Large Scale Annotation, Storage and Analysis of Digitised Sri Lankan Tamil Content

The project aims to build a platform capable of doing large scale content analysis of digitised Sri Lankan Tamil Texts. This is related to the field of semantic culturomics in which researchers data mine large digital archives to investigate cultural phenomena reflected in language and word usage. It is a form of computational lexicology that studies human behaviour and cultural trends through the quantitative analysis of digitised texts. The underlying data is from Noolaham Foundation, a Digital Archive and a Digital Library undertaking the critical work of documenting, digitally preserving and providing free and open access to knowledge bases and cultural heritage of Sri Lankan Tamil speaking communities. The archive contains digitised text from Sri Lankan newspapers, books, magazines, pamphlets etc from various sources totalling up to approximately 100,000+ documents. It also includes a web archive and born-digital data in text format which would be included in our pipeline.

Figure 1 illustrates the component model of the project consisting of a language pre-processing layer, language resource layer, processing resource layer and finally the knowledge engineering component. As first steps, the goal is to define and initiate multiple sub-projects to build an annotated corpus in text format as shown in the language pre-processing layer. Each and every project is described in detail below.

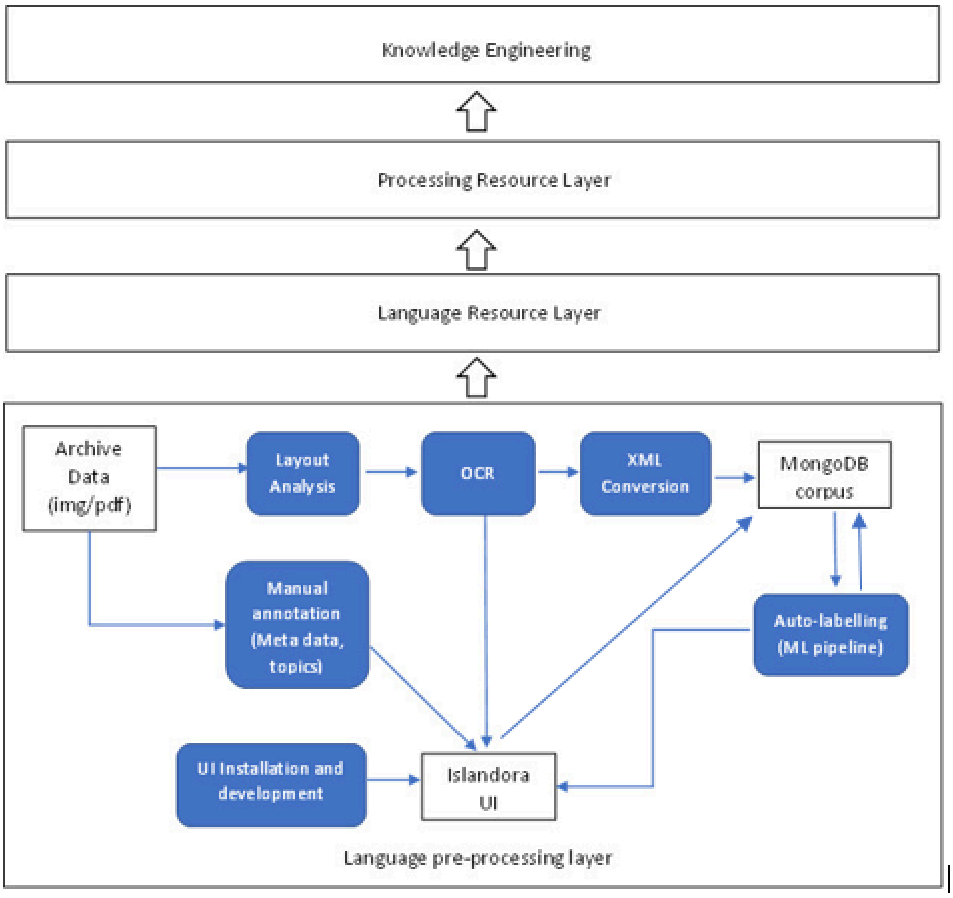
Knowledge Engineering
Processing Resource Layer
Language Resource Layer
Archive Data (img/pdf)
Layout Analysis
OCR
XML Conversion
MongoDB corpus
Manual annotation (Meta data, topics)
Auto-labelling (ML pipeline)
UI Installation and development
Islandora UI
Language pre-processing layer

Figure 1: Component model showing the different layers in the platform from language pre-processing to knowledge engineering

Section 1- Language Pre-processing Layer

User Interface development (Project 1)
In this project we would use Islandora, a collaborative open source framework to manage digitised assets and develop it to showcase digitised content from noolaham in image, pdf and text formats. The UI will also show metadata and keywords for each document from manual annotations done by annotators. We aim to do keyword labelling via an automated machine learning model for the whole document collection which would be later displayed in the UI.

Detailed Proposal :
https://docs.google.com/document/d/117NY-YDRz6nx5w6SxMFIB1wA5zaBGEDM/edit?usp=sharing&ouid=109406652623621046836&rtpof=true&sd=true

Mentors : Prashanth Sirinivasan
Partners :
Donors :

Metadata Creation (Project 2)
We require certain metadata to be attached to the documents displayed in the Islandora UI. eg: For a newspaper article the meta data would include, the name of the news outlet, date, title and author of the article. This project requires a few human annotators to manually add metadata to all the documents available from Noolaham in the Islandora platform. Additional annotations such as keywords or topics for the documents would also be manually annotated initially.

Detailed Proposal :
Mentors : Kopinath Thillainathan ; Thamilini Jothilingam ;
Partners :
Donors :

Layout Analysis (Project 3)
All documents available in the digital archive need to be converted to text format via OCR. For newspaper articles it's important to perform document layout analysis before the image is being sent to OCR. It is the process of identifying and categorising the regions of interest in the scanned image of a text document. A reading system requires the segmentation of text zones from non-textual ones and the arrangement in their correct reading order. We aim to use software such as CCS DocWorks or Azure Forms Recogniser for this purpose, or build a custom pipeline from cloud service providers.

Detailed Proposal :
Mentors : Vaheesan Selvarajah
Partners :

Donors :

Optical Character Recognition - OCR (Project 4)

This project would focus on converting all document images to text via an OCR system. The text would be displayed in Islandora UI along with all other available file formats. The text will also be stored in an open source document-based database system such as MongoDB.

Apart from newspapers for all other types of documents we can feed the scanned document (pdf) or image file directly into a Tamil OCR tool via an automated pipeline. We are considering tesseract-ocr (https://github.com/tesseract-ocr/tesseract) which is currently being maintained by Google and offers the most accurate results in current benchmarks. For newspapers once the Layout analysis is complete, the resulting chunks of images should be fed to the OCR tool.

Detailed Proposal :
Mentors : Saatviga Sudhahar ;
Partners :
Donors :

Document type Storage (Project 5)

A document-type storage database is required to store all document texts which have been extracted from the previous steps along with the manual/automatic annotations associated with a document. This database needs to be in direct sync with the User interface system that is being developed so that when manual annotations are added to a document via the UI, it is also updated in the backend in the relevant document object. Similarly when we generate annotations through the auto-labelling module, they have to be inserted into the relevant document as well. There are three ways in which data needs to be inserted/updated for the document records.

- Firstly all documents being displayed in the Islandora UI needs to have a corresponding row in the document db with the right identifier and any available manual annotations.
- When a document goes through the layout analysis, OCR process the output text would have to be added to the corresponding record in the database
- Annotations that become available via the auto-labelling module have to be updated in the corresponding record in the database

There needs to be bi-directional communication between the database and the UI so that information is updated in both ways. For example when new annotations become available for a given document, they have to be displayed in the UI in the relevant document along with the already available manual annotations. We are considering using MongoDB for this project but we still have to research and do a feasibility study for the integration between Islandora UI and MongoDB.

Detailed Proposal :

Mentors : Saatviga Sudhahar
Partners :
Donors :

XML Conversion (Project 6)

In this project we would make the OCRed content saved and available in XMLschema based metadata standard formats such as METS or ALTO for future use. The content from the web archive and the born-digital data would be also converted to XML. The METS standard is a flexible schema for describing a complex digital object like a digitised newspaper issue. METS describes the structure of the object but does not encode the actual textual content of the object. The ALTO standard fills this void by encoding the textual content of a digitised page in great detail, including styles and layouts.

The meta data annotation and OCRed text which would become available from previous steps would be fed to this module in which they would be converted to METS and ALTO formats. A whole set of tools and libraries in Java/Python are available in https://www.loc.gov/standards/mets/mets-tools.html. Example formats of these files can be found in https://www.loc.gov/standards/mets/METSOverview.v2.html.

The METS document contains several sections including,

- METS Header - metadata describing the METS document itself, including such information as creator, editor, etc.
- Descriptive Metadata
- Administrative Metadata
- Filesection - OCRed text files in ALTO format
- Structural links

This would involve the following steps:

- Convert all OCRed text to ALTO format files
- Generate METS format files for each document with different sections mentioned above and linking the text file in ALTO format in the Filesection part of the METS file.
- We might have several files for one large document and they would be grouped in this part of the METS file.

Detailed Proposal :
Mentors : Saatviga Sudhahar
Partners :
Donors :

Auto-labelling (Project 7)

This project would build a machine learning model to auto-label documents with specific annotations such as keywords or topics. We would train a model using annotations performed manually and use the system to auto-label new documents. This would be cross-checked and corrected using a human in the loop which then adds to the training set of the model. This approach is referred to as Active learning in this field, and often used to build ML systems when limited high confidence labels are available to train the model. The resulting labels would be then added to the annotations in the relevant document displayed in Islandora UI.

Detailed Proposal :
Mentors : Saatviga Sudhahar ;
Partners :
Donors :

Section 2 - Language Resource Layer

Language Resources refer to data-only resources such as corpuses, lexicons, tagsets, models and dictionaries or thesaurus which are required to process the language. The resulting corpus created in the previous step would be added to the language resources in the platform as Srilankan Tamil text corpus with annotations.

Detailed Proposal :
Mentors : Saatviga Sudhahar ;
Partners :
Donors :

Section 3 - Processing Resource Layer

Processing resources refer to resources whose character is principally programmatic or algorithmic, such as tokenisers, chunkers or parsers used to process the Tamil language. This component in the platform would include all necessary tools to parse Sri lankan tamil text such as sentence splitter, tokeniser, POS tagger, dependency parser, Morphological analyser/generator, named entity recogniser, coreference resolver, pronominal resolver and word sense disambiguator. Figure 2 below shows how a corpus could be processed by a set of processing resources and used for Analytics.

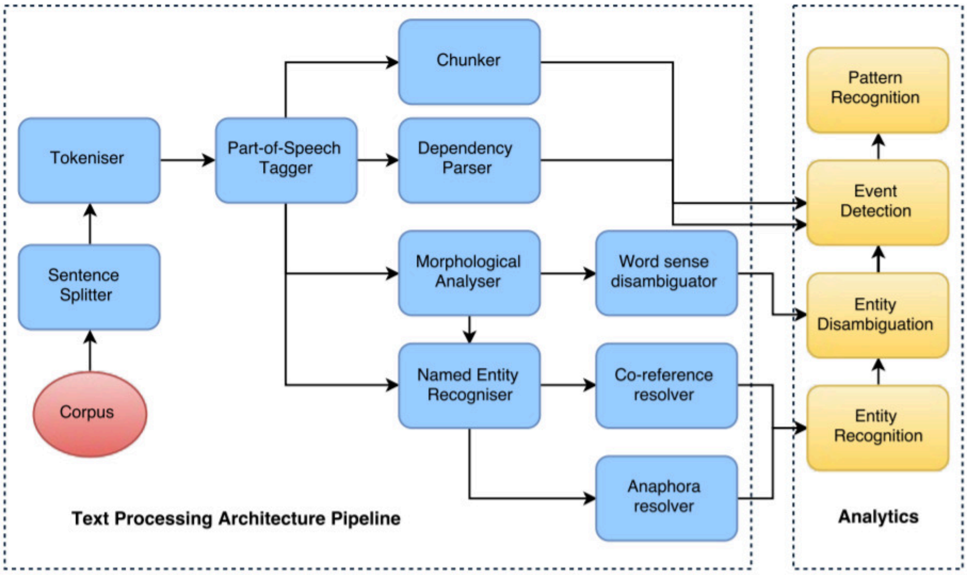

Figure 2: Text Processing Architecture Pipeline

Detailed Proposal :
Mentors : Saatviga sudhahar ;
Partners :
Donors :

Section 4 - Knowledge Engineering

Knowledge engineering would be the ultimate point we would reach when all the previous resources are in place. Using the language and processing resources there is the possibility of analysing the huge amount of Tamil text data for new insights. Simple content analysis of the newspaper articles could detect key events with high accuracy. Beyond just counting words we could detect references to named entities, such as individuals, locations etc and their mentions and trends over a period of time. In addition, we could compare the results from newspapers with text from books written at the time, in order to determine whether newspapers could be “more sensitive to certain culture shifts”, as newspapers had a closer relation to current events.