, N. Semenova, A. Kadurin, E. Tutubalina, Biomedical entity representation with graph-augmented multi-objective transformer, in: Findings of the Association for Computational Linguistics: NAACL 2024, 2024, pp. 4626–4643.
[17] W. Zhou, K. Huang, T. Ma, J. Huang, Document-level relation extraction with adaptive thresholding and localized context pooling, in: Proceedings of the AAAI conference on artificial intelligence, volume 35, 2021, pp. 14612–14620.
[18] P.-T. Lai, C.-H. Wei, L. Luo, Q. Chen, Z. Lu, Biorex: improving biomedical relation extraction by leveraging heterogeneous datasets, Journal of Biomedical Informatics 146 (2023) 104487.
[19] P.-T. Lai, C.-H. Wei, S. Tian, R. Leaman, Z. Lu, Enhancing biomedical relation extraction with directionality, arXiv preprint arXiv:2501.14079 (2025).
[20] J. Koutsikakis, I. Chalkidis, P. Malakasiotis, I. Androutsopoulos, Greek-bert: The greeks visiting sesame street, in: 11th Hellenic Conference on Artificial Intelligence, SETN 2020, Association for Computing Machinery, New York, NY, USA, 2020, p. 110–117. URL: https://doi.org/10.1145/3411408.3411440.
[21] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, arXiv preprint arXiv:1711.05101 (2017).
[22] O. Bodenreider, The unified medical language system (umls): integrating biomedical terminology, Nucleic acids research 32 (2004) D267–D270.

[23] N. Loukachevitch, A. Sakhovskiy, E. Tutubalina, Biomedical concept normalization over nested entities with partial umls terminology in russian, in: Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), 2024, pp. 2383–2389.
[24] V. Davydova, N. Loukachevitch, E. Tutubalina, Overview of bionne task on biomedical nested named entity recognition at bioasq 2024, CLEF Working Notes (2024).