MEDDOPLACE Task Guide

Short introduction to the sub-tasks, formats and submission

https://temu.bsc.es/meddoplace

<salvador.limalopez@bsc.es>

Task Overview

MEDDOPLACE is a shared task that challenges participant to detect, normalize and classify locations and related entities in a corpus of 1,000 medical documents in Spanish. It will be held as part of the IberLEF 2023 Workshop, celebrated within the SEPLN 2023 Conference. For more information about the task, links to the Gold Standard data, gazetteers and annotation guidelines, please check the official website at <https://temu.bsc.es/meddoplace>.

This document provides information for participants about the evaluation setting and the different sub-tasks, including details on input/output formats and general tips.

Evaluation Setting

MEDDOPLACE offers four different sub-tasks, summarized in the Table 1. Each of the first three sub-tasks focuses on a different aspect (entity recognition, toponym resolution/entity linking, entity classification), evaluating them individually. The fourth sub-task (end-to-end) puts together the previous three sub-tasks and evaluates them collectively.

The task's evaluation and submission will be done via CodaLab, available at <https://codalab.lisn.upsaclay.fr/competitions/13017>.

To allow the individual evaluation of normalization and classification systems, the evaluation will be divided in two phases (explained in Table 2 below). For Phase 1, only the test text files will be released. Once it's finished, the list of entities found in the test files will be published so that participants can create predictions for all entities' normalization and classification during Phase 2. The deadlines for each phase are on the task's website: <https://temu.bsc.es/meddoplace/schedule>.

Additionally, we've prepared an evaluation library so that you can evaluate your results on your own. The evaluation library is available on GitHub at <https://github.com/TeMU-BSC/meddoplace_scoring_script>.

***Table 1.* Overview of MEDDOPLACE sub-tasks**

Sub-task	Description	Main metric	Additional metrics
Sub-task 1: Named Entity Recognition	Locate relevant mentions in text and return their start/end position	Strict precision, recall, micro F-1	Overlapping precision, recall, micro F-1
Sub-task 2.1: Normalization to GeoNames (Toponym Resolution)	Normalize mentions of GPE_NOM and GEO_NOM to a GeoNames identifier	Accuracy	Accuracy@161km, Area Under the Curve (AUC), Mean and Median Error
Sub-task 2.2: Normalization to PlusCodes (POIs Toponym Resolution)	Normalize mentions of FAC_NOM to a PlusCodes identifier	Accuracy	Accuracy@161km, Area Under the Curve (AUC), Mean and Median Error
Sub-task 2.3: Normalization to SNOMED CT (Entity Linking)	Normalize remaining mentions to a SNOMED CT identifier	Accuracy	-
Sub-task 3: Entity Classification	Classify GPE, GEO and FAC mentions into four predefined classes	Micro-averaged accuracy	-
Sub-task 4: End-to-end	Detect, normalize and classify entities sequentially using one or multiple systems	Strict precision, recall, micro F-1	-

***Table 2.* Overview of MEDDOPLACE evaluation phases**

Sub-task	Sub-tasks
Phase 1	Sub-task 1 (NER) + Sub-task 4 (End-to-End)
Phase 2	Sub-tasks 2 (Normalization) + Sub-task 3 (Classification)

Sub-tasks Explanation

This section gives some details about the purpose of each task, metrics used, input and output format, as well as some tips.

As a general comment, the MEDDOPLACE corpus is released on three different formats (.ANN, which is used by the annotation tool brat, .JSON and .TSV). For simplicity, on this guide we will always refer to the .TSV version. All submissions must be made in .TSV format as well.

- *Sub-task 1: Named Entity Recognition*

This a classic NER task whose **aim** is to detect entities in text using the full-text documents as input.

The **input .TSV file** for this task is called "*meddoplace_tsv_train_subtask1*". Each row contains an individual entity with the following columns:

- *filename*: Name of the file associated with the annotated mention
- *ann_id*: Internal ID of the annotation set by brat on a document-basis. This is kept for traceability and can be safely ignored.
- *label*: Name of the tag the annotation belongs to.
- *start_span*: Position where the annotation starts in the text (in characters).
- end_span: Position where the annotation ends in the text (in characters).
- *text*: String of the annotation.

The **output .TSV file** sent for submission should have the same columns as the input .TSV file except for the *ann_id* field which can be ignored. Participants must provide predictions for the columns *label, start_span, end_span* and *text*. It is important that the file contains headers. All in all, these are the columns that should be included: *filename, label, start_span, end_span* and *text*.

The **test set to be used** for this task will be released for Phase 1 and contains simply the text files.

The **main evaluation metrics** for this subtask are exact-match, micro-averaged precision, recall and F-1 score.

We will also provide as **additional evaluation metrics** the overlapping-matches, micro-averaged precision, recall and F-1 score. This metric is a bit more permissive as it considers a match mentions that are overlap with the Gold Standard mentions.

As a general **tip/food for thought** for this task, keep in mind that there are 10 different labels and some of them might overlap with each other at times. Is it better to train separate systems for each tag or to train them all together? How would you tackle overlapping spans? Could the model work better if you group semantically-related entities for training?

- *Sub-task 2.1: Normalization to GeoNames*

This a geocoding/toponym disambiguation/geographical entity normalization task whose **aim** is to normalize a given list of GPE and GEO named places to GeoNames identifiers. GeoNames is an established database of geographical locations available on <https://www.geonames.org/>.

The **input .TSV file** for this task is called "*meddoplace_tsv_train_subtask2-1*". Each row contains an individual entity with the following columns:

- *filename*: Name of the file associated with the annotated mention
- *ann_id*: Internal ID of the annotation set by brat on a document-basis. This is kept for traceability and can be safely ignored.
- *label*: Name of the tag the annotation belongs to.
- *start_span*: Position where the annotation starts in the text (in characters).
- end_span: Position where the annotation ends in the text (in characters).
- *text*: String of the annotation.
- *normalization*: Assigned GeoNames ID for the annotation.
- *source*: Origin of the normalization (in this case, it will always be GN, i.e. GeoNames)

The **output .TSV file** sent for submission should have the same columns as the input .TSV file except for the *ann_id* and *source* field which can be ignored. Since most columns will already be given, participants must only predict the *normalization* column. It is important that the file contains headers. All in all, these are the columns that should be included: *filename, label, start_span, end_span*, *text*, *normalization* and *source* (in this case, it's always GN).

Participants are free to use any features extracted from the text document or any additional resources they may see fit (they must specify it in their system description though).

As a **reference for GeoNames**, participants may download a GeoNames dump from <https://download.geonames.org/export/dump/>. Specifically, you should download the file *allCountries.zip* which includes the complete database.

The **test set to be used** for this task will be released for Phase 2 and contains a list of all relevant entities and their position in the text.

The **main evaluation metric** for this subtask is accuracy, understood as the percentage of correctly normalized mentions out of the total of mentions to be normalized. We will refer to this metric as "accuracy (percentage correct)" to distinguish it from the coordinate-based Accuracy at 161 km presented below.

To provide a better evaluation of geocoding systems, we will use some **additional evaluation metrics** based on distance and coordinates. Namely, we will provide:

- Accuracy at 161 km (percentage of references resolved within an error distance of 161 kilometers)
- Area Under the Curve (AUC; this metric uses the log of the error distances to soften the effect of outlier errors).
- Mean Error Distance and Median Error Distance (both in kilometers).

If you are not familiar with these distance metrics and want to learn more, we recommend taking a look at the following paper:

> Gritta, M., Pilehvar, M.T. & Collier, N. A pragmatic guide to geoparsing evaluation. *Lang Resources & Evaluation* **54**, 683–712 (2020). https://doi.org/10.1007/s10579-019-09475-3 [pages 694-697]

Some pointers about these metrics:

- For A@161km, the higher the value, the better the score. The maximum score is 1.
- For AUC, the lower the value, the better the score. The lowest value is 0.
- The AUC is calculated as described in the paper above, with 20039 being 1/2 of Earth's circumference:

$$Area\ Under\ the\ Curve = \frac{\int_0^{dim(x)} \ln(x)dx}{dim(x) * ln(20039)}$$

To calculate these, we will obtain the distance between your proposed code and the Gold Standard code. Because of this, please make sure that the codes you provide are **valid GeoNames codes**. Since we need a set of coordinates to calculate them, the scorer will default any invalid codes to the Gold Standard coordinates's antipodes (i.e. the point on Earth opposite to it).

As a general **tip**, keep in mind that the GeoNames reference files give a lot of information about every entry, including coordinates, population, administrative division type, … You may able to integrate them in your system in some way (or not).

- *Sub-task 2.2: Normalization to PlusCodes*

This a geocoding/toponym disambiguation/geographical entity normalization task whose **aim** is to normalize a given list of FAC named places to PlusCodes codes. PlusCodes is a way of encoding coordinates based on latitude and longitude, and displayed as a string of numbers and letters. More information is available on <https://maps.google.com/pluscodes/>.

The **input .TSV file** for this task is called "*meddoplace_tsv_train_subtask2-2*". Each row contains an individual entity with the following columns:

- *filename*: Name of the file associated with the annotated mention

- *ann_id*: Internal ID of the annotation set by brat on a document-basis. This is kept for traceability and can be safely ignored.
- *label*: Name of the tag the annotation belongs to.
- *start_span*: Position where the annotation starts in the text (in characters).
- end_span: Position where the annotation ends in the text (in characters).
- *text*: String of the annotation.
- *normalization*: Assigned PlusCodes code for the annotation.
- *source*: Origin of the normalization (in this case, it will always be PC, i.e. PlusCodes)

The **output .TSV file** sent for submission should have the same columns as the input .TSV file except for the *ann_id* and *source* field which can be ignored. Since most columns will already be given, participants must only predict the *normalization* column. It is important that the file contains headers. All in all, these are the columns that should be included: *filename, label, start_span, end_span, text*, *normalization* and and *source* (in this case, it's always PC).

Participants are free to use any features extracted from the text document or any additional resources they may see fit (they must specify it in their system description though).

Since PlusCodes is just a way to encode coordinates, there is **no reference file**. However, you can use the OpenLocationCode library (available on GitHub at <https://github.com/google/open-location-code>) to encode and decode both PlusCodes and coordinates. A few important considerations here:

- You will need to use the OpenLocationCode library to check that your codes are valid.
- We will only accept full codes – no shortened versions.
- If a code is not valid and can't be converted back into coordinates by the OpenLocationCode library the scoring program will return a warning to tell you about it so that you can fix it. In addition, just as in the GeoNames task, you will be penalized in the distance metrics as we will calculate the

distance between the Gold Standard coordinates and their antipodes (i.e. the point on Earth opposite to it) instead of the incorrect code.

The **test set to be used** for this task will be released for Phase 2 and contains a list of all relevant entities and their position in the text.

The **main evaluation metric** for this subtask is also accuracy (percentage). Since the space represented by PlusCodes is quite small, we will only compare PlusCodes up to the plus sign. This means that we will use a slightly bigger reference area, being a bit more lenient in the evaluation.

As in the GeoNames task we will also use some **additional evaluation metrics** based on distance. Namely, we will provide Accuracy@161km, Area Under the Curve, Mean Error Distance and Median Error Distance.

Again, to learn more about these distance metrics, we recommend taking a look at the following paper:

> Gritta, M., Pilehvar, M.T. & Collier, N. A pragmatic guide to geoparsing evaluation. *Lang Resources & Evaluation* **54**, 683–712 (2020). https://doi.org/10.1007/s10579-019-09475-3 [pages 694-697]

As a general **tip**, keep in mind that, again, PlusCodes is just a way to encode coordinates. You will most likely need to convert the codes to coordinates to work with them, as well convert the predicted coordinates back to PlusCodes for submission. If you don't know where to start with this sub-task, it might be a good idea to take a look at open geocoding services such as Nominatim (<https://nominatim.org/>) or WikiLocation (<https://wikilocation.org/documentation/index.html>).

- *Sub-task 2.3: Normalization to SNOMED CT*

This a entity linking/normalization task whose **aim** is to normalize a given list of entities to SNOMED CT identifiers. SNOMED CT is a general purpose clinical terminology considered to be one of the most comprehensive in the world (more info on <https://www.snomed.org>).

The **input .TSV file** for this task is called "*meddoplace_tsv_train_subtask2-3*". Each row contains an individual entity with the following columns:

- *filename*: Name of the file associated with the annotated mention
- *ann_id*: Internal ID of the annotation set by brat on a document-basis. This is kept for traceability and can be safely ignored.
- *label*: Name of the tag the annotation belongs to.
- *start_span*: Position where the annotation starts in the text (in characters).
- end_span: Position where the annotation ends in the text (in characters).
- *text*: String of the annotation.
- *normalization*: Assigned SNOMED CT code for the annotation.
- *source*: Origin of the normalization (in this case, it will always be SCTID, i.e. SNOMED CT ID)

The **output .TSV file** sent for submission should have the same columns as the input .TSV file except for the *ann_id* and *source* field which can be ignored. Since most columns will already be given, participants must only predict the *normalization* column. It is important that the file contains headers. All in all, these are the columns that should be included: *filename, label, start_span, end_span*, *text*, *normalization* and *source* (in this case, it's always SCTID).

Participants are free to use any features extracted from the text document or any additional resources they may see fit (they must specify it in their system description though).

A **SNOMED CT reference file** is provided by us and available to download together with the Gold Standard data [https://doi.org/10.5281/zenodo.7707566].

The **test set to be used** for this task will be released for Phase 2 and contains a list of all relevant entities and their position in the text.

The **main evaluation metric** for this subtask is accuracy (percentage correct)..

- *Sub-task 3: Entity Classification*

This sub-task's **aim** involves further classifying location entities in text inside pre-defined classes of clinical relevance. The possible categories are LUGAR-NATAL (*birth place*), RESIDENCIA (*living place*), MOVIMIENTO (*movement*) and ATENCION (*healthcare attention*). Location entities that don't belong into any of these categories are classified as OTHER. For reference, these are the corpus' location tags: GPE_NOM, GPE_GEN, GEO_NOM, GEO_GEN, FAC_NOM and FAC_GEN.

The **input .TSV file** for this task is called "*meddoplace_tsv_train_subtask3*". Each row contains an individual entity with the following columns:

- *filename*: Name of the file associated with the annotated mention
- *ann_id*: Internal ID of the annotation set by brat on a document-basis. This is kept for traceability and can be safely ignored.
- *label*: Name of the tag the annotation belongs to.
- *start_span*: Position where the annotation starts in the text (in characters).
- end_span: Position where the annotation ends in the text (in characters).
- *text*: String of the annotation.
- *class*: Assigned class to every annotation.

The **output .TSV file** sent for submission should have the same columns as the input .TSV file except for the *ann_id* field which can be ignored. Since most columns will already be given, participants must only predict the *class* column. It is important that the file contains headers. All in all, these are the columns that should be included: *filename, label, start_span, end_span, text* and *class*.

Participants are free to use any features extracted from the text document or any additional resources they may see fit (they must specify it in their system description though).

The **main evaluation metric** for this subtask is micro-averaged accuracy (i.e. calculated for all mentions at once).

- *Sub-task 4: End-to-End*

This sub-task challenges participants to do **all three previous sub-tasks at once**. That is, they must detect entities in text, normalize them to the corresponding ontology and classify appropriate entities. The main difference with the previous sub-tasks is that the normalization and classification systems will not be evaluated on a vacuum, but instead depend on the mentions detected by the NER system (Task 1). This means that the scores will change and be closer to a real-world system application. Participants are free to use separate systems for each task or a real end-to-end system, as well as to reuse the systems used in the other sub-tasks.

The **input .TSV file** for this task is called "*meddoplace_tsv_train_complete*". Each row contains an individual entity with the following columns:

- *filename*: Name of the file associated with the annotated mention
- *ann_id*: Internal ID of the annotation set by brat on a document-basis. This is kept for traceability and can be safely ignored.
- *label*: Name of the tag the annotation belongs to.
- *start_span*: Position where the annotation starts in the text (in characters).
- end_span: Position where the annotation ends in the text (in characters).
- *text*: String of the annotation.
- *normalization*: Assigned code for the annotation.
- *source*: Origin of the normalization (can be either GN (GeoNames), PC (PlusCodes) or SCTID (SNOMED CT))
- *class*: Assigned class for the annotation; annotations that aren't from a location label have "None" in this field.

The **output .TSV file** sent for submission should have the same columns as the input .TSV file except for the *ann_id* field which can be ignored. It is important that the file contains headers. All in all, these are the columns that should be included: *filename, label, start_span, end_span, text*, *normalization*, *source* and *class*.

The **test set to be used** for this task will be released for Phase 1, which includes only the test text files. As a reminder:

- In normalization, GPE_NOM and GEO_NOM entities are normalized to GeoNames; FAC_NOM entities are normalized to PlusCodes; the rest are normalized to SNOMED CT.
- In classification, only location (GPE_NOM, GPE_GEN, GEO_NOM, GEO_GEN, FAC_NOM and FAC_GEN) entities must be classified.

Again, participants are free to use any features extracted from the text document or any additional resources they may see fit (they must specify it in their system description though).

As for the **metrics**, an F-score will be given for each individual step (detection, each of the three normalization types and classification), as well as an average of all scores.