CODI - A Conversation Disentanglement Microservice

Edoardo Riggio

Prof. Dr. Michele Lanza
Advisor
michele.lanza@usi.ch

Marco Raglianti
Co-Advisor
marco.raglianti@usi.ch

Motivation

In literature, conversation disentanglement – i.e., the ability to identify conversation given the contents and metadata of a series of messages – has been studied for years. Many different algorithms for conversation disambiguation are available. Their main problem is that they are challenging to set up and use. For this reason, we decided to build *CODI*, a RESTful microservice that is both simple to set up and easy to use.

Conversation Disentanglement

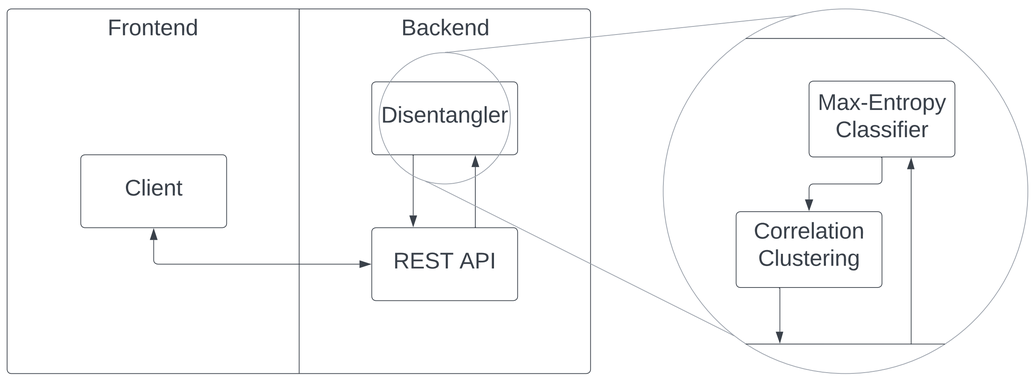
Conversation disentanglement is the task of clustering messages into a set of conversations. A human reader is only marginally capable of reconstructing the conversation flow in real-time. But, for a machine, it represents a highly complex task. Instant Messaging platforms can reach throughputs of several hundred messages per hour during active conversations. This means that multiple messages can arrive at almost the same time, resulting in interleaving messages about different simultaneous conversations.

T1	Chauncey	the human touch ?
T1	Gale	Chauncey what do you mean?
T2	Nestor	ok
T2	Nestor	i got it solved
T2	Nestor	thanks for the help guys
T2	Nestor	(if you ask how, i made a mini bash script)
T2	Nestor	goodbye.
T1	Chauncey	Paulita: I'm looking for the end to that sentence...

As shown in the figure above, the conversation between *Chauncey* and *Gale* is interrupted by *Nestor*, who is following up on a possible coding question he had sent previously. The first conversation is resumed as soon as *Nestor* finishes.

Architecture

CODI is a web server composed of both frontend and backend and divided into several modules. The fronted consists of the **client** module, while the backend includes both the **disentangling** and **REST API** modules.

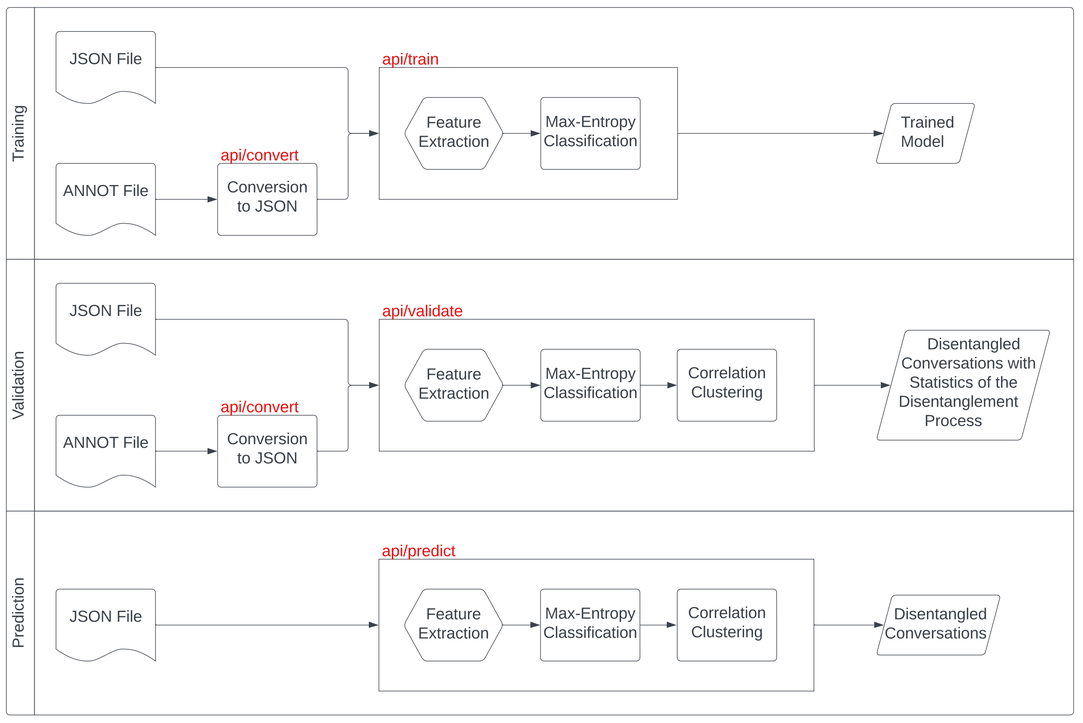


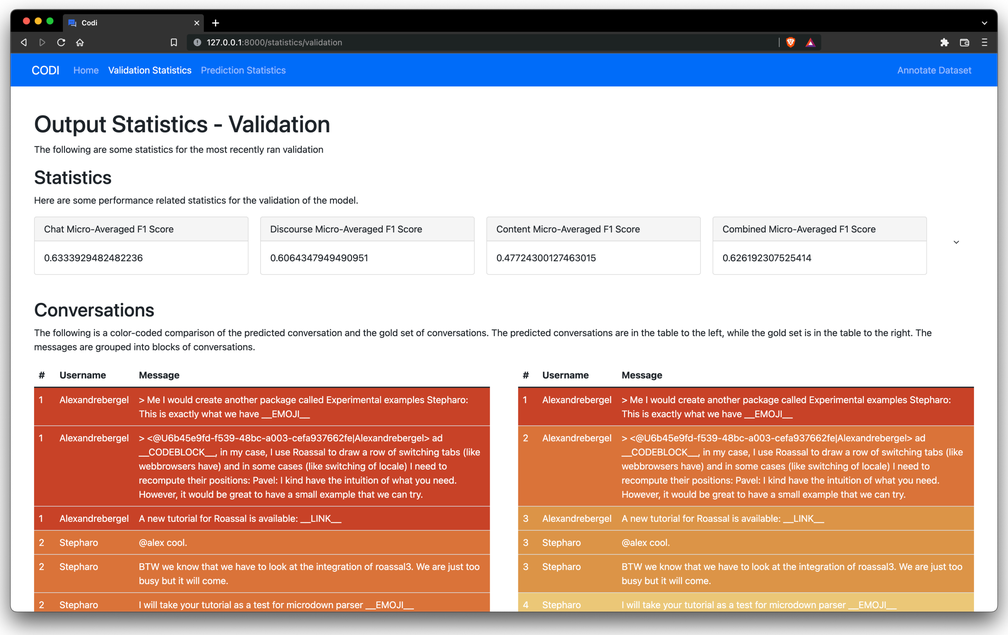
On the one hand, the **REST API** module is responsible for connecting the frontend with the **disentangling** module. On the other hand, the **disentangling** module provides all the logic and algorithms for pre- and post-processing the messages and the two-step algorithm containing both the max-entropy classifier and the correlation clustering algorithm.

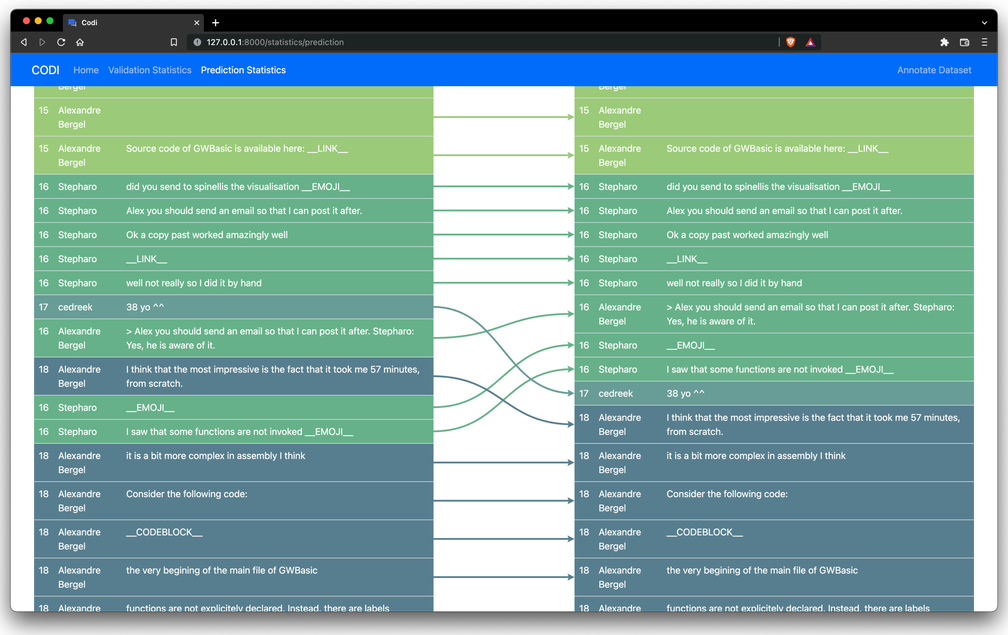
Data Flow

The operations that *CODI* can carry out are:

1. **Training:**
   This operation is used to train the disentangler's classification model.
2. **Validation:**
   This operation creates the conversation clusters and compares them to a gold dataset.
3. **Prediction:**
   This operation simply creates the conversation clusters.







Experiments

We carried out the first experiments utilizing annotated datasets provided by the original authors, which can be obtained from their GitHub repository. These were done to test the correctness of our implementation. More tests were subsequently carried out, this time using tailored annotated datasets to stress-test the algorithm and determine any potential limitations of such an implementation.

	Accuracy	Precision	Recall	F1-Score	Micro-Averaged F1-Score
python	60	86	62	72	78
clojure	67	94	68	79	88
pharo	68	68	77	79	63

The table above are the results produced by *CODI*. By running these tests, We've found that the two implementations – *CODI* and Chatterjee et al. – give comparable results.