

pyTLEX 1.0: A Python Library for Timeline Extraction

Students: Raul Garcia Breijo, Kevin Fontela, Jorge Segredo & Akul Singh

Mentor: Mark Finlayson, Mustafa Ocal, Florida International University

Instructor/Faculty: Dr. Masoud Sadjadi, Florida International University

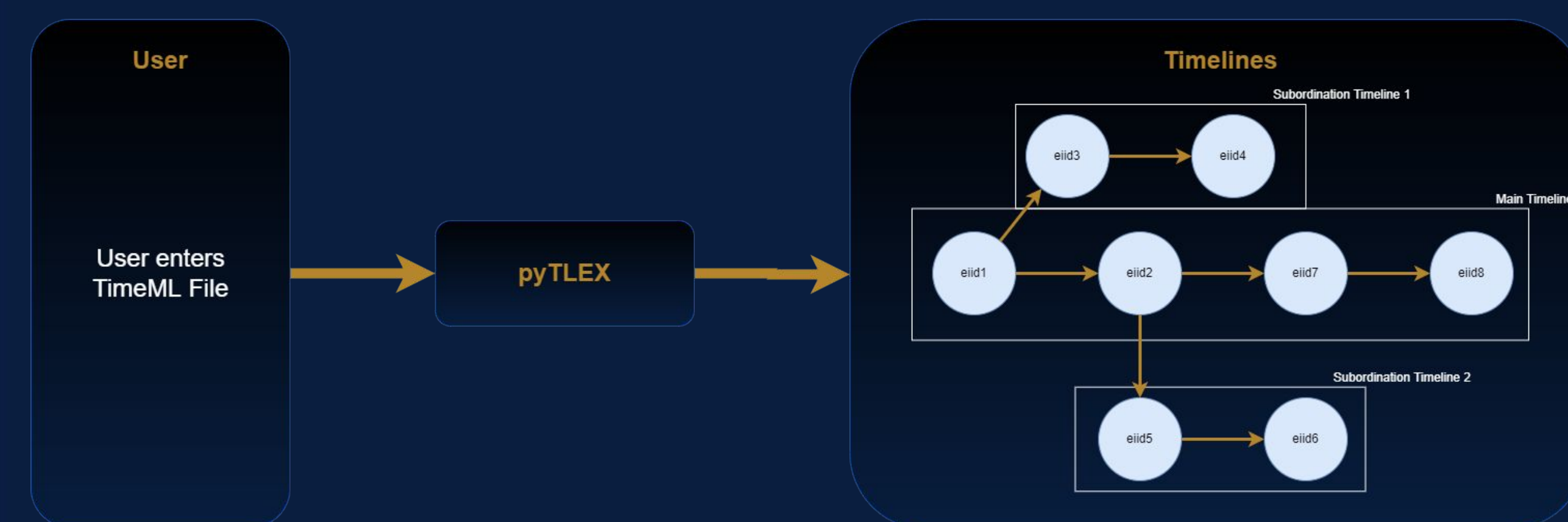
PROBLEM

Qualitative temporal graphs that are derived from annotations on text (TimeML annotations) explicitly reveal only the partial orderings of events and times. For many purposes, however, a total ordering (a timeline) is more useful. Unfortunately, timelines are rarely explicit in text, and usually cannot be extracted directly.

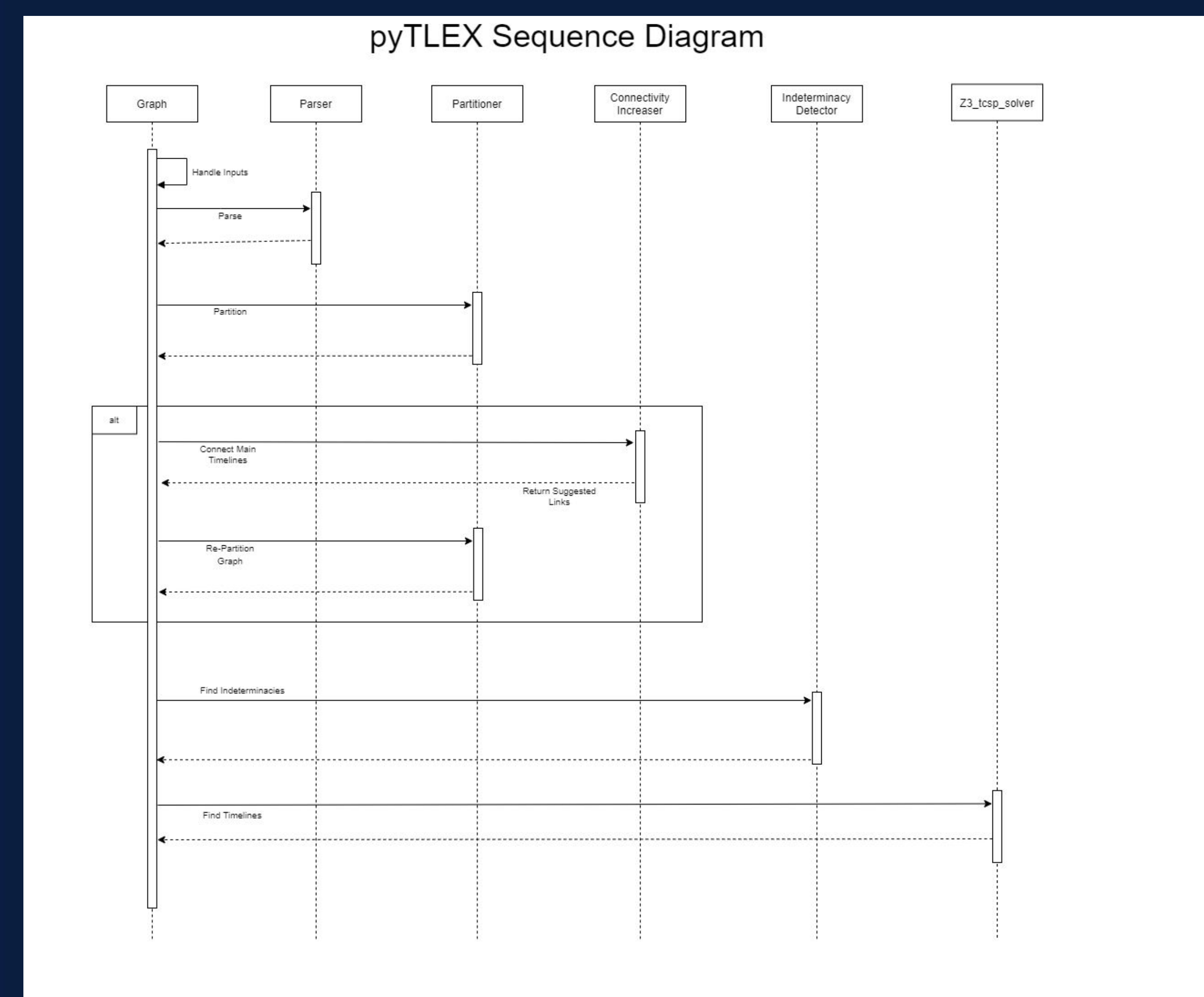
The TLEX algorithm specifies an exact, end-to-end solution, to the task of extracting multiple timelines from TimeML annotated texts. TLEX transforms TimeML annotations into a collection of main and subordinated timelines arranged into a trunk-and-branch style structure.

pyTLEX offered the solution proposed by the TLEX algorithm into an easy-to-use python library.

SYSTEM DESIGN



SEQUENCE DIAGRAM



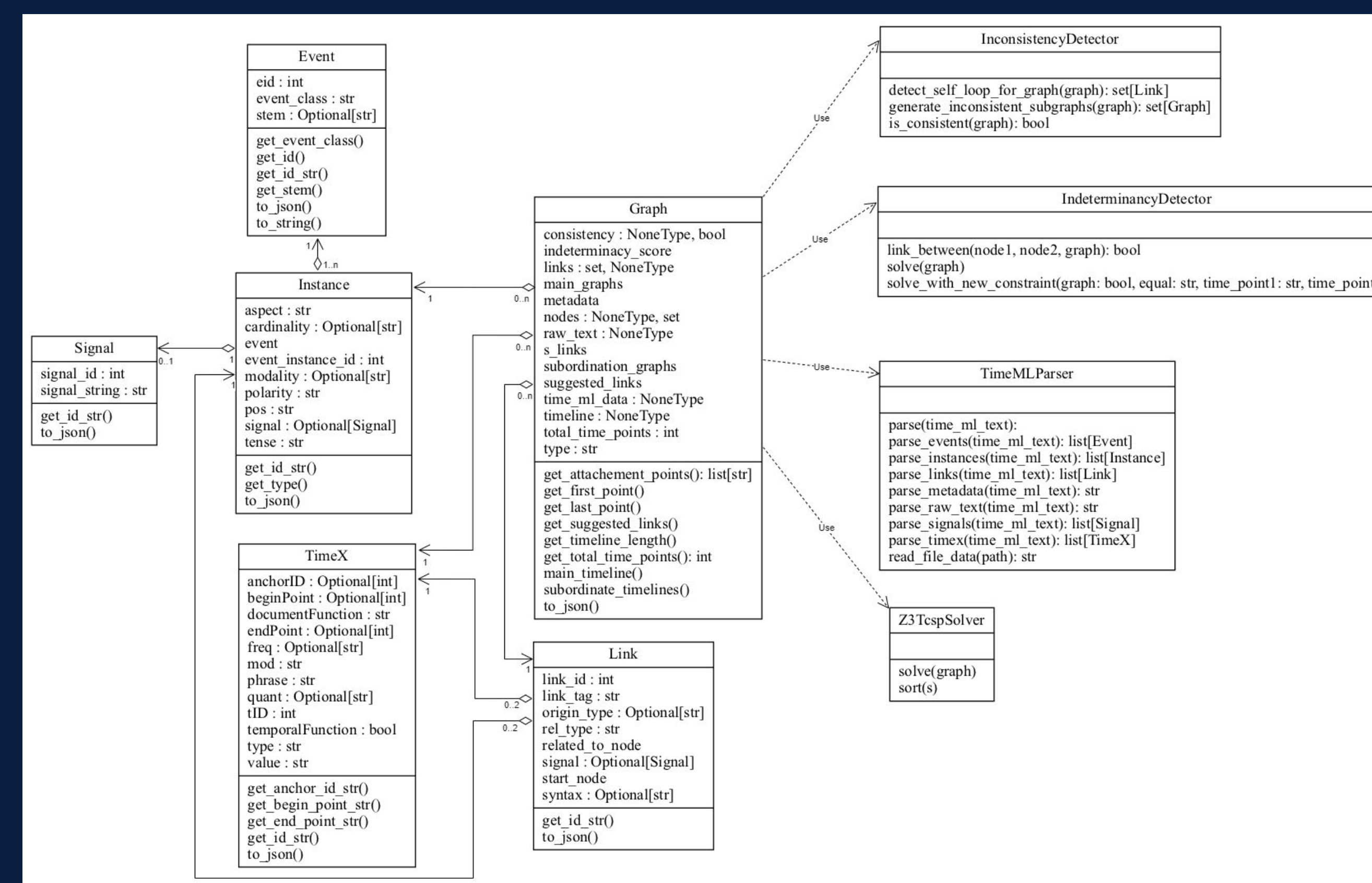
CURRENT SYSTEM

The work done for the jTLEX java library is now being converted into a Python library pyTLEX. The conversion from a Java implementation to a Python implementation has not been straightforward as many of the Java libraries used are not offered in Python.

Because of this and other differences between the two languages we have had to change our approach towards the implementation of the algorithm.

The current system has had five classes translated by the team including: Signal, Event, Instance, TimeX, and Link.

OBJECT DESIGN



VERIFICATION

Purpose	Method	Preconditions	Expectation	Result
(1) Check the python dataclasses' functionality	Create pytest unit tests that creates multiple objects with all possible options	Valid Signal, Event, Instance, Timex, and Link Inputs	All Objects Created Correctly	All Objects Created, Successful
(2) Check the python dataclasses' invalid input handling	Create pytest unit tests that tests all possible invalid inputs for all dataclasses	Invalid Signal, Event, Instance, Timex, and Link Inputs	All Objects Failed and Threw Exceptions	All Objects Failed Creation, Successful
(3) Check the json output of the python dataclasses	Conduct manual testing to compare the expected json output from jTLEX to the output from pyTLEX	Valid Signal, Event, Instance, Timex, and Link Inputs	All JSON is Equal	All JSON Equal, Successful
(4) Show that the pyTLEX output matches jTLEX	Run both jTLEX and pyTLEX to receive json output	Valid TimeML File	Equivalent JSON output	Equivalent JSON Output Received

IMPLEMENTATION



The pyTLEX library was mainly constructed through the use of Python coded in PyCharm. We first assigned tasks and organized sprints through the use of Trello. From there, when working on the library, we utilized the Z3 library for constraint solving in order to formulate timelines and pytest for testing. Finally, we would then share our code through Subversion

SUMMARY

The pyTLEX library was created by translating the existing jTLEX classes with several added improvements and tested using pytest.

It utilizes a new way of parsing TimeML files in a way that is much more adaptable and able to convert the files into Python objects.

It also utilizes a new constraint solver, Z3, in order to find the best fit timeline.

REFERENCES

- Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11), 832-843.
- Pustejovsky, J., Hanks, P., Sauri, R., See, A., Gaizauskas, R., Setzer, A., ... & Lazo, M. (2003, March). The timebank corpus. In *Corpus linguistics* (Vol. 2003, p. 40).
- Sauri, R., Littman, J., Knippen, B., Gaizauskas, R., Setzer, A., & Pustejovsky, J. (2006). TimeML Annotation Guidelines Version 1.2. 1.

REQUIREMENTS

- Python 3.9
- pytest_cov >= 2.5.1
- pytest >= 3.4.2
- z3-solver >= 4.8.12.0

ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by the Cognac Laboratory and MITRE. We thank Mustafa Ocal and Dr. Mark Finlayson for their assistance and mentorship that we received throughout the Capstone 1 project.