DATA601
Project Report

Genre Classification in Historical Newspapers: A Project Using the National Library of New Zealand's Papers Past Open Data

Author: Karin Stahel (47274804)
Submitted: 11 February 2022
Word count: 10,447

Abstract. Interest in the application of quantitative approaches to text analysis for the humanities is growing and methods such as genre classification offer new ways to explore and gain insights from large collections of digital text. This report presents the methodology and outcomes of a project designed to explore feature sets and machine learning methods for classifying the genre of newspaper articles in the National Library of New Zealand's Papers Past open data set. The motivations for genre classification of digitised historical newspapers are explored through a review of related studies. This review also provides a starting point for identifying candidate feature sets and machine learning methods. Data processing, labelling, and feature extraction phases are followed by trial, evaluation, and refinement of the features and methods. The best performing models are then tested on new data sets. The results and discussion address questions of which combination of features and machine learning algorithm perform best in classifying genres in the Papers Past data and if the best performing models can be shown to be robust across topic, text length, time, and newspaper. The genre classification pipeline is documented in a series of Jupyter notebooks designed to allow other researchers to easily apply and build on the results of this project. In addition, stand-alone notebooks are developed for poetry, fiction, family notices, and letters to the editor, allowing the pipeline to be run end-to-end to discover examples of those genres in the dataset.

1 Introduction

Advances in optical character recognition (OCR) technology have enabled many libraries, museums, and newspaper publishers to successfully undertake large-scale digitisation projects of their historic newspaper collections (Holley, 2009). A wealth of data is now available to researchers in fields such as the digital humanities and journalism history. However, these large datasets present considerable challenges in terms of querying and categorising the texts in ways that go beyond the limitations of a basic key word search (Bilgin, et al., 2018).

The ability to retrieve historic newspaper articles by genre, as opposed to topic, can significantly enhance the accessibility and usefulness of large collections of digital text (Onan, 2018). For example, trying to search a diverse collection of texts for poetry or fiction by keyword is almost impossible, yielding many irrelevant results and missing many that would be of interest. For queries where a particular topic is of interest, say in relation to a specific person or event, the ability to narrow a keyword search by genre further refines the results and makes the search process

considerably more effective and efficient (Onan, 2018). Due to the size of the datasets, manual labelling is prohibitive in terms of both time and financial cost, and researchers are increasingly exploring methods for automated genre prediction as a way to categorise texts (Bilgin, et al., 2018).

This project explores the use of supervised machine learning algorithms for genre classification of the digitised historical newspaper articles made available through the National Library of New Zealand's Papers Past open data pilot (The National Library of New Zealand). A key goal of the project is to produce a transparent pipeline from sampling of raw data, through feature extraction and model training, testing, and evaluation, to applying the best performing models on unseen data. The pipeline was developed and deployed through a series of Jupyter notebooks, designed to be worked through sequentially.[1] In addition, stand-alone notebooks for the best performing models for detecting poetry, fiction, family notices, and letters to the editor were developed.[2] These notebooks sample a given number of issues from the Papers Past open dataset and return a dataframe of articles ranked by probability for the selected genre.

This project was carried out through the UC Arts Digital Lab at the University of Canterbury. It was supervised by Chris Thomson, James Williams, and Joshua Black.

The remainder of this report is organised as follows. Section 2 is a literature review that explores the motivations and methods of published research in the field of genre classification. Section 3 presents the project research questions, provides an overview of the data, and discusses ethical considerations. Section 4 outlines the project methodology. Section 5 presents the results of the experiments. Section 6 discusses the results of the experiments, along with ideas for future work. Section 7 provides concluding remarks.

2 Literature review

Definition of genre

Definitions of genre vary considerably in scope and complexity. Finn and Kushmerick (2006) explore a range of definitions from the relatively straightforward "class; form; style esp. in literature" provided by *Webster's Third New International Dictionary* to a broader working definition from Swales, "A genre is defined as a class of communicative events where there is some shared set of communicative purposes". Gregory (2018) states that classification is central to the concept of genre, "in that genre still depends on the idea that texts can be meaningfully organized and grouped according to their shared attributes".

The distinction between the concepts of topic, genre, and style is an important consideration even though there can be substantial correlation between them (Karlgren & Cutting, 1994; Petrenz & Webber, 2010). Karlgren and Cutting (1994) state, "Texts about certain topics may only occur in certain genres, and texts in certain genres may only treat certain topics; most topics do, however, occur in several genres". Finn and Kushmerick (2006) explicitly seek to separate the identification of the topic and genre of a document and describe their understanding of genre as "what kind of document it is rather than what topic the document is about".

[1] https://jupyter.org/
[2] The reasons for selecting these genres for development of the stand-alone notebooks are discussed in the results and discussion sections of this report.

Motivation and value of genre classification

As the volume of searchable collections of digital text grows, researchers are highlighting the need for search tools that go beyond keyword and Boolean query searches to efficiently identify documents of interest (Allen, Waldstein, & Zhu, 2008; Broersma & Harbers, 2016; Feldman, Marin, Ostendorf, & Gupta, 2009; Kessler, Nunberg, & Schutze, 1997; Rauber & Merkl, 2003). Genre classification is considered a useful solution, and an alternative or complement to the more extensively studied subject and topic-based classification approaches (Lim, Lee, & Kim, 2005).

In addition to enhancing the retrieval of the most relevant documents for a particular user and purpose, classification by genre provides a further layer of information that can open up opportunities for new research areas and insights. Broersma and Harbers (2016) discuss the benefits of genre classification of text in historical newspapers for journalism scholars and media historians who seek to undertake longitudinal research on changes in the norms and structure of journalistic discourse. They state, "We join in calls for journalism scholars to move beyond keyword search and manual content analysis and take full advantage of the available digitized newspaper material" (Broersma & Harbers, 2016).

Murphy (2019) presents a genre classification scheme for a segment of Early English Books Online – Text Creation Partnership (EEBO-TCP) that is designed to enable a comparison of the language of Shakespeare with that of his contemporaries. Murphy (2019) also cites examples of studies that could use the genre classification scheme to investigate historical change. These examples include exploring "lexical innovation" such as the creation of a scientific form of language, and comparing styles to "investigate whether language became more securalized [*sic*]" over a certain time period (Murphy, 2019).

Kilner and Fitch (2017) focus on the challenges that researchers face in discovering examples of poetry and other literary material in large newspaper databases such as the National Library of Australia's collection available through the Trove website.[3] They point out that:

> "Researchers need to know what they are looking for to use Trove effectively. They are unable to search for all poetry in a particular newspaper, for example, though they are able to search for particular keywords that appear in poetry to expose individual poems." (Kilner & Fitch, 2017)

Kilner and Fitch (2017) suggest that developing a method for identifying literary work in New Zealand's Papers Past corpus could be useful to "assist with research into the trans-Tasman flow of literature between Australia and New Zealand."

Genre classification methods

Most of the reviewed methods for text genre classification take a supervised learning approach using one or a combination of well-recognised methods such as Naïve Bayes (NB), Support Vector Machines (SVM), Random Forest (RF), logistic regression (LR), and discriminant analysis. The approaches are generally similar to those used in authorship attribution studies, which aim to capture and quantify distinctive features of text in order to differentiate between authors (Zhang, Wu, Niu, & Ding, 2014).

In an early automated genre classification (AGC) study, Karlgren and Cutting (1994) used discriminant analysis on a set of texts from the Brown Corpus that had already been pre-selected by

[3] https://trove.nla.gov.au/newspaper/

"crude semantic analysis" to further classify the texts according to genre. Kessler, Nunberg and Schutze (1997) experimented with genre classification of texts from the Brown Corpus using logistic regression and neural networks. Finn and Kushmerick (2006) selected the C4.5 decision tree as their learning alogrithm because the set of rules it generates can be easily interpreted by a human.

In general, genre classification studies aim to develop robust models that will perform reliably in classifying text across different topics, time periods, and document lengths (Feldman, Marin, Ostendorf, & Gupta, 2009; Kruger, Lukowiak, Sonntag, Warzecha, & Stede, 2017; Stamatatos, 2018). Various pre-processing and feature selection methods are used to try and achieve this goal. The use of parts-of-speech (POS) tags as features is very common, along with other "surface based" features such as length of the document (characters and sentences), length of sentences (characters and words), type/token ratio, frequency of long words, and frequency of function words (Kruger et al., 2017).

Feldman et al. (2009) used POS histogram statistics to "indirectly provide syntactic information without the cost of parsing". They used a NB baseline but achieved the best results with a quadratic discriminant classifier (Feldman et al., 2009). Finn and Kushmerick (2006) used POS statistics, bag-of-words (BOW), and text statistics (TS) feature sets, along with a combined "multiview ensemble" (MVE). Accuracy was used as the metric. The best performing feature set varied depending on the specific task and they concluded that it was possible to build a genre classifier that performed well in a single-topic domain, but achieving robust domain transfer results were more challenging as their techniques did not provide a complete separation of genre and topic (Finn & Kushmerick, 2006).

Onan (2018) used an ensemble approach based on language function analysis (LFA) and feature engineering to classify documents from two different domains, camera reviews and book reviews, as personal, commercial, or informational. Five different feature sets and their possible combinations were evaluated in the feature engineering stage. In the classification stage, five classification algorithms (NB, SVM, LR, KNN, and RF) were used as base learning methods. These base methods were then combined in ensemble learning methods (AdaBoost, Bagging, and Random Subspace) to improve predictive performance. The feature list using all five features resulted in the highest classification accuracy in conjunction with the Random Subspace ensemble for RF (Onan, 2018).

Allen, Waldstein, and Zhu (2008) used segmentation and genre identification to classify text in digitised historical newspapers with numerous OCR errors. Their approach involved using an XML wrapper that identified information such as date, page, and coordinates of the newspaper segments, and accumulated information as it passed down the processing pipeline. For genre classification they relied on matching words associated with specific genres, using the IPTC genres as the basis for their categories. They noted that their pipeline simplified the distinction between genre and topic and stated, "the vagueness between genres and subjects makes a clear separation between them difficult" (Allen, Waldstein, & Zhu, 2008). Precision and recall were used as metrics and better results were obtained for genre and topic combinations (such as ads: medicine) than for broader genres such as literature or opinion (Allen, Waldstein, & Zhu, 2008).

The work of Kruger et al. (2017) focused on classifying genres in the BLIIP WSJ and NLTK Brown corpora, with a focus on general linguistic features. They aimed to differentiate their study from earlier work and demonstrate the robustness of their model by experimenting with training and test data that covered several different genres (news, editorials, letters to the editor) and newspapers. They used the WEKA machine learning toolkit with the following methods: SVO (SVM using sequential minimal optimisation and trialled with two different kernels: a polynomial kernel and a linear kernel), ADTree, a Bayesian Network, logistic regression, and a multilayer perceptron. F_1-

score was used as the metric. In all test cases they used the linear SMO classifier, which they found to provide the best results at the best performance rates. The study found that models using linguistic features outperformed standard bag-of-lemma approaches when applied to texts that were stylistically different from the training set. They also found that classifiers based on just POS tags or lemmas performed poorly on shorter texts. In these cases, the best results were obtained with a combination of features, including linguistic features (Kruger et al., 2017).

Kilner and Fitch (2017) used a NB model for their work on classifiying poetry in Australian historical newspapers. Their classifier was based on a spam detection model and used the 15 most discriminating words to determine the probability of a given article being poetry. The initial results included a large number of false positives so they added features such as variation in left indents and line lengths, and frequency of rhyming lines. The rhyming lines feature was found to be particularly susceptible to OCR errors, which they improved using the *overProof* correction algorithm.[4] The recall score for their final model was around 88%, although precision was less than 40% (Kilner & Fitch, 2017).

Some researchers have highlighted the importance of transparency in the processing pipelines and methods used for text classification so that decisions and results can be more easily interpreted and analysed by humans (Bilgin, et al., 2018; Schulman & Barbosa, 2018). Bilgin et al. (2018) argue that a transparency-driven environment for genre detection is important given that those using and interpreting the models may often be historians (in their case journalism historians) rather than developers or data scientists. They present a workflow for creating and using a transparent machine learning pipeline, along with a case study on automatic genre classification in historical journalism using eight genres (Background, Column, Interview, News, Op-ed, Report, Review, and Feature) (Bilgin, et al., 2018). Bilgin et al. (2018) also raise the important point that many text classification studies use prediction accuracy as a metric and highlight that this may not provide a true assessment of model performance, particularly when there are multiple and unbalanced classes. Optimisation of accuracy alone may not be the only indicator of a model's usefulness, as misclassifications and comparison of predictions between cases can provide interesting and useful insights.

Issues and considerations for this project

Based on the literature review, the following points were identified as key considerations for this project:

- Selection and definition of the genres to be classified.
- Selection and encoding of features to explore separation of genre and topic.
- Transparency of the machine learning pipeline.
- Dealing with class imbalance.
- Selection of performance metrics.
- Robustness of model across topic, text length, domain/newspaper.

[4] https://overproof.projectcomputing.com/

3 Research questions, data, and ethics

Research questions

Building on previous studies that have used the Papers Past open data[5], and drawing on published research methods in the field of genre classification, this project aims to address the following research questions:

1. What combination of features and machine learning algorithm(s) perform best in classifying a defined set of genres for the Papers Past data?
2. Can the best performing classifier(s) be shown to be robust across topic, text length, time, and newspaper?
3. How can the genre classification pipeline be documented and shared to best enable other users to easily apply, adapt, and interpret its performance on other datasets?

Data

The dataset for this project is a subset of the National Library of New Zealand's Papers Past collection of digitised historical newspapers that has been made available through the Papers Past open data pilot project. Although the pilot project has now finished, the data is still available for download from the Papers Past website.[6] The data is compressed in a collection of TAR Gzip archive files for each newspaper and year combination. The total download size of the open data pilot files is around 235 GB.

The dataset is organised by newspaper and year with a total of 79 New Zealand newspaper titles covering the period 1839-1899. Approximately half the newspaper titles in the data set are from the North Island and half from the South Island. Otago is the most represented region with 18 newspaper titles, followed by Canterbury with 10. Figure 1 shows the regional distribution of the newspapers in the Papers Past open data set. In total, there are 306,538 newspaper issues and 1,471,384 pages.[7]

Each newspaper issue is stored as a set of METS (Metadata Encoding and Transmission Standard) and ALTO (Analyzed Layout and Text Object) XML files. METS and ALTO are XML standards maintained by the Library of Congress and are the current industry standard for newspaper digitisation (Veridian Software). The METS file describes the structure of a digital object (in this case the digitised newspaper issue) but does not encode the content. The corresponding ALTO files encode the content, including information such as styles, layout, and spatial coordinates of columns, lines, and words on each page (Veridian Software). Figure 2 provides a snippet of the contents of the METS and ALTO files for the 25 May 1861 issue of the Christchurch Press.

As described by Black (2021), the Papers Past newspaper dataset is created from microfilm images that have been processed using optical character recognition (OCR) software. Some of this processing was carried out more than ten years ago and has a high number of OCR errors (Black,

[5] The work of Joshua Black in his DATA601 project, '*Using Data Science Methods to Investigate Philosophical Discourse in Early New Zealand Newspapers*' was particularly informative in relation to methods and code for extracting data from the newspaper METS/ALTO XML files into a dataframe, as well as for general information and advice regarding the structure and characteristics of the dataset (Black, Newspaper Philosophy Methods, 2021).

[6] https://natlib.govt.nz/about-us/open-data/papers-past-metadata/papers-past-newspaper-open-data-pilot/dataset-papers-past-newspaper-open-data-pilot

[7] The dataset statistics are calculated in Notebook 4: Data Exploration.

2021). These errors have an impact on the features and performance of text classification models, which will be addressed further in the results and discussion sections of this report.

The data in the Papers Past open data pilot is out of copyright and may be copied and otherwise re-used in New Zealand without copyright-related restrictions (The National Library of New Zealand).

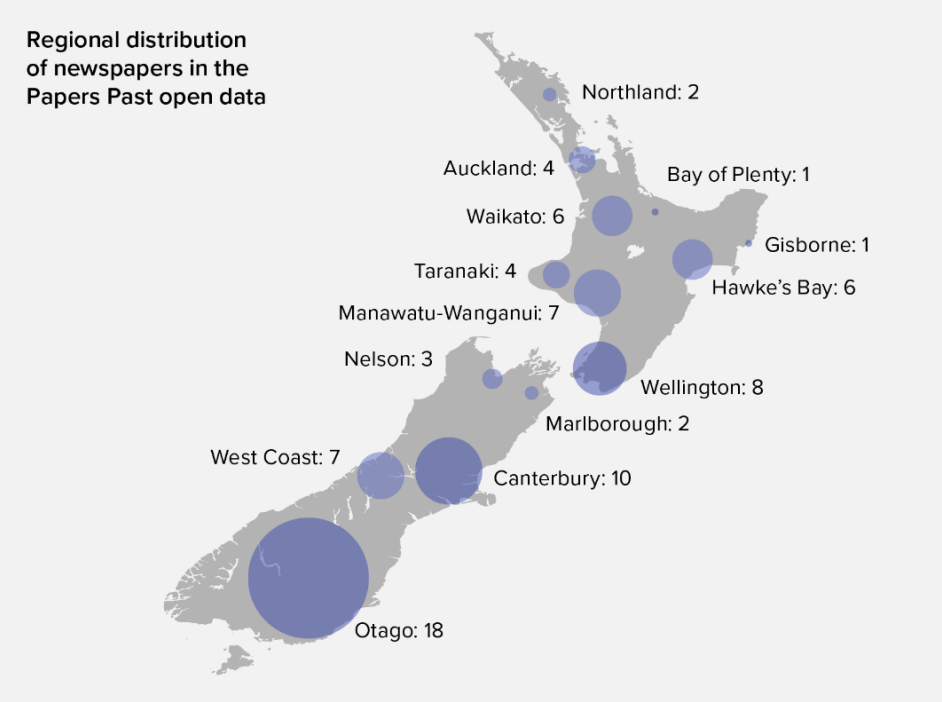


Figure 1. Regional distribution of newspapers in the Papers Past open data.

Figure 2. Portion of the METS file (left) and page one ALTO file (right) for the 25 May 1861 issue of the Christchurch Press.

Ethics

The dataset used in this project consists of publicly available digitised newspaper articles that were published in New Zealand newspapers during the period 1839-1899. Greater access to historical newspaper collections can play an important role in broadening understanding of a nation's history and heritage (Stone, 2012).

In terms of the Papers Past open data, the content of the articles in these newspapers predominantly reflects the concerns and perspectives of European settlers in New Zealand. As noted by Nairn et al. (2017) newspapers of the time promoted a collective narrative amongst settlers and some, such as the *New Zealand Gazette* published by the New Zealand Land Company, also aimed to support commercial interests by promoting "progress and thriving settlements" to the company's investors. It is important that the content returned by the models developed in this project is read and analysed in the context of its time, place, and purpose to avoid perpetuating bias or discriminatory perspectives.

4 Methodology

Pipeline

One of the aims of this project is to document and share a genre classification pipeline that allows other users to easily interpret its performance and adapt it to suit their needs. This is in line with the approach of researchers such as Bilgin et al. (2018) who promote a "transparency-driven environment" where the methods and predictions of a machine learning pipeline can be explained to domain experts such as historians.

The pipeline for this project is implemented using a series of Jupyter notebooks and the annotation software Prodigy.[8, 9] The notebooks are designed to be run in sequence, each building on the previous one to complete part of the pipeline. Breaking down the pipeline in this way, with distinct inputs and outputs for each notebook, allows each stage to be explored and modified in manageable chunks.

Figure 3 illustrates the genre classification pipeline developed for this project. Each stage is described in more detail in the following sections of the methodology.

[8] The Jupyter notebooks are provided as supplementary files.
[9] https://prodi.gy/

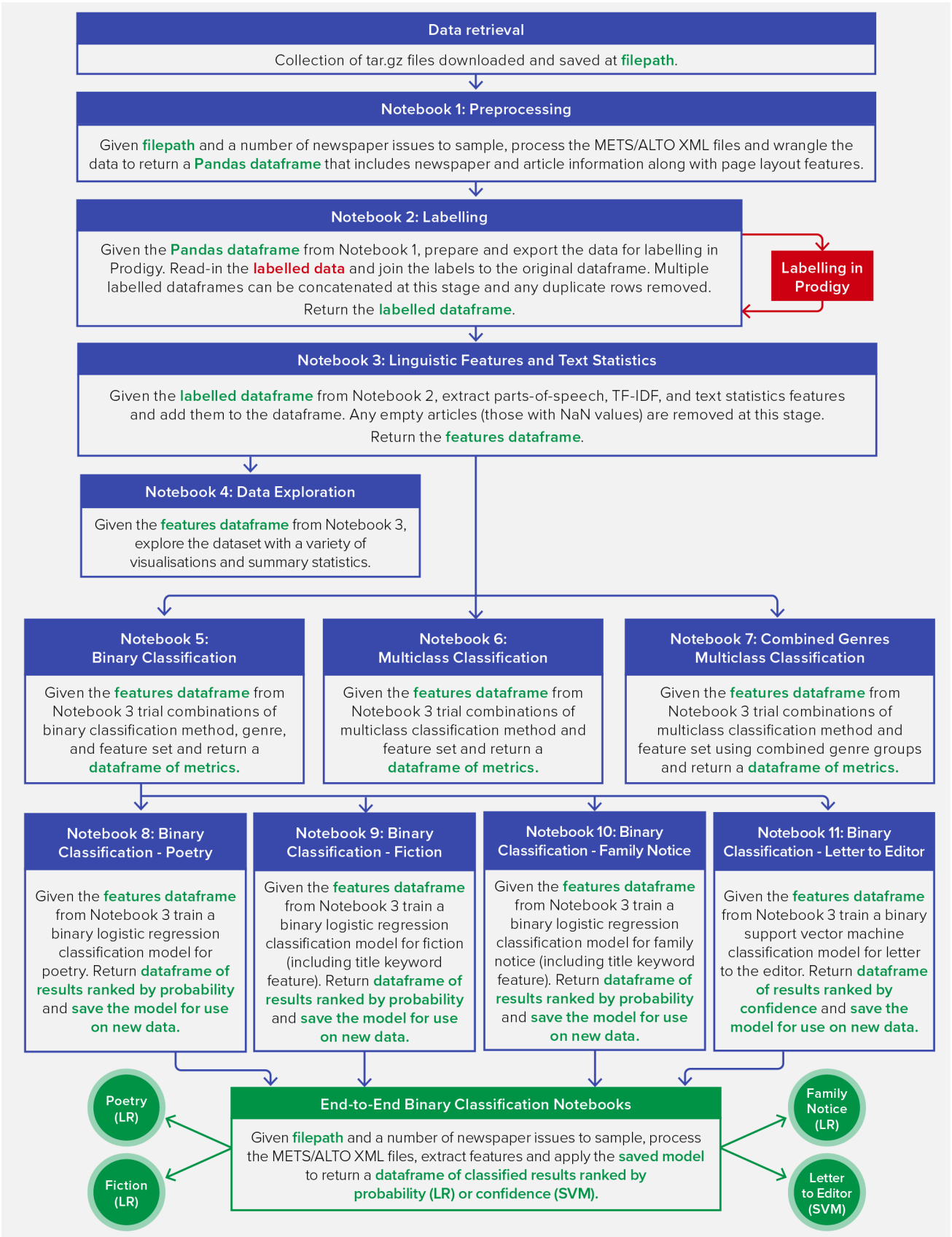


Figure 3. Flow diagram of the genre classification pipeline used in this project highlighting inputs and outputs for each stage.

Data retrieval, preprocessing, and structural/spatial feature extraction

The entire collection of TAR Gzip archive files for each newspaper and year combination was downloaded from the Papers Past dataset web page[10] using the 'Downthemall' browser extension.[11] The files were saved to a directory on the MADS network drive and left in their compressed form.

To extract articles from the archive files, variables for the file path of the raw data directory and the number of newspaper issues to randomly sample are set in Notebook 1, with the option to use a random seed for tracking and reproducibility. The process to extract the article IDs, titles, and text from the METS/ALTO files for each issue and return them in a Pandas[12] dataframe was defined in a series of Python[13] functions based on the code developed for a previous Papers Past project (Black, Newspaper Philosophy Methods, 2021). This code was further developed to extract page layout features such as line widths and horizontal position of text blocks and lines. The page layout data is summarised as statistical features i.e., the average, maximum, minimum, and range of line widths and the average, maximum, and minimum offsets of lines from the text block for each article.

Some of the files in the dataset are corrupt, in which case the issue is skipped and an error message printed with the failed file path. When sampling is complete, the final number of articles retrieved is printed and any duplicate rows can be removed from the dataframe.[14]

The next stage of preprocessing in Notebook 1 involves extracting individual data items (newspaper ID, date, article ID) from the index column and then removing the redundant column, converting the date column to date format, and reordering columns. Full newspaper names are added using a dictionary of newspaper codes mapped to newspaper name. Notebook 1 also facilitates a preliminary exploration of the returned data by providing views of the full-text of selected articles and displaying the newspapers and count of articles per newspaper in the sample.

The final dataframe is saved in pickle format for use in the next stage of the pipeline.[15] The columns and datatypes in the dataframe returned at this stage shown in Appendix A (Table 1).

Labelling

Defining the set of genre labels to be used is fundamental to the classification process. The set used for this project was developed through a process of researching newspaper genres, drafting a candidate list, exploring the validity of the selected genres for the Papers Past dataset through a series of labelling trials in Prodigy, and refining the genre set based on the results. The genres used in the final labelled dataset for this project are as follows:

- **News:** Recent or breaking news. Factual and does not express a distinct opinion or angle.
- **Report:** A report of a meeting or event. Describes the event and outcomes in detail without the author expressing a distinct opinion. Includes reports of parliamentary and court proceedings, public events (including sporting events), council and community meetings, and social occasions.

[10] https://natlib.govt.nz/about-us/open-data/papers-past-metadata/papers-past-newspaper-open-data-pilot/dataset-papers-past-newspaper-open-data-pilot
[11] https://www.downthemall.net/
[12] https://pandas.pydata.org/docs/index.html
[13] Python 3.8.5 (default, Sep 3 2020, 21:29:08) [MSC v.1916 64 bit (AMD64)]
[14] No duplicates have been encountered in any of the trials run as part of this project.
[15] https://docs.python.org/3/library/pickle.html#

- **Speech:** A printed speech. Note, where only short portions of a speech are intermingled with a general report, the article is classified as a report. The 'Speech' label was used where most of the article was the speech printed verbatim.
- **Feature:** A factual article that covers an issue or topic in depth. May contain a variety of sources and angles. The author does not express a personal opinion or angle.
- **Obituary:** An article that announces a person's death and provides more detailed information about their life than a simple death notice.
- **Opinion:** An article where the author expresses an obvious opinion or angle (includes editorials).
- **Letter to Editor:** A printed letter to the editor.
- **Review:** An article that gives the details and author's opinion of an event, book, exhibition etc.
- **Family Notice:** A short notice announcing a personal/family event such as birth, death, or marriage.
- **Notice:** A public information notice, for example shipping arrivals and departures, upcoming events, postal deadlines, etc.
- **Advertisement:** An article that promotes an item or service that can be purchased.
- **Fiction:** A work of literary prose such as short stories or serialised fiction. Jokes, 'moral tales', and non-fictional narrative or dialogue is not included (these forms were assigned to the 'Other' class for this project).
- **Poetry:** Any form of verse ranging from a few lines to hundreds of words and distinct from other genres in its patterns of meter, rhyme, line breaks, and stanzas (SuperSummary, n.d.).
- **Results:** A notice or article that is predominantly a list of results. E.g. sports, horse racing, sales, exam results, agricultural competitions. Where results are intermingled with commentary, a judgement call is made regarding whether the article sits better under 'Report' or 'Results'.
- **Other:** Anything that does not obviously fit in the above categories. This includes articles that can't be classified due to a high number of OCR errors and articles that may include a number of genres, such as news, opinion, and advertising in a single run-on article. Genres such as recipes, gossip, and entertainment (jokes and puzzles) are included in 'Other' in this project. Future work may seek to explore classification of these genres as they are represented in reasonable numbers in the Papers Past dataset and could have research and general interest value.

It is worth noting the distinction between those genres listed above that can be readily discovered through a keyword search compared to those where this is very difficult, if not impossible. As in the National Library of Australia's digitised newspaper collection described by Kilner and Fitch (2017), the genre or form of most articles in the Papers Past collection is not identified. The content is simply categorised as either 'article', 'advertisement', or 'illustration' and the content is searchable online by keyword. A keyword search can be effective for finding news articles or reports about a specific event or person, features about a given topic, or results for a sport such as cricket. However, the keyword search is not useful for the researcher who wants to discover examples of a particular type of content, such as poetry, fiction, or letters to the editor, regardless of topic.

The labelling stage of the genre classification pipeline uses Notebook 2 and the annotation software Prodigy. The dataframe saved at the end of the preprocessing stage is read-in to Notebook 2 and prepared for labelling in Prodigy by extracting the article text and title as columns in a csv

file.[16] The final stage of Notebook 2 imports the genre label annotations created in Prodigy (in the form a jsonl file) and joins them to the original dataframe.

Linguistic feature extraction

Following labelling, parts-of-speech (POS) and text statistic features are extracted and added to the dataframe in Notebook 3. For efficiency, articles labelled as 'Other' are removed before linguistic feature extraction. The article text column, which contains unnecessary symbols that are the result of OCR errors, is cleaned with a function that uses regular expressions to parse only alphanumeric strings and hyphens (to include hyphenated words) (Rao, 2020). Observations with missing values are also removed at this stage.

The textstat and textfeatures Python libraries are used to add features for word count, character count, average word length, syllable count, and frequencies of stopwords, monosyllabic and polysyllabic words.[17, 18]

A spaCy pipeline is used to add POS tags, which are then converted to frequencies.[19] In early trials, a basic POS feature set from the Universal POS tag set was used.[20] This set included proper nouns, verbs, nouns, adjectives, pronouns, and numbers. A more finely grained POS feature set was added later, using the Penn Treebank tags.[21] The selection of tags for this set is based on the findings of previous research that identified certain POS frequencies which improved predictive performance and did not impair model stability (Webber, 2011). The tags included in this feature set are plural proper nouns, base form verbs, singular or mass nouns, adjectives, cardinal numbers, personal pronouns, adverbs, coordinating conjunctions, singular proper nouns, past tense verbs, and third-person singular present verbs.

As a goal of this project is to produce models that are robust across topic, features that introduce elements of topic were initially avoided. As the project progressed, a decision was made to explore the influence of two features that are related to topic: a term frequency-inverse document frequency (TF-IDF) feature and a feature that creates a binary variable ('title_keyword') based on the appearance of keywords in the article's title. The TF-IDF feature is the sum of the TF-IDF scores for a given number of top words and is applied in the binary and multiclass classification experiments. The 'title_keyword' feature was experimented with to refine the results of the top performing binary classification models for specific genres. The influence of these topic-related features is discussed later in this report.

As in previous notebooks, the final dataframe is saved in pickle format for use in the next stage of the pipeline. The columns and datatypes in the dataframe returned at this stage are shown in Appendix A (Table 2).

Data exploration

Notebook 4 explores the given dataset (a Pandas dataframe saved in pickle format) through summary statistics and data visualisation. The insights provided in this notebook include: dataset

[16] The code used to run the Prodigy labelling pipeline and return the annotated text is provided in the supplementary file 'pp_prodigy.sh'.
[17] https://pypi.org/project/textstat/
[18] https://towardsdatascience.com/textfeatures-library-for-extracting-basic-features-from-text-data-f98ba90e3932
[19] https://spacy.io/usage/processing-pipelines
[20] http://universaldependencies.org/u/pos/
[21] https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

statistics, counts of articles by genre, data types, distribution of articles across time, genre, and newspaper, word counts by genre, and distribution of genres over time. Also included are pairs plots, density plots, and boxplots for selected features by genre. Some of the plots from this notebook are included in later sections of this report.

Binary classification

In Notebook 5 combinations of genre, feature set, and binary classification method are trialled and evaluated. Details of the feature sets are included in Appendix A (Table 3).

The binary classification methods selected for evaluation are commonly used in text classification and most of those trialled here were used in the genre classification studies reviewed as part of this project. The scikit-learn library for Python is used to implement this stage of the pipeline with the following classifiers: [22]

- Stochastic Gradient Descent
- Random Forest
- Decision Tree
- AdaBoost
- GradientBoosting
- Gaussian Naïve Bayes
- K Nearest Neighbour
- Logistic Regression
- Support Vector Machine
- Dummy classifier (included as a baseline comparison).

A train/test split of 70/30 is used, and the data is scaled using scikit-learn's standard scaler. Where allowed by the method, the parameter class_weight is set to "balanced" and a random state set at "3". The balanced class parameter "uses the values of y to automatically adjust weights inversely proportional to class frequencies in the input data" (scikit-learn developers).[23]

The performance of each combination of method, genre, and feature set is evaluated against the metrics accuracy, precision, recall, F1, and AUROC. If an "average" parameter is available for the metric this is set to "binary". A dataframe of the metrics for all combinations is returned (sorted by highest AUROC score) and exported as a csv file for further analysis.

Multiclass classification

Notebook 6 trials and evaluates combinations of feature set and multiclass classification method to determine the best performing model. The same feature sets and models are used as in the binary classification experiment, apart from logistic regression which is only suitable for binary classification.

Again, a train/test split of 70/30 is used and the data is scaled using scikit-learn's standard scaler. In the multiclass experiment the class_weight parameter is not included as it was found to impair model performance.

---

[22] https://scikit-learn.org/stable/

[23] The methods were trialled with and without the class_weight = "balanced" parameter. When it was included, the metrics were considerably improved for binary classification task and slightly impaired for multiclass classification.

For the multiclass classification trial, the combinations of method and feature set are evaluated against the metrics accuracy, precision, recall, and F1. If an "average" parameter is available for the metric it is set to "weighted". A dataframe of the metrics for all combinations is returned (sorted by highest F1 score) and exported as a csv file for further analysis.

Multiclass classification – combined genre classes

Notebook 7 trials and evaluates the same combinations of feature set and multiclass classification methods as those in Notebook 6, using the same preprocessing and settings. The difference in this case is that some genres have been combined in broader "Opinion" and "Reportage" genre groups, which reduces the number of classes. These broader groups are in line with those selected by researchers such as Kruger et al. (2017), who classified newspaper text into two classes: "opinion" (editorials, commentary, letters to the editor) and "neutral" (reports). The genre groupings used in Notebook 7 are shown Table 1. It is worth drawing attention to two genre grouping decisions that may raise questions. First, obituary is included in the reportage genre group rather than family notice because the style of writing and format of the long-form obituary is more aligned with reports than with notices. Second, family notice is a subset of the notice genre differentiated only by topic but it was retained as a distinct genre due to its potential interest for researchers such as genealogists.

Combined genre groups
Family Notice
Opinion: Letter to Editor, Opinion, Review, Speech
Reportage: Feature, News, Report, Results, Obituary
Fiction
Poetry
Notice

Table 1. Genre groups used in Notebook 7.

A dataframe of the metrics for all combinations is returned (sorted by highest F1 score) and exported as a csv file for further analysis.

Binary classification – best performing combinations

Notebooks 8, 9, 10 and 11 train and save the best performing models for binary classification of poetry, fiction, family notice, and letter to the editor. These genres were selected for further exploration for a few reasons. First, they are among the genres that were most successfully classified by the machine learning models trialled in Notebook 5.[24] Second, they are genres that were identified during discussions about the potential outcomes of the project as being useful and interesting to researchers. Third, in the case of poetry, fiction, and letter to the editor it is not possible to retrieve meaningful results from a keyword search as these genres can cover many topics. The family notice genre is different in this third aspect. As mentioned previously, it is a sub-genre of notice differentiated by the topic (births, deaths, and marriages). Although this deviates from the goal of developing models that are robust across topic, the potential value to researchers and librarians of being able to extract family notices as a distinct type of content meant it was included as a separate genre.

[24] See Section 6 of this report for further discussion of the results.

The notebooks print the accuracy, precision, recall, F1, and AUROC scores for the model, along with an AUROC chart. The trained model is saved for later use and a dataframe is returned showing the predicted classes and prediction probabilities for the train and test data. For the logistic regression models, the coefficients are converted from log odds to odds so that they can be more easily interpreted and charts of the top five positive and negative coefficients are produced (Benton, 2020).

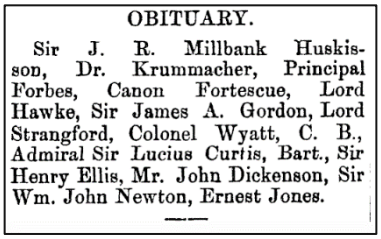
End-to-end binary classification notebooks

The best performing models saved at the previous stage of the pipeline are finally put to use in stand-alone binary classification notebooks. Variables for the file path of the raw data and the number of issues to sample are set, along with a filename for the export of the final csv file. A random seed can also be set for reproducibility. The full pipeline is run from sampling and data extraction, through feature extraction, to applying the saved model to classify the data for the selected genre. A dataframe of articles is returned and exported as a csv file, with the results sorted by the highest probability (or distance from the decision boundary in the case of the SVM classifier) for the positive class. The distance from the decision boundary is provided by the "decision_function" method in scikit-learn (Joshi, 2015). This method is preferred over probability as a confidence score for the SVM classifier for reasons of computational efficiency and to avoid issues of inconsistency between probability estimates and the predicted classes (scikit-learn developers). A link to the newspaper issue for each article is constructed and added to a column of the returned dataframe, allowing a scan of the original article to be easily located and viewed.

5 Results

Data retrieval, preprocessing, labelling, and feature extraction

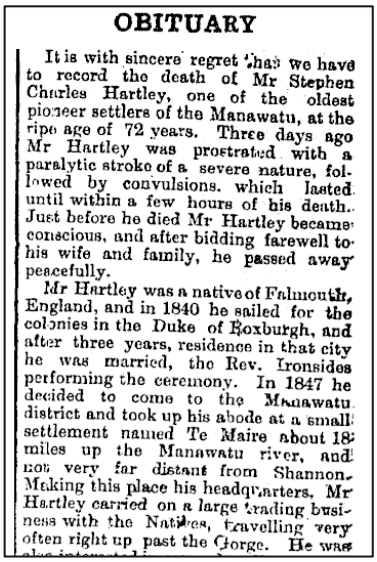
The data retrieval, preprocessing, labelling, and feature extraction stages were trialled with small samples of articles. These trials tested the code and checked the validity of the outputs. The labelling trials allowed the suitability of the genres to be evaluated and refinements were made as a result. For example, it was found that many articles titled "Obituary" were simple notices that stated a name (or list of names), sometimes along with the place or time of death (see example in Figure 4). This is in contrast to the long-form obituary that provides a detailed account of the person's life and achievements (see Figure 5). With consideration of potential use cases, the decision was made to create separate family notice and obituary genres, with a judgement call made during labelling regarding the best fit for a particular example.



OBITUARY.

Sir J. R. Millbank Huskisson, Dr. Krummacher, Principal Forbes, Canon Fortescue, Lord Hawke, Sir James A. Gordon, Lord Strangford, Colonel Wyatt, C. B., Admiral Sir Lucius Curtis, Bart., Sir Henry Ellis, Mr. John Dickenson, Sir Wm. John Newton, Ernest Jones.

Figure 4. A 'family notice' style obituary from The Press, 23 March 1869.



OBITUARY

It is with sincere regret that we have to record the death of Mr Stephen Charles Hartley, one of the oldest pioneer settlers of the Manawatu, at the ripe age of 72 years. Three days ago Mr Hartley was prostrated with a paralytic stroke of a severe nature, followed by convulsions, which lasted until within a few hours of his death. Just before he died Mr Hartley became conscious, and after bidding farewell to his wife and family, he passed away peacefully.

Mr Hartley was a native of Falmouth, England, and in 1840 he sailed for the colonies in the Duke of Roxburgh, and after three years, residence in that city he was married, the Rev. Ironsides performing the ceremony. In 1847 he decided to come to the Manawatu district and took up his abode at a small settlement named Te Maire about 18 miles up the Manawatu river, and not very far distant from Shannon. Making this place his headquarters, Mr Hartley carried on a large trading business with the Natives, travelling very often right up past the Gorge. He was

Figure 5. A snippet of a long-form obituary from the Manawatu Times, 24 June 1897.

A larger dataset was then sampled and labelled in three stages, with the goal of obtaining a minimum of 30 examples of each genre. This was successful for all genres apart from 'Speech', for which only 25 examples were found.[25] Table 2 provides a breakdown of the number of issues sampled and the number of articles retrieved for construction of the labelled dataset.

Number of issues sampled	Number of articles retrieved
100	2,476
50	1,430
25	771

Table 2. Sampling stages in the construction of the labelled dataset.

Articles labelled as 'Other' were removed, along with 'empty' articles (those that had a title but no text) leaving a labelled dataset of 3,518 articles across 14 genres. The distribution of the genres in the labelled dataset is shown in Figure 6. Report was the most represented genre with 1,001 examples, and speech the least represented with 25 examples. In total 57 different newspaper titles are represented in the labelled dataset, which covers 72% of the titles in the full Papers Past open data. The distribution of articles across newspapers in the labelled dataset is shown in Figure 7. The distribution of articles is shown by year in Figure 8.

[25] With consideration of time constraints for the project, the decision was made to continue with the development of the pipeline and models rather than spend further time sampling to boost the number of speeches in the dataset.

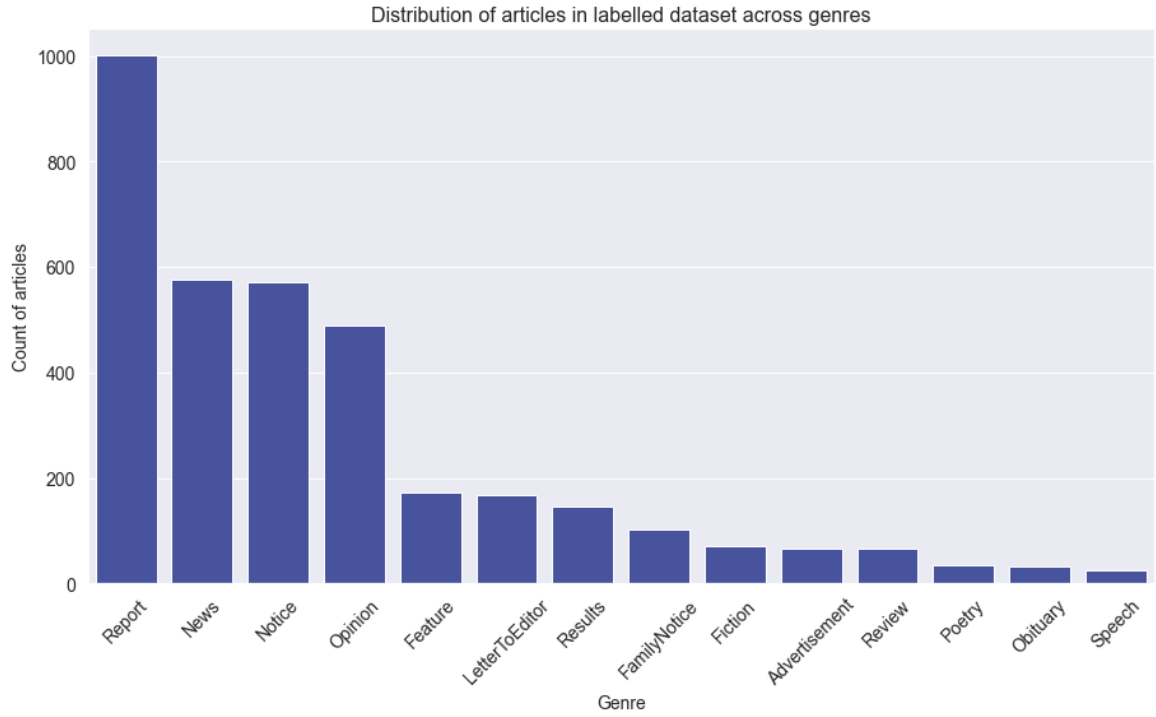


Figure 6. Distribution of genres in the labelled dataset.

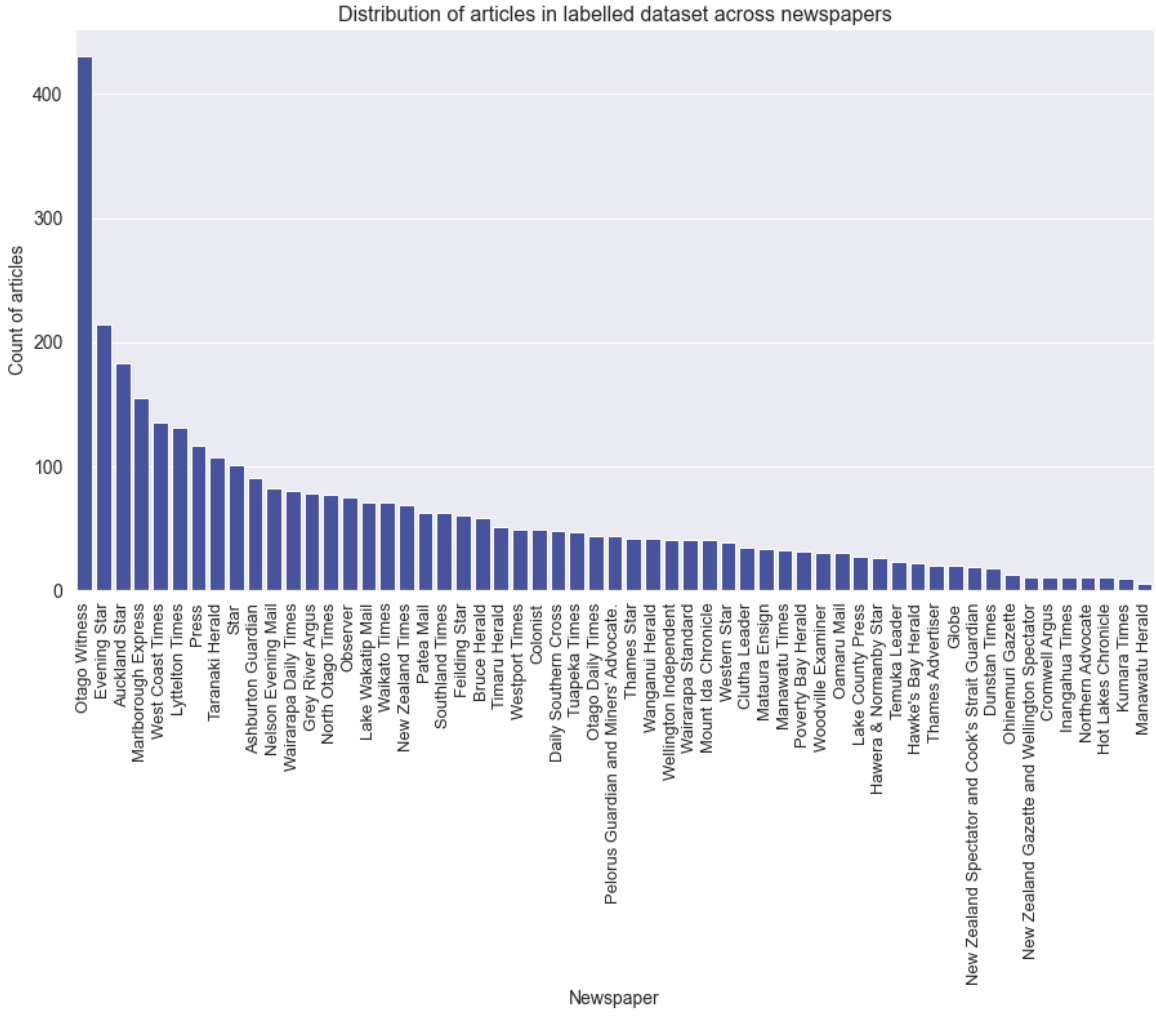


Figure 7. Distribution of articles by newspaper in the labelled dataset.

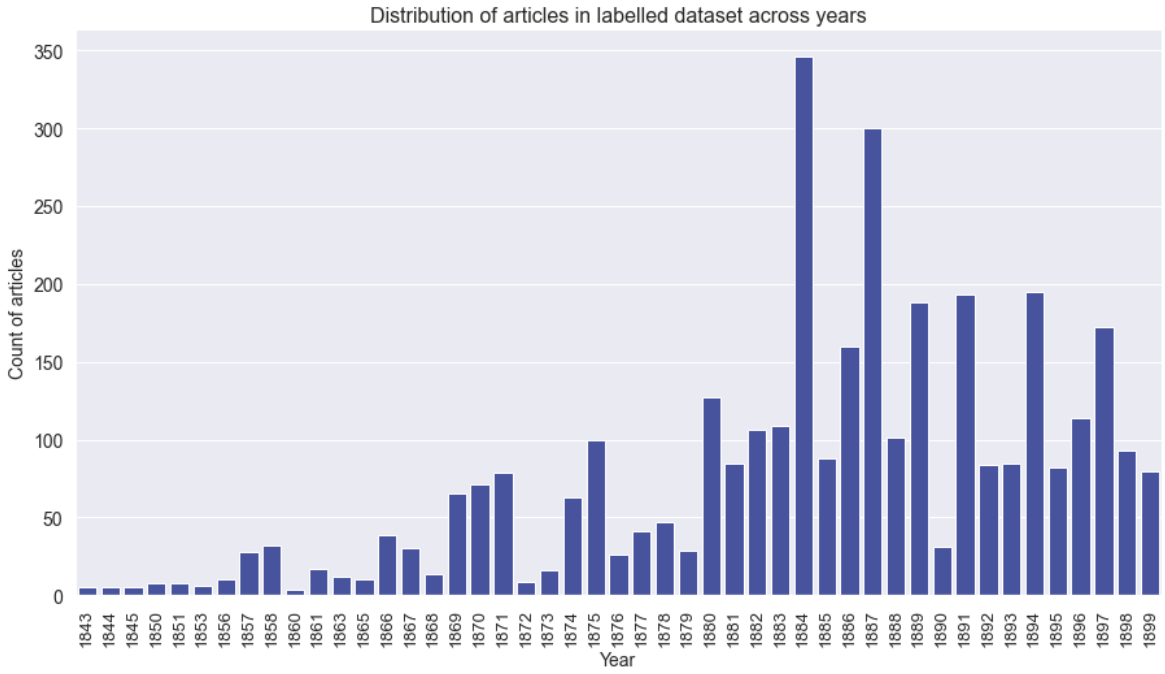


Figure 8. Distribution of articles by year in the labelled dataset.

Binary classification comparisons

The output of the binary classification trials in Notebook 5 is a dataframe of all combinations of classification method, feature set, and genre ranked by AUROC score. The metrics for all combinations are provided as a supplementary Microsoft Excel file with filters for easy exploration of the results.[26] The most successful model was a logistic regression classifier for poetry using all features, which achieved an AUROC of 0.95 (2 d.p.). The AUROC chart for this model is shown in Figure 9.

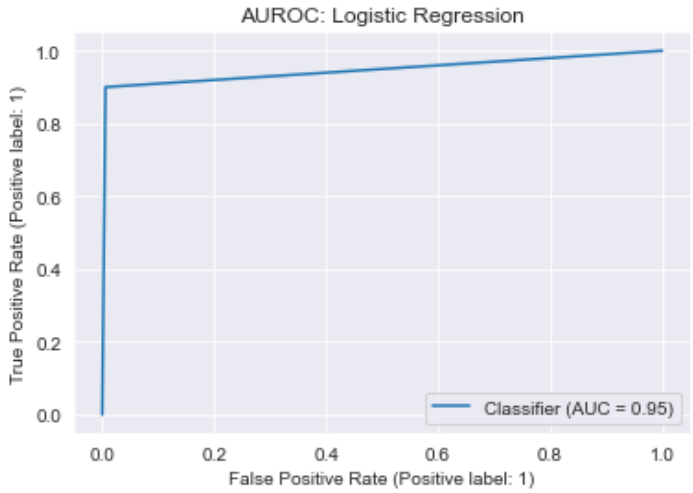


Figure 9. AUROC chart for the logistic regression classifier for poetry, using all features.

[26] 20220113_PP_3518articles_BinaryMetrics.xlsx

The metrics for the top performing binary classification models for each genre are shown in Table 3. The 'all_features' feature set, or variations of it that exclude one or other of the POS feature sets or the TF-IDF feature, performs strongly across all genres, as did the two POS frequency feature sets either individually or in combination. Logistic regression and support vector machine were the only methods that appeared in the top-ranking combinations by AUROC, although stochastic gradient descent and Gaussian Naïve Bayes also produced good results, particularly for classifying fiction. In deciding which models to pursue further, the end use was kept top of mind i.e., discovering relevant examples of a genre in way that overcomes the limitations of a keyword search. As such, recall, transparency, and interpretability were the key considerations. The full list of feature sets is described in Table 3 of Appendix A.

Genre	Feature set	Method	Accuracy	Precision	Recall	F1	AUROC
Poetry	all_features	LR	0.99	0.60	0.90	0.72	0.95
Fiction	all_features	LR	0.98	0.54	0.90	0.68	0.94
FamilyNotice	pos_freq_univ	SVM	0.94	0.33	0.94	0.49	0.94
Speech	pos_freq_penn	SVM	0.95	0.13	0.88	0.23	0.92
Notice	pos_freq_combo	LR	0.90	0.64	0.91	0.75	0.91
LetterToEditor	all_features	SVM	0.92	0.35	0.86	0.50	0.89
Obituary	pos_freq_combo	LR	0.87	0.06	0.90	0.12	0.89
Review	all_features_excl_univ	LR	0.86	0.11	0.90	0.19	0.88
News	all_features_excl_univ	SVM	0.87	0.57	0.84	0.68	0.86
Results	all_features_excl_penn	SVM	0.90	0.28	0.82	0.41	0.86
Opinion	all_features_excl_tfidf	SVM	0.81	0.42	0.88	0.57	0.84
Feature	all_features_excl_tfidf	LR	0.81	0.19	0.88	0.31	0.84
Advertisement	pos_freq_combo	LR	0.84	0.09	0.80	0.16	0.82
Report	all_features_excl_univ	SVM	0.79	0.59	0.83	0.69	0.80

Table 3. Best performing combination of method and feature set (by AUROC) for the binary classification task for each genre.

Binary classification – best performing combinations

As noted previously, Notebooks 8, 9, 10 and 11 train and save the best performing models for binary classification of poetry, fiction, family notice, and letter to the editor. At this stage, the impact of the 'title_keyword' feature was trialled to see if the performance of the binary classifiers could be further improved with information from the article's title. This feature is a binary label that indicates the presence of certain keywords relevant to the genre in the title of the article.

The keywords were selected through a manual inspection of the titles of examples of the genre in the labelled dataset. For example, some newspapers included a regular "Poets' Corner" or published poetry under the title, "Select Poetry". In the case of fiction, if it was published in serialised form the title often included the word "Chapter". Short stories are sometimes identified as such in the title, and some newspapers published fiction under the titles of "Fiction" or "Literature". Letters to the editor are often titled "Correspondence", and family notices almost always have titles in the form of "Births", "Deaths" (or "Died"), or "Marriages" (or "Married"). "Obituary", "Funeral", and "Memoriam" were also included in the family notice keywords.

The 'title_keyword' feature significantly improved the performance of the logistic regression/all features model for family notice. Although this model originally ranked second behind the SVM/pos_freq_univ model for family notice in terms of AUROC, it was tested here because it is the same method and feature combination as the top performing models for poetry and fiction. The metrics for the family notice logistic regression/all features model with the 'title_keyword' feature

included are: 0.99 accuracy, 0.83 precision, 0.94 recall, 0.88 F1, and 0.96 AUROC (2 d.p.). The AUROC chart for this model is shown in Figure 10. These results outperform the original SVM/pos_freq_univ model for family notice. The recall of the SVM model deteriorated when the 'title_keyword' feature was included, although precision was excellent at 0.96 (2 d.p.).[27]

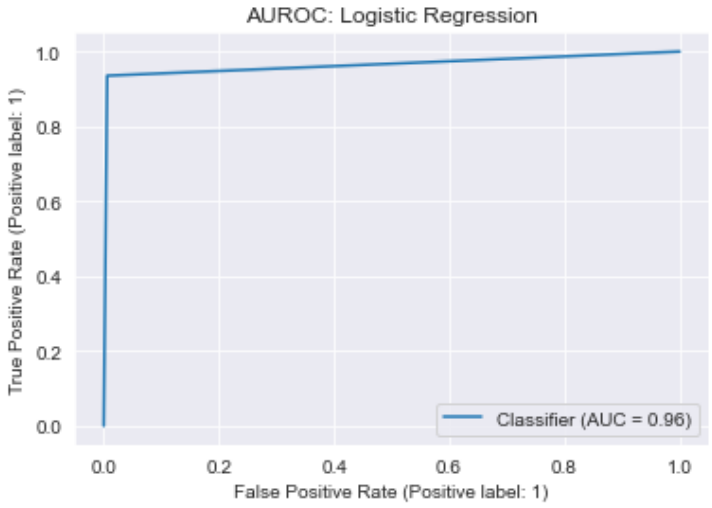


Figure 10. AUROC chart for the logistic regression classifier for family notice, using all features plus the 'title_keyword' feature.

The logistic regression/all features model for poetry also suffered in terms of recall when the 'title_keyword' feature was added, although accuracy and precision improved.[28] The fiction classifier (logistic regression/all features) was slightly improved in terms of precision, which increased from 0.54 to 0.57 (2 d.p.) while the other metrics were unaffected. The performance of the SVM/all features model for letter to the editor deteriorated when the 'title_keyword' feature was included.

The logistic regression models provide the opportunity to extract the coefficients, which are converted from log odds to odds, with charts of the top five positive and negative coefficients produced. Figure 11 displays the coefficient charts for the poetry and family notice classifiers. These results show that the strongest predictor of poetry is the frequency of monosyllabic words. For every unit increase in the frequency of these words, the odds that the article is poetry is more than six times as large as the odds that it is not poetry (Benton, 2020). At the other end of the scale, the frequency of stopwords is the strongest predictor that an article is not poetry. For family notice, the two strongest predictors are the two topic-related features, the title keyword feature and the sum of the top five TF-IDF scores. The strongest negative predictor of family notice is the frequency of verbs (a Universal POS tag set feature). For fiction, the strongest predictor is the frequency of pronouns (from the Universal POS tag set) with odds of over 20, while the strongest negative predictor is the minimum line width. Tables 4, 5, and 6 in Appendix A provide the full list of coefficients converted to odds for the poetry, fiction, and family notice logistic regression models.

It is important to note that while examining the model coefficients provides interesting insights, the influence of correlated features such as the average and maximum line widths must also be

[27] Metrics for the SVM/pos_freq_univ model for family notice with the 'title_keyword' feature included are: 1 accuracy, 0.96 precision, 0.87 recall, 0.91 F1, and 0.94 AUROC (2 d.p.)
[28] Metrics for the logistic regression/all features model for poetry with the 'title_keyword' feature included are are: 1 accuracy, 0.78 precision, 0.70 recall, 0.74 F1, and 0.85 AUROC (2 d.p.)

taken into consideration. Correlated features have similar predictive relationships to the outcome and therefore the sign and value of the coefficients should be interpreted with caution (Bruce, Bruce, & Gedeck, 2020). Scatterplots showing the relationship between pairs of Universal POS features and pairs of page layout features are provided in Appendix B.

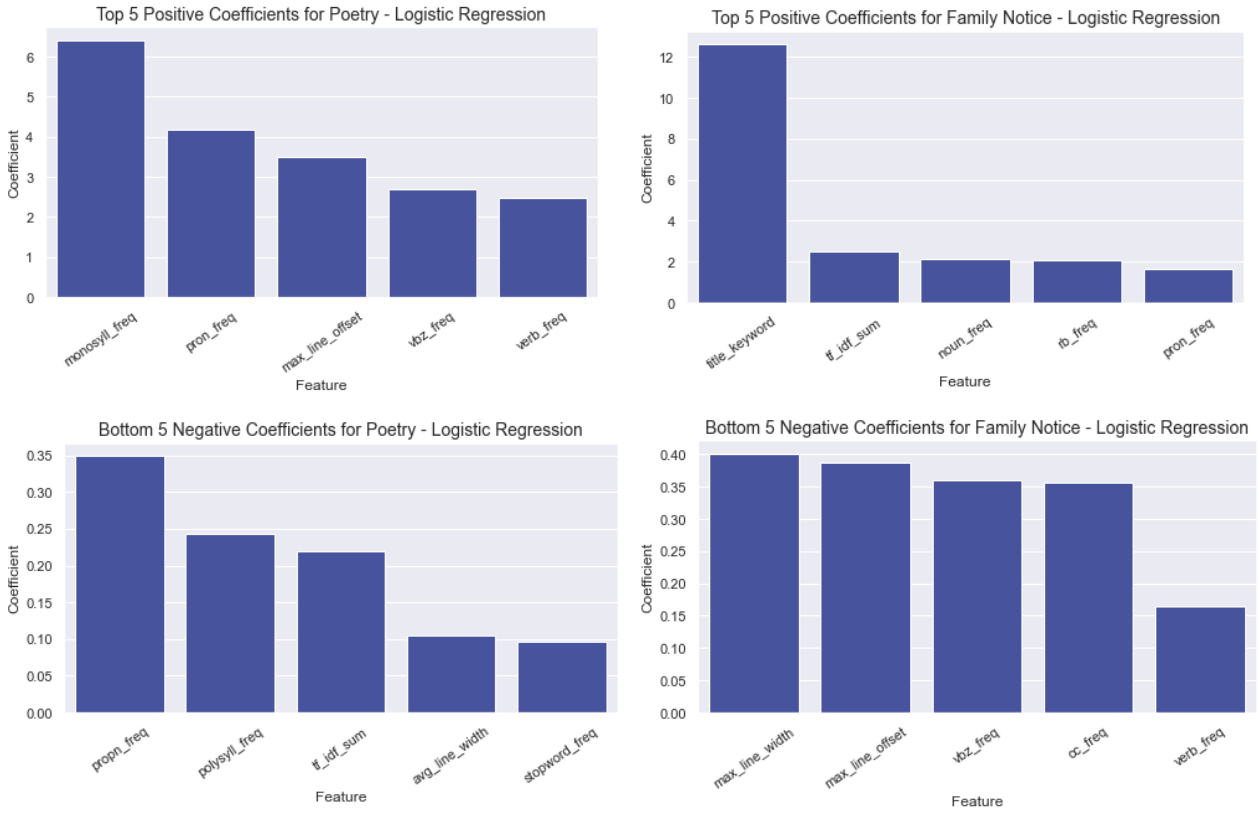


Figure 11. The top five positive and bottom five negative coefficients (odds) for the logistic regression poetry classifier (left) and family notice (right).

End-to-end binary classification notebooks

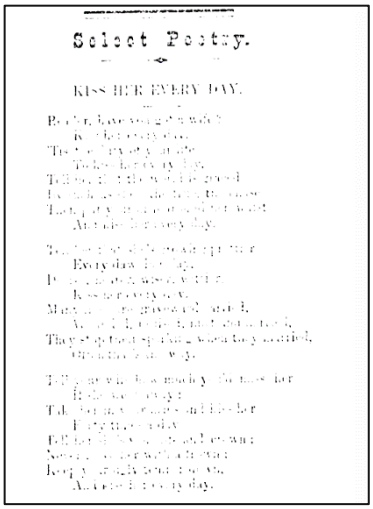
The standalone notebooks for poetry, fiction, family notice, and letter to the editor were used to sample and classify new data from the Papers Past open dataset. Each notebook was used to sample articles from 307 issues (0.1% of the issues in the Papers Past open dataset) using the same random seed. In each instance, a dataframe of 8,016 valid articles was returned with the results sorted by probability of the positive class for the three logistic regression models and by confidence for the SVM model.[29] These dataframes are exported as csv files that can be further explored using spreadsheet software.[30]

The performance of these models on new data is considered in terms of a practical end use case where they might be used by a researcher or librarian to collect examples of the genre in a way that goes beyond the capabilities of the existing keyword search on the Papers Past website. In this

[29] Refer to the 'End-to-end binary classification notebooks' section of the methodology for an explanation of the confidence score for the SVM model.
[30] The csv files are provided as supplementary files: '20220127_NewPoetry_307issues_seed1_df.xlsx', '20220127_NewFiction_307issues_seed1_df.xlsx', '20220127_NewFamilyNotice_307issues_seed1_LR_df.xlsx', '20220127_NewLetterToEd_307issues_seed1_SVM_df.xlsx'

scenario, the classification label assigned to an article by the model is less important than the ability of the model to return a large percentage of articles that are true examples of the genre in the top results when sorted by probability or confidence. It is at the researcher's discretion to determine at which point "scrolling down" becomes cost prohibitive as the number of irrelevant results increases.

For poetry, 645 of the 8,016 articles returned were labelled as the positive class. The top 100 articles were manually examined and of these 69 were correctly classified. In the top 50 articles, 45 were correctly classified as poetry. Some surprising results were achieved, such as the example shown in Figure 12. This example illustrates the influence of the line offset and line width features (and possibly also the frequency of monosyllabic words) on the poetry classifier. The image on the left shows the scanned page, while the one on the right is the OCR text. Note that the title, "Select Poetry" is not included as a feature in this model.



Select Poetry.

I j kl-< ii: i; i-:\ w:\ i 1 \^. i i ! v •... . t: l' ; 1 r Kv. l v !. - ~ l.- r.v •: > v. l ~v, 1 , !!:'H ij M l.'T V'. > 1 .r. • -.n -l '■ . * i" * i : \ ii* u • j i!j: i? i ■ l • til' • > i!; i • j. : lilt ii%. • i. : • \; : k t; . Wi. :• "ɑ:i .u ui'.ii !- 4 ,, '\. . .. : - •'ru 4 ; y, ii.. \■t •. i:l l. i,A 1. l 'iNi.it l'll, \ A- • • > i!i >:.\ ' ■ i.' % • v . V.. : %, , .., ■ 'l !i• !: -• n. », ,!i;. hi . ■ - v. -■' w. ; • • . l; t , (. • . - A:' ■ : : • . • : . 'i.i v.jV. (l. : . l. l <• !..- 1 . !...■ i:..!. : v. >V. '! ; , '; •i. ;;• ;.i i• v\\ i. ; ■ i! ■ • i • :i, .• •• V -t : \! . . l, • i" . 111 - j i i i .. !' i t:'. -' 1 h ; i; . :■■ .■.!_■ ... !i in ;;■' v. ii:ii..w l : i ■ 11 111.-. l 'll tir u:i \. J//

Figure 12. An example of correctly classified (but unreadable) poetry from the 21 June 1895 issue of the Lake Wakatip [sic] *Mail.*

The family notice and fiction classifiers were similarly successful in terms of returning a good number of examples of the genre high-up in the sorted results. For the family notice classifier, 47 of the top 50 results were true family notices, and for fiction 40 of the top 50 results were examples of what a human reader would classify as fiction. The letter to the editor classifier was the least successful on new data, with no results predicted as the positive class. However, in the top 50 results there were still 33 examples of letters to the editor, even though they were not labelled as such by the model.

Multiclass classification

The multiclass classifier was initially trialled using the full set of genres. As can be seen in Table 4, variations of all features or POS feature sets performed best, in combination with support vector machine or random forest classifiers.[31]

[31] The results for all combinations are included in the supplementary file '20220126_PP_3518articles_metrics_multiclass.csv'.

Feature set	Method	Accuracy	Precision	Recall	F1
all_features	Support Vector Machine	0.66	0.65	0.66	0.63
all_features	Random Forest	0.65	0.65	0.65	0.62
all_features_excl_tfidf	Support Vector Machine	0.64	0.64	0.64	0.61
all_features_excl_univ	Random Forest	0.64	0.62	0.64	0.61
all_features_excl_univ	Support Vector Machine	0.64	0.63	0.64	0.61
all_features_excl_tfidf	Random Forest	0.64	0.63	0.64	0.61
pos_freq_combo	Support Vector Machine	0.62	0.61	0.62	0.59
pos_freq_combo	Random Forest	0.61	0.59	0.61	0.58
pos_freq_penn	Support Vector Machine	0.61	0.60	0.61	0.58
all_features_excl_penn	Support Vector Machine	0.61	0.59	0.61	0.58

Table 4. Best performing combination of multiclass classifier and feature set (by F1-score).

Because the dataset is very imbalanced, an alternative approach was explored where similar genres were combined in six labelled groups as shown below:[32]

- 1: Family Notice
- 2: Opinion-based = Letter to Editor, Opinion, Review, Speech
- 3: Reportage-based = Feature, News, Obituary, Report, Results
- 4: Fiction
- 5: Notice
- 6: Poetry

This approach yielded better results, as shown in Table 5.[33] A confusion matrix for the best performing model of SVM using all features is shown in Figure 13, illustrating where the classifier is making mistakes. A significant number of misclassifications occur between the two most represented genres: opinion and reportage. Of the articles with a true label of 'opinion', 25% were misclassified as 'reportage' and 11% of articles labelled 'reportage' were misclassified as 'opinion'. Misclassification between reportage and notice also occurred, with around 20% of notices classified as 'reportage'. Family notices were often misclassified as notices (26%) and reportage (23%).

The confusion between notices, family notices, and reportage is not surprising, as news and results can be short articles similar in form and style to a notice. Also, as described previously in this report, family notice is a sub-genre of notice and the 'title_keyword' feature that was shown to be a strong predictor of family notices in the binary models was not included in this multiclass experiment.

[32] The rationale behind the separation of these groups is described in the 'Multiclass classification – combined genre classes' section of the methodology.
[33] The results for all combinations are included in the supplementary file '20220114_PP_3518articles_metrics_multiclass_combinedgenres.csv'.

Feature set	Method	Accuracy	Precision	Recall	F1
all_features	Support Vector Machine	0.81	0.81	0.81	0.81
all_features_excl_tfidf	Support Vector Machine	0.80	0.80	0.80	0.80
all_features_excl_univ	Support Vector Machine	0.80	0.80	0.80	0.79
all_features_excl_tfidf	Random Forest	0.79	0.79	0.79	0.79
all_features	Random Forest	0.79	0.79	0.79	0.78
all_features_excl_univ	Random Forest	0.78	0.79	0.78	0.78
all_features_excl_penn	Support Vector Machine	0.78	0.78	0.78	0.78
all_features	K Nearest Neighbor	0.78	0.78	0.78	0.77
pos_freq_combo	Random Forest	0.78	0.77	0.78	0.77
all_features_excl_tfidf	K Nearest Neighbor	0.77	0.78	0.77	0.77

Table 5. Best performing combination of multiclass classifier and feature set for combined genre groups (by F1-score).

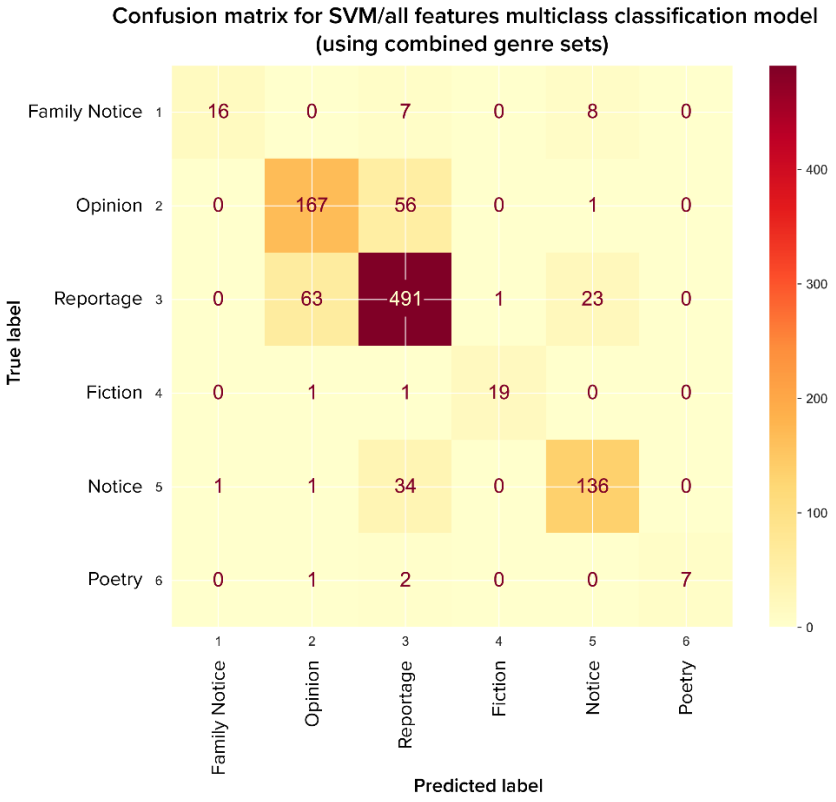


Figure 13. Confusion matrix for the best-performing multiclass classifier (SVM/all features) for the combined genre groups.

6 Discussion

This project has shown that it is possible to use machine learning algorithms to identify the genre of articles in the Papers Past open dataset with good levels of recall. Researchers in fields such as literary history or genealogy can use the models to discover examples of genres such as poetry, fiction, and family notices in a way that overcomes the limitations of a keyword search. The following discussion addresses the three research questions presented at the start of this report.

What combination of features and machine learning algorithm(s) perform best in classifying a defined set of genres for the Papers Past data?

This project has shown that there is no one-size-fits-all combination of features and machine learning algorithm that performs best for all genres. Labelling and feature selection are critical components of the successful model and the transparent pipeline developed in this project enables further analysis and refinement of these areas. When ranked by AUROC, logistic regression and support vector machine were consistently the best performing algorithms across all the genres, in combination with variations of all features or POS feature sets.

There is a varying relationship between topic and genre, shown in this study by the effect of the two topic-related features: TF-IDF scores and title keyword. For a genre such as family notice, where the topic is embedded in the genre's definition, the topic-related features were important predictors. For other genres such as poetry, fiction, and letter to the editor, the effect of the topic-related features was either insignificant or they impaired the model's performance in terms of recall. As described in the literature review, different researchers and domains have different concepts of genre and it would be difficult to build a model that serves everyone. A benefit of the transparent pipeline developed for this study is that it provides a platform that can be readily adapted to suit different domains and use-cases.

Can the model be shown to be robust across topic, text length, time, and newspaper?

As mentioned previously, the only binary classification model that was significantly improved by the introduction of a topic-based feature (the title keywords feature) was family notice. This is encouraging and suggests the performance of the other models is relatively robust across topic.

Features related to year of publication and newspaper title were not included in the models, and the trials of the classifiers on new data show results from across the range of newspapers and the time period of the dataset. In terms of text length, features such as sentence count, word count, and character count were included as features and the results show that they can be useful predictors for some genres. Figure 14 shows the distribution of article word count by genre in the labelled dataset of 3,518 articles.

A true test of the stability and robustness of these models would be an experiment on a completely new dataset of digitised newspaper articles that uses the same METS/ALTO XML format as the Papers Past open data.

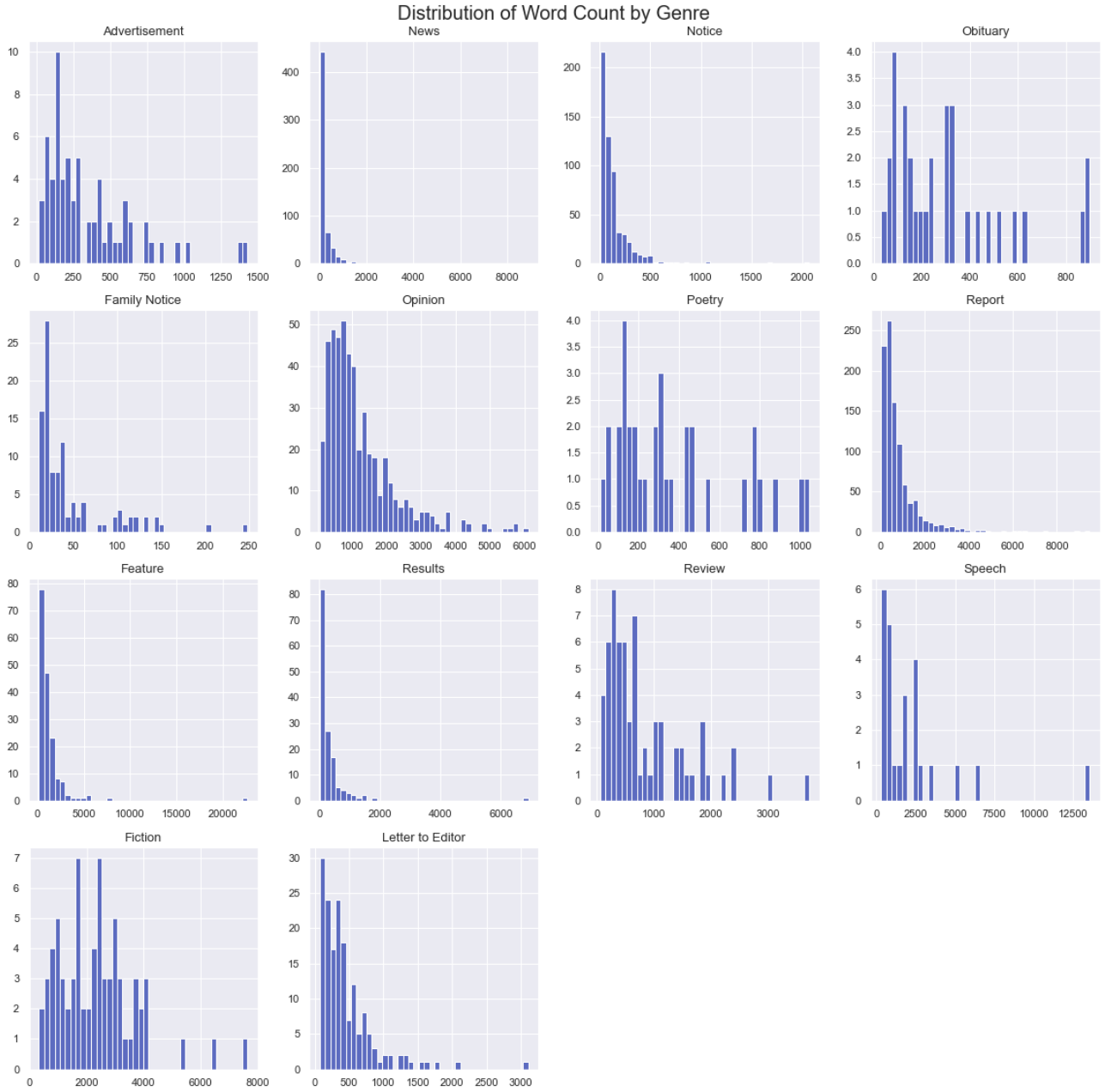


Figure 14. Distributions of article word count by genre in the labelled dataset.

How can the genre classification pipeline be documented and shared to best enable other users to easily apply, adapt, and interpret its performance on other datasets?

The transparency and usability of the genre classification pipeline were key considerations throughout the development of this project. A series of notebooks, each implementing a distinct phase of the pipeline, has been created. Preprocessing, labelling, feature extraction, and model development are presented as distinct steps with tangible inputs and outputs in the form of Pandas dataframes. This process allows phases of the pipeline to be modified and adjusted in manageable chunks to suit different requirements or datasets. The end-to-end notebooks can be easily run by a user with limited understanding of programming and require only four inputs: a random seed (if required), the directory where the tar.gz files are saved in newspaper/year combinations, the number of newspaper issues to sample, and a filename for the final exported csv file.

Other studies, such as the work of Kilner and Fitch (2017), have highlighted the motivation and value of using automated methods to uncover examples of creative writing in historic newspaper archives and the fiction and poetry classifiers developed here enable similar exploration of the Papers Past open data. A use-case example is a literary historian interested in finding fiction printed only in a particular newspaper. They can save the tar.gz files for that newspaper in a directory folder and use the end-to-end notebook for fiction to sample issues of the newspaper, run the classification model, and return the dataframe of sorted results as a csv file. The final dataframe includes the full article text and links to the online version of the newspaper issue where the scanned version of the article can be viewed. The results can be further analysed or refined using the researcher's preferred method and tools.

In terms of transparency of the pipeline and interpretability of results, particularly for non-data scientists, the success of the logistic regression models was an encouraging and pleasing outcome. As noted by Bilgin et al. (2018) uncovering insights about the influence of certain features in classifying a genre can help domain experts to, "bridge the gap between their abstract understanding of genres and the concrete features needed to automatically classify them". The fiction and poetry classifiers showed that these genres have non-topic related features that are strong predictors for genre classification. In the case of poetry, the strength of the frequency of monosyllabic words as a feature of the genre is consistent with findings from research dating back as far as 1921 (Sturtevant, 1921).

Future work

The methodology and results outlined in this report can be considered a proof-of-concept and there are many opportunities for the pipeline to be refined and the results of the classification models improved. Ideas for improvement and future work include:

- Explore common misclassifications across the existing genres and use these to develop alternative genre definitions to overcome these problems. For example, a narrative genre could capture non-fictional stories or accounts that are currently misclassified as fiction.[34]
- Refine the existing feature sets and try alternatives. For example, investigate and address correlated variables, try different values of 'n' for the sum of top-n TF-IDF scores, and explore new features such as frequencies of named entities.
- Further explore the possibilities of using the spatial and structural data provided in the METS/ALTO XML files as features. For example, extend the single line width and offset features used in this project to examine the effect of block indentations such as the frequencies of indented to non-indented blocks of text.
- Use the best performing models for the underrepresented genres to sample a larger number of examples that can be used to create a more balanced dataset. Use this balanced dataset to train and test updated binary and multiclass classification models.
- Explore the impact of different pre-processing and feature selection steps such as alternative scaling and normalisation techniques, recursive feature elimination, and principal component analysis (PCA) (Brownlee, 2021).
- Tune the hyperparameters of selected binary and multiclass classifiers (Brownlee, 2021).

[34] In examples such as historical narrative, determining fiction versus non-fiction requires human interpretation.

- Experiment with the pipeline on a different dataset of newspapers that have been digitised in a similar way using the METS/ALTO standards.
- Combine the end-to-end notebooks in a locally-hosted web application with a straightforward user interface for required inputs such as genre selection, number of issues to sample, and output filename. The app could be bundled with all the necessary files, such as the saved models, to further simplify the process for a non-technical user.

7 Conclusion

This project has addressed questions of which combination of features and machine learning algorithm perform best in classifying genres in the Papers Past open data and if the best performing models can be shown to be robust across topic, text length, time, and newspaper. The results presented in this report have shown that classification models can be developed to successfully identify specific genres within the dataset. Different feature sets using POS frequencies, page layout information, text features and statistics were developed and trialled in combination with the defined set of genres. The classification of poetry and fiction using binary classification models of logistic regression with all features was particularly successful. The family notice genre was found to benefit significantly from the inclusion of a topic-related feature from the article's title, and this was acknowledged to be acceptable and expected for that genre.

The genre classification pipeline is documented in a series of Jupyter notebooks that allow other researchers to easily apply and build on the results of this project. In addition, stand-alone notebooks were developed for poetry, fiction, family notices, and letters to the editor, allowing the pipeline to be run end-to-end to discover examples of those genres in the Papers Past open dataset.

This project has shown that genre classification of articles in digitised historical newspapers is an effective way to find examples of genres that may otherwise go undiscovered. It is hoped that this work opens the door for more researchers to discover the hidden treasures contained within the National Library of New Zealand's Papers Past open data.

References

Allen, R. B., Waldstein, I., & Zhu, W. (2008). Automated Processing of Digitized Historical Newspapers: Identification of Segments and Genres. In G. Buchanan, M. Masoodian, & S. J. Cunningham (Ed.), *Digital Libraries: Universal and Ubiquitous Access to Information* (pp. 379-386). Berlin Heidelberg: Springer-Verlag.

Benton, J. (2020, June 30). *Interpreting Coefficients in Linear and Logistic Regression*. Retrieved January 27, 2022, from Towards Data Science: https://towardsdatascience.com/interpreting-coefficients-in-linear-and-logistic-regression-6ddf1295f6f1

Bilgin, A., Hollink, L., Tjong, E., Sang, K., Smeenk, K., Harbers, F., & Broersma, M. (2018). Utilizing a Transparency-driven Environment toward Trusted Automatic Genre Classification: A Case Study in Journalism History. *IEEE 14th International Conference on e-Science*, (pp. 486-496).

Black, J. (2021, November 21). *Newspaper Philosophy Methods*. Retrieved November 2021, from Github: https://github.com/JoshuaDavidBlack/newspaper-philosophy-methods/blob/main/Notebooks/Preprocessing%20Stage.ipynb

Black, J. (2021). *Using Data Science Methods to Investigate Philosophical Discourse in Early New Zealand Newspapers*. DATA601 Project Report, University of Canterbury, UC Arts Digital Lab, Christchurch.

Broersma, M., & Harbers, F. (2016). Exploring Machine Learning to Study the Long-Term Transformation of News. *Digital Journalism, 6*(9), 1150-1164.

Brownlee, J. (2021). *Machine Learning Mastery with Python*. Jason Brownlee.

Bruce, P., Bruce, A., & Gedeck, P. (2020). *Practical Statistics for Data Scientists* (Second ed.). Sebastopol, California, United States of America: O'Reilly Media.

Bucco, R. (2021, July 23). *Python - Using TF-IDF to summarise dataframe text column*. Retrieved January 12, 2022, from stackoverflow: https://stackoverflow.com/questions/68459166/python-using-tf-idf-to-summarise-dataframe-text-column

Corrales, D. C., Ledezma, A., & Corrales, J. C. (n.d.). From Theory to Practice: A Data Quality Framework for Classification Tasks. *Symmetry, 10*.

Dewitt, A. (2015). Advances in the Visualization of Data: The Network of Genre in the Victorian Periodical Press. *Victorian Periodicals Review, 48*(2), 161-182.

Feldman, S., Marin, M. A., Ostendorf, M., & Gupta, M. R. (2009). Part-of-Speech Histograms for Genre Classification of Text. *IEEE International Conference on Acoustics, Speech and Signal Processing*.

Finn, A., & Kushmerick, N. (2006). Learning to Classify Documents According to Genre. *Journal of the American Society for Information Science and Technology, 57*(11), 1506-1518.

Garrido, A. L., Bobed, C., Cardiel, O., Aleyxendri, A., & Quilez, R. (2017). Optimization in Extractive Summarization Processes Through Automatic Classification. In A. Gelbukh (Ed.), *Computational Linguistics and Intelligent Text Processing. CICLing 2017. Lecture Notes in Computer Science* (Vol. 10762, pp. 506-521). Springer.

Gorlach, M., & Gorlach, M. (2004). Text Types and the History of English. Retrieved from ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/canterbury/detail.action?docID=325657

Gregory, M. V. (2018). Genre. *Victorian Literature and Culture, 46*(3-4), 715-719.

Holley, R. (2009, March/April). How Good Can it Get? Analysing and Improving OCR Accuracy in Large Scale Historic Newspaper Digitisation Programs. *D-Lib Magazine, 15*(3/4).

Joshi, P. (2015, December 15). *How To Compute Confidence Measure For SVM Classifiers*. Retrieved January 20, 2022, from Perpetual Enigma: https://prateekvjoshi.com/2015/12/15/how-to-compute-confidence-measure-for-svm-classifiers/

Karlgren, J., & Cutting, D. (1994). Recognizing Text Genres with Simple Metrics Using Discriminant Analysis. *arXiv:cmp-lg/9410008*.

Kessler, B., Nunberg, G., & Schutze, H. (1997). Automatic Detection of Text Genre. *Proceedings ACL/EACL* (pp. 32-38). Madrid: arXiv:cmp-lg/9707002.

Kilner, K., & Fitch, K. (2017). Searching for My Lady's Bonnet: Discovering Poetry in the National Library of Australia's newspapers database. *Digital Scholarship in the Humanities, 32*(1).

Kruger, K. R., Lukowiak, A., Sonntag, J., Warzecha, S., & Stede, M. (2017). Classifying news versus opinions in newspapers: Linguistic features for domain independence. *Natural Language Engineering, 23*(5), 687-707.

Lim, C. S., Lee, K. J., & Kim, G. C. (2005). Multiple sets of features for automatic genre classification of web documents. *Information Processing & Management, 41*, 1263-1276.

Lorang, E. M., Soh, L.-K., Datla, M. V., & Kulwicki, S. (2015). Developing an Image-Based Classifier for Detecting Poetic Content in Historic Newspaper Collections. *Faculty Publications, UNL Libraries, 340*.

Manrique-Losada, B., Zapata-Jaramillo, C. M., & Venegas-Velásquez, R. (2019). Applying rhetorical analysis to processing technical documents. *Acta Scientiarum*.

Murphy, S. (2019). Shakespeare and his contemporaries: Designing a genre classification scheme for Early English Books Online 1560–1640. *ICAME Journal, 43*, 59-82.

Nairn, R., McCreanor, T., & Moewaka Barnes, A. (2017). *Mass media representations of indigenous peoples*. Massey University, SHORE & Whariki Research Centre.

Onan, A. (2018). An ensemble scheme based on language function analysis and feature engineering for text genre classification. *Journal of Information Science, 44*(1), 28-47.

Peters, R. (n.d.). *Classification is Easy with SciKit's Logistic Regression*. Retrieved January 25, 2022, from Sweetcode: https://sweetcode.io/easy-scikit-logistic-regression/

Petrenz, P., & Webber, B. L. (2010). Stable Classification of Text Genres. *Computational Linguistics, 37*(2), 385-393.

Rao, P. (2020, May 2). *Turbo-charge your spaCy NLP pipeline*. Retrieved December 15, 2021, from https://prrao87.github.io/blog/spacy/nlp/performance/2020/05/02/spacy-multiprocess.html

Rauber, A., & Merkl, D. (2003). Text Mining in the SOMLib Digital Library System: The Representation of Topics and Genres. *Applied Intelligence, 18*, 271-293.

Salgado, R. (2019, March 31). *Multiclass Text Classification From Start To Finish*. Retrieved 12 01, 2021, from Medium.com: https://medium.com/@robert.salgado/multiclass-text-classification-from-start-to-finish-f616a8642538

Schulman, A. D., & Barbosa, S. E. (2018). Text Genre Classification using only Parts of Speech. *International Conference on Computational Science and Computational Intelligence (CSCI)*.

scikit-learn developers. (n.d.). *1.4. Support Vector Machines*, 1.0.2. Retrieved January 19, 2022, from scikit-learn: https://scikit-learn.org/stable/modules/svm.html#scores-probabilities

scikit-learn developers. (n.d.). *sklearn.linear_model.LogisticRegression*, scikit-learn 1.0.2. Retrieved December 18, 2021, from scikit-learn.org: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

Sharoff, S. (2018). Functional Text Dimensions for the annotation of web corpora. *Corpora, 13*(1), 65-95.

Stamatatos, E. (2018). Masking Topic-Related Information to Enhance Authorship Attribution. *Journal of the Association for Information Science and Technology, 69*(3), 461-473.

Stone, J. (2012). Historic Oregon Newspapers Online: Bringing Oregon's "first rough draft of history" into a New Era of Public Accessibility. *Oregon Historical Quarterly, 113*(1), 90-104.

Stubbe, A., Ringlstetter, C., & Schulz, K. U. (2007). Genre as noise: noise in genre. *International Journal on Document Analysis and Recognition (IJDAR), 10*, 199-209.

Sturtevant, E. H. (1921, December 19). On the Frequency of Short Words in Verse. *The Classical Weekly, 15*(10), 73-76.

SuperSummary. (n.d.). *Prose and Verse*. Retrieved February 7, 2022, from SuperSummary: https://www.supersummary.com/prose/#prose-and-verse

The National Library of New Zealand. (n.d.). *Copyright and re-use — Papers Past newspaper open data pilot*. Retrieved 11 22, 2021, from National Library: https://natlib.govt.nz/about-us/open-data/papers-past-metadata/papers-past-newspaper-open-data-pilot/copyright-and-re-use-papers-past-newspaper-open-data-pilot

The National Library of New Zealand. (n.d.). *Papers Past newspaper open data pilot*. Retrieved November 15, 2021, from https://natlib.govt.nz/about-us/open-data/papers-past-metadata/papers-past-newspaper-open-data-pilot

Veridian Software. (n.d.). *What is METS/ALTO?* Retrieved December 03, 2021, from veridiansoftware.com: https://veridiansoftware.com/knowledge-base/metsalto/

Webber, B. L. (2011, June). Stable Classification of Text Genres. *Computational Linguistics, 37*(2), 385-393.

Zhang, C., Wu, X., Niu, Z., & Ding, W. (2014). Authorship identification from unstructured texts. *Knowledge-Based Systems*, 99-111.

Appendix A

Column name	Datatype
Date	datetime64[ns]
newspaper_id	string
newspaper	string
article_id	int32
avg_line_width	float64
min_line_width	float64
max_line_width	float64
line_width_range	float64
avg_line_offset	float64
max_line_offset	float64
min_line_offset	float64
title	string
text	string

Table 1. Columns and datatypes of the dataframe returned from Notebook 1.

Column name	Datatype
date	datetime64[ns]
newspaper_id	string
newspaper	string
article_id	int32
avg_line_width	float64
min_line_width	float64
max_line_width	float64
line_width_range	float64
avg_line_offset	float64
max_line_offset	float64
min_line_offset	float64
title	string
text	string
genre	string
sentence_count	int64
clean_text	string
word_count	int64
syll_count	int64
polysyll_count	int64
monosyll_count	int64
stopwords_count	int64
avg_word_length	float64
char_count	int64
propn_count	int64
verb_count	int64
noun_count	int64
adj_count	int64
nums_count	int64
pron_count	int64
nnps_count	int64

vb_count	int64
nn_count	int64
jj_count	int64
cd_count	int64
prp_count	int64
rb_count	int64
cc_count	int64
nnp_count	int64
vbd_count	int64
vbz_count	int64
propn_freq	float64
verb_freq	float64
noun_freq	float64
adj_freq	float64
nums_freq	float64
pron_freq	float64
nnps_freq	float64
vb_freq	float64
nn_freq	float64
jj_freq	float64
cd_freq	float64
prp_freq	float64
rb_freq	float64
cc_freq	float64
nnp_freq	float64
vbd_freq	float64
vbz_freq	float64
polysyll_freq	float64
monosyll_freq	float64
stopword_freq	float64
tf_idf	object
tf_idf_sum	float64

Table 2. Columns and datatypes of the dataframe returned from Notebook 3.

Feature set name	Description	Included features (column names)
pos_freq_penn	Frequencies of the Penn Treebank POS tags and stopwords	"nnps_freq", "vb_freq", "nn_freq", "jj_freq", "cd_freq", "prp_freq", "rb_freq", "cc_freq", "nnp_freq", "vbd_freq", "vbz_freq", "stopword_freq"
pos_freq_univ	Frequencies of the Universal POS tags and stopwords	"propn_freq", "verb_freq", "noun_freq", "adj_freq", "nums_freq", "pron_freq", "stopword_freq"
pos_freq_combo	Frequencies of the combined POS tags and stopwords	"nnps_freq", "vb_freq", "nn_freq", "jj_freq", "cd_freq", "prp_freq", "rb_freq", "cc_freq", "nnp_freq", "vbd_freq", "vbz_freq", "propn_freq", "verb_freq", "noun_freq", "adj_freq", "nums_freq", "pron_freq", "stopword_freq"
line_offsets	Average and maximum offsets of article lines from the article block	"avg_line_offset", "max_line_offset"
line_widths	Average, minimum, maximum and range of article lines	"avg_line_width", "min_line_width", "max_line_width", "line_width_range"
syllable_freq	The frequency of polysyallabic and monosyllabic words	"polysyll_freq", "monosyll_freq"
tf_idf	The sum of TF-IDF scores for the top 'n' words (in this experiment n = 5.	"tf_idf_sum"
text_stats	Basic text statistics: counts of sentences, words, and characters in the article, and the average word length.	"sentence_count", "word_count", "avg_word_length", "char_count"
all_features	All of the above features combined.	"propn_freq", "verb_freq", "noun_freq", "adj_freq", "nums_freq", "pron_freq", "nnps_freq", "vb_freq", "nn_freq", "jj_freq", "cd_freq", "prp_freq", "rb_freq", "cc_freq", "nnp_freq", "vbd_freq", "vbz_freq", "stopword_freq", "avg_line_offset", "max_line_offset", "avg_line_width", "min_line_width", "max_line_width", "line_width_range", "polysyll_freq", "monosyll_freq", "sentence_count", "word_count", "avg_word_length", "char_count", "tf_idf_sum"

Feature set name	Description	Included features (column names)
all_features_excl_penn	All of the above features excluding the Penn Treebank POS frequencies.	"propn_freq", "verb_freq", "noun_freq", "adj_freq", "nums_freq", "pron_freq", "stopword_freq", "avg_line_offset", "max_line_offset", "avg_line_width", "min_line_width", "max_line_width", "line_width_range", "polysyll_freq", "monosyll_freq", "sentence_count", "word_count", "avg_word_length", "char_count", "tf_idf_sum"
all_features_excl_univ	All of the above features excluding the Universal POS frequencies.	"nnps_freq", "vb_freq", "nn_freq", "jj_freq", "cd_freq", "prp_freq", "rb_freq", "cc_freq", "nnp_freq", "vbd_freq", "vbz_freq", "stopword_freq", "avg_line_offset", "max_line_offset", "avg_line_width", "min_line_width", "max_line_width", "line_width_range", "polysyll_freq", "monosyll_freq", "sentence_count", "word_count", "avg_word_length", "char_count", "tf_idf_sum"
all_features_excl_tfidf	All of the above features excluding the TF-IDF sum.	"nnps_freq", "vb_freq", "nn_freq", "jj_freq", "cd_freq", "prp_freq", "rb_freq", "cc_freq", "nnp_freq", "vbd_freq", "vbz_freq", "stopword_freq", "avg_line_offset", "max_line_offset", "avg_line_width", "min_line_width", "max_line_width", "line_width_range", "polysyll_freq", "monosyll_freq", "sentence_count", "word_count", "avg_word_length", "char_count"
title_keyword	This additional binary feature was trialled for the top performing binary classification models and included in the final logistic regression/all_features models for family notice and fiction. The feature is '1' if the article title contains specific keywords (such as "Birth", "Death", or "Marriage" in the case of family notice) and '0' otherwise.	

Table 3. Details of the feature sets.

Feature	Coefficient (odds)
monosyll_freq	6.41247895
pron_freq	4.174933324
max_line_offset	3.481910482
vbz_freq	2.682030798
verb_freq	2.479378157
jj_freq	2.092396442
adj_freq	1.984206675
avg_line_offset	1.968844878
max_line_width	1.965381585
noun_freq	1.689472256
nn_freq	1.63047326
vb_freq	1.614339324
line_width_range	1.264793403
char_count	1.183893473
cc_freq	1.171517149
min_line_width	1.159877501
word_count	1.084401236
rb_freq	1.066156738
vbd_freq	0.970907733
prp_freq	0.762873471
sentence_count	0.705812822
avg_word_length	0.599392689
nnps_freq	0.595108609
nums_freq	0.403754017
cd_freq	0.403751144
nnp_freq	0.354776712
propn_freq	0.349383444
polysyll_freq	0.243574297
tf_idf_sum	0.219446991
avg_line_width	0.104529094
stopword_freq	0.095960146

Table 4. Coefficients for the logistic regression/all features poetry classifier converted from log odds to odds. Odds less than 1 are negative coefficients (the strongest predictors that the article is not poetry).

Feature	Coefficient (odds)
pron_freq	20.5423609
line_width_range	8.508024966
adj_freq	6.543803942
avg_word_length	3.720444368
vbd_freq	3.503639818
noun_freq	3.334691236
word_count	3.260950172
nn_freq	2.995001466
verb_freq	2.767612728
cc_freq	2.711279337
vb_freq	2.191561893
rb_freq	1.959550393
char_count	1.583284787
vbz_freq	1.536636689
monosyll_freq	1.519937981
title_keyword	1.370164455
stopword_freq	1.314435901
prp_freq	0.959713498
nnp_freq	0.958853072
propn_freq	0.921956394
tf_idf_sum	0.920654946
avg_line_width	0.73550071
max_line_width	0.545609712
nums_freq	0.543566543
cd_freq	0.543478042
nnps_freq	0.521478781
max_line_offset	0.498434174
jj_freq	0.272122422
sentence_count	0.270583458
avg_line_offset	0.252651624
polysyll_freq	0.187301187
min_line_width	0.057943637

Table 5. Coefficients for the logistic regression/all features fiction classifier converted from log odds to odds. Odds less than 1 are negative coefficients (the strongest predictors that the article is not fiction).

Feature	Coefficient (odds)
title_keyword	12.62190817
tf_idf_sum	2.52834888
noun_freq	2.128843421
rb_freq	2.05100191
pron_freq	1.634255605
polysyll_freq	1.628437519
jj_freq	1.562806735
monosyll_freq	1.493111633
vb_freq	1.418123117
nnps_freq	1.31864837
adj_freq	1.268194399
propn_freq	1.189756922
nnp_freq	1.170169487
vbd_freq	1.105016905
min_line_width	0.858043615
stopword_freq	0.857282545
avg_line_offset	0.845995312
cd_freq	0.740175822
nums_freq	0.740175756
line_width_range	0.693306935
word_count	0.592859862
char_count	0.588005521
prp_freq	0.577863871
sentence_count	0.546618088
nn_freq	0.504553497
avg_word_length	0.502000462
avg_line_width	0.41928355
max_line_width	0.400753411
max_line_offset	0.38675269
vbz_freq	0.359757456
cc_freq	0.356022714
verb_freq	0.164087595

Table 6. Coefficients for the logistic regression/all features family notice classifier converted from log odds to odds. Odds less than 1 are negative coefficients (the strongest predictors that the article is not a family notice).

Appendix B

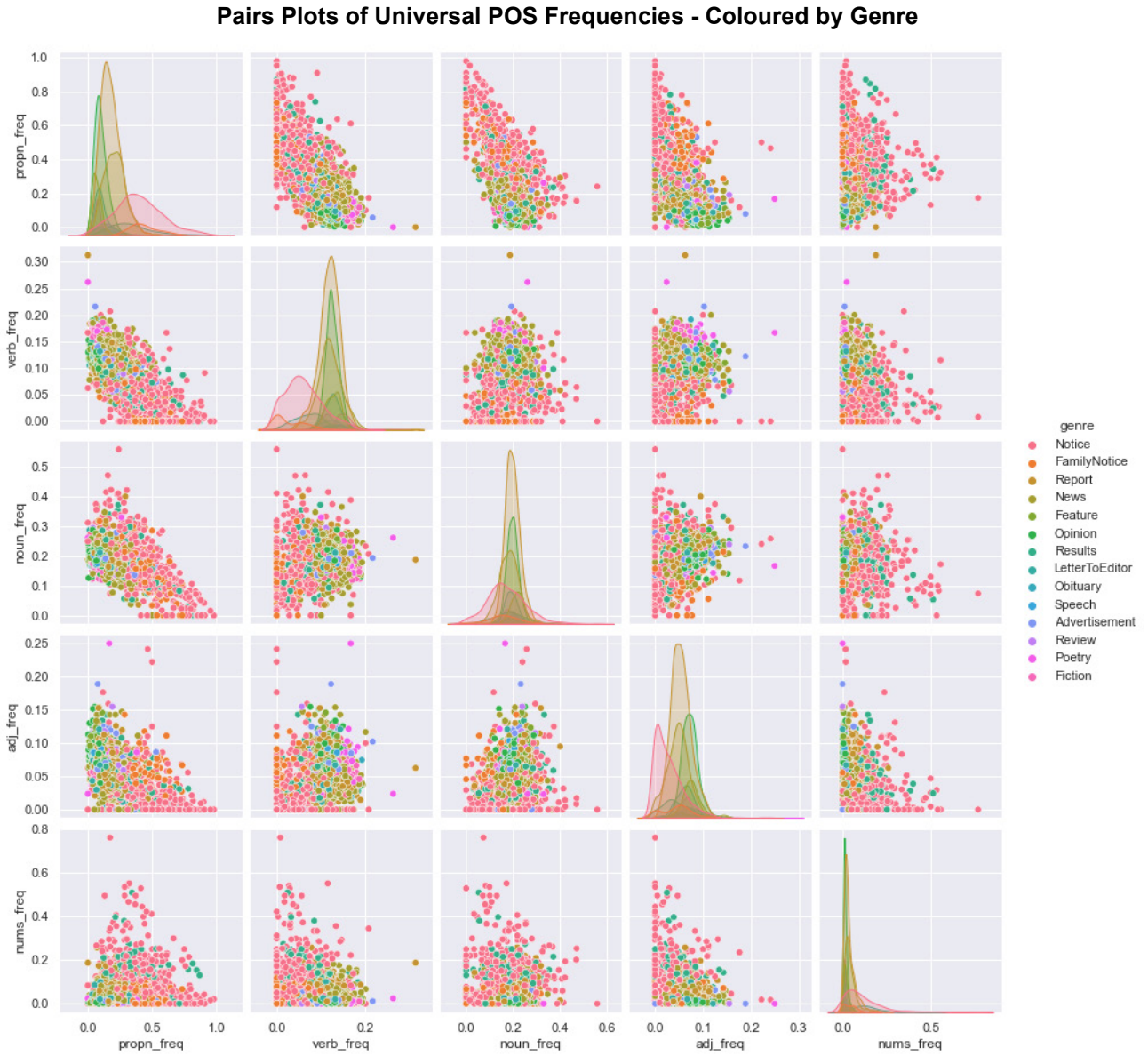


Figure 1. Scatterplots of pairs of Universal POS frequencies in the labelled dataset, coloured by genre.

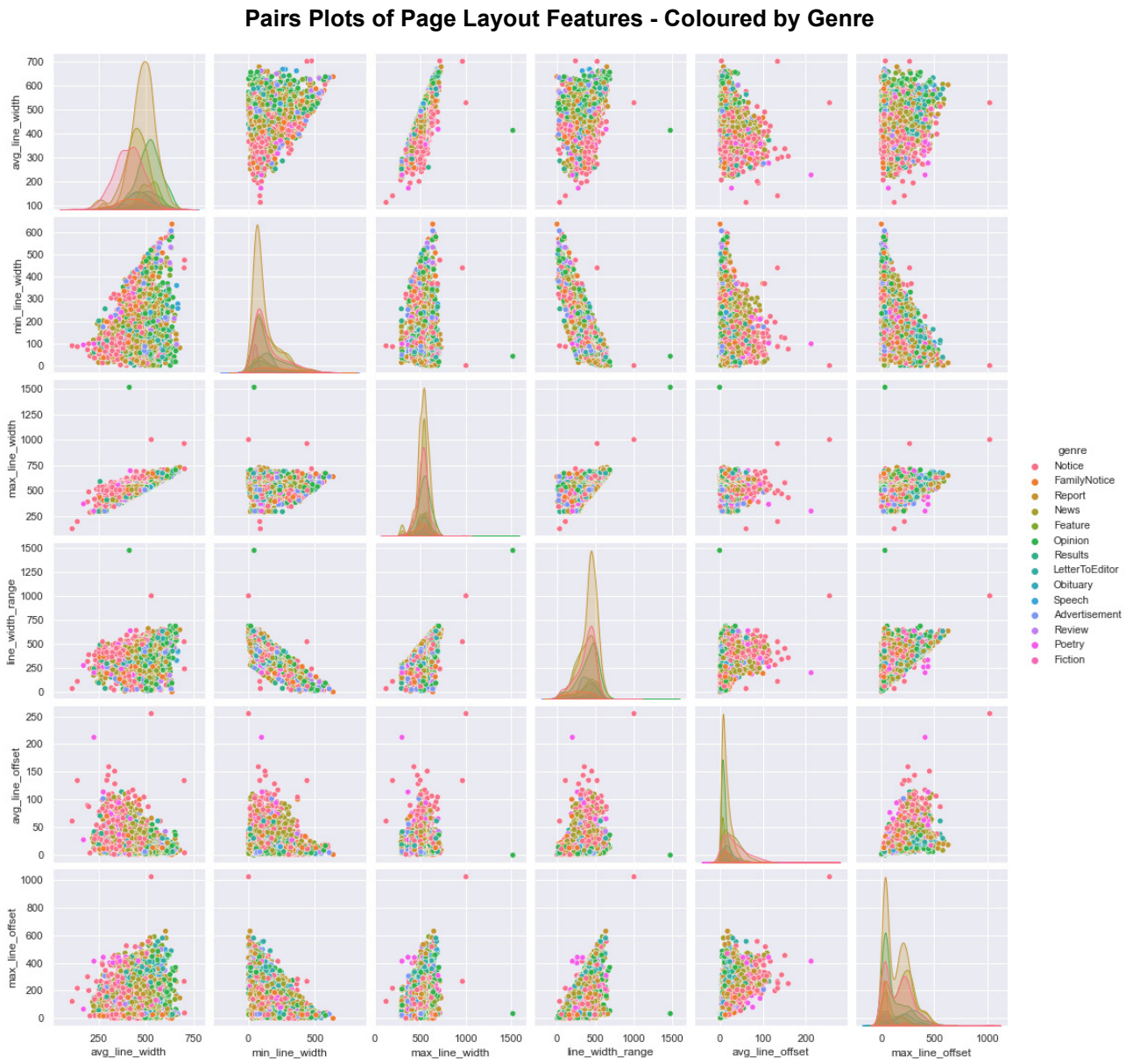


Figure 2. Scatterplots of pairs of page layout features in the labelled dataset, coloured by genre.