Genre Classification in Historical Newspapers:

A Project Using the National Library of New Zealand's Papers Past Open Data

UNIVERSITY OF CANTERBURY
Te Whare Wānanga o Waitaha
CHRISTCHURCH NEW ZEALAND

Project goals

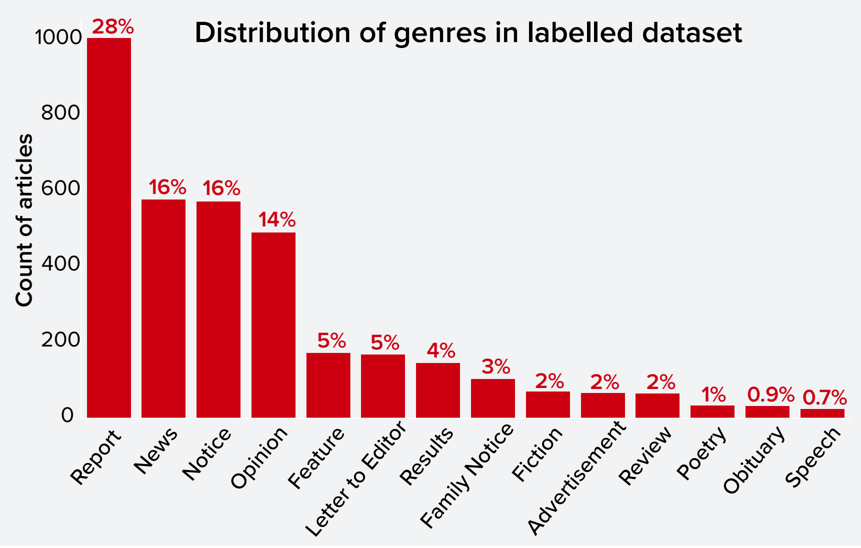
1. Understand the combination of features and machine learning algorithm(s) that perform best in classifying genres of newspaper articles in the Papers Past dataset.
2. Determine if the classifier is robust across topic, time, and newspaper.
3. Document and share the genre classification pipeline in a way that allows other users to easily apply, understand, and adapt it.

Papers Past open data

79 New Zealand newspaper titles
306,538 issues, **1,471,384** pages
Covering the period **1839-1899**

Project data

3,518 articles labelled with **14** genres
57 newspaper titles
Covering the period **1843-1899**


Distribution of genres in labelled dataset

Best binary classification models

Poetry LR (all features)

Accuracy	Precision	Recall	F1	AUROC
0.99	0.60	0.90	0.72	0.95

Fiction LR (all features + title keyword feature)

Accuracy	Precision	Recall	F1	AUROC
0.98	0.57	0.90	0.68	0.94

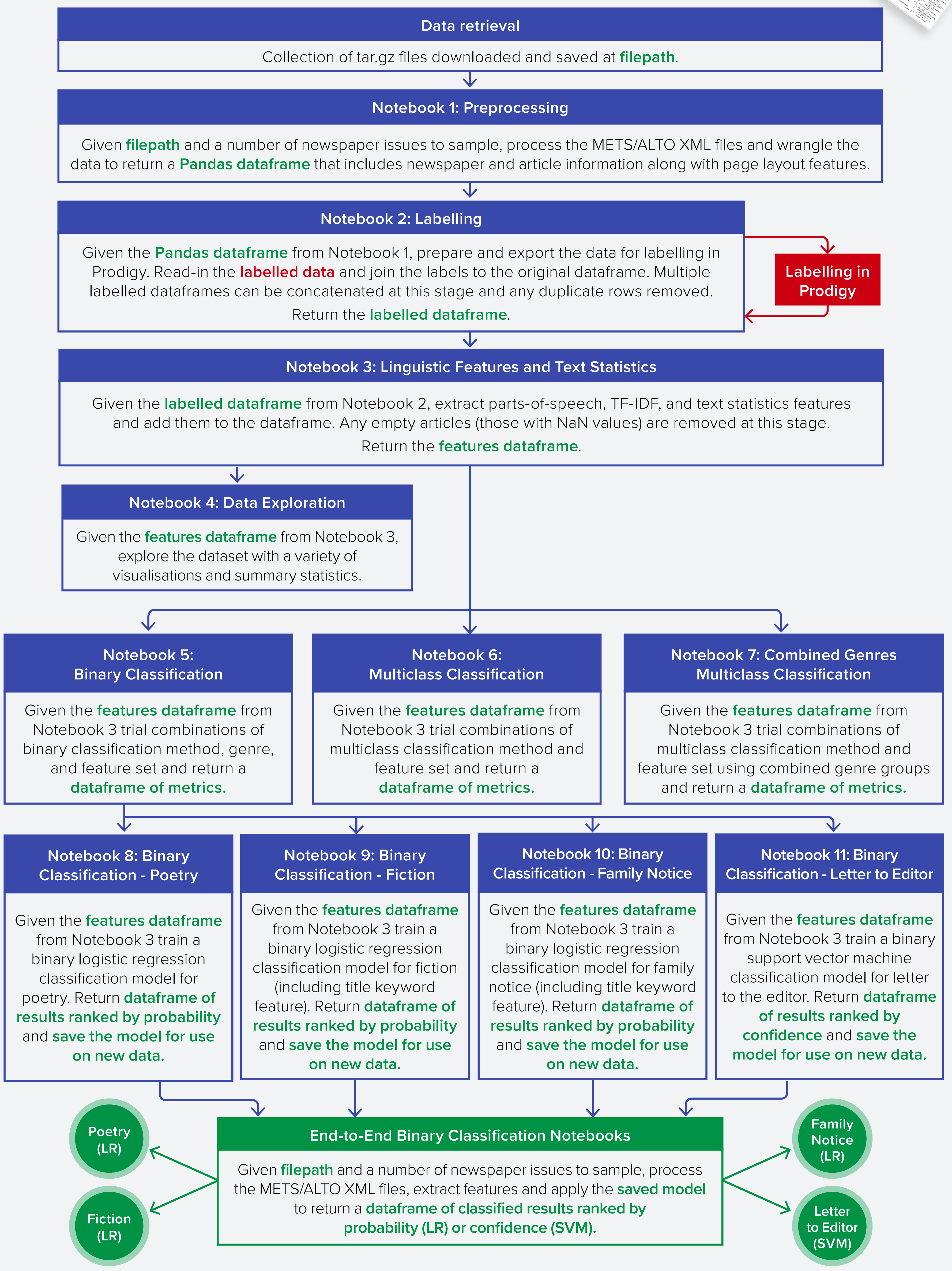
Family Notice LR (all features + title keyword feature)

Accuracy	Precision	Recall	F1	AUROC
0.99	0.83	0.94	0.88	0.97

Letter to Editor SVM (all features)

Accuracy	Precision	Recall	F1	AUROC
0.92	0.35	0.86	0.50	0.89

Use case Researchers, librarians, and genealogists can discover examples of genres such as poetry, fiction, and family notices in the Papers Past dataset in way that is impossible using a keyword search.



Data retrieval
Collection of tar.gz files downloaded and saved at filepath.

Notebook 1: Preprocessing
Given filepath and a number of newspaper issues to sample, process the METS/ALTO XML files and wrangle the data to return a Pandas dataframe that includes newspaper and article information along with page layout features.

Notebook 2: Labelling
Given the Pandas dataframe from Notebook 1, prepare and export the data for labelling in Prodigy. Read-in the labelled data and join the labels to the original dataframe. Multiple labelled dataframes can be concatenated at this stage and any duplicate rows removed.
Return the labelled dataframe.

Labelling in Prodigy

Notebook 3: Linguistic Features and Text Statistics
Given the labelled dataframe from Notebook 2, extract parts-of-speech, TF-IDF, and text statistics features and add them to the dataframe. Any empty articles (those with NaN values) are removed at this stage.
Return the features dataframe.

Notebook 4: Data Exploration
Given the features dataframe from Notebook 3, explore the dataset with a variety of visualisations and summary statistics.

Notebook 5: Binary Classification
Given the features dataframe from Notebook 3 trial combinations of binary classification method, genre, and feature set and return a dataframe of metrics.

Notebook 6: Multiclass Classification
Given the features dataframe from Notebook 3 trial combinations of multiclass classification method and feature set and return a dataframe of metrics.

Notebook 7: Combined Genres Multiclass Classification
Given the features dataframe from Notebook 3 trial combinations of multiclass classification method and feature set using combined genre groups and return a dataframe of metrics.

Notebook 8: Binary Classification - Poetry
Given the features dataframe from Notebook 3 train a binary logistic regression classification model for poetry. Return dataframe of results ranked by probability and save the model for use on new data.

Notebook 9: Binary Classification - Fiction
Given the features dataframe from Notebook 3 train a binary logistic regression classification model for fiction (including title keyword feature). Return dataframe of results ranked by probability and save the model for use on new data.

Notebook 10: Binary Classification - Family Notice
Given the features dataframe from Notebook 3 train a binary logistic regression classification model for family notice (including title keyword feature). Return dataframe of results ranked by probability and save the model for use on new data.

Notebook 11: Binary Classification - Letter to Editor
Given the features dataframe from Notebook 3 train a binary support vector machine classification model for letter to the editor. Return dataframe of results ranked by confidence and save the model for use on new data.

End-to-End Binary Classification Notebooks
Given filepath and a number of newspaper issues to sample, process the METS/ALTO XML files, extract features and apply the saved model to return a dataframe of classified results ranked by probability (LR) or confidence (SVM).

Poetry (LR)
Fiction (LR)
Family Notice (LR)
Letter to Editor (SVM)

Genre classification pipeline

Key Python libraries: pandas, scikit-learn, spaCy, textstat, textfeatures.

UC ARTS DIGITAL LAB

DATA601 2021
Student: Karin Stahel
Supervisors: Chris Thomson, James Williams, Joshua Black