ISSN 2161-3974
eScience in Action

Journal of eScience Librarianship 12(1): e626
DOI: https://doi.org/10.7191/jeslib.626

Journal of eScience Librarianship
putting the pieces together: theory and practice

The Open Science of Deep Learning: Three Case Studies

Chreston Miller, Virginia Tech, Blacksburg, VA, US, chmille3@vt.edu
Leah Hamilton, Virginia Tech, Blacksburg, VA, US, hleah@vt.edu
Jacob Lahne, Virginia Tech, Blacksburg, VA, US, jlahne@vt.edu

Abstract

Objective: An area of research in which open science may have particularly high impact is in deep learning (DL), where researchers have developed many algorithms to solve challenging problems, but others may have difficulty in replicating results and applying these algorithms. In response, some researchers have begun to open up DL research by making their resources available (e.g., code, datasets and/or pre-trained models) to the research community. This article describes three case studies in DL where openly available resources are used and we investigate the impact on the projects, the outcomes, and make recommendations for what to focus on when making DL resources available.

Methods: Each case study represents a single project using openly available DL resources for a research project. The process and progress of each case study is recorded along with aspects such as approaches taken, documentation of openly available resources, and researchers' experience with the openly available resources. The case studies are in multiple-document text summarization, optical character recognition (OCR) of thousands of text documents, and identifying unique language descriptors for sensory science.

Received: July 1, 2022 **Accepted**: November 4, 2022 **Published**: February 15, 2023

Competing Interests: The authors declare that they have no competing interests.

Data Availability: There is no affiliated data with this submission. The case studies present projects that they themselves have data. The data from the case studies may be available at request.

Results: Each case study was a success but had its own hurdles. Some takeaways are well-structured and clear documentation, code examples and demos, and pre-trained models were at the core to the success of these case studies.

Conclusions: Openly available DL resources were the core of the success of our case studies. The authors encourage DL researchers to continue to make their data, code, and pre-trained models openly available where appropriate.

Introduction

In the field of machine learning (ML), deep learning (DL) has become a very powerful tool that has been incorporated into many new technologies, e.g., self-driving cars (Daily et al. 2017) and natural language processing (NLP) (Vaswani et al. 2017; Devlin et al. 2019; Sutskever, Vinyals, and Le 2014). DL refers to a subset of ML accomplished using deep neural networks. Many materials and codebases for DL are openly available through platforms like GitHub. This allows those interested to use, improve, and build upon existing work to tackle new problems without starting from scratch. These openly available resources are at the heart of open science, which furthers research by allowing researchers to directly and indirectly collaborate to accomplish new goals. DL has proven to be a great collection of tools for addressing challenging problems. Putting the two together has great potential.

One common task in open science using DL is the process of one researcher taking another researcher's freely available DL code and applying it to a new problem space with different data. This is different from using a flexible open source framework or library (Brownlee 2019; "Machine Learning Education" 2022; "Free Online Course to Learn the Basics of Deep Learning with Keras" 2022), because the initial codebase is usually developed for a specific task (e.g., to replicate results from a paper). We aim to compare multiple examples of using openly available materials for DL with varying degrees of documentation and extensibility.

This paper presents three case studies in which a Data and Informatics Consultant (DIC) embedded within University Libraries Data Services identified and adapted existing DL codebases to support research groups with new research goals. The case studies are presented in the order of increasing effort and complexity for the researchers to achieve their goals from the openly available resources. The involvement and support of the DIC will be described for each case study in later sections, along with the details, implications, and applications towards the library itself.

In the first case study, we present a workflow for adapting an open source tool to summarize thousands of news articles. The outcome, generalizable to many situations, is a pipeline that can concisely report key facts and events from the articles. In the second case study, we describe the development of an Optical Character Recognition (OCR) pipeline for archival research of typed notecards, here documenting a curated collection of thousands of clothing items. In the last case study, we describe the process of applying an NLP tool for resumé skill extraction to save time on a novel task: identifying descriptive language for whiskies from thousands of free-form text reviews. These case studies resulted in working solutions to challenging problems, thanks to researchers embracing open science.

Literature Review

The idea of open science is not new. Woelfle et.al. explained it in 2011 as analogous to asking your colleague for help. Openly sharing research between researchers accelerates the research process and makes science more transparent (Woelfle, Olliaro, and Todd 2011). Fecher and Friesike (2014) discussed five schools of thought on open science. The one most pertinent to our case studies is the democratic school of thought, focusing on "principal access to the products of research." However, Heise and Pearce (2020) point out the challenges of making things open, including the slight differences between open access ("pure access to published knowledge") and open science ("complete access to the entire scientific process").

The ML and DL communities, inside and outside formal research groups, have a wide variety of open resources. Erickson et. al. (2017) provides a review of freely available ML/DL libraries. There are many websites dedicated to providing pre-trained models and datasets (e.g., huggingface, Kaggle). On these sites, especially Kaggle, other researchers provide not only datasets but also code notebooks to run the experiments and explain the pipeline.

In the academic sphere, Computer Scientists have long made their research openly available online, independent of academic journals, a practice already widespread by 1998 (Giles, Bollacker, and Lawrence 1998). Today, there are prominent ML/DL journals using both open access and subscription-based publishing models. The Journal of Machine Learning Research, one of the highest-impact ML/DL journals ("Journal Rankings on Artificial Intelligence" 2022; Google Scholar 2022), is an open access journal whose establishment in 2001 led many editors to resign from the subscription-based Machine Learning Journal (Journal of Machine Learning Research 2001; Lewis 2001). Many ML/DL researchers continue to prioritize open access (Hutson 2018), and when ML/DL research is published in subscription-based journals, it is often still freely available as a pre-print. Since 2018, Computer Science is the most-published category on ArXiv.org ("Submissions by Category since 2009+ | ArXiv E-Print Repository" 2022).

As for librarian activities around these computational resources, there are several examples exemplifying such work and partnership. Lamba and Madhusudhan in (Lamba and Madhusudhan 2021a) describe the use of text mining in various contexts, most notably within libraries as presented in Chapter 1, "The Computational Library" (Lamba and Madhusudhan 2021b). Some Computer Science researchers work in the area of digital libraries with a close partnership to their respective academic libraries. One example is ("Digital Library Research Laboratory" 2022), a lab dedicated to partnering with academic libraries for research in information retrieval. A notable member of the lab is the Assistant Dean and Director of

Information Technology within the Virginia Tech University Libraries (Ingram 2022). Another entity supporting computation in libraries is the Online Computer Library Center ("OCLC Research" 2022) which has a Research Library Partnership (RLP) that connects research libraries that support these libraries with 21st Century challenges.

Case study partnerships with libraries have occurred for multiple kinds of events and projects. Hackathons have become popular which entails creating a prototype using hardware and/or software within a short amount of time, generally around 24 hours. The Ohio State University library hosted such an event (Longmeier, Dotson, and Armstrong 2022). Librarians and a Computer Science faculty member partnered to design and teach a class on algorithm bias (Ramachandran, Cutchin, and Fu 2021).

Such partnerships and interactions between a university library and external entities, such as faculty members and students, show support for the positive outcomes such partnerships create. In our case, the openly available DL resources available, along with the skillset of the DIC to use them, provides an excellent opportunity for partnership and support for the university. Our presented case studies not only exemplify this, but also show the positive outcome of such openly available resources being increasingly accessible.

Case Studies

Text Summarization

The first case study is a semester-long project in a Computer Science class offered simultaneously at the graduate and senior capstone level. The focus of the project was to summarize thousands of news articles into a coherent, short summary. This "multi-document summarization" was a novel task at the time of the case study, as opposed to single-document summarization. The professor gave suggestions of how to process the articles with the ultimate goal being to make an abstractive summary as opposed to an extractive summary. An extractive summary takes pieces of the articles and uses them verbatim to create the summary. An abstractive summary is able to use words not present in the initial text to aid in summarization, more closely mimicking how humans produce a summary. The goal was to use one of the many available DL algorithms to produce an abstractive summary.

Background

For this case study, the research group was interested in off-the-shelf, ready-to-use solutions. The single-semester time limit made developing something from scratch impractical. Various techniques for automated summarization were introduced to the class. The need for text summarization is supported by the challenge of having vast volumes of text available from various sources in which it is impractical to expect an individual to read and synthesize from many sources (Allahyari et al. 2017).

The library support for this project was provided by the DIC as the DIC was taking the class at the same time. During the case study, the DIC was able to utilize their training and resources within the library to support the research group.

When this case study took place in fall 2018, the state-of-the-art in summarization using DL was sequence-to-sequence (Sutskever, Vinyals, and Le 2014). Another recent and successful deep learning approach was the Pointer-Generator Network (PGN) (Abigail See, Liu, and Manning 2017). Fall 2018 also saw the release of the transformer architecture by Google ("BERT" 2018), which was published shortly before (Vaswani et al. 2017), and was a game changer for language models. However, this was released partway through the semester, well into the case study researcher's development phase, so it was not used. The group researched and tried several open source implementations of sequence-to-sequence and PGN algorithms, with the PGN being the most successful for the task. A PGN is a hybrid between abstractive and extractive text summarization and provides an end-to-end solution for the research group. A main reason for this choice is that the researchers were unable to successfully get the identified sequence-to-sequence repositories to work within the tight deadlines of the class.

Data

The corpus to be summarized consisted of approximately 12,000 articles related to the #NeverAgain hashtag scraped from the internet: a movement seeking to end gun violence in schools. The Graduate Teaching Assistant (GTA) performed scraping of the articles using the Twitter API to collect tentatively-relevant URLs followed by a web crawler. When duplicate, empty, or irrelevant articles were removed, around 3,600 articles remained. Article relevance was determined using Latent Dirichlet Allocation (LDA) topic modeling (Blei, Ng, and Jordan 2003).

Code

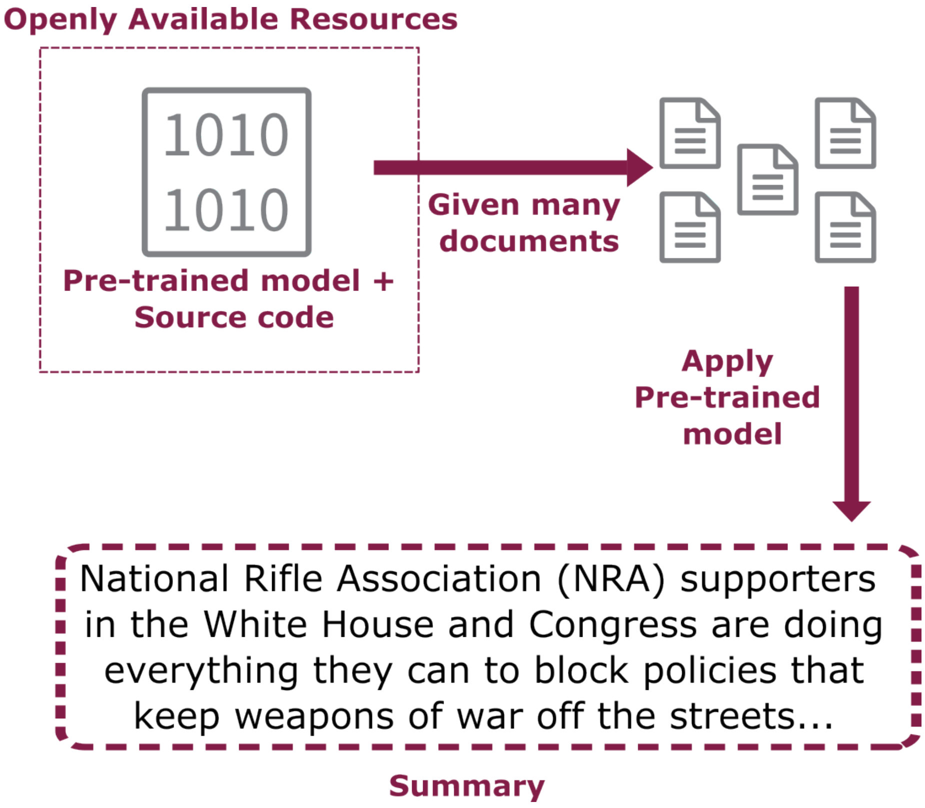
The model used with the PGN was trained on the CNN/Daily Mail dataset (Hermann et al. 2015; Nallapati et al. 2016) and was downloaded and used for this case study, instead of training a model from scratch. The link can be found at this GitHub repository (Abi See 2017). This pre-trained model saved the case study researchers days of training on a computing cluster. They tried this themselves at first, and it took 3 to 4 days. If they needed to re-build the model to tweak some parameters, each retraining would take another 3 to 4 days. Hence, the pre-trained model saved much time.

In order to perform summarization, the articles needed to be converted into a specific binary format. The code to perform this was open source but assumed that the user was going to convert the CNN/Daily Mail dataset. The researchers updated the code in a few days so as to specify what set of articles to convert, making it more extensible. This produced the input format for the PGN. The code is openly available (Miller 2018).

The abstractive summary was made using code from a GitHub repository (Abi See 2017) implementing the PGN DL algorithm to perform multi-document summarization. Minimal coding was needed to adapt this open source code to the #NeverAgain corpus.

Documentation

This project was successful partly because the PGN documentation was very thorough, including required library version numbers, links to pre-trained models, and a full set of example Python commands to train and evaluate the model. The version numbers are very important, as some libraries like TensorFlow, as well as Python itself, are not always backwards-compatible. The availability of pre-trained models and an example script ensured the project was completed on time.



National Rifle Association (NRA) supporters in the White House and Congress are doing everything they can to block policies that keep weapons of war off the streets...

Figure 1: Text Summarization pipeline overview.

Results

The researchers started with a pre-trained model and source code and applied them to a collection of documents, producing a summary. The resulting data pipeline can be seen in Figure 1, including a portion of the final summary. Generally, the quality of a summary is calculated as a ROUGE score (Lin 2004). The

ROUGE scores were low for this summary, with entity coverage only being 7.25%. Entity coverage is how many entities, e.g., people, places, names, are identified in comparison to a "gold standard" summary, here created by human classmates. Even though the scores were low, we direct the reader to the summary result in Figure 1 which exemplifies that the summary was coherent and on-topic. Further details can be found in the final report (Arora et al. 2018). Either way, the openly available pre-trained model and source code allowed for this pipeline to be successfully created and resulted in a successful project for the class.

Challenges

In terms of open science and open access, the challenges were minimal. There were some technical challenges in providing a processing environment for computationally performing the summarization. This was overcome by a computer made available to the DIC within the library that could be dedicated to the computational processing required by the project.

Separate from that, the conversion code for converting the articles into the desired binary format for the PGN took a couple of days to understand and adjust for the researchers' purpose. This was less of a challenge with open resources and more of a time commitment to learning another researcher's code. The original conversion code could be improved to be more extensible, which is exactly what the DIC was able to accomplish.

Another challenge was the quality and conciseness of documentation. One repository identified was not used since the documentation had excess information making it harder to understand. The examples were too specific and did not document the general use case.

Discussion

The researchers of this case study explored several approaches for text summarization with mixed success. They had to judge each based on two criteria: the quality of the benchmark summaries and the clarity of documentation. Poor documentation ruled out several summarization algorithms, with only the PGN being able to be adapted within the semester timeline.

In the end, they selected the TensorFlow version of PGN for reasons directly related to the key principles of open science. The directions were clear and concise, and the commands to run the PGN were simple. There was a provided pre-trained model. In DL, training a model can take days or weeks, so having an openly available model saves valuable time. This also provided more assurance that the model was correctly tested and verified, as it was created by the researchers who designed the algorithm. Having the code, model, and documentation all openly available made it possible for the group in the process of learning about DL to successfully apply a DL approach in a constrained time frame. Even data conversion was possible due to clearly written code.

An interesting point is that the tools for performing the summarization were openly available resources, but the article data was only open within the institution. This limits who could have possibly done this particular research project; however, the main focus in this project was the openly available DL resources that made the project a success. Hence, with the given open resources, other researchers could conduct a similar project with other available data.

There were several DL frameworks available to perform text summarization, with varying qualities of documentation. The availability of pre-trained models was also critical as it saved much time for the researchers since training a model could take days. Thankfully, the authors of the PGN also made a pre-trained model openly available, saving the researchers time and allowing the researchers to focus on the problems specific to the project. For more details on the project, the reader is directed to the full project report found at (Arora et al. 2018).

This project built expertise within the library to support other researchers at the university with similar needs. Tools to summarize vast amounts of documents could also positively impact library services by facilitating faster understanding of large archival collections and published text provided by the library.

Optical Character Recognition

The second case study is an optical character recognition (OCR) project focused on extracting text from scanned images of notecards describing a curated costume collection (~5100 notecards). OCR is the process of identifying text within an image and converting it into accessible text. The notecards contain different pieces of information of interest, such as the donor and description of an item. This information is organized in different layouts on the notecards. In order to identify the information, the researchers needed to automatically identify the layout, i.e., where this information is on the notecards.

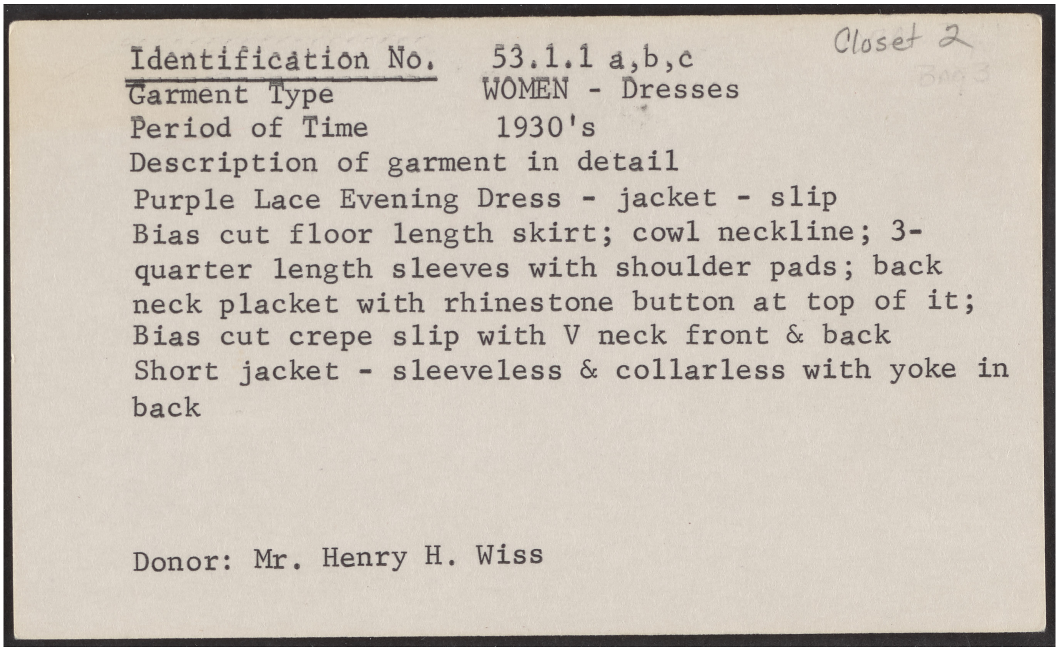
To perform this task, the researchers used an open source version of a Masked Recurrent Convolutional Neural Network (MRCNN) ("Mask R-CNN for Object Detection and Segmentation" 2017), a DL approach to identify objects within an image. In this case, the objects were each a piece of information from the notecards. This was done for two different layouts which comprised approximately 80% of the entire notecard collection. The rest of the notecards were left to be processed at a later date. This layout segmentation provided the input to the OCR algorithm chosen (discussed next).

Background

Performing accurate OCR has been a challenge for many years (Hamad and Kaya 2016; Ahmed and Abidi 2019). One area with specific needs for OCR support is the humanities (Henry 2014) and this case study is an example of how one can address this need. At the time of this case study (Spring/Summer 2020), there were only a handful of open source OCR solutions. While exploring solutions such as Google's open source tesseract ("Tesseract OCR" 2014), the team discovered the open source OCR provided by Clova which had placed high in multiple competitions ("Focused Scene Text - Robust Reading Competition" 2019; "ICDAR2017 Robust Reading Challenge on COCO-Text" 2017; "ICDAR2019 Robust Reading Competition" 2019; "ICDAR2019 - ReCTS - Robust Reading Competition" 2019). Clova provides two tools for an OCR pipeline: text identification and text extraction processes. First, it identifies all of the individual pieces of text (normally single words) within an image using the DL approach (Y. Baek et al. 2019). Text extraction, or traditional OCR, is then conducted using a pre-trained DL model (J. Baek et al. 2019) applied to each identified piece of text. This tool is more efficient as it performs OCR on single "patches" of an image instead of processing the entire image through the OCR engine. This reduces the resources needed for processing each image as only a "subset" of an image is processed. A major reason why Clova OCR was chosen is that the Clova authors developed a framework to test different DL algorithms in conjunction to identify the best combination. This was unique to the Clova OCR solution and allowed the usage of state-of-the-art solutions.

Data

A faculty member within the Fashion Merchandising and Design Department approached the library with the need to OCR thousands of descriptive scanned notecards in JPEG format. Each notecard describes a particular accessory or garment within the curated collection ("The Oris Glisson Historic Costume and Textile Collection" 2022). An example can be seen in Figure 2. The text is typeset and generally clean with some potential blurring and extra lines (e.g., "Identification No." is underlined).



Closet 2
Bag 3

Identification No. 53.1.1 a,b,c
Garment Type WOMEN - Dresses
Period of Time 1930's
Description of garment in detail
Purple Lace Evening Dress - jacket - slip
Bias cut floor length skirt; cowl neckline; 3-
quarter length sleeves with shoulder pads; back
neck placket with rhinestone button at top of it;
Bias cut crepe slip with V neck front & back
Short jacket - sleeveless & collarless with yoke in
back

Donor: Mr. Henry H. Wiss

Figure 2: Example notecard from the garment collection.

The faculty member sought the expertise of the library, specifically the DIC, to aid in performing this task. The DIC worked alongside the faculty member throughout the entire project to support and aid where necessary.

Code

The text in the notecard photos was digitized using code from two GitHub repositories ("Clovaai/CRAFT-Pytorch" 2019; "Clovaai/Deep-Text-Recognition-Benchmark" 2019). One codebase identified where text is in an image and the second converted the identified text areas to machine readable text (performs the OCR). The researchers of this case study wrote "glue" code to integrate both repositories together and create a working OCR pipeline from input to final output. The OCR solution has support for GPU acceleration through the use of PyTorch.

Documentation

The documentation helped the research team piece everything together. Each code repository had a well-organized README accompanying the code repository and demo code with a low learning curve. The researchers started with the demo code for each and connected them together to create the final OCR pipeline.

Results

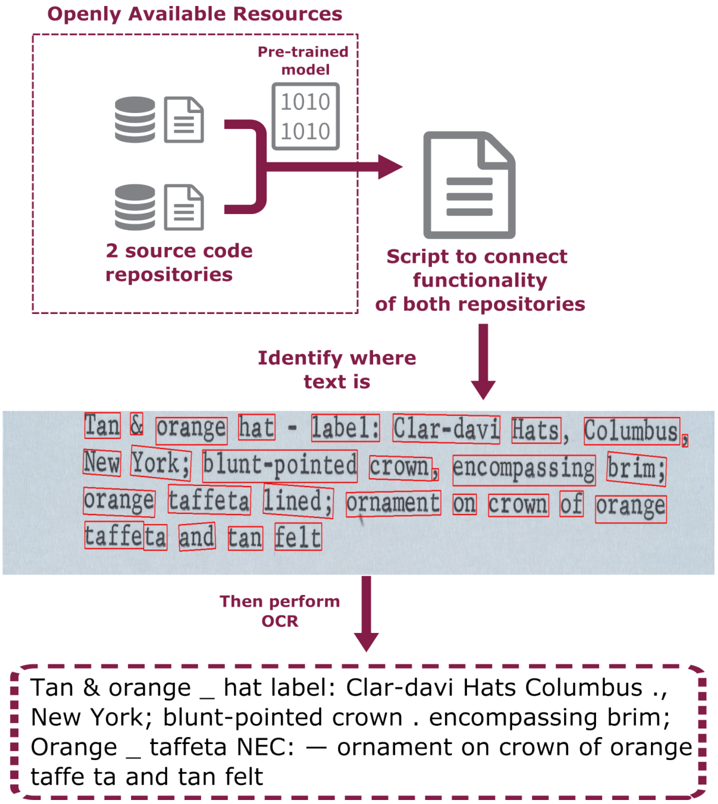
The resulting data pipeline developed for this case study can be seen in Figure 3. The researchers "glued" together the two repositories and used an available pre-trained model to first identify where text was, and then perform the OCR.

The resulting output of the data pipeline can be seen as an example in Figure 4. The text identification code provided the location of each word on the JPEG, i.e., the location and dimensions of the bounding box of each word identified. Given this information, the researchers were able to use the Python library FPDF (Reingart 2013) to create a PDF and specify the locations to put the identified words. Here the font size is based on the dimensions of the bounding box for each word and the shade of blue represents the confidence level (probability) that the OCR is correct. A darker shade means higher confidence while lighter is less confident. As can be seen, the formatting is not ideal and fixing this is an area of future research for the team.

Given the openly available code repositories and a pre-trained model, the researchers were able to accomplish OCR on thousands of scanned notecards describing a physical collection. The creation of the pipeline took roughly two weeks to complete and resulted in an end-to-end solution.

Challenges

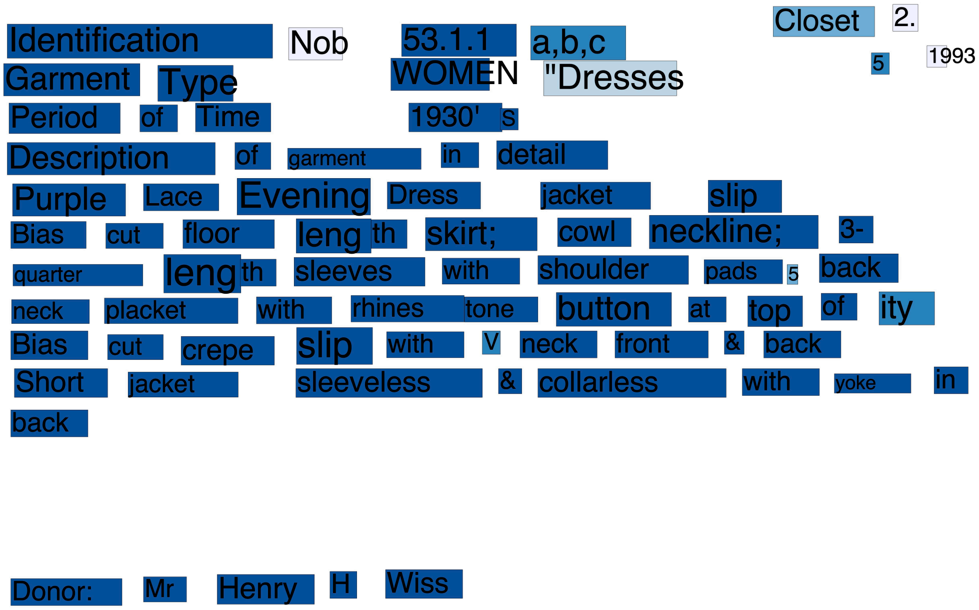
In terms of open science and open access, the challenges were a little more than the first case study as the researchers had to work with two repositories that required the creation of new code that connected


Openly Available Resources
Pre-trained model
1010
1010
2 source code repositories
Script to connect functionality of both repositories
Identify where text is
Then perform OCR

Tan & orange hat - label: Clar-davi Hats, Columbus,
New York; blunt-pointed crown, encompassing brim;
orange taffeta lined; ornament on crown of orange
taffeta and tan felt

```
Tan & orange _ hat label: Clar-davi Hats Columbus .,
New York; blunt-pointed crown . encompassing brim;
Orange _ taffeta NEC: — ornament on crown of orange
taffe ta and tan felt
```

Figure 3: OCR pipeline overview.



```
                                                              Closet   2.
Identification   Nob   53.1.1   a,b,c
                                                                   5   1993
Garment Type           WOMEN "Dresses
Period of Time         1930' s
Description of garment in detail
Purple Lace Evening Dress jacket slip
Bias cut floor leng th skirt; cowl neckline; 3-
quarter leng th sleeves with shoulder pads 5 back
neck placket with rhines tone button at top of ity
Bias cut crepe slip with V neck front & back
Short jacket sleeveless & collarless with yoke in
back

Donor: Mr Henry H Wiss
```

Figure 4: Clova OCR output for one notecard. The shade of blue represents the confidence level (probability) that the OCR is correct. Darker shades means higher confidence while lighter is less confident.

them. As the repository authors did not provide a way for the text identification and OCR repositories to work together, it was up to the researchers to create a solution. The repositories could be improved by the repository authors providing a demo that tied them together. This would have been very beneficial.

To overcome any computational resource requirements, a specialized computer was made available to the DIC through the library. With this computer, the potential challenges of performing the OCR, which is very computationally intense, was solved.

Discussion

The search for an OCR solution was short, ending with a GitHub repository that had ranked high in OCR competitions and linked to another open code repository able to identify where text is in an image. In the context of this paper, having such high quality open resources provided a robust OCR solution for the project.

It took mere minutes to test each repository separately, and two weeks to integrate the tools together. Given the complexity of DL code, this is a testament to the quality of the documentation. Each repository included well-documented pre-trained models and demo code requiring minimal set up. The initial results using the two libraries and the “glue” code were very promising, with the speed of implementation being a major advantage.

The openness of these available resources allowed the researchers to have a successful project and exemplify what is possible when such resources are made freely accessible.

The knowledge and skills learned and developed during this case study strengthened the OCR support the library could offer. This is a crucial service that can be utilized within the library when collections require OCR. Having a framework already in place allows for digitization through OCR to be more readily available and possible.

Descriptor Identification

The final case study involves extracting flavor descriptors from reviews of beverages. Flavor descriptors are words that describe the sensory profile of a food item, specifically whiskey, for this project. The set of possible flavor descriptors for a food product (its “flavor lexicon”) can be difficult to extract due to the unique nature of words used as descriptors. They are commonly, but not always adjectives, and not all adjectives are useful descriptors. Frequency is not always key as some useful descriptors can be infrequent, and the topmost common adjectives often describe color (e.g., “red”, “black”, “brown”), intensity (e.g., “very”), or liking (e.g., “nice”, “pleasant”) (Bécue-Bertaut, Álvarez-Esteban, and Pagès 2008).

The descriptor extraction was the most challenging project, as only the core code was available online through a blog post (Intuition Engineering 2018). The researchers had to study the problem deeper to better understand what code “glue” and other pieces needed to be developed. The work required to accomplish this task took months, as the researchers were newer to developing DL algorithms. For example, the code that was available required a more involved understanding of DL models in order to define the model and format the input data (text). There are whole books written on text data preparation for ML and DL algorithms such as (Brownlee 2020a, 2020b). This exemplifies the need to study techniques for accomplishing this preparation. However, the success of the project would have potentially taken longer if not for the available open resources.

As in the OCR project, a faculty member approached the library looking for a collaborator with the skillset to aid in this project. Fortunately, the DIC within the library was able to partner with the faculty member and support the DL part of the project.

Background

The research team was not able to use another researcher’s pre-trained model, as this case study had a unique challenge. To the author’s knowledge, at the time of the project (Summer 2019 - Fall 2021), there were no other trained models for identifying unique sensory terms. In the domain of sensory science, flavor lexicons are made using an experimental methodology called Descriptive Analysis (DA). DA uses a trained human panel that tastes a variety of products within a category and provides descriptive sensory terms for each product (Heymann, King, and Hopfer 2014). There are a few examples, however, of corpora of food descriptions being used to identify flavor descriptors for a lexicon (Ickes, Lee, and Cadwallader 2017).

The second, less common method of extracting descriptors is more similar to other keyword extraction problems that have some solutions using NLP such as RAKE (Rose et al. 2010) and YAKE (Campos et al. 2020). The blog (Intuition Engineering 2018) described an analogous problem identifying skills in resumés. Terms describing skills can likewise be very unique to the domain, such as words containing all capital letters or symbols, e.g., SQL or C++. Hence, why this analogous problem was used as a template. Only the core code for the resumé skill extraction tool was available, with functions to define the DL architecture in Python using Keras along with a few other helper functions.

Data

Our data set consisted of 8000+ whiskey reviews for training and testing. The data was scraped from WhiskyAdvocate (4288 reviews), WhiskyCast (2309 reviews), The Whiskey Jug (1095 reviews), and Breaking Bourbon (344 reviews). WhiskyAdvocate and WhiskyCast reviewers are professionals, with whiskey writing being a primary income source, while Breaking Bourbon and the Whiskey Jug are hobbyist blogs run by “semi-professional” reviewers.

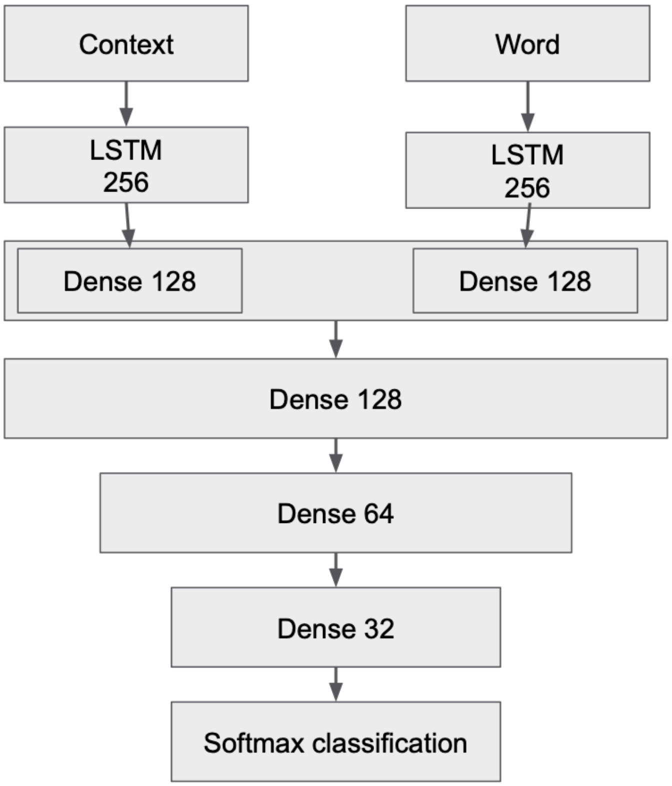


Figure 5: DL architecture used for descriptor identification.

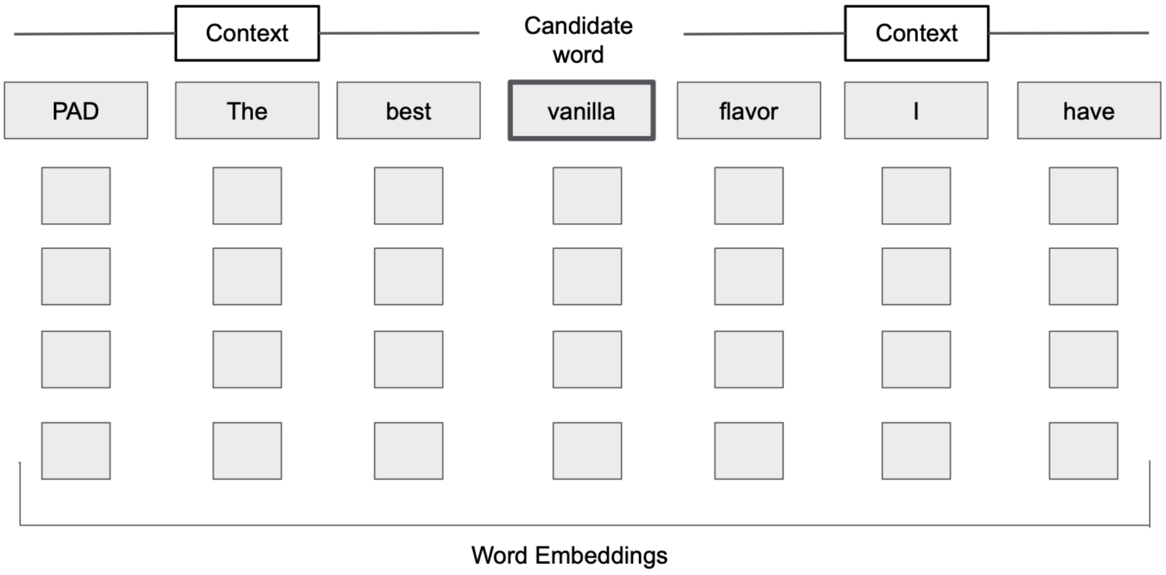


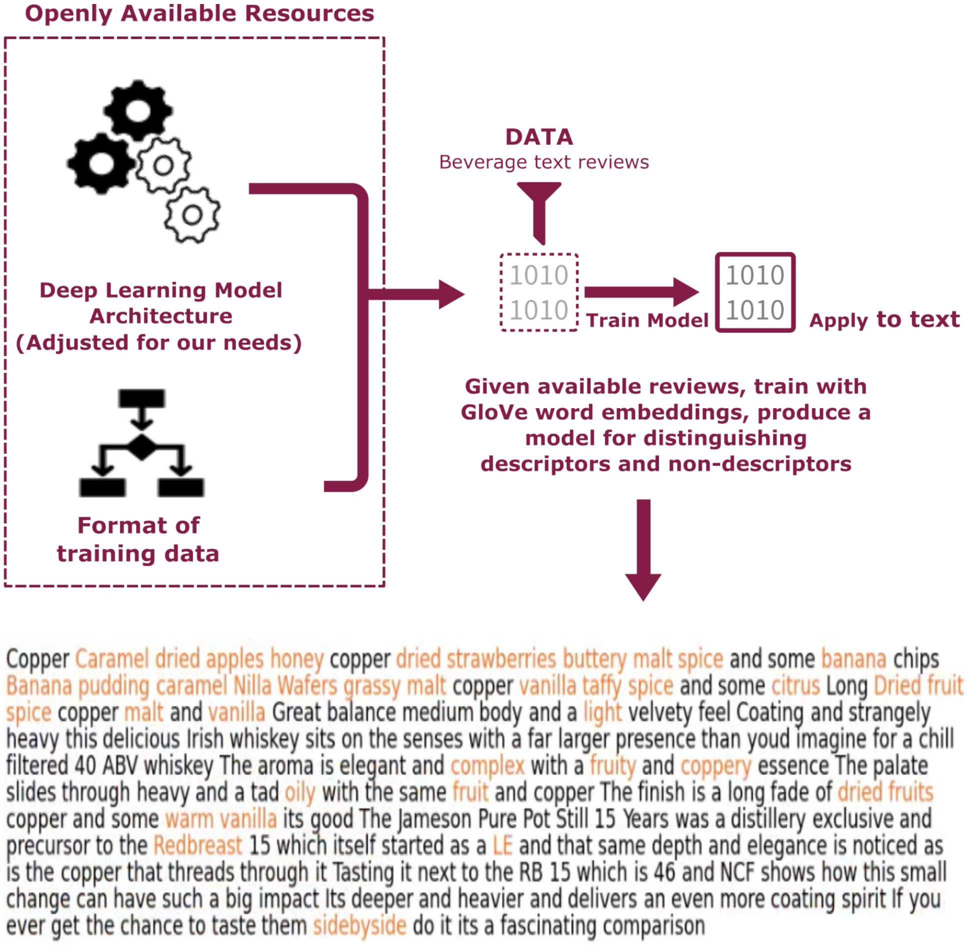
Figure 6: Format described by the blog post for input data for training the model.

Code

This case study required the most custom coding by far. The code snippets publicly available on the blog post were primarily focused on defining the DL neural network architecture (Figure 5) and the format of the input data for training (Figure 6). The research team needed to fill in the rest, which included adjusting the model architecture to use their feature set, pre-processing the data into the desired format using GloVe word embeddings (Pennington, Socher, and Manning 2014), and developing code to train the model and evaluate the results.

Documentation

The description of the problem and DL solution in the blog post had clear steps, most of which had some code snippets to represent what was discussed. Certain steps, however, were only described in prose without accompanying code. How to implement these processes was left for the reader to figure out.


Openly Available Resources
Deep Learning Model Architecture (Adjusted for our needs)
Format of training data
DATA
Beverage text reviews
1010
1010
Train Model
1010
1010
Apply to text
Given available reviews, train with GloVe word embeddings, produce a model for distinguishing descriptors and non-descriptors

Copper Caramel dried apples honey copper dried strawberries buttery malt spice and some banana chips Banana pudding caramel Nilla Wafers grassy malt copper vanilla taffy spice and some citrus Long Dried fruit spice copper malt and vanilla Great balance medium body and a light velvety feel Coating and strangely heavy this delicious Irish whiskey sits on the senses with a far larger presence than youd imagine for a chill filtered 40 ABV whiskey The aroma is elegant and complex with a fruity and coppery essence The palate slides through heavy and a tad oily with the same fruit and copper The finish is a long fade of dried fruits copper and some warm vanilla its good The Jameson Pure Pot Still 15 Years was a distillery exclusive and precursor to the Redbreast 15 which itself started as a LE and that same depth and elegance is noticed as is the copper that threads through it Tasting it next to the RB 15 which is 46 and NCF shows how this small change can have such a big impact Its deeper and heavier and delivers an even more coating spirit If you ever get the chance to taste them sidebyside do it its a fascinating comparison

Figure 7: Descriptor identification pipeline overview along with resources that were developed. The words in the last step highlighted in orange are the words identified as descriptors.

The partial code available that became the core of this project, as mentioned earlier, is in a blog post on TowardDataScience.com, which allows free access for some articles. The author(s) can also require a Medium.com membership to view their work, which is the case here as of the writing of this paper. Now that the case study researchers have successfully adapted these available resources, they have published an open access journal article describing the full technical details of the project (Miller, Hamilton, and Lahne 2021).

Results

The resulting data pipeline developed for this case study can be seen in Figure 7 where the words in the last step highlighted in orange are the words identified as descriptors. The openly available resources were the DL model architecture and the format of the training data. From there, the researchers were able to train using the openly available GloVe word embeddings, then apply to text and identify which words are descriptors.

The result was a trained model that can accurately identify descriptors within a corpus of whisky review texts with a train/test accuracy of 99% and precision, recall, and F1-scores of 0.99. This shows that even though the available resources were limited, the researchers were still able to use them to solve a challenging problem.

Since one measure of success was whether the model can distinguish between if a word is a descriptor or not, the researchers applied a t-SNE to visualize if there was any clustering of descriptors and non-descriptors. A t-SNE (van der Maaten and Hinton 2008) is a dimensionality reduction algorithm used to visualize high dimensional data in a two-dimensional space. Word embeddings are a high-dimensional mathematical vector representation of words, one vector per word. It represents a conceptual space where words with similar meaning have vectors that are closer to each other. The result can be seen in Figure 8 where the brown "X"'s represents a word tagged as a descriptor and a blue dot those words tagged as not being a descriptor. The clusters represent that the model was able to identify the descriptor space, i.e., where in the conceptual space do the descriptors lie. There are some "speckles" of descriptors throughout the blue dot cluster which shows that some words may be conceptually similar, but their contextual meaning varies.

Challenges

This was the most challenging case study, as the use of partially-open resources without a full demo required the most programming and engineering from the researchers and DIC. The project was successful and resulted in a full description of the problem in an open access journal (Miller, Hamilton, and Lahne 2021) with access to the code and data at request of the researchers. What could have been improved was more available code to aid in the creation of the entire pipeline. The only demo available was an online example for identifying skills from a resumé. This is beneficial to showcasing the solution, but does not help in engineering a solution.

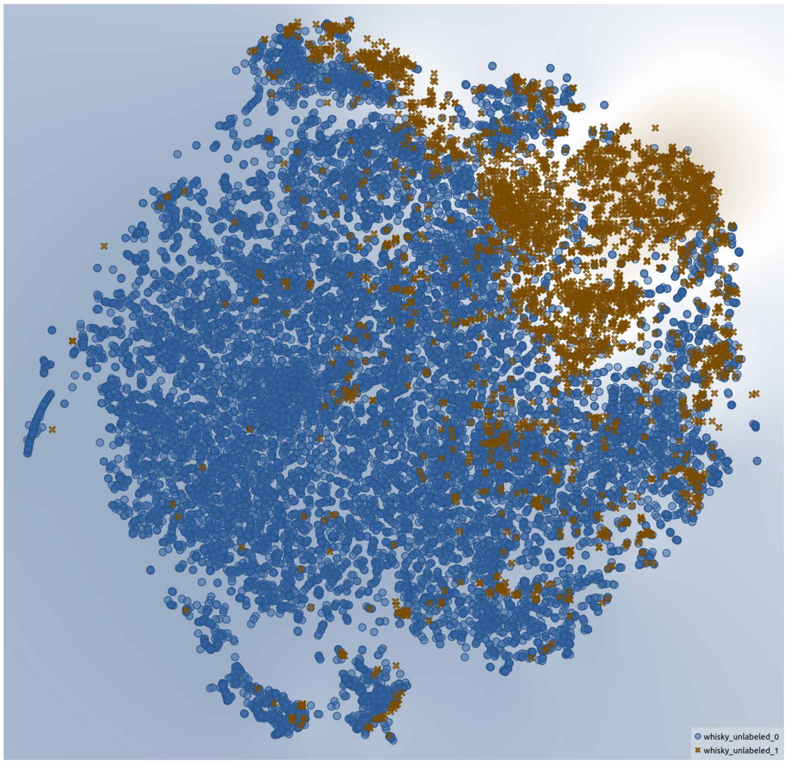


Figure 8: Application of trained model to the corpus (minus the training and test set). Brown "X"'s represent a word labeled as a descriptor and blue dots are labeled as non-descriptors.

Last is providing the computational resources needed for training an NLP model from scratch. Once again, a specialized computer was provided to the DIC through the library to address this challenge and allow the researchers to focus on the research itself.

Discussion

The high accuracy, precision, recall, and F-1 scores shows that this case study was a great success. No pre-trained model was available, and while this increased the project timeline, the researchers' model successfully learned some nuances of the sensory language of their domain as demonstrated by the t-SNE visualization.

Given these challenges, this case study exemplifies how a partially-open resource can inspire a project direction but also hinder it, e.g., the absence of code for all parts of the project. This contrasts with the other two case studies, which were able to proceed faster with fully-open DL resources. Despite these limitations, this case study demonstrates that research can still benefit from even partially-open resources.

The implications to the library from this project are fairly unique. The ability to develop a pipeline for pre-processing text, identifying training data, developing a custom language model, and training the model has many potential connections to text library collections. Custom models can be developed to identify unique information within collections and especially for vast archived collections. For example, these skills and techniques can be applied to historic collections to better understand elements of the collection, extracting key items and insights from the collections. The developed skill sets provide desired resources for people from outside the library along with those within.

Lessons Learned

A key part of open science is providing enough information for the scientific investigation to be replicated: a description of how an experiment was run or documentation for how a software tool works. However, not all documentation is created equal. Even with meticulous directions, success was not always possible. In the text summarization case study, the researchers ruled out one promising solution on GitHub because the README contained an excess of detail on how to use the software for only some specific cases and the commands were confusing to adapt to the researchers' needs. A useful README contains step-by-step instructions of how to install the software, a list of dependencies, several base use cases, and ideally several pre-packaged demos. Too much detail can make more work for the end user who has to sift through it, especially if the documentation isn't well-structured or is overly specific to the original use case. Functionality that exists in an openly-available library is only useful to the end user if the documentation explains when the functionality is useful and provides run commands demonstrating the functionality.

Beyond documentation, another tool to open science is demonstration–this could be in the form of a recorded or documented experiment or a set of code notebooks that walk through a tool. The OCR case study demonstrates the use of these tools in bolstering open science: a well-developed demo to showcase its functionality made it possible to adapt the original tool's functionality for the researcher's needs.

Access to pre-trained models was vital to the success of the first two case studies. These models allowed the researchers to progress through their projects and minimize time needed for set-up and application. Without pre-trained models, the researchers would need to perform the entire training pipeline themselves, which may not always be possible due to the lack of computing resources and/or expertise. In contrast, in the third case study the need to develop parts of the model and train it proved a significant barrier—although it was possible to overcome this, the time required (and the work that was presumably redundant to that done by the original authors of the model (Intuition Engineering 2018)) was much greater than in the other two case studies.

Conclusion

Using open resources, the researchers of these case studies were able to successfully pursue research projects with positive outcomes. Each case study started with the researchers identifying available resources, with each requiring different approaches. The support from, and partnership with, the library provided key eScience elements allowing each case study to be that much more successful.

Our observation is that more and more ML and DL researchers (and some companies) are making their resources (data, pre-trained models, code) openly available for the wider research community. This is a reason these case studies were able to find workable solutions without the need to be experts in the respective ML and DL topics and techniques.

Some takeaways are that documentation, clear code examples and demos, and pre-trained models were at the core to the success of these case studies. Also that eScience skills within the library can provide crucial support to making some projects more possible and allow such technologies discussed in this paper more accessible by those within and coming to the library.

We hope this collection of case studies will provide compelling evidence as to why the spirit of open science should be embraced not only by ML and DL researchers and practitioners, but also by other domains as well.

Acknowledgements

The content of this article is based on the Poster presentation at RDAP Summit 2022, available from Open Science Framework: https://osf.io/76a3b.

Data Availability

There is no affiliated data with this submission. The case studies present projects that they themselves have data. The data from the case studies may be available at request.

Competing Interests

The authors declare that they have no competing interests.

References

Ahmed, Muna, and Ali Abidi. 2019. REVIEW ON OPTICAL CHARACTER RECOGNITION.

Allahyari, Mehdi, Seyedamin Pouriyeh, Mehdi Assefi, Saeid Safaei, Elizabeth D. Trippe, Juan B. Gutierrez, and Krys Kochut. 2017. "Text Summarization Techniques: A Brief Survey." arXiv:1707.02268. arXiv. https://doi.org/10.48550/ARXIV.1707.02268.

Arora, Anuj, Chreston Miller, Jixiang Fan, Shuai Liu, and Yi Han. 2018. "Big Data Text Summarization for the NeverAgain Movement." December. Virginia Tech. https://vtechworks.lib.vt.edu/handle/10919/86357.

Baek, Jeonghun, Geewook Kim, Junyeop Lee, Sungrae Park, Dongyoon Han, Sangdoo Yun, Seong Joon Oh, and Hwalsuk Lee. 2019. "What Is Wrong With Scene Text Recognition Model Comparisons? Dataset and Model Analysis." In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 4714–4722. Seoul, Korea (South): IEEE. https://doi.org/10.1109/iccv.2019.00481.

Baek, Youngmin, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. 2019. "Character Region Awareness for Text Detection." In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9357–9366. Long Beach, CA, USA: IEEE. https://doi.org/10.1109/cvpr.2019.00959.

Bécue-Bertaut, Mónica, Ramón Álvarez-Esteban, and Jérôme Pagès. 2008. "Rating of Products through Scores and Free-Text Assertions: Comparing and Combining Both." *Food Quality and Preference* 19(1): 122–134. https://doi.org/10.1016/j.foodqual.2007.07.006.

"BERT." 2018. Google Research. https://github.com/google-research/bert.

Blei, David M., Andrew Y. Ng, and Michael I. Jordan. 2003. "Latent Dirichlet Allocation." *Journal of Machine Learning Research* 3(Jan): 993–1022.

Brownlee, Jason. 2019. "Your First Deep Learning Project in Python with Keras Step-By-Step." Machine Learning Mastery. July 23. https://machinelearningmastery.com/tutorial-first-neural-network-python-keras.

———. 2020a. Data Preparation for Machine Learning. https://machinelearningmastery.com/data-preparation-for-machine-learning.

———. 2020b. "8 Top Books on Data Cleaning and Feature Engineering." Machine Learning Mastery. June 30. https://machinelearningmastery.com/books-on-data-cleaning-data-preparation-and-feature-engineering.

Campos, Ricardo, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. 2020. "YAKE! Keyword Extraction from Single Documents Using Multiple Local Features." *Information Sciences* 509: 257–289. https://doi.org/10.1016/j.ins.2019.09.013.

"Clovaai/CRAFT-Pytorch." 2019. Clova AI Research. https://github.com/clovaai/CRAFT-pytorch.

"Clovaai/Deep-Text-Recognition-Benchmark." 2019. Clova AI Research. https://github.com/clovaai/deep-text-recognition-benchmark.

Daily, Mike, Swarup Medasani, Reinhold Behringer, and Mohan Trivedi. 2017. "Self-Driving Cars." *Computer* 50(12): 18–23. https://doi.org/10.1109/mc.2017.4451204.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." arXiv. https://doi.org/10.48550/ARXIV.1810.04805.

"Digital Library Research Laboratory." 2022. Accessed November 15. https://dlib.vt.edu/content/dlib_vt_edu/en/index.html.

Erickson, Bradley J., Panagiotis Korfiatis, Zeynettin Akkus, Timothy Kline, and Kenneth Philbrick. 2017. "Toolkits and Libraries for Deep Learning." *Journal of Digital Imaging* 30(4): 400–405. https://doi.org/10.1007/s10278-017-9965-6.

Fecher, Benedikt, and Sascha Friesike. 2014. "Open Science: One Term, Five Schools of Thought." In *Opening Science: The Evolving Guide on How the Internet Is Changing Research, Collaboration and Scholarly Publishing*, edited by Sönke Bartling and Sascha Friesike, 17–47. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-00026-8_2.

"Focused Scene Text - Robust Reading Competition." 2019. https://rrc.cvc.uab.es/?ch=2&com=evaluation&task=3.

"Free Online Course to Learn the Basics of Deep Learning with Keras." 2022. Simplilearn.Com. Accessed June 23. https://i9simplex.simplilearn.com/learn-keras-for-beginners-free-course-skillup.

Giles, C. Lee, Kurt D. Bollacker, and Steve Lawrence. 1998. "CiteSeer: An Automatic Citation Indexing System." In *Proceedings of the Third ACM Conference on Digital Libraries - DL '98*, 89–98. Pittsburgh, Pennsylvania, United States: ACM Press. https://doi.org/10.1145/276675.276685.

Google Scholar. 2022. "Artificial Intelligence - Google Scholar Metrics." Accessed June 22. https://scholar.google.com/citations?view_op=top_venues&hl=en&vq=eng_artificialintelligence.

Hamad, Karez, and Mehmet Kaya. 2016. "A Detailed Analysis of Optical Character Recognition Technology." *International Journal of Applied Mathematics, Electronics and Computers* 4(Special Issue-1): 244–244. https://doi.org/10.18100/ijamec.270374.

Heise, Christian, and Joshua M. Pearce. 2020. "From Open Access to Open Science: The Path From Scientific Reality to Open Scientific Communication." *SAGE Open* 10(2): 2158244020915900. https://doi.org/10.1177/2158244020915900.

Henry, Geneva. 2014. "Data Curation for the Humanities: Perspectives From Rice University." In *Research Data Management: Practical Strategies for Information Professionals*, 347–374. Purdue University Press.

Hermann, Karl Moritz, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. "Teaching Machines to Read and Comprehend." *Advances in Neural Information Processing Systems* 28. https://proceedings.neurips.cc/paper/2015/hash/afdec7005cc9f14302cd0474fd0f3c96-Abstract.html.

Heymann, Hildegarde, Ellena S. King, and Helene Hopfer. 2014. "Classical Descriptive Analysis." In *Novel Techniques in Sensory Characterization and Consumer Profiling*, CRC Press, 9–40. https://doi.org/10.1201/b16853.

Hutson, Matthew. 2018. "Why Are AI Researchers Boycotting a New Nature Journal—and Shunning Others?" May 17. https://www.science.org/content/article/why-are-ai-researchers-boycotting-new-nature-journal-and-shunning-others.

"ICDAR2017 Robust Reading Challenge on COCO-Text." 2017. https://rrc.cvc.uab.es/?ch=5&com=evaluation&task=2.

"ICDAR2019 - ReCTS - Robust Reading Competition." 2019. https://rrc.cvc.uab.es/files/ICDAR2019-ReCTS.pdf.

"ICDAR2019 Robust Reading Competition." 2019. https://rrc.cvc.uab.es/files/ICDAR2019-ArT.pdf.

Ickes, Chelsea M., Soo-Yeun Lee, and Keith R. Cadwallader. 2017. "Novel Creation of a Rum Flavor Lexicon Through the Use of Web-Based Material." *Journal of Food Science* 82(5): 1216–1223. https://doi.org/10.1111/1750-3841.13707.

Ingram, William. 2022. "Researcher Profile | Virginia Tech." https://experts.vt.edu/5675-william-a-ingram.

Intuition Engineering. 2018. "Deep Learning for Specific Information Extraction from Unstructured Texts." Medium. July 31. https://towardsdatascience.com/deep-learning-for-specific-information-extraction-from-unstructured-texts-12c5b9dceada.

Journal of Machine Learning Research. 2001. "History of JMLR." https://www.jmlr.org/history.html.

"Journal Rankings on Artificial Intelligence." 2022. Accessed June 22. https://www.scimagojr.com/journalrank.php?category=1702&order=h&ord=desc.

Lamba, Manika, and Margam Madhusudhan. 2021a. *Text Mining for Information Professionals: An Uncharted Territory*. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-85085-2.

———. 2021b. “The Computational Library.” In *Text Mining for Information Professionals: An Uncharted Territory*, 1–31. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-85085-2_1.

Lewis, David D. 2001. “On Less Restrictive Access to Archival Research Literature.” *SIGIR Forum* 35(2). http://sigir.org/files/forum/F2001/sigirFall01Letters.html.

Lin, Chin-Yew. 2004. “ROUGE: A Package for Automatic Evaluation of Summaries.” In T*ext Summarization Branches Out*, 74–81. Barcelona, Spain: Association for Computational Linguistics. https://aclanthology.org/W04-1013.

Longmeier, Meris M., Daniel S. Dotson, and Julia N. Armstrong. 2022. “Fostering a Tech Culture through Campus Collaborations: A Case Study of a Hackathon and Library Partnership.” *Science and Technology Libraries* 41(2): 152–173. https://doi.org/10.1080/0194262x.2021.1963388.

Maaten, Laurens van der, and Geoffrey Hinton. 2008. “Viualizing Data Using T-SNE.” *Journal of Machine Learning Research* 9(November): 2579–2605.

“Machine Learning Education.” 2022. TensorFlow. Accessed June 23. https://www.tensorflow.org/resources/learn-ml.

“Mask R-CNN for Object Detection and Segmentation.” 2017. Matterport, Inc. https://github.com/matterport/Mask_RCNN.

Miller, Chreston. 2018. “Process_data_for_pointer_summrizer.” https://github.com/chmille3/process_data_for_pointer_summrizer.

Miller, Chreston, Leah Hamilton, and Jacob Lahne. 2021. “Sensory Descriptor Analysis of Whisky Lexicons through the Use of Deep Learning.” *Foods* 10(7): 1633. https://doi.org/10.3390/foods10071633.

Nallapati, Ramesh, Bowen Zhou, Cicero Nogueira dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. “Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond.” arXiv. https://doi.org/10.48550/ARXIV.1602.06023.

“OCLC Research.” 2022. OCLC. November 1. https://www.oclc.org/research/home.html.

Pennington, Jeffrey, Richard Socher, and Christopher D Manning. 2014. “GloVe: Global Vectors for Word Representation.” https://nlp.stanford.edu/pubs/glove.pdf.

Ramachandran, Shalini, Steven Matthew Cutchin, and Sheree Fu. 2021. “Raising Algorithm Bias Awareness among Computer Science Students through Library and Computer Science Instruction.” In *2021 ASEE Virtual Annual Conference Content Access*. https://peer.asee.org/37634.

Reingart, Mariano. 2013. “Pyfpdf: FPDF for Python.” https://github.com/reingart/pyfpdf.

Rose, Stuart, Dave Engel, Nick Cramer, and Wendy Cowley. 2010. "Automatic Keyword Extraction from Individual Documents." In *Text Mining*, edited by Michael W. Berry and Jacob Kogan, 1–20. Chichester, UK: John Wiley & Sons, Ltd. https://doi.org/10.1002/9780470689646.ch1.

See, Abi. 2017. "Abisee/Pointer-Generator." https://github.com/abisee/pointer-generator.

See, Abigail, Peter J. Liu, and Christopher D. Manning. 2017. "Get To The Point: Summarization with Pointer-Generator Networks." arXiv. https://doi.org/10.48550/ARXIV.1704.04368.

"Submissions by Category since 2009+ | ArXiv E-Print Repository." 2022. Accessed June 22. https://arxiv.org/about/reports/submission_category_by_year.

Sutskever, Ilya, Oriol Vinyals, and Quoc V Le. 2014. "Sequence to Sequence Learning with Neural Networks." In *Advances in Neural Information Processing Systems*. Vol. 27. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2014/hash/a14ac55a4f27472c5d894ec1c3c743d2-Abstract.html.

"Tesseract OCR." 2014. C++. tesseract-ocr. https://github.com/tesseract-ocr/tesseract.

"The Oris Glisson Historic Costume and Textile Collection." 2022. Accessed June 17. https://liberalarts.vt.edu/content/liberalarts_vt_edu/en/departments-and-schools/apparel-housing-and-resource-management/experience/collections/the-oris-glisson-historic-costume-and-textile-collection.html.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. "Attention Is All You Need." *Advances in Neural Information Processing Systems* 30. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Woelfle, Michael, Piero Olliaro, and Matthew H. Todd. 2011. "Open Science Is a Research Accelerator." *Nature Chemistry* 3(10): 745–748. https://doi.org/10.1038/nchem.1149.