University of Glasgow | School of Computing Science

Honours Individual Project Dissertation

Multilingual News Collection and Classification

Adam Fairlie

January 30, 2023

Abstract

Every abstract follows a similar pattern. Motivate; set aims; describe work; explain results.

"XYZ is bad. This project investigated ABC to determine if it was better. ABC used XXX and YYY to implement ZZZ. This is particularly interesting as XXX and YYY have never been used together. It was found that ABC was 20% better than XYZ, though it caused rabies in half of subjects."

Education Use Consent

Contents

1 Introduction **1**
1.1 Motivations and Aims 1
1.2 Chapter outline 1

2 Background and Research **2**
2.1 Digital News Surveillance Systems 2
2.2 Web Scraping Technologies 2
2.3 Database and Visualisation 4
2.4 News Article Classification 4
2.4.1 Datasets 4
2.4.2 Models 4
2.5 Web Interface 4
2.6 Ethical / Legal Considerations 4

3 Requirements **5**
3.1 Functional Requirements 5
3.2 Non-functional Requirements 5

4 Implementation **6**
4.1 Design Choices 6
4.1.1 Web Scraping Technology 6
4.1.2 Classifier Model 6
4.1.3 Visualisation 6
4.1.4 Web Interface 6
4.2 Web Scraping System 6
4.3 Database 6
4.4 Visualisations 6
4.5 News Article Classification 6
4.5.1 Research Questions 6
4.5.2 Data Collection 6
4.5.3 Model and Parameter Choices 6
4.5.4 Data Transformations 7
4.5.5 Evaluation 7
4.6 Web Interface 7

5 Evaluation **8**
5.1 Automatic Web Scraping System 8
5.2 News Article Classification 8
5.3 Web Interface and Visualisations 8
5.3.1 Accessibility 8
5.3.2 Usability Testing 8
5.4 Project Limitations 8

6 Conclusion **9**
6.1 Project Summary 9
6.2 Future Work 9

Appendices **10**

Bibliography **10**

1 Introduction

Why should the reader care about what are you doing and what are you actually doing?

1.1 Motivations and Aims

Motivate first, then state the general problem clearly.

1.2 Chapter outline

2 Background and Research

What did other people do, and how is it relevant to what you want to do?

2.1 Digital News Surveillance Systems

BioCaster

This project reworks and extends some of the digital news surveillance system currently used in BioCaster (2023). The system collects news articles through a variety of RSS, feeds using a Perl script, across different languages including English, French and Mandarin Chinese (Collier et al. 2008). It uses a machine translation server to translate the headlines and descriptions into English, and a machine learning model built on PubMedBERT (Gu et al. 2021) to classify whether news articles are related to disease outbreaks (Meng et al. 2022). It also uses text mining approaches with BioSyn (Sung et al. 2020) to perform entity extraction and linking, in order to determine details of the disease outbreak such as the disease type and who it is currently affecting. It also uses Early Aberration and Reporting System (EARS) algorithms, will classify the risk level of the outbreak (Collier 2011).

The later stages of this process are outside the scope of this project. The current BioCaster data collection system is dated, relying on PERL and only collecting data through RSS feeds, which are only provided by some (typically larger, more common language) news sources. This greatly limits the amount of data which can be extracted, particularly from smaller sources in minority languages, which are underrepresented in disease tracking systems (citation needed). This project will develop a more modern system for collecting the RSS data in Python, as well as functionality for using web scraping to collect news articles from webpage HTML. The system will be designed so that it is easily extensible to other sources of data which may be more prominent in different countries, such as social media posts. This also allows for entire article text to be collected instead of just the headline and a short description, which may allow for a richer understanding of the news content and more effective performance in ML tasks.

Other BioMedical Systems

HealthMap, GPHIN, ProMED Mail, MedISys, Argus, EpiSPIDE, EpiCore. Web crawler with keyword filtering (Zhang et al. 2009)

Other Systems

Meltwater, Cision, Muck Rack, Agility PR Solutions (Marketing, brand management/PR etc.).

2.2 Web Scraping Technologies

The main free and open-source technologies for scraping news articles from websites in Python were Newspaper3K and news-please, which is built on top of Newspaper3K and adds some extra

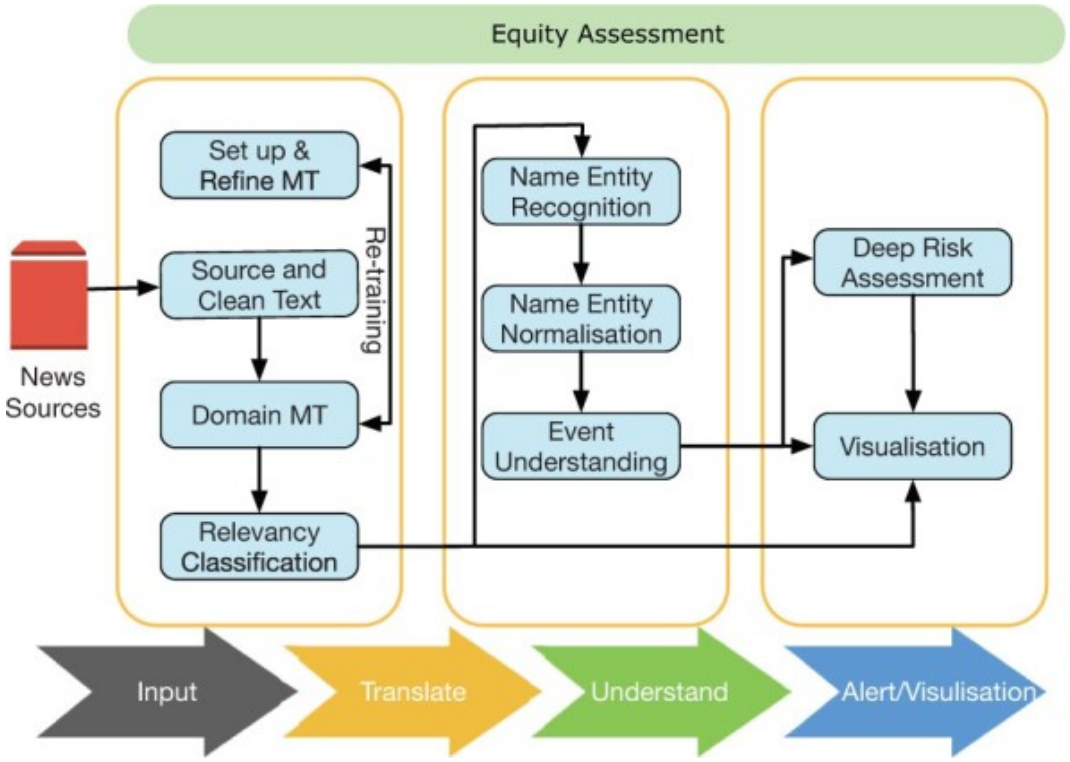

Figure 2.1: *The full architecture of the current BioCaster app. Image obtained from Meng et al. (2022)*

features (Ou-Yang 2020; Hamborg 2022). Another option considered was Newscatcher (Bugara and Sugonyaka 2023), but the free API is limited in how many calls can be made and this made it unsuitable for this project. Finally, I looked at pygooglenews (Bugara 2021) which provided some promise in using google news to find articles under certain subjects, keywords, languages and regions. For this project, I found it desirable to have better control of the exact sources collected, instead of filtering through keywords and relying on Google's source selection, but a scraper using this library could easily be added to extend the capabilities of the current system. I decided to move forward with the former two libraries and conducted an experiment to compare their capabilities.

I compared the features present in each of the two libraries. Notably, Newspaper3K can perform full website scraping in Python, whereas news-please can only do this using its Command Line Interface (CLI). I attempted to scrape 3 articles from each of the 109 previously selected websites, across 10 languages, and compared the number of successful scrapes (without error) and the average speed. The results are shown below:

(Results table)

Newspaper3K scraped 103 of the 109 websites (94.5%) without error, whereas news-please scraped 102/109 (93.58%). The average scraping times are similar in both libraries, but Newspaper3K was faster at scraping in 8 of the 10 languages, and average scraping time per article was 14.82% lower, Based on these factors, I chose to use Newspaper3K for the scraping system.

2.3 Database and Visualisation

The visualisation was mainly inspired by the current BioCaster (2023) interface, available on their website. This design had to be updated to reflect the changes from my project to the current system, including the addition of news article topics and different source types of information retrieved. I also did not have the specific region or disease information to create the alerts seen on the BioCaster interface. In addition, I chose to change the article density to cover the whole country, to make the source of the data more visible (so that it does not obstruct the view of other countries) and created a monochromatic green colour map, where darker colours represented more articles retrieved as is common practice in world density maps (Our World in Data 2022; Office for National Statistics 2022).

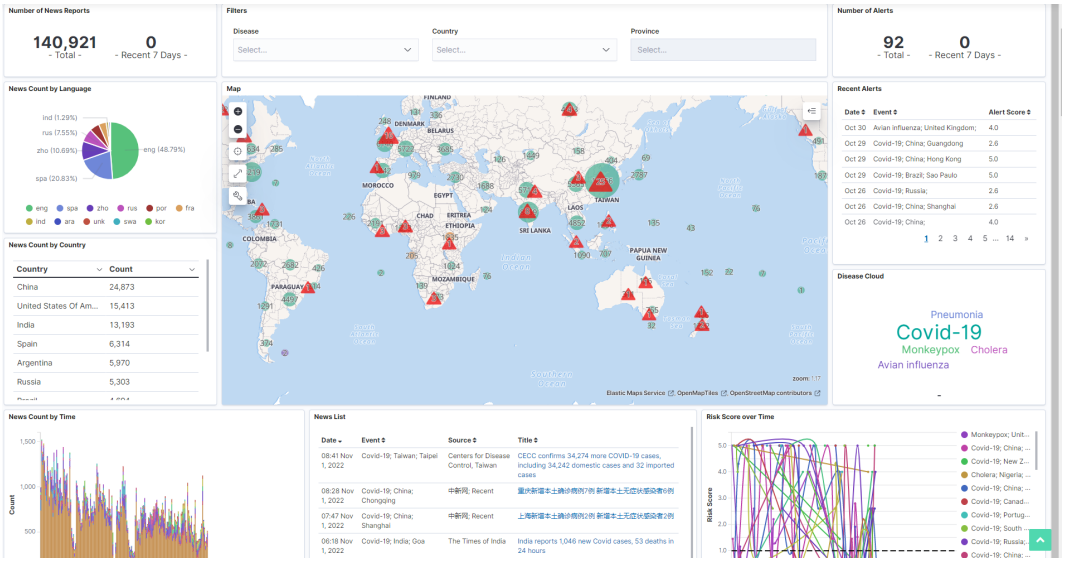

Figure 2.2: BioCaster Kibana Visualisation (from the homepage)

The BioCaster visualisations are created using the Elastic stack, storing the data with elasticsearch and creating visualisations using Elastic Kibana (Elastic 2023). After originally using MySQL with the Python MySQL connector to store the data from the scraping system, and considering standard python visualisation libraries such as matplotlib (Hunter 2007) and geoplotlib (Cuttone 2019) for geographical data, I decided to also use the Elastic stack, using the elasticsearch Python library to integrate the scraping system into the database.

2.4 News Article Classification

2.4.1 Datasets

2.4.2 Models

2.5 Web Interface

2.6 Ethical / Legal Considerations

3 Requirements

3.1 Functional Requirements

3.2 Non-functional Requirements

4 Implementation

4.1 Design Choices

4.1.1 Web Scraping Technology

4.1.2 Classifier Model

4.1.3 Visualisation

4.1.4 Web Interface

4.2 Web Scraping System

4.3 Database

4.4 Visualisations

4.5 News Article Classification

4.5.1 Research Questions

The research question I aim to answer is *"Are multilingual models effective for multilingual multi-topic news classification?"* and *"What is the most effective model in this setting?"*

4.5.2 Data Collection

Data will be collected through the scraping system previously developed, using articles which have been labelled by the source as one of 6 categories: Health, Sports, Entertainment, Business/Finance, Politics and Technology. For some sources, categories with similar names (e.g. "Wellness" and "Health") have been merged. A full list of sources and category keywords used can be found in the appendices.

4.5.3 Model and Parameter Choices

Previous research for monolingual research has shown that Multinomial Naive Bayes models have been effective in multi-topic news classification. I will optimise the alpha (smoothing) parameter through a randomised grid search. I will compare this model to a fine-tuned uncased BERT model. For multilingual classification, I will compare 2 fine-tuned BERT-based models: Multilingual uncased BERT and XML-RoBERTa.

4.5.4 Data Transformations

In the case of the monolingual models, sentences will be translated into English before classification. As is common in article classification research, the article headline and body will be concatenated into one column. In the Naive Bayes model, words will be tokenised and stemmed, and tokens will be converted into TF-IDF feature vectors. In the deep learning models (Monolingual/multilingual bert, XML-RoBERTa) The text will be converted into word embeddings by the BERT tokeniser. In all cases, data will be truncated to 256 tokens, and padded if necessary.

4.5.5 Evaluation

The performance of each model will be evaluated on its accuracy and micro-f1 score. Confusion matrices for each model type will also be shown in the appendices. I will use a most frequent dummy classifier as a baseline for model effectiveness.

4.6 Web Interface

5 Evaluation

5.1 Automatic Web Scraping System

5.2 News Article Classification

5.3 Web Interface and Visualisations

5.3.1 Accessibility

5.3.2 Usability Testing

5.4 Project Limitations

6 Conclusion

Summarise the whole project for a lazy reader who didn't read the rest (e.g. a prize-awarding committee).

6.1 Project Summary

6.2 Future Work

6 Bibliography

BioCaster. Biocaster, 2023. URL http://biocaster.org/.

A. Bugara. pygooglenews, 2021. URL https://github.com/kotartemiy/pygooglenews.

A. Bugara and M. Sugonyaka. Newscatcher news api, 2023. URL https://newscatcherapi.com/news-api.

N. Collier. Towards cross-lingual alerting for bursty epidemic events. *Journal of Biomedical Semantics*, 2(5):1–11, 2011.

N. Collier, S. Doan, A. Kawazoe, R. M. Goodwin, M. Conway, Y. Tateno, Q.-H. Ngo, D. Dien, A. Kawtrakul, K. Takeuchi, et al. Biocaster: detecting public health rumors with a web-based text mining system. *Bioinformatics*, 24(24):2940–2941, 2008.

A. Cuttone. geoplotlib, 2019. URL https://github.com/andrea-cuttone/geoplotlib.

Elastic. Elastic stack: Elasticsearch, kibana, beats logstash | elastic, 2023. URL https://www.elastic.co/elastic-stack/.

Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, and H. Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23, 2021.

F. Hamborg. news-please, 2022. URL https://github.com/fhamborg/news-please.

J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3): 90–95, 2007. doi: 10.1109/MCSE.2007.55.

Z. Meng, A. Okhmatovskaia, M. Polleri, Y. Shen, G. Powell, Z. Fu, I. Ganser, M. Zhang, N. B. King, D. Buckeridge, et al. Biocaster in 2021: automatic disease outbreaks detection from global news media. *Bioinformatics*, 38(18):4446–4448, 2022.

Office for National Statistics. Population density - census maps, ons, 2022. URL https://www.ons.gov.uk/census/maps/choropleth/population/population-density/population-density/persons-per-square-kilometre.

L. Ou-Yang. Newspaper3k: Article scraping curation, 2020. URL https://github.com/codelucas/newspaper.

Our World in Data. Population density, 2022, 2022. URL https://ourworldindata.org/grapher/population-density.

M. Sung, H. Jeon, J. Lee, and J. Kang. Biomedical entity representations with synonym marginalization. *arXiv preprint arXiv:2005.00239*, 2020.

Y. Zhang, Y. Dang, H. Chen, M. Thurmond, and C. Larson. Automatic online news monitoring and classification for syndromic surveillance. *Decision Support Systems*, 47(4):508–517, 2009.