# GetGut: A Multi-Stage Pipeline for Gut-Brain Axis Information Extraction

Notebook for the GutBrainIE Task at the BioASQ Lab on Gut-Brain Interplay Information Extraction at CLEF 2026

Ernesto Garavito Molina[†], Sophus Jørgensen[†], Juan Manuel Rodriguez and Daniele Dell'Aglio

*Department of Computer Science, Aalborg University, Aalborg, Denmark*

**Abstract**

This paper presents the GetGut@AAU pipeline, a multi-stage, waterfall information extraction architecture designed for the GutBrainIE Task within the BioASQ Lab at CLEF 2026. Our solution addresses the extraction of structured knowledge across four subtasks: Named Entity Recognition (NER), Named Entity Recognition and Disambiguation (NERD), Mention-Level Relation Extraction (M-RE), and Concept-Level Relation Extraction (C-RE). To address the disparities in annotation density and quality across the training datasets, we introduce a Weighted Funnel Fine-Tuning methodology for our NER and M-RE models. We progressively train our pipeline by starting with large-scale, weakly supervised data and gradually fine-tuning on smaller, expert-annotated datasets. This allows learning broad biomedical terminology before refining precise entity boundaries. Furthermore, we address data quality on a per-task basis. We mitigate positive-class bias in M-RE through a controlled negative sampling mechanism, and we boost NERD accuracy by excluding distantly supervised noise from our reference dictionary. Our proposed system achieved third place in both the M-RE and C-RE subtasks with Micro F1 scores of 0.4054 and 0.2020, respectively, and achieved 6th and 5th place in the NER and NERD subtasks with Micro F1 scores of 0.8014 and 0.5517. All code regarding the implementation can be found at: https://github.com/SophusJ/GetGut-AAU.

**Keywords**

Information Extraction, Named Entity Recognition, Relation Extraction, Funnel Fine-Tuning, Gut-Brain Axis, Label Disambiguation

## 1. Introduction

The study of gut microbiota and its influence on neurological conditions, often referred to as the "gut-brain axis", has garnered significant attention in recent years. Emerging research indicates strong correlations between microbiota composition and neurodegenerative diseases such as Alzheimer's, Parkinson's, Multiple Sclerosis, and Amyotrophic Lateral Sclerosis [1, 2, 3]. While these findings are critical for the medical community, the rapid development of literature in this domain has created an information overload, making it increasingly difficult for experts to manually synthesise the expanding body of evidence.

To cope with the increasing volume of available literature, researchers are focusing on tools that can automatically analyse and combine it. Advances in Information Extraction (IE) [1], a subfield of Natural Language Processing (NLP), provide a robust framework for this task. Unlike standard retrieval methods, IE is designed to identify and extract specific entities and semantic relationships from unstructured text. By transforming dense scientific prose into structured, machine-readable formats, IE allows researchers to extract meaningful data about the gut-brain axis.

Initiatives like the GutBrainIE Task [4] within the BioASQ Lab [5], introduced at CLEF 2026, reach experts in IE, NLP and the gut-brain axis to develop new solutions and tools. This task aims to leverage

*CLEF 2026 Working Notes, 21-24 September 2026, Jena, Germany*

[†] These authors contributed equally.

ernegaravito@gmail.com (E. Garavito Molina); sophus-dog@hotmail.com (S. Jørgensen); jmro@cs.aau.dk (J. M. Rodriguez); dade@cs.aau.dk (D. Dell'Aglio)

0000-0002-1130-8065 (J. M. Rodriguez); 0000-0003-4904-2511 (D. Dell'Aglio)

NLP techniques to automate the extraction of structured knowledge regarding the gut-brain interplay through four subtasks:

- **Subtask 6.1.1 - Named Entity Recognition (NER):** The identification and classification of text spans into 13 predefined categories (e.g., *bacteria*, *chemical*, *microbiota*).
- **Subtask 6.1.2 - Named Entity Recognition and Disambiguation (NERD):** An extension of NER that requires linking identified mentions to Uniform Resource Identifiers (URI) within standardised biomedical ontologies.
- **Subtask 6.2.1 - Mention-Level Relation Extraction (M-RE):** The extraction of semantic triples relating two specific entity mentions within a document context.
- **Subtask 6.2.2 - Concept-Level Relation Extraction (C-RE):** The identification of relationships between normalised concepts, abstracting away from specific lexical surface forms to capture knowledge-level connections.

To facilitate model training and evaluation, the organisers provide a multi-tiered dataset characterised by varying levels of supervision:

- **Gold Collection:** High-quality, fully supervised data manually annotated by domain experts.
- **Silver Collections (Standard & 2025):** Weakly supervised datasets annotated by trained students, introducing a trade-off between volume and annotation precision.
- **Bronze Collection:** A distantly supervised dataset generated via automated heuristics, containing no manual verification.

Across all tiers, the data collections comprise comprehensive metadata, including authors, titles, and abstracts, with the abstracts serving as the primary textual corpus for extraction. While the underlying schema of entities, relations and URI definitions remains consistent, navigating the disparities in annotation density and supervision quality across these collections presents a significant hurdle.

The extraction of structured knowledge from gut-brain axis literature is hindered by the inconsistencies in annotation quality across datasets. Noisy and weakly supervised annotations increase the difficulty of training reliable NER, NERD, and Relation Extraction (RE) models capable of producing accurate biomedical relationships. The problem we focus on is how to account for these known differences in annotation quality when fine-tuning models to extract structured biomedical knowledge.

We, the **GetGut@AAU** team, propose a sequential, waterfall-style pipeline. It utilises funnel-finetuning for the NER and M-RE models, while supporting the NERD and C-RE models with a URI dictionary. Each model was specifically selected for its respective stage to seamlessly ingest, analyse, and output entity inferences, disambiguation, and relationship extractions. We also examine whether Quantized Low-Rank Adaptation (QLoRA)-tuned Large Language Modelss (LLMs) can bridge the performance gap that emerges when replacing the funnel-finetuning approach of smaller, domain-specific models such as BiomedBERT [6].

As part of our solution, we introduce a Weighted Funnel Fine-Tuning methodology for our NER and M-RE models to address the disparities in annotation density and supervision quality across the multi-tiered dataset. We address positive-class bias in the M-RE task by implementing a controlled negative sampling mechanism that effectively prevents overprediction of relationships. For the NERD and C-RE subtasks, we propose a hybrid scoring approach that explicitly accounts for data quality and that excludes distantly supervised data from our reference dictionaries, significantly boosting entity linking accuracy. We experimentally compare Parameter Efficient Fine-Tuning (PEFT) (QLoRA) on large generalist LLMs and Full Fine-Tuning (FFT) on smaller, domain-specific models, highlighting the structural trade-offs required to address dataset class imbalances under hardware constraints.

Our primary contributions include the development of a multi-stage waterfall pipeline, a Weighted Funnel Fine-Tuning methodology, and a controlled negative sampling mechanism, which collectively enabled our system to achieve 3rd place in both the M-RE and C-RE subtasks with Micro F1 scores of 0.4054 and 0.2020, respectively. The remainder of this paper is organised as follows: Section 3 details the preprocessing steps, Section 4 outlines the model architecture, and Section 5 presents the evaluation of our system.

## 2. Related Works

Historically, the field of IE gravitated toward joint modelling, in which NER and RE were treated as a single, unified task rather than separate steps [7]. This approach relied on a single shared network to predict both entities and relations simultaneously, driven by the assumption that this would mitigate error propagation from NER into RE. Zhong and Chen [7] challenged this by introducing a “frustratingly easy" pipeline that leverages two independent, highly optimised encoders for NER and RE. They separated the representations and injected specific entity markers into the relation model’s input and established new state-of-the-art results, suggesting that independent architectures can outperform joint models when contextual entity boundaries are clearly defined.

The work done in Chen et al. [8] is another example where building on independent architectures has proven highly effective for complex domains. They propose a fine-tuned MC-BERT approach along with downstream BiLSTM, CNN, and CRF layers, establishing a highly robust NER system capable of navigating complex, domain-specific text structures. This shows promise that a singular fine-tuned specialised model is capable of solving complex NER tasks, with the right model selection, fine-tuning approach and downstream layers.

**LLM Integration and Text Simplification in Biomedicine** Research in biomedical IE has increasingly explored the integration of LLMs to address the ambiguity inherent in highly descriptive medical terminology. For instance, Borchert et al. [9] propose to use generative LLMs to improve entity linking by simplifying complex, multi-token entity mentions into canonical forms before candidate retrieval.

While this mention-level simplification is effective for normalising isolated clinical symptoms, applying such an approach in the GutBrainIE task presents challenges. Given that our objective depends strictly on exact matching, altering the representation of entities can result in a loss of precise matches. We prioritise preserving the original textual spans while employing robust encoders for disambiguation.

**Comparing FFT with PEFT** Selecting the optimal fine-tuning strategy requires balancing efficiency against fidelity. PEFT methods, such as Low-Rank Adaptation (LoRA) [10], operate by freezing the base model and restricting weight updates to a small, targeted subset of parameters within specific layers. Because PEFT methods only train a fraction of the network, these methods offer significant computational advantages, drastically reducing the memory and processing power required. Conversely, FFT involves calculating gradients and updating the weights across every single layer of the entire network. While this comprehensive update demands substantially more memory and compute, it provides the distinct advantage of fundamentally altering the model’s internal representations, thereby achieving superior domain adaptation and higher overall performance.

Sreenivas et al. [11] highlight this disparity, noting a significant performance gap across the MACCROBAT dataset depending on the fine-tuning methodology. Their study showed that applying a supervised FFT approach with BERT achieved an F1 score of 0.708, whereas applying LoRA [10], a PEFT method, resulted in a drastically lower F1 score of just 0.05 on the same task. Sreenivas et al. attribute this to “LoRA’s underperformance to the limited size of the dataset and the need for precise domain adaptation, which favours Full Fine-Tuning”. This is further supported by the development of Med42 [12], a clinical Large Language Model. The authors observed that while linear approaches to model fine-tuning yield competitive results with lower overhead, FFT consistently outpaces PEFT in overall performance and domain adaptation accuracy.

**Efficient Fine-Tuning Under Resource Constraints** Despite the empirical advantages of FFT, applying it to highly parameterised LLMs is often computationally prohibitive. To address memory constraints, Dettmers et al. [13] propose QLoRA, an efficient fine-tuning approach that backpropagates gradients through a frozen, 4-bit quantised pretrained language model into Low Rank Adapters. In their experiments, they show that QLoRA allows for the training of massive models (up to 65B parameters) on single GPUs without significant performance degradation compared to 16-bit full fine-tuning baselines.

**Learning with Noisy Labels and Weak Supervision** Finally, the varying level of annotation quality in the GutBrainIE dataset must be addressed. Traditional models are highly sensitive to label noise, which is abundant in distantly supervised (Bronze) and weakly supervised (Silver) collections. Andersen et al. [14] show that models are prone to over-indexing on noisy, heuristically generated predicates. To prevent this while still leveraging the massive volume of lower-quality datasets, specialised training regimens must be designed. This challenge directly motivates our proposed Weighted Funnel Fine-Tuning methodology.

# 3. Data analysis

In this section, we present an in-depth analysis of the GutBrainIE dataset, detailing its structural components, semantic properties, and statistical distributions. We begin by outlining the high-level architecture and formatting of the document entries in Section 3.1, defining the scope of metadata, entities, and varying levels of relational data. Section 3.2 details the 13 biomedical entity categories and linking via URI to resolve lexical variations. Following this, Section 3.3 provides an exploratory data analysis of annotation density across the corpus, utilising Kernel Density Estimation (KDE) plots to examine the correlation between document length and annotation frequency. Finally, overarching dataset statistics are evaluated in Section 3.4, highlighting key insights regarding entity density consistency, relation variability across supervision tiers, and the long-tail learning challenges introduced by class imbalances.

## 3.1. Data Structure and Format

In the provided dataset, each key corresponds to a unique PubMed Article IDentifier (PMID). Each document entry encapsulates metadata alongside structured annotations for entities and various levels of relations. The high-level architecture of a single document entry is defined as follows:

- **Metadata:** Contains bibliographical information including the *title*, *author*, *journal*, *year*, *abstract*, and the *annotator*.
- **Entities:** A list of identified entities within the text. Each entity object records its precise location (`start_idx`, `end_idx`, `location`), the exact `text_span`, its semantic `label`, and its corresponding `uri`.
- **Relations:** Detailed relational data containing the full span, location, label, and URI for both the subject and the object, connected by a specific `predicate`.
- **Mention-Level Relations:** A simplified representation focusing strictly on the surface text, containing the `subject_text_span`, `subject_label`, `predicate`, `object_text_span`, and `object_label`.
- **Concept-Level Relations:** An abstracted representation focusing on knowledge, comprising the `subject_uri`, `subject_label`, `predicate`, `object_uri`, and `object_label`.

## 3.2. Entity Categorisation and Semantic Linking

The GutBrainIE task defines a classification consisting of 13 biomedical entity categories: *Anatomical Location, Animal, Bacteria, Biomedical Technique, Chemical, Dietary Supplement, Disease/Disorder/Finding (DDF), Drug, Food, Gene, Human, Microbiome,* and *Statistical Technique.*

A central component of the advanced subtasks (NERD and C-RE) is the normalisation of these surface-level entity mentions to standardised reference resources using **URI**. A URI is a unique string that identifies a specific concept, for example within a biomedical ontology (such as MeSH, NCIT, or CHEBI). For instance, while the surface texts "patients" and "human subjects" vary lexically, they are both resolved to the same ontological concept via the URI `http://id.nlm.nih.gov/mesh/D010361`. This semantic linking process abstracts lexical variations, enabling the evaluation of knowledge-level connections rather than string-matching.

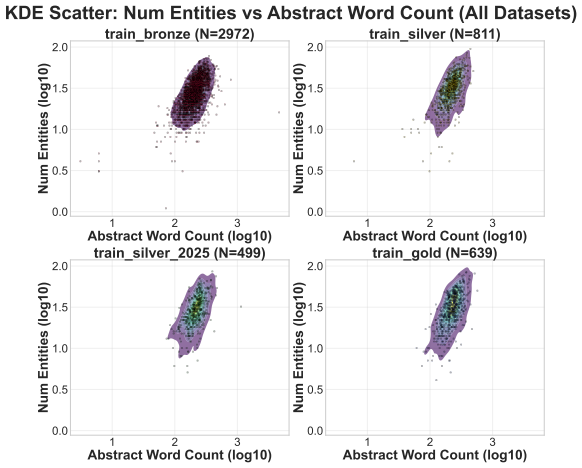
### 3.3. Exploratory Data Analysis

An initial analysis of the annotation density reveals variability across the document corpus. While every article successfully contains at least one identified entity, a subset of the data lacks relational annotations. Specifically, 147 articles do not contain a single relationship. The distribution of these zero-relation articles across the different collection tiers is detailed in Table 1.

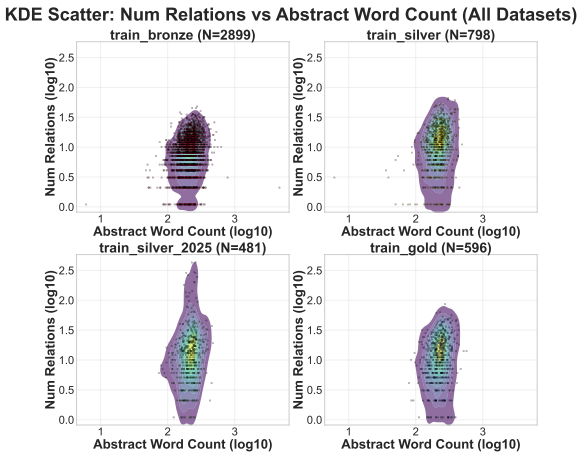
**Table 1**
Distribution of articles containing zero relations across the dataset collections. High-level annotation density across training collections obtained from the official Task overview [5].

| Collection | Articles | Zero-Relation Articles | Percentage | Avg Entities | Avg Relations |
|---|---|---|---|---|---|
| Gold | 639 | 43 | 6.73% | 32.13 | 13.39 |
| Silver | 811 | 13 | 1.60% | 32.22 | 13.45 |
| Silver (2025) | 499 | 18 | 3.61% | 30.61 | 21.27 |
| Bronze | 2972 | 73 | 2.46% | 30.28 | 9.99 |
| **Total** | **4921** | **147** | **2.99%** | **31.31** | **14.52** |

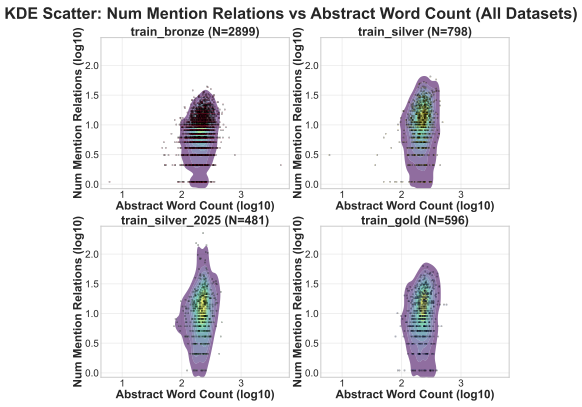
To visually explore the relationship between document length and annotation density, KDE scatter plots 1 were generated across all four collections. Both the abstract word count (x-axis) and the respective annotation counts (y-axis) are presented on a base-10 logarithmic scale to accommodate the wide variance in the data.



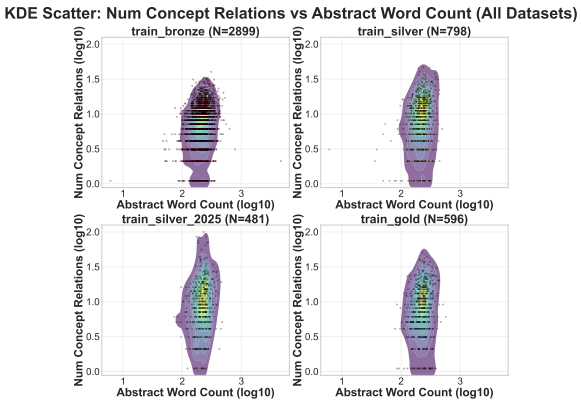

(a) KDE: Entities




(b) KDE: Relations




(c) KDE: Mention Relations




(d) KDE: Concept Relations

Figure 1: KDE combined distribution analysis.

**Correlation in Entity Density:** Across all tiers, the density shapes into a diagonal ellipse. This suggests that as the length of an abstract increases, the number of annotated entities scales at a

predictable rate.

**Variability in Relation Density:** In contrast, the distributions for Relations, Mention Relations, and Concept Relations behave differently from the entities. Rather than a diagonal trend, they show a vertical shape. This suggests a weak correlation between the length of an abstract and the number of relationships it contains.

### 3.4. Dataset Statistics and Insights

An analysis of the dataset statistics provided by the organisers [4] gives insights regarding the distribution and density of annotations across the different supervision tiers. Table 1 shows the high-level statistics of the training collections. A review of these distributions yields three primary observations:

- **Entity Density Consistency:** Despite the stark differences in annotation methodologies, ranging from expert curation (Gold) to automated heuristics (Bronze), the average number of entities identified per document remains remarkably stable, hovering tightly between 30.28 and 32.22. This suggests a consistent linguistic density of relevant biomedical terms across the PubMed abstracts.
- **Relation Density:** Relationship extraction is highly sensitive to the supervision method. The Silver 2025 collection exhibits a significant spike in relation density (21.27 per document), while the distantly supervised Bronze collection drops to just 9.99 relations per document.
- **Class Imbalance:** Across all tiers, the dataset is heavily skewed toward specific entity types. *Disease, Disorder, or Finding (DDF)* is consistently the most dominant category, accounting for 32% to 38% of all entities, followed by *Chemical* (∼14-16%) and *Microbiome* (∼9-10%). Conversely, categories like *Gene*, *Food*, and *Compared To* relations represent less than 2% of the data, presenting a significant "long-tail" learning challenge for predictive models.

### 3.5. Relation Distance Analysis and Bias Mitigation

While data volume was optimal for entity recognition, applying this same logic to RE poses a risk. An analysis of the relation distribution across the dataset reveals a structural imbalance between intra-sentence (same-sentence) and inter-sentence (cross-sentence) relationships. Out of 53,874 verified relations across all tiers, 49,066 ($91.07\%$) occur within the boundaries of a single sentence, leaving only 4,808 ($8.93\%$) that span across multiple sentences.

Training the RE module heavily on the Bronze dataset risks introducing a strong bias, causing the model to assume that valid relationships are strictly confined to single sentences. This could limit its ability to identify the complex, multi-sentence dependencies found in the Gold standard. Consequently, even though cross-sentence relationships represent a minority of the data, the training strategy must explicitly account for them to ensure robust performance.

## 4. Methodology

This section presents our solution to the GutBrainIE CLEF 2026 challenge. We present a waterfall pipeline, which can be seen in Figure 2. Our pipeline consists of 4 stages, one for each of the 4 subtasks:

1. **Named Entity Recognition (Subtask 6.1.1):** A specialised NER model is deployed to identify and classify entity spans found within the title and abstract from the PubMed documents.
2. **Named Entity Recognition and Disambiguation (Subtask 6.1.2):** This stage uses the inferred entities alongside the title and abstract contained within each document. By applying similarity measures to the context surrounding each entity, a concept-URI is added and output as the prediction.
3. **Mention-Level Relation Extraction (Subtask 6.2.1):** To address M-RE, the extracted entities from subtask 6.1.1 is reused. By checking the extracted entities, it is evaluated whether or not

entities within a context window have a valid relation or no relation, outputting those that have a valid relation as a mention-level pairing.

4. **Concept-Level Relation Extraction (Subtask 6.2.2):** The C-RE stage utilize the output from the prior subtask 6.2.1. By reading each entity from the relations as well as their context, a prediction is made in a similar fashion to the NERD subtask 6.1.2.

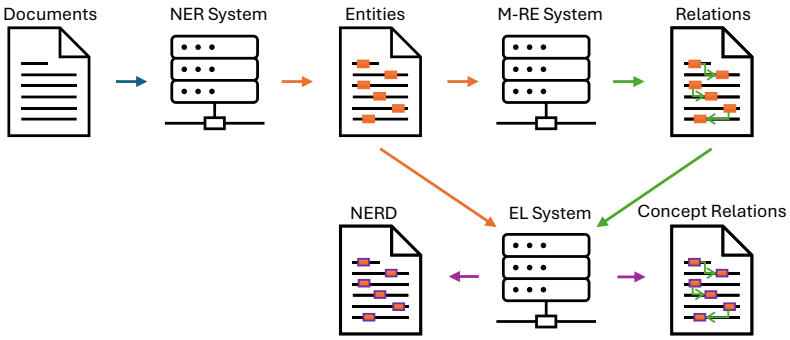



**Figure 2:** Extraction pipeline overview

## 4.1. Baseline data evaluation: Data Quantity vs. Quality in NER

Before designing our final training pipeline, we conduct preliminary baseline experiments to isolate and quantify the impact of annotation quality versus dataset volume on NER. To reduce variance due to random initialisation, we run five independent fine-tuning runs per dataset (Gold, Silver, Silver 2025, and Bronze).

For the model architecture, we utilise the `bert-base-uncased` [15] encoder, which processes each sentence into contextualised embeddings to capture relationships with surrounding words. On top of this encoder, we appended a token classification layer as a classification head to evaluate the embeddings produced for each token; this is the component that is updated during training.

To address tokenisation mismatches, where longer terms are fragmented into multiple subwords (e.g., splitting "acetaminophen" into "acet", "##amino", and "##phen"), we apply a label alignment strategy. The ground-truth NER label is assigned exclusively to the first subword fragment of the tokenised text. Subsequent subword fragments are assigned a masking index of $-100$, ensuring they are ignored for training and loss calculation purposes.

The results support the hypothesis that, for foundational surface-level extraction (NER), dataset scale can outweigh strict annotation quality, likely because larger corpora provide broader exposure to specialised biomedical vocabulary and to variation in entity boundaries. At the same time, the relatively small difference between Silver Standard and Gold suggests that quality-focused datasets remain important for later-stage calibration, especially for structurally harder downstream relation tasks.

## 4.2. Selection of Models

Due to the high amount of resources required to train a model from scratch, we use a pre-trained specialised models for different parts of the pipeline. All selected models are pretrained on biomedical literature, ensuring that they are well-suited to the terminology and linguistic structures present in the GutBrainIE corpus. We selected the following models:

- `Ihor/gliner-biomed-large-v1.0` [16] for the NER task, GLiNER treats NER as a span-matching problem using a bi-encoder architecture instead of relying on a fixed label set [17].

The biomedical large variant provides strong zero-shot and few-shot generalisation while being specialised for biomedical terminology, making it effective for overlapping multi-token entities such as bacterial strains and dietary supplements.

- `SapBERT-from-PubMedBERT-fulltext` [18] for the NERD and C-RE tasks. SapBERT uses a self-alignment metric learning framework to cluster synonymous biomedical entities while separating unrelated concepts [18]. Built upon PubMedBERT-fulltext, it is pretrained entirely on biomedical literature, enabling robust handling of lexical variability in biomedical entity linking.
- `BiomedNLP-BiomedBERT-large-uncased-abstract` [6] for Relation Extraction (RE). As the GutBrainIE corpus primarily consists of PubMed abstracts, the model is well adapted to biomedical terminology and scientific phrasing. Following the "frustratingly easy" pipeline approach [7], the RE task is decoupled from NER, allowing BiomedBERT-Large to specialise in modelling the semantic dependencies required for Mention-Level (M-RE) extraction.

## 4.3. Weighted Funnel Fine-Tuning

To bridge the gap between structural complexity, relation distance, and label noise, we propose a training paradigm: **Weighted Funnel Fine-Tuning**. This approach adapts our parsing architecture by introducing a staged learning phase that shifts its reliance on data volume versus data quality.

Our pipeline is structured sequentially to leverage the proven strengths of the multi-tiered dataset:

1. **Stage 1 (Domain Syntax & Local Relations):** Capitalising on our baseline NER findings 4.1, the NER model is initially fine-tuned on the massive Bronze collection ($n = 2972$). During this stage, the model focuses on acquiring the foundational biomedical vocabulary and intra-sentence relationships. Similarly, for sentence-level RE batches, Bronze data is heavily utilised to teach local sentence relationships.
2. **Stages 2 & 3 (Weak Supervision & Cross-Sentence Introduction):** The funnel narrows to the Silver and Silver 2025 collections ($n = 1310$). As the model processes data annotated by trained students, training step sizes for entity and weights for relationship tasks are continuously updated.
3. **Stage 4 (Expert Calibration):** In the final stage, the funnel is restricted entirely to the Gold collection ($n = 639$). Maximum loss weights are applied across all tasks.

## 4.4. Funnel Fine-Tuning of GliNER for NER

For the Named Entity Recognition component of our pipeline, we fine-tuned our `gliner-biomed-large-v1.0` model using the technique described in section 4.3 to adapt the model to the specific challenge and entities.

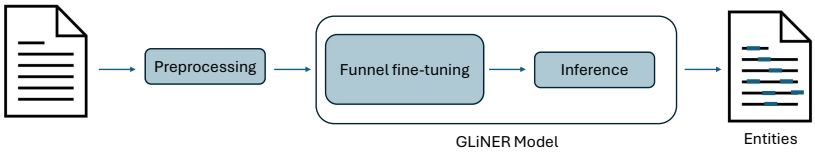



**Figure 3:** NER overview

### 4.4.1. Preprocessing: Data preparation and alignment

The GLiNER model's architecture does not accept raw strings, meaning the documents have to be converted into a token-aligned format. To tokenise the documents, the titles and abstracts are split, and

then a regular expression (`\w+|[^\w\s]`) is applied to separate each word and punctuation, outputting a list of tokens. These tokens are then further separated into a dictionary structure that is usable for GLiNER, as it has a strict $[start\_token\_idx, end\_token\_ix, LABEL]$ schema.

#### 4.4.2. Fine-tuning: Preserving knowledge

To optimise the fine-tuning across the datasets of varying annotation quality, we apply our funnel fine-tuning approach. Rather than applying an explicit mathematical weight, we attempt to use an implicit weighting mechanism in step sizes. The training curriculum is divided into four stages: bronze (1,000 steps), silver2025 (1,500 steps), silver (2,000 steps), and gold (3,000 steps). By training on the lower-quality data first with fewer steps, a rough initialisation of the target entities is established. Exposing the model to the high-quality gold annotations for the longest duration ensures that the final parameter updates are done with the most reliable data. This is to exploit recency bias and overwrite early noise that is found in the bronze dataset.

Furthermore, as `gliner-biomed-large-v1.0` is already a highly trained model, it is important to ensure we preserve the encoder's foundational understanding of biomedical semantics. Therefore, the learning rate of the encoders is restricted to $1 \times 10^{-5}$, while the classification layer is allowed to adapt to our specific entities with a learning rate of $5 \times 10^{-5}$.

#### 4.4.3. Inference: Sliding windows

To ensure that no information is lost during inference, we implement a sliding window with a window size of 450 tokens and an overlap of 50 tokens. The 450-token limit is to conform to the underlying encoder's maximum sequence length of 512 tokens, leaving a necessary buffer for GLiNER's prepended entity labels and special tokens without causing truncation. The model extracts entities locally within the constrained sliding windows. However, post-extraction, the $start\_index$ and $end\_index$ of the extracted entities must match. Therefore, a coordinate-mapping function calculates the character offset for each sliding window, ensuring it is possible to calculate character offsets for the submission as well as downstream tasks.

### 4.5. A Frustratingly Easy Approach for NERD and C-RE

When approaching the NERD and C-RE subtasks, our strategy relies on building a reference dictionary of the entities mapped to their respective URIs. Our objective is to utilise these mappings to predict URIs for new entities, with relatively low computational power.

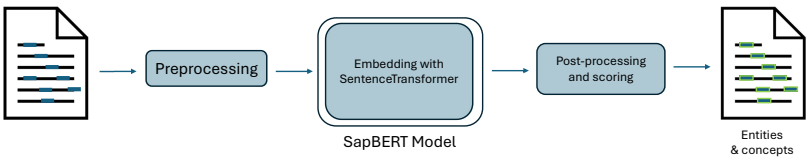



**Figure 4:** NERD/C-RE overview

#### 4.5.1. Preprocessing: Extracting context and conflict resolutions

To capture the contextual semantics, we implement a sentence-level splitting technique before embedding anything into a vector space. Therefore, each document is divided into sentences. Each entity within a sentence is paired with the sentence. Or as follows. Let us assume the input text "Patient was given *Aspirin*. He reported no pain." where *aspirin* is an entity. The text is divided into two sentences.

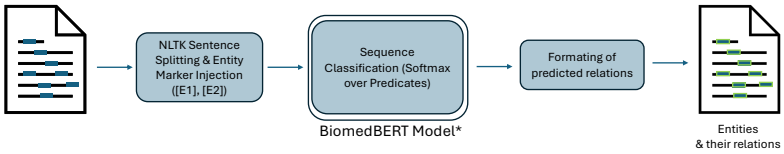



**Figure 5:** M-RE overview, BiomedBERT Model* refers to `BiomedNLP-BiomedBERT-large-uncased-abstract`

The first one, i.e. “Patient was given Aspirin.", contains the entity “aspirin". Hence, the resulting representation of the entity is “aspirin [SEP] Patient was given Aspirin"; we call this string *context_string*. Then, the associated URI is paired with the string; *(context_string, uri)*.

To ensure that a *context_string* does not represent different URIs, we implement a counting solution. If we have any cases where a context pair is not unique, such that *(context_string, uri)* is equal to another or multiple *(context_string, uri)* pairs, we count whichever URI is most common in combination with the specific *context_string*. This is then the chosen pair that is added to a temporary dataframe, before it is embedded by the `SapBERT-from-PubMedBERT-fulltext` model.

#### 4.5.2. Vectorization via SapBERT

We encode the text using `SapBERT-from-PubMedBERT-fulltext` as it is trained to encode medical words with similar meanings in such a way that they are closely related. For example, “aspirin” and “painkiller” are quite different words, but they are closely related. A machine that cannot recognise this similarity will encode these in a way where they are not similar, leading to a low score when comparing with a similarity measure.

Furthermore, when encoding, we have to ensure *context_string* is kept in the encoding. We want the embedding to cover our entire *context_string*. To solve this issue, we utilise the SentenceTransformers[19] library, which lets us encode the strings with the `SapBERT-from-PubMedBERT-fulltext` model, as it adds a pooling layer on top of the encoder. Using the pooling layer, we ensure a fixed-size sentence embedding.

#### 4.5.3. Post-processing: Hybrid Scoring

During inference, URIs must be assigned to incoming entities with unknown URIs. While the [SEP]-tag in the *context_string* ensures we always know the entity, initial testing revealed that querying the dictionary using only the entity or the full sentence leads to less than favourable results. If the full sentence is used, the scoring of the sentence would overpower the entity, and using only the entity does not work, as there is no context for that specific entity in that specific context.

To resolve this, we use a hybrid scoring approach as a post-processing action. For any incoming prediction, the system isolates the entity and its host sentence. These are then encoded with `SapBERT-from-PubMedBERT-fulltext` and compared to our dictionary using cosine similarity, such that two scores are returned. These scores are the similar entities($s_e$), and the score for the similar contexts($s_c$). Our hybrid scoring then applies an experimentally chosen weight $w_e$ and $w_c$ to each of our scores.

$$\textbf{Final Score} = (w_e \times s_e) + (w_c \times s_c)$$

We utilise these scores in our hybrid scoring approach, and return the top-1 result from the Final score as the predicted URI. This is done for each individual entity that is predicted.

### 4.6. M-RE, Fine-Tuning BiomedBERT for Relation Extraction

For the RE component of our pipeline, we fine-tune the `BiomedNLP-BiomedBERT-large-uncased-abstract` model, treating relation classification as a sequence classification task to predict semantic relationships

between entity pairs.

#### 4.6.1. Preprocessing: Entity markers and negative sampling

To enable the model to identify which entities within a sentence are under evaluation, the text inputs are marked with special tokens: `[E1]`, `[/E1]`, `[E2]`, and `[/E2]`. These markers are injected around the subject and object entities before tokenisation. Any sequences exceeding BERT's constraints are securely truncated to a maximum length of 512 tokens.

#### 4.6.2. Figuring out What is Not a Relationship

While the datasets provide a rich example of what relationships look like, training only on them may introduce a bias towards positive relationships. To test this hypothesis, `BiomedNLP-BiomedBERT-large-uncased-abstract` was trained only in the positive relationship instances. After an experimental run, we confirm that training only on positive relationships is not an ideal solution for this task.

To mitigate this imbalance, a negative sampling mechanism is implemented. We sample entity combinations that have no relation to create `NO_RELATION` instances. A controlled negative-to-positive ratio of 3:1 is enforced, and in contexts entirely devoid of positive relations, the mechanism caps the sampling to a maximum of 3 negative relations per positive one.

#### 4.6.3. Fine-tuning: Staged curriculum and structural weighting

Rather than combining all our corpora, we construct a funnel training curriculum similar to our NER strategy, progressing through datasets of increasing annotation quality. Training sequentially progresses through four autonomous stages: bronze, silver, silver 2025, and finally gold datasets.

In addition to the funnelling, we implement an explicit mathematical weighting during the loss calculation to reflect corpora reliability. Via a custom `WeightedSequenceClassificationTrainer`, modifying the cross-entropy function, the bronze dataset contributes to the loss scaled by a weight of 0.75, the silver datasets by 1.0, and the gold dataset is amplified with a weight of 1.25, following the findings of Andersen et al. [14].

For each quality tier, the checkpoint is initialised from the most successful model (evaluated by F1 score) of the preceding tier. The model underwent 3 epochs per stage using a learning rate of $2 \times 10^{-5}$ and a weight decay of 0.01. This ensures that final parameter updates prioritise the most reliable data representations while earlier stages absorb broad domain variance.

#### 4.6.4. Evaluation: Tracking metrics and sequence classification

To track improvements dynamically during training, we integrate an evaluation routine triggered every epoch using the development dataset. It computes macro-averaged metrics, including Precision, Recall, and F1-score, alongside universal Accuracy. A linear sequence classification head translates the final hidden states into logit predictions across the constructed relation vocabulary. Finally, an early-stopping and saving mechanism retains the top-performing checkpoint utilising the highest macro F1-score as the deciding metric.

#### 4.6.5. Inference: Sentence-level evaluation with contextual boundaries

For inference, the system isolates entity pairs across individual sentences by utilising the Natural Language Toolkit (NLTK) [20] parser to accurately construct sentence boundaries. Within these sentence limits, all combinations of extracted entities implicitly generate ordered subject and object pairs. Local character offsets are inserted into sentence-relative coordinates, correctly injecting the `[E1]` and `[E2]` markers surrounding the selected entities' bounds before inference sequences are truncated. The underlying classification head processes each marker-enriched sentence pairing, converting output

via softmax into a predicted predicate, while filtering out any permutations designated as NO_RELATION. Finally, contextual entity data is mapped precisely back to the target format.

# 5. Experiments and Results

In this section, the experimental setup, pipeline considerations, and task results are presented. We begin by outlining the hardware environment, followed by Section 5.1, which presents the results of the baseline data quantity versus data quality experiment. Section 5.2 details the evaluation of the NER models and analyses the impacts of class imbalance on precision and recall. In Section 5.3, we present a systematic optimisation study for NERD, which includes exploring similarity measures, dataset dictionary quality, and entity-to-context weight distributions. Sections 5.4 and 5.5 analyse the performance of the M-RE and C-RE tasks, respectively, highlighting the strengths of domain-specific architectures and balanced datasets. Section 5.6 presents the effects of the waterfall pipeline error propagation. Finally, the final leaderboard results are presented in Section 5.7.

All experiments and training were run on the AAU AI LAB, which has the following specifications:

- 8 NVIDIA L4 GPUs, each with 24GB of RAM.
- 2 AMD EPYC 7543 32-Core processors.

## 5.1. Baseline Evaluation: Data Quantity vs. Quality Results

As mentioned in Section 4.1, we conduct experiments using the `bert-base-uncased` [15] encoder. In this setup, we use a token classification layer as a classification head to evaluate the embeddings produced for each token. These experiments were done to quantify the trade-off between annotation quality and volume on NER. Table 2 presents the results, where the Bronze collection ($n = 2972$) achieved the strongest validation performance, with a mean final validation F1 of $\mathbf{0.7643}$. The Silver collection ($n = 811$) reached $\mathbf{0.6497}$, slightly above the Gold collection ($n = 639$), which reached $\mathbf{0.6445}$. Silver 2025 ($n = 499$) obtained the lowest performance at $\mathbf{0.5935}$.

**Table 2**
NER baseline over five runs per dataset. Mean and sample standard deviation are reported for final validation F1; best-epoch F1 is included for completeness.

| Dataset | Size ($n$) | Final F1 | Best F1 | Final Rank |
|---|---|---|---|---|
| Bronze | 2972 | 0.7643 | 0.7658 | 1 |
| Silver | 811 | 0.6497 | 0.6530 | 2 |
| Gold | 639 | 0.6445 | 0.6445 | 3 |
| Silver 2025 | 499 | 0.5935 | 0.5935 | 4 |

## 5.2. NER Experiments

Once all the models were trained and predictions generated, an evaluation script was run to measure the performance of each model. Duplicate entries were removed, and an exact match approach was used in this and all other tasks.

### FFT vs QLoRA

As an alternative approach to the Full Fine-Tuning for the NER task, we trained a large generalist model using QLoRA[13], a PEFT method. We selected `Meta-Llama-3.1-8B` [21] due to its strong contextual understanding and ability to generalise across diverse language patterns. Operating under hardware constraints made QLoRA an ideal solution, allowing us to compare the performance of this massive model against a fully fine-tuned, specialised model less than a third of its size.

**Table 3**
NER Average scores, **Funnel BiomedBERT refers to Funnelled BiomedNLP-BiomedBERT-large-uncased-abstract with a token-classification header and funnel GLiNER refers to Ihor/gliner-biomed-large-v1.0*

| Average Type | Macro-P | Macro-R | Macro-F1 | Micro-P | Micro-R | Micro-F1 |
|---|---|---|---|---|---|---|
| Baseline | 0.7114 | **0.7480** | 0.7267 | 0.7782 | **0.8221** | **0.7996** |
| Llama3.1 8B QLoRA | 0.5825 | 0.7222 | 0.6449 | 0.4193 | 0.3854 | 0.4016 |
| *Funnel BiomedBERT | 0.6175 | 0.6576 | 0.6306 | 0.6841 | 0.7438 | 0.7127 |
| *Funnel GLiNER | **0.7424** | 0.7468 | **0.7369** | **0.7797** | 0.8033 | 0.7913 |

As shown in Table 3, Micro scores are higher than Macro scores for almost all models, which is standard behaviour when a dataset exhibits class imbalance. However, Llama3.1 8B QLoRA presents a notable anomaly. Its Macro-F1 (0.6449) is significantly higher than its Micro-F1 (0.4016). This indicates that while the Llama model struggles significantly with the most frequent, dominant classes in the dataset, it maintains moderate performance on the rare or minor classes. Additionally, the *Funnel BiomedBERT variant represents a step backwards across all metrics compared to both the Baseline and its GLiNER counterpart.

When comparing the top performers, Funnel GLiNER achieves higher Macro-P and Micro-P than the Baseline, yielding fewer false positives and demonstrating higher accuracy on its positive predictions. Conversely, the Baseline's higher recall indicates better overall sensitivity and coverage. Consequently, the F1 scores present a clear trade-off rather than a definitive superior model. The Baseline achieves a higher Micro-F1, making it statistically the most reliable for frequent classes; however, Funnel GLiNER secures a higher Macro-F1, proving it to be the more adequate model for predicting rare and minor classes.

## 5.3. A Frustratingly Easy Approach for NERD

We conducted an experimental study to optimise the NERD pipeline's performance. In doing so, we identified three primary factors impacting the prediction accuracy.

- The weights of the hybrid scoring.
- The quality of the data upon which the dictionary was built.
- The algorithm/method used for the scoring of the similarity.

To determine the optimal configuration, we systematically evaluated these factors. Our final NERD pipeline was configured utilising cosine similarity, the Gold/Silver/Silver2025 dictionary and an entity/context weighted scoring system of 0.85/0.15. The sections below present the top-1 precision achieved during these optimisation phases, and how changing the factors impact the precision of the configuration. The optimisation results were tested and evaluated using the *dev.json* evaluation dataset, as it worked as the optimal input, compared to a predicted input.

### Similarity Metrics

For similarity metrics, we experimented with four different measures: cosine similarity, Euclidean distance, Manhattan distance, and dot product. However, as can be observed in Table 4, there is no major difference among three of the four tested methods, all scoring 0.727.

**Table 4**
NERD Testing of methods and their results on the **dev.json** evaluation dataset

| Methods | Top-1 precision |
|---|---|
| Cosine similarity | **0.727** |
| Euclidean distance | **0.727** |
| Manhattan distance | **0.727** |
| Dot Product | 0.721 |

### Data quality

Similar to our initial funnel-training decision of utilising bronze as a "quick-learn", we wanted to test which datasets gave the best dictionary for predictions. During testing, we created the dictionary for all datasets based on the intuition that more data provides more context. However, we found that the bronze set introduced too much noise into the dictionaries, regardless of how we combined the datasets used.

This finding led to a prioritisation of data quality over data quantity in the entity linking task. As seen in table 5, expanding the dictionaries' volume does not yield a better result. While the combination of Gold, Silver and Silver2025 achieved our Top-1 precision of 0.727, introducing the bronze dataset dropped this precision to 0.689. Removing the quality features of Silver and Silver2025 from the dictionary yielded our lowest recorded score, 0.669, despite the dictionary being of high quantity.

This suggests that while the bronze dataset is high in volume, the entity-to-concept linking is not up to the quality of the human-annotated datasets. Furthermore, we suspect that due to the volume of the bronze dataset, the boundaries between concepts might have been diluted, making it harder for similarity measures to separate related entities and concepts.

**Table 5**
NERD Testing of dataset dictionaries and their results

| Datasets used | Top-1 precision |
|---|---|
| Gold and Bronze | 0.669 |
| Gold, Silver, Silver2025 | **0.727** |
| Gold and Silver | 0.701 |
| Gold, Silver, Silver2025, Bronze | 0.689 |
| Gold and Silver2025 | 0.681 |
| Silver and Silver2025 | 0.697 |

### Weighted scoring

When testing weight distributions, we observed that a higher context weight led to the wrong entity text spans being retrieved, which led to the wrong concept being returned. However, we knew from our initial testing that returning just the entity with no context led to the wrong concepts being returned, as they changed depending on the context of the sentence. Therefore, we separated the entity from the text span, ensuring we would always return the correct entity while using the sentence as an enforcing factor for context.

To determine the optimal balance between these two weights, we evaluated various weight distributions, which are presented in Table 6. A weight skewed heavily towards the entity performed better, with a 0.85/0.15 mix scoring 0.727 and a 0.50/0.50 mix scoring 0.688. Our results suggest that even though the context is important, the semantic noise of the context can overshadow the entity's importance. We suspect this is due to the specialised semantics contained within PubMed abstracts and the connections made between those semantics made by SapBERT. However, we cannot completely ignore the context, as it is an important part of differentiating between entities with multiple concepts, which was initially seen in our first non-weighted experiments.

**Table 6**
NERD Testing of weights and their results

| Weights (Entity/Context) | Top-1 precision |
|---|---|
| 0.85/0.15 | **0.727** |
| 0.75/0.25 | 0.715 |
| 0.50/0.50 | 0.688 |

### Final results

Table 7 presents the final evaluation of our dictionary SapBERT-based approach against the baseline, with our approach utilising the predictions passed from the NER task. With our SapBERT-based approach showing a relative overall improvement of ≈38-43% across all scores despite having a similar NER baseline from subtask 6.1.1.

**Table 7**
NERD scores *Dictionary SapBERT refers to SapBERT-from-PubMedBERT-fulltext using our Dictionary

| Average Type | Macro-P | Macro-R | Macro-F1 | Micro-P | Micro-R | Micro-F1 |
|---|---|---|---|---|---|---|
| Baseline | 0.3820 | 0.4045 | 0.3916 | 0.4281 | 0.4522 | 0.4398 |
| *Dictionary SapBERT | **0.5527** | **0.571** | **0.5572** | **0.5998** | **0.6176** | **0.6086** |

## 5.4. M-RE

To tackle the M-RE task, `BiomedNLP-BiomedBERT-large-uncased-abstract`, as described in Section 4.6, was trained on all datasets, along with the corresponding negative-relation dataset. This dataset has a 1:3 ratio, with 3 negative relationships for every positive one. A funnelled strategy was followed in this scenario as well, allowing for a quick pickup of how relationships look for later, heavily leaning on the ones we know are right in the gold dataset. This approach achieved an F1-micro score of 0.4016, with more fine-grained results in the table below:

In Table 8, we can observe a difference between Micro and Macro averages. Micro-F1 (0.4016) is significantly higher than Macro-F1 (0.3278). This usually indicates a class imbalance. The model performs better on the "majority" classes (those with more samples) and poorly on the "minority" classes. Since the Macro average treats every class equally, the poor performance on rare classes is dragging that score down.

We can also observe that Macro Precision (0.3665) is slightly higher than Macro Recall (0.3437), and Micro Precision (0.4193) is notably higher than Micro Recall (0.3854). This could indicate that the model is slightly "conservative." It is a bit better at ensuring that when it predicts a label, it is correct (Precision) than it is at finding all possible instances of that label (Recall). Overall, while the Funnel BiomedBERT model demonstrates a clear improvement over the baseline across nearly all metrics, further calibration of the model, e.g. adjusting the classification threshold or fine-tuning the negative-relation ratio, could help balance Precision and Recall, potentially recovering some of the instances currently being missed.

**Table 8**
M-RE Average scores, **Funnel BiomedBERT refers to Funneled BiomedNLP-BiomedBERT-large-uncased-abstract*

| Average Type | Macro-P | Macro-R | Macro-F1 | Micro-P | Micro-R | Micro-F1 |
|---|---|---|---|---|---|---|
| Baseline | 0.3660 | 0.2862 | 0.3003 | **0.4462** | 0.3453 | 0.3893 |
| *Funnel BiomedBERT | **0.3665** | **0.3437** | **0.3278** | 0.4193 | **0.3854** | **0.4016** |

## 5.5. C-RE

After generating an M-RE output, a similar approach to NERD was used to obtain concept relation extraction on the existing relationships.

**Table 9**
C-RE Average scores, **Dictionary SapBERT refers to SapBERT-from-PubMedBERT-fulltext using our Dictionary*

| Average Type | Macro-P | Macro-R | Macro-F1 | Micro-P | Micro-R | Micro-F1 |
|---|---|---|---|---|---|---|
| Baseline | 0.1009 | 0.1021 | 0.0966 | 0.1409 | 0.1292 | 0.1348 |
| *Dictionary SapBERT | **0.2250** | **0.2349** | **0.2146** | **0.2617** | **0.2632** | **0.2625** |

As we can see in Table 9, the Dictionary SapBERT model outperforms the Baseline across the board. Depending on the metric, you are seeing performance improvements of 85% to 130%. Integrating the domain-specific dictionary with SapBERT clearly provided a critical boost to the model's understanding.

Macro-Recall saw the highest leap overall (≈130%), meaning the implementation is better at finding relevant instances across all classes without ignoring the rarer ones. These results highlight the substantial benefit of incorporating domain-specific resources into the C-RE pipeline. Unlike the M-RE task where class imbalance visibly hindered the recognition of minority classes, the Dictionary SapBERT approach appears to mitigate this issue, successfully capturing rarer concept relations using the vocabulary. While this represents a definitive leap forward from the baseline, the absolute F1 metrics indicate that there is still room for optimisation. Future improvements could focus on further expanding or refining the dictionary entries, or enhancing the quality of the preceding M-RE outputs, as cascading errors from the initial mention extraction inevitably bottleneck the concept-level performance.

## 5.6. Waterfall-effect

Table 10 illustrates the compounding effect of error propagation within the waterfall pipeline. While the initial NER subtask shows identical performance across both conditions, the gap between cascaded and perfect inputs widens significantly with each stage. For instance, the NERD subtask experiences a moderate micro-F1 degradation of approximately 0.12, but by the final C-RE stage, this disparity expands to over 0.28. Ultimately, the downstream C-RE task achieves less than half the performance under cascaded conditions (0.2625) compared to perfect inputs (0.5427), starkly highlighting how early-stage inaccuracies magnify and severely degrade the pipeline's overall efficacy.

**Table 10**
Cascaded error micro-F1 compared to perfect input micro-F1.

| Subtask | Cascaded Input | Perfect Input | Difference |
|---|---|---|---|
| 6.1.1 NER | 0.7913 | 0.7913 | 0.0000 |
| 6.1.2 NERD | 0.6086 | 0.7267 | 0.1181 |
| 6.2.1 M-RE | 0.4016 | 0.5531 | 0.1515 |
| 6.2.2 C-RE | 0.2625 | 0.5427 | 0.2802 |

## 5.7. Final Leaderboads

Table 11 illustrates the final leaderboards for each of the subtasks among the top teams in the CLEF 2026 challenge. Our proposed information extraction pipeline (GetGut@AAU) achieved third place in both relation extraction subtasks (M-RE and C-RE).

In the M-RE and C-RE tasks, the Micro F1 score differences between second place and our results are 0.0169 and 0.0432, respectively. In the NER and NERD subtasks, our approach was competitive, trailing the third-place teams by margins of only 0.0124 and 0.0119. This corresponds to a difference in performance of just 1.55% and 2.16%, respectively. Furthermore, our proposed solution consistently

**Table 11**
CLEF 2026 General leaderboard. All scores report Micro F1.

| | NER | | NERD | | M-RE | | C-RE | |
|---|---|---|---|---|---|---|---|---|
| **Rank** | **Team** | **F1** | **Team** | **F1** | **Team** | **F1** | **Team** | **F1** |
| 1 | YesYoung | 0.8420 | YesYoung | 0.6169 | YesYoung | 0.4525 | YesYoung | 0.2632 |
| 2 | Graphwise | 0.8385 | Graphwise | 0.6053 | SMTE | 0.4223 | Graphwise | 0.2452 |
| 3 | TEXA | 0.8138 | TUGW | 0.5636 | GetGut@AAU | 0.4054 | GetGut@AAU | 0.2020 |
| – | GetGut@AAU | 0.8014 | GetGut@AAU | 0.5517 | GetGut@AAU | 0.4054 | GetGut@AAU | 0.2020 |
| – | Baseline | 0.7996 | Baseline | 0.4398 | Baseline | 0.3886 | Baseline | 0.1345 |

outperformed the official baseline across all four subtasks, demonstrating notable improvements of 0.1119 in the NERD subtask and 0.0675 in the C-RE subtask.

## 6. Conclusions and future work

This paper presents the GetGut@AAU pipeline, a multi-stage information extraction architecture developed for the GutBrainIE Task at the BioASQ Lab for CLEF 2026. To overcome the significant disparities in annotation density and supervision quality across the provided collections, we introduce a Weighted Funnel Fine-Tuning methodology for our NER and M-RE models. This staged training strategy proved effective. By initially training on high-volume, weakly supervised data and progressively narrowing the focus to low-volume, high-quality expert annotations, the models successfully acquire a broader biomedical vocabulary while relying on recency bias to calibrate precise entity and relation boundaries.

To prevent the M-RE model from over-predicting relationships, we implement a negative sampling mechanism. Using `NO_RELATION` instances, we enforce a controlled negative-to-positive ratio, successfully mitigating positive-class bias and improving extraction fidelity.

For the NERD subtask, prioritising data quality over data quantity is essential. We deliberately excluded the distantly supervised Bronze collection from our reference dictionary because its noise actively lowered our top-1 precision scores. Furthermore, our experimental results show the efficacy of a hybrid scoring approach. We found that while weighting the entity significantly higher than its surrounding context (0.85/0.15) yielded the best predictive accuracy, preserving the sentence context remains a necessary factor for differentiating ambiguous entities.

Finally, our study provides insights into the trade-offs of fine-tuning regimes under hardware constraints. When comparing FFT against PEFT, our results suggest that fully fine-tuning a smaller, domain-specific model fundamentally outperforms applying QLoRA to a massive, generalist LLM, particularly when addressing dataset class imbalances. Overall, our pipeline achieved 3rd place in both the M-RE and C-RE subtasks, alongside 6th place in NER and 5th place in NERD, beating the GutBrainIE Baseline in all subtasks.

Despite these encouraging results, our implementation presents several limitations that motivate future work. First, the class imbalance, particularly the long-tail distribution of rare entity categories, remains a persistent challenge that limited our Macro-F1 performance. Future iterations must explore advanced re-weighting or data augmentation strategies to better capture minority classes. Second, to mitigate positive-class bias in relation extraction, we employed a fixed 1:3 negative-to-positive sampling ratio to establish non-relation baselines; systematically experimenting with different negative sampling ratios could further optimise our model's boundary detection capabilities. Finally, our NERD pipeline relied specifically on SapBERT for entity-to-concept linking. Investigating how different biomedical encoders affect semantic representations and downstream disambiguation accuracy will be a key focus in refining our hybrid scoring methodology.

We identified two main limitations that can lead to improvements to our pipeline.

Firstly, during the FFT of our RE model, we went with a 1:3 ratio of negative relations; this was to teach the system how a relation does not look; however, due to time constraints, we didn't manage to experiment with further ratios of negative relations. We believe there is value in doing negative relations, and teaching a model when it is wrong, especially in tasks with a large amount of comparisons. There is an argument for testing different ratios, to comparison tasks in general machine learning, not limited to just the GutBrain axis or Relation Extraction.

Secondly, we identified a a substantial disparity in the number of classes In some cases we saw DDF making up 38% of all entities, while others such as Gene or Food only made up 2% of all entities. We often see that our FFT models are performing worse in the Macro-F1 score compared to the Micro-F1 score. It would be interesting to see how the models would perform if not only the datasets were weighted and considered for quality, but the individual labels were also weighted in a way to make up for the disparity.

## Acknowledgments

We are grateful for the support of CLAAUDIA through AI-Lab, which provided the computational infrastructure for conducting the experiments in this study.

This work is partially supported by the HEREDITARY Project, as a part of the European Union's Horizon Europe research and innovation programme under grant agreement No GA 101137074.

## Declaration on Generative AI

During the preparation of this work, we used Gemini-3.1 and Grammarly to perform grammar and spelling checks. After using these tools/services, we reviewed and edited the content as needed and take full responsibility for the publication's content.

## References

[1] C. Chen, G.-q. Wang, D.-d. Li, F. Zhang, Microbiota–gut–brain axis in neurodegenerative diseases: molecular mechanisms and therapeutic targets, Molecular Biomedicine 6 (2025) 64. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC12436269/. doi:10.1186/s43556-025-00307-1.

[2] A. Jain, S. Madkan, P. Patil, The Role of Gut Microbiota in Neurodegenerative Diseases: Current Insights and Therapeutic Implications, Cureus 15 (2023) e47861. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10680305/. doi:10.7759/cureus.47861.

[3] S. K. Chakrabarti, D. Chattopadhyay, The Gut–brain–immune Triad in Neurodegeneration: An Integrated Perspective, Journal of Translational Gastroenterology 3 (2025) 136–154. URL: https://www.xiahepublishing.com/m/2994-8754/JTG-2025-00027. doi:10.14218/JTG.2025.00027.

[4] M. Martinelli, V. Bonato, G. M. Di Nunzio, G. Faggioli, N. Ferro, O. Irrera, B. Kantz, S. Marchesin, L. Menotti, S. Merlo, R. Michelotto, G. Tussardi, F. Vezzani, P. Waldert, G. Silvello, Overview of GutBrainIE@CLEF 2026: Gut-Brain Interplay Information Extraction, in: E. S. Salido, A. Barrón-Cedeño, A. G. S. de Herrera, S. MacAvaney, J. M. Struß (Eds.), CLEF 2026 Working Notes, 2026.

[5] A. Nentidis, G. Katsimpras, A. Krithara, M. Krallinger, M. Rodríguez-Ortega, E. Rodriguez-López, N. Loukachevitch, I. Rozhkov, 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 2026: The fourteenth BioASQ Challenge on Large-Scale Biomedical Semantic Indexing and Question Answering, Lecture Notes in Computer Science, Springer, 2026.

[6] R. Tinn, H. Cheng, Y. Gu, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Fine-tuning large neural language models for biomedical natural language processing, Patterns 4 (2023) 100729. URL: https://doi.org/10.1016/j.patter.2023.100729. doi:10.1016/J.PATTER.2023.100729.

[7] Z. Zhong, D. Chen, A Frustratingly Easy Approach for Entity and Relation Extraction, in: K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, Y. Zhou (Eds.), Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Online, 2021, pp. 50–61. URL: https://aclanthology.org/2021.naacl-main.5/. doi:10.18653/v1/2021.naacl-main.5.

[8] P. Chen, M. Zhang, X. Yu, S. Li, Named entity recognition of Chinese electronic medical records based on a hybrid neural network and medical MC-BERT, BMC medical informatics and decision making 22 (2022) 315. doi:10.1186/s12911-022-02059-2.

[9] F. Borchert, I. Llorca, M.-P. Schapranow, Improving biomedical entity linking for complex entity mentions with LLM-based text simplification, Database 2024 (2024) baae067. URL: https://academic.oup.com/database/article/doi/10.1093/database/baae067/7721591. doi:10.1093/database/baae067.

[10] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, LoRA: Low-Rank Adaptation of Large Language Models, in: The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022, OpenReview.net, 2022. URL: https://openreview.net/forum?id=nZeVKeeFYf9.

[11] S. C. Sreenivas, S. Chowdhury, M. Masum, Enhancing Clinical Named Entity Recognition via Fine-Tuned BERT and Dictionary-Infused Retrieval-Augmented Generation, Electronics 14 (2025) 3676. URL: https://www.mdpi.com/2079-9292/14/18/3676. doi:10.3390/electronics14183676.

[12] C. Christophe, P. Kanithi, P. Munjal, T. Raha, N. Hayat, R. Rajan, A. A. Mahrooqi, A. Gupta, M. U. Salman, M. A. Pimentel, S. Khan, B. B. Amor, Med42 - evaluating fine-tuning strategies for medical LLMs: Full-parameter vs. parameter-efficient approaches, in: AAAI 2024 Spring Symposium on Clinical Foundation Models, 2024. URL: https://openreview.net/forum?id=oulcuR8Aub.

[13] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, QLoRA: Efficient Finetuning of Quantized LLMs, in: A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine (Eds.), Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023, 2023. URL: http://papers.nips.cc/paper_files/paper/2023/hash/1feb87871436031bdc0f2beaa62a049b-Abstract-Conference.html.

[14] L. R. Andersen, M. H. Dolmer, M. I. Gardshodn, J. M. Rodriguez, D. Dell'Aglio, Trusting gut instincts: Transformer-based extraction of structured data from gut-brain axis publications, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum, CLEF 2025, Madrid, Spain, 9-12 September 2025, volume 4038 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2025, pp. 99–119. URL: https://ceur-ws.org/Vol-4038/paper_6.pdf.

[15] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: J. Burstein, C. Doran, T. Solorio (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. URL: https://aclanthology.org/N19-1423/. doi:10.18653/v1/N19-1423.

[16] A. Yazdani, I. Stepanov, D. Teodoro, Gliner-biomed: a suite of efficient models for open biomedical named entity recognition, Bioinform. 42 (2026). URL: https://doi.org/10.1093/bioinformatics/btag322. doi:10.1093/BIOINFORMATICS/BTAG322.

[17] U. Zaratiana, N. Tomeh, P. Holat, T. Charnois, GLiNER: Generalist Model for Named Entity Recognition using Bidirectional Transformer, in: Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), Association for Computational Linguistics, Mexico City, Mexico, 2024, pp. 5364–5376. URL: https://aclanthology.org/2024.naacl-long.300. doi:10.18653/v1/2024.naacl-long.300.

[18] F. Liu, E. Shareghi, Z. Meng, M. Basaldella, N. Collier, Self-alignment pretraining for biomedical entity representations, in: Proceedings of the 2021 Conference of the North American Chapter of

the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Online, 2021, pp. 4228–4238. URL: https://www.aclweb.org/anthology/2021.naacl-main.334.

[19] sentence-transformers (Sentence Transformers), 2026. URL: https://huggingface.co/sentence-transformers.

[20] S. Bird, E. Klein, E. Loper, Natural language processing with Python: analyzing text with the natural language toolkit, " O'Reilly Media, Inc.", 2009.

[21] meta-llama/Llama-3.1-8B · Hugging Face, 2024. URL: https://huggingface.co/meta-llama/Llama-3.1-8B.