Project Biography
Leah Marszalek
Data Extract of PDFs in Colab Notebook

Links:

Github link: **https://lmarszalek-suffolk.github.io/ctl/DataExtractionProjectBiography.pdf**
Colab Notebook link: https://colab.research.google.com/drive/100xn_JXOkXGrpmYYuPpWR45mcD-BJWwi?usp=sharing (shared with suffolklitlab@gmail.com)
Google Drive Main Folder Link (containing articles): https://drive.google.com/drive/folders/1mWvSondvX9vtE1qTzZaZ8-XXCG5AxQ9Q?usp=sharing (shared with suffolklitlab@gmail.com)
Google Drive Sub-Folder – Background Only: Excel Export of Reworded Prompt Attempt Results, Link: https://drive.google.com/drive/folders/1IH17DFNdLOUexpX-osEoy-Slo1-OtjvY?usp=sharing (shared with suffolklitlab@gmail.com)
Google Drive Sub-Folder – CSV Export of Coding Results (will populate with exported CSV from notebook once final step is run), Link: https://drive.google.com/drive/folders/1ab-aLl6SHSWlIzlsNk4-EqwsY78sa5i8?usp=sharing (shared with suffolklitlab@gmail.com)

Framing:

A major problem that legal practitioners face is time, especially when you are an attorney at a big law firm that has billable hour requirements. Reviewing documents is a huge time sink, and unfortunately, that is one of the main tasks of an attorney. Many large firms require that their attorneys, in addition to billable work, take on *pro bono* work. But when you find yourself billing 8-10 hours a day on case work, how can you possibly have time to fulfill the *pro bono* requirements (typically 20 hours a year)? As a litigation associate, specifically at Ropes & Gray, the main *pro bono* project for the firm is its asylum cases. For each of those asylum cases, the court requires something called a "country conditions report." This report consists of relevant quotes from publicly available articles that report out on the conditions of the specific country you are researching—typically, the country your client is seeking asylum from. Some of these articles are short, i.e., blog posts, while others are longer, i.e., government reports. In any event, this is the most time-consuming task for an asylum matter, and is typically passed off to summer associates or first-year associates because they have a lower billable rate. In order to cut the time for those individuals and higher billing associates, this Colab notebook will reduce the time by nearly half, by reviewing the downloaded, text searchable documents, and extracting out any language the associate is targeting. Even further, it exports those quotes into a csv file, with respective columns, which takes additional work off the attorney for copy and pasting relevant statements from those articles, into the format required by the court. The Colab notebook can be updated at any time to target different information, and the linked Google Drive folder can also be easily maintained. While the tool would most likely be used by associates, it can easily be initially set up with the appropriate targeted questions, and troubleshooted, by paralegals or litigation technology staff at the firm. The tool is cost effective, and also allows for restricted access or shared access. Since the articles required for country condition reports are public facing and accessible, there is a diminished data security risk. However, if the targeted questions inputted into the notebook are sensitive, then controls can be put in place to restrict access to this notebook—the firm should also review Google Drive data security features, and if they are problematic, then update the tool to pull from a more secure document foldering system.

Research, Ideation, Prototyping, User Testing, Refinement:

Existing Platforms. There are several platforms that export data from PDFs based off of LLMs, including ChatGPT. I did use ChatGPT to test my prompts and see if they populated good results from my articles. However, this software requires that you use a third-party site outside of ChatGPT, i.e., Zapier, to convert the documents you are trying to search/extract, into a plain text document. Then, to send the raw text file to ChatGPT. Our solution, in Colab Notebooks, only requires that you make the document text recognizable if it is a PDF, which can easily be done in batches in Adobe. Further, our solution allows you to store all documents, whether used in that current running of the notebook or stored for future use, into one Google Drive folder that is shareable with others, if applicable. Therefore, future users can see documents you may have already pulled information from and the extraction results, and do not have to reinvent the wheel. ChatGPT is helpful because it does not require you to write the actual background code, just simply ask it a question. But, this can also be a downside. In Colab Notebook, you can enter information for the purpose of guiding the user, and the user will know exactly why certain information is being pulled over other information, and can troubleshoot it or update it. There is no way of knowing what the code for a ChatGPT request is, how to troubleshoot it, or how to update it. On the site itself is a disclaimer that the site does not know everything. While some versions of ChatGPT are free, more advanced features are not. In the Colab Notebook, it is possible to get free credits through OpenAPI tokens, which takes a good amount of time to be used up. After that, you can reload your credits at a small rate.

In addition to ChatGPT, there are several other AI generated data extraction solutions, but none that are as noteworthy. For instance, Docugami, which is a platform that extracts data from commercial insurance carrier documents—which is not a relevant topic for a litigation associate working on asylum matters. Next, Liner, a ChatGPT PDF Summarizer, is also not helpful for this purpose because it will summarize for you, but it will not pull exact quotes, and you cannot batch upload PDF files. Like ChatGPT, you cannot see the background code, and there are also monthly and yearly fees for simply highlighting articles. Next, is Amazon Textract. Though the website claims this solution improves security and compliance through robust data privacy, encryption, security controls, and support compliance standards such as HIPAA, GDPR, and more, it is not clear that this platform would be safe for legal data. It all extracts text and structured data from tables and forms, it is unclear whether it would be able to accurately pull data from unstructured blog posts or articles that are void of headings. Further, there are payment requirements and restrictions on the amount of pages that can be reviewed per month, i.e., 1,000 pages per month , or 100 pages per month. With our solution, it can interpret hundreds or even thousands of pages in one run (though note they have to be small documents, but can have a unlimited amount of total documents.) The list goes on and on and there are dozens more websites that purport to extract text from PDFs, but none in a substantial, and customized way, like our solution can. (Further solutions found during research but not noteworthy include: Astera, Parseur, Microsoft, Adobe, Taskade, AlgoDocs, Nanonets, OneAI, Parsio, Popai Pro, etc.) I surveyed several student attorneys and former attorney colleagues about these platforms, and no one had used any of them aside from ChatGPT.

In partnering with myself, these are all things I considered when determining what I would want to see in this solution. As a former litigation paralegal that managed all of the firm's asylum cases and training, to being a litigation summer associate who had to research and draft the country condition reports, to an incoming litigation associate, I have both prior experience, as

well as future opportunities to draft these country condition reports and utilize this tool. As the partner, it was important for me to take note of my observations and changes I would like to make after performing user-testing. Please see the below observations I made as the partner, that I problem solved as the developer.

Observation #1

I observed that the notebook could only read documents under 10 pages, so I needed to find shorter articles. But, that the notebook can read unlimited amounts of articles, i.e., can read 100, 4-page articles, but cannot read 4, 100-page articles. *Suggested Update in the Future*: Increase the max tokens/word count/page count so that more accredited, lengthy government articles could be uploaded to the notebook. This will also save additional time for the user, because they will not need to pay attention to the page count, when deciding what articles to upload to the Google Drive and in turn imported into the notebook. However, as the notebook stands, it is a viable product for shorter articles that still contain very useful information.

Observation #2

I realized that I was asking for too many requests/prompts according to the API key, so I spoke with Professor Colarusso and we were able to implement the pausing with python time sleep code. This solved a reoccurring error that would prevent a user from running multiple prompts across multiple articles. Though it increases the amount of time for the notebook to run the prompts, it merely adds an additional ~23 seconds in between each prompt, and can be loading in the background as you do other work. It is also user-friendly to those that do not have a payment on file. *Note:* it is helpful to put a card on file with OpenAPI so that you do not run into errors. This is a nominal cost to the user and can be reloaded any time, or automatically once the account is depleted.

Observation #3

When saying "`prompt_text = """Below you will be provided with the text of a report on conditions in Uganda. You're looking to find quotes about sexual violence. That is, whether or not the report has quotes including sexual violence.`" I realized that it only pulled information that had quotation marks around it within the article. So I had to reword the prompt to "`prompt_text = """Below you will be provided with the text of a report on conditions in Uganda. You're looking to find quotes about gender-based violence. That is, whether or not the report has sentences including the words "gender-based violence". The sentences do not need to be in quotation marks.`" But, this still did not pull great results (*see* Attempt #3 Screenshot at the end of Biography)

So, I again changed the prompts to include something like "`Find all references to sexual violence in these articles`"

Rewording the prompts to pull "all references" seemed to bring up an error and a lengthy quotation. It pulled a long unwieldy response, of several paragraphs. So, I needed to streamline it again. This time, changing the language to, "`two sentences from the article referencing domestic violence.`" This format for the prompts brought back great results (*see* Attempt 4 Screenshot at the end of Biography).

Observation #4

While I downloaded more articles to include in Google Drive, I noticed that some included "Intimate Partner Violence" in the titles, and so on the last iteration of the solution, I added a 5th set of prompts for this key term. By adding this, my results were much more precise.

Observation #5

During my final couple of days of testing I ran into several errors. First, a rate limit error, where I needed to add a credit card to my OpenAPI account, in addition to already adding in a time sleep code so it would lapse 23 seconds before going to the next call. Next, I ran into a 502 API Error, which I walked through with Professor Colarusso.

From these errors, more populated, such as producing results that had "{output:" at the beginning of each column in the results. Professor Colarusso and I walked through the coding and removed the "sample=" and instead inputted "f=" followed by the filename of one of the documents, in order to test out a few options on just one document before it crashes after spending a lot of time on the others. After this, we also removed the "return a json" prompt, and instead included a simple prompt, i.e., "`Return your answer below. If you can't find _if it mentions intimate partner violence_ in the text of the above, answer simply yes or no.""". format(text"` for all of the result prompts. By making these changes, the error messages disappeared, the results in the CSV were formatted, and the amount of time it took the solution to run through the prompts was significantly shorter, i.e., under a minute, rather than 15+ minutes. I then commented out the "f=" so future users could user-test with one document if they made changes, and in this final iteration, it would read all the documents in the folder, which was 10. It also cut down on the cost, and only cost about $6 in OpenAPI tokens. Which, in comparison to what an attorney's billable rate would be, e.g., $200-$250 an hour, is a significant cost savings.

As the partner, I also found it difficult to know how I would update this notebook in the future without the knowledge gained by the developer in discussions with the professor. So, throughout the notebook, specifically before the prompts, I added background notes in green with instructions. I also discussed with my partner that it would be helpful in the future to create country specific folders in the Google Drive, so future users can make a selection of various folders to link to the notebook, while still maintaining the information in the folders that were not being used at that time. *Suggested Update in the Future*: depending on the user, update the notebook to increase or decrease the amount of sentences being pulled from the article, or fine tune the language so it pulls exactly what you are looking for. It should be noted, that decreasing the amount, or pulling a blanket amount of sentences, from the article could lead to missing valuable information. Which, would increase the amount of time needed to cite check the work being extracted. In the next version of this notebook, the language should be fine tuned to make sure that the user is able to get all relevant quotes from the articles, and not just a couple. Currently, the notebook has 5 different topic prompts, with two prompts per topic. See below:

- Topic 1 – Domestic Violence (DV)
    - Prompt 1 – does the article mention Domestic Violence, yes or no
    - Prompt 2 – if yes, pull a couple statements from the article that reference Domestic Violence
- Topic 2 – Violence Against Women (VAW)
    - Prompt 1 – does the article mention Violence Against Women, yes or no

    - Prompt 2 – if yes, pull a couple statements from the article that reference Violence Against Women
- Topic 3 – Sexual Violence (SV)
    - Prompt 1 – does the article mention Sexual Violence, yes or no
    - Prompt 2 – if yes, pull a couple statements from the article that reference Sexual Violence
- Topic 4 – Gender-Based Violence (GBV)
    - Prompt 1 – does the article mention Gender-Based Violence, yes or no
    - Prompt 2 – if yes, pull a couple statements from the article that reference Gender-Based Violence
- Topic 5 – Intimate Partner Violence (IPV)
    - Prompt 1 – does the article mention Intimate Partner Violence, yes or no
    - Prompt 2 – if yes, pull a couple statements from the article that reference Intimate Partner Violence

These four topics and prompts were chosen because those terms are frequently mentioned or used in articles to describe a women's experience in the country they are seeking asylum from, due to domestic violence from their partners or spouses, or other forms of violence against women by third-parties and government officials, such as the police and army forces—which are typically corrupt, especially in Uganda.

Real World Viability and Sustainability:

As mentioned above, this solution could be implemented now as a viable product for extracting relevant statements from short (under 10 page) PDFs. In the future, the coding would need to change to allow for larger documents, such as government reports that typically are over 100 pages, to be reviewed and extracted. There are clear instructions in green within the notebook, of what fields would need to be changed if they would like to change the prompt or columns of the exported csv, by choosing a different country, topic, or question.

Further steps would need to be taken by the firm to ensure data security. However, as previously discussed, since the majority of information included in country condition reports stems from publicly available documents, and not case specific information, I do not anticipate there would be many security concerns with this solution. However, prompts would need to be tailored to publicly available information in order to stay data security compliant.

This solution has the ability to be shared, or restricted, just as the Google Drive (or other document storage platform may be used). The Google Drive, depending on storage available to the main user, can contain an unlimited amount of documents that can be used in the notebook at one time, or saved for later use. The CSV data extract file also is programmed to upload to a subfolder in that Google Drive, so everything is in one place. Further subfolders can be created to include instruction documents, preferred coding/prompt language, history of failed/suboptimal coding/prompt language, and additional resources. Users will need to establish an OpenAPI account in order to use the notebook, but as previously stated, there are free trials, and there are nominal costs with increasing usage. If this solution expands in scope, the user(s) will need to evaluate whether this is a cost effective solution based on token rates, or if re-coding is necessary to run the tool without an API key. Additionally, user(s) will benefit from having Adobe Acrobat downloaded on their devices so they can OCR/text recognize documents.

As previously stated, paralegals, litigation technology support, or associates can work on maintaining and updating the notebook. Updating the prompts is straightforward enough with LLMS and RegEx, that it does not require user(s) to have coding experience. If they require background, there are ample training videos on the internet of how to become familiar with Collaboratory Notebooks. When maintaining and updating, those individuals will want to make sure that folder structures in google drive that are referenced in the notebook (e.g., folder for csv extract; folder hosting PDFs) have not been renamed. Prompts will want to be updated with new language that may capture the information, e.g., if popular wording is established outside of "domestic violence," "gender-based violence" etc., when discussing applicable asylum matters. And subfolders should be created that host articles organized by year or country, since this criteria changes with each new case and each new country condition report. A date range for one client, even if from the same country, might be different for another client, because it is based on the years the alleged abuse/torture occurred. Providing information regarding the condition of that country for a date that is not relevant, will not set over well with a hearing officer/judge, and will be irrelevant to the case.

I am willing as an incoming litigation associate, to continue to use this solution and maintain/update it, because it directly relates to my anticipated *pro bono* work. Just by performing user-testing, I was able to see the amount of time I would need to dedicate reviewing documents decrease. There were several sample documents that I downloaded without reading, that claimed to discuss domestic violence, but the solution was able to spot that it in fact did not, i.e., the "4 page_COPY_Uganda-2018" document. In others, it was able to extract relevant information about domestic violence that did not mention the words "domestic violence," but by training the system, it knew that the sentence was related to domestic violence, and therefore relevant.

CSV Data Extracts after Prompt Re-Wording Attempts

Attempt 1: in shared Google Drive folder here: https://drive.google.com/file/d/16GlfawNZx6B3EPFo-jzebBnhFRKyRGIn/view?usp=drive_link

Attempt 2: in shared Google Drive folder here: https://drive.google.com/file/d/11oeG3-4PvI9eSvlxnCDss9vv6QOjcYsn/view?usp=drive_link

Attempt 3: in shared Google Drive folder here: https://drive.google.com/file/d/1ELcbrH8aVc7L7Z6pqix3xMdVFWKMDN41/view?usp=drive_link

Attempt 4: in shared Google Drive folder here: https://drive.google.com/file/d/10yxfhWPHDZHkQ5gYv7nt6dh9ITy4zeoK/view?usp=drive_link

Attempt 5 (Final—will be similar to user-testing): in shared Google Drive folder here: https://drive.google.com/file/d/1RZYH0IQCGc3XiIZXrYKgZSbmANkwNIlL/view?usp=drive_link

From: Leah Marszalek
To: Leah Marszalek
RE: Partner Letter re Satisfaction and Upkeep

Leah, great work on putting together this Colab for extracting data from OCR'ed PDFs. This tool will be extremely useful for efficiently and timely managing our *pro bono* asylum programs. This solution significantly decreases the amount of time of reviewing the documents, and is a drop in the bucket for cost, i.e., ~$6A in API tokens. Although there is always room for improvement, and additional capabilities, this minimal viable product will enable myself and others at the firm to quickly search for articles, upload them to this Colab Notebook, and run a solution that will (1) tell me if there even is a reference to the terms needed, and (2) if yes, a search for the key words of necessary phrases to enter into Country Condition Reports.

PROS: I like that you can upload an unlimited amount of files into the Google Drive folder, and that the simplification of the prompt in your Attempt #5 made it so the solution could review the documents and spit out an answer in tables and graphs in a matter of seconds, rather than over 15 minutes (as seen in the earlier attempts). Further, the ability to export those findings into a csv file, with customized column headings, and also preferred answers, i.e., "yes or no" or "N/A – no quotes re X found." This saves a lot of time reading lengthy articles that may only have 1 or 2 quotes, by more efficiently targeting articles in the first instance that have the most mentions. This will enable me, as the reader, to prioritize certain articles first. I also appreciate that all files can be kept in one place on google drive, and you do not need to re-upload files each time you run the solution, like some other platforms require. It removes another unnecessary step, and allows for future users to see which documents have already been run through, what the results are, and to store them in an organized manner. Thank you for also adding in instructions throughout the prompt cells, once you finalized the tool, so that future users know what they can and cannot change, and how to leave comments themselves when they add in new features.

UPDATES IN THE FUTURE: Ability to add larger articles to search. While there are many accredited websites that review the conditions of countries, a majority of articles produced by the government are hundreds of pages and may have the most valuable pieces of information. I know this was an obstacle you faced while creating the solution, by having to find only articles that were less then 10 pages, and then also running into issues when a 9 page document had too many words, so that went over the token amount. In the future it would also be great to add in coding for a bluebook citation for the source, and page numbers, since this is required to include in a country condition report. Fine tuning the prompt language so it is pulling all reference, rather than just two sentences, will also be helpful because it minimizes the amount of time that someone has to cite check to make sure all quotes were pulled from the article.

CONTINUED UPKEEP: Collaborating with more partners at the firm, so they can continuously upload documents to google drive that they have already downloaded (whether or not helpful) for their cases. This will save attorneys time by limiting—or eliminating all together—the first step of researching past articles. If attorneys are able to continuously upload articles to these folders, and organize by countries, topics, and dates, the use of this Colab will expand greatly past just articles regarding the country of Uganda, and the topic of Domestic Violence, Sexual Violence, Gender-Based Violence, Violence Against Women, and Intimate Partner Violence. There are many more countries and many more topics that appear in our asylum matters and for which we have to compile country conditions. In addition to adding more articles, more prompts can be added to the Colab notebook, so the tool can expand on the information it is pulling out.

Colab Text Extraction for Country Condition Reports

by Leah Marszalek

Country Condition Reports

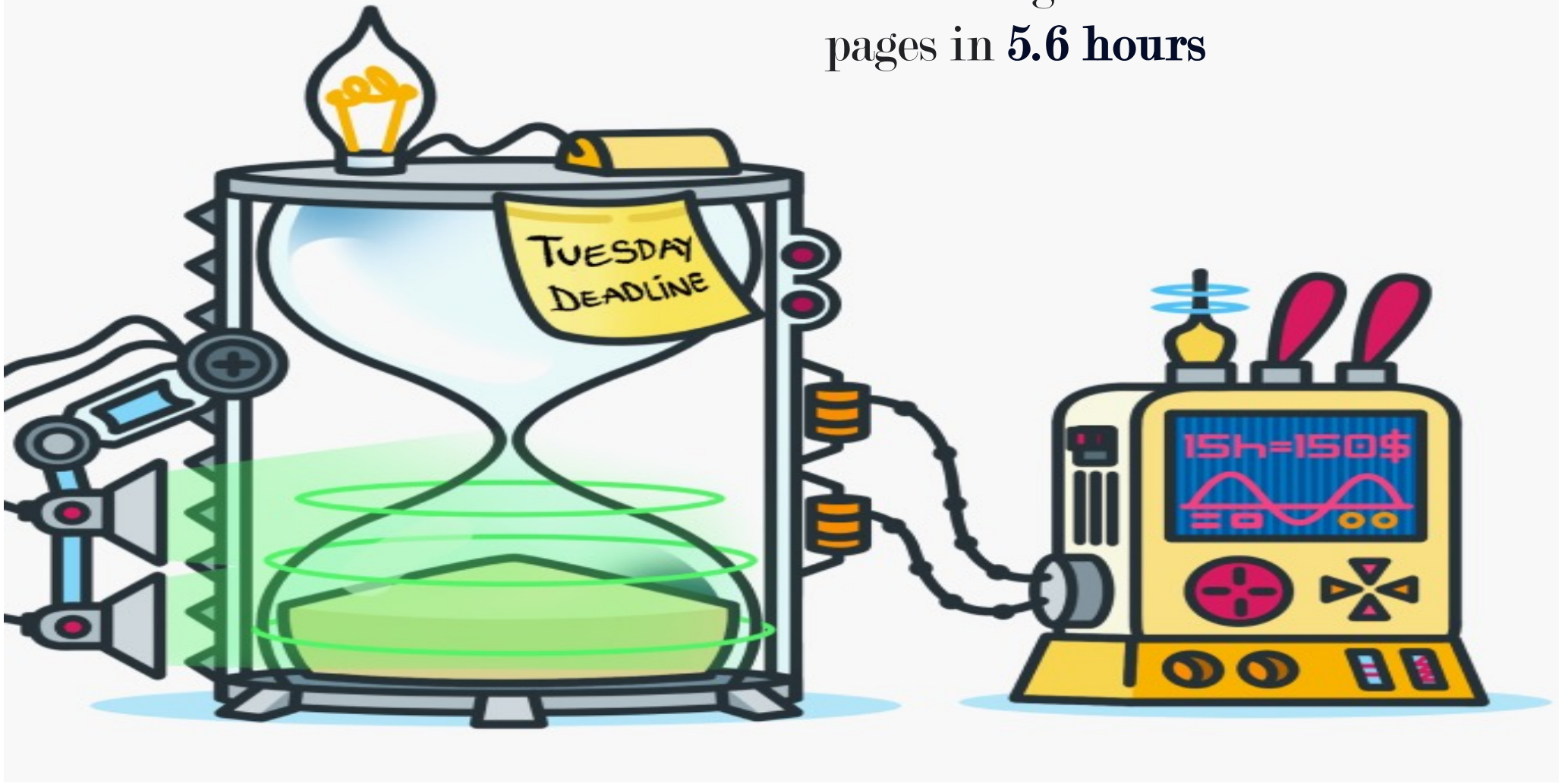
To win asylum, an applicant must demonstrate that they fear being seriously harmed in their country of origin.

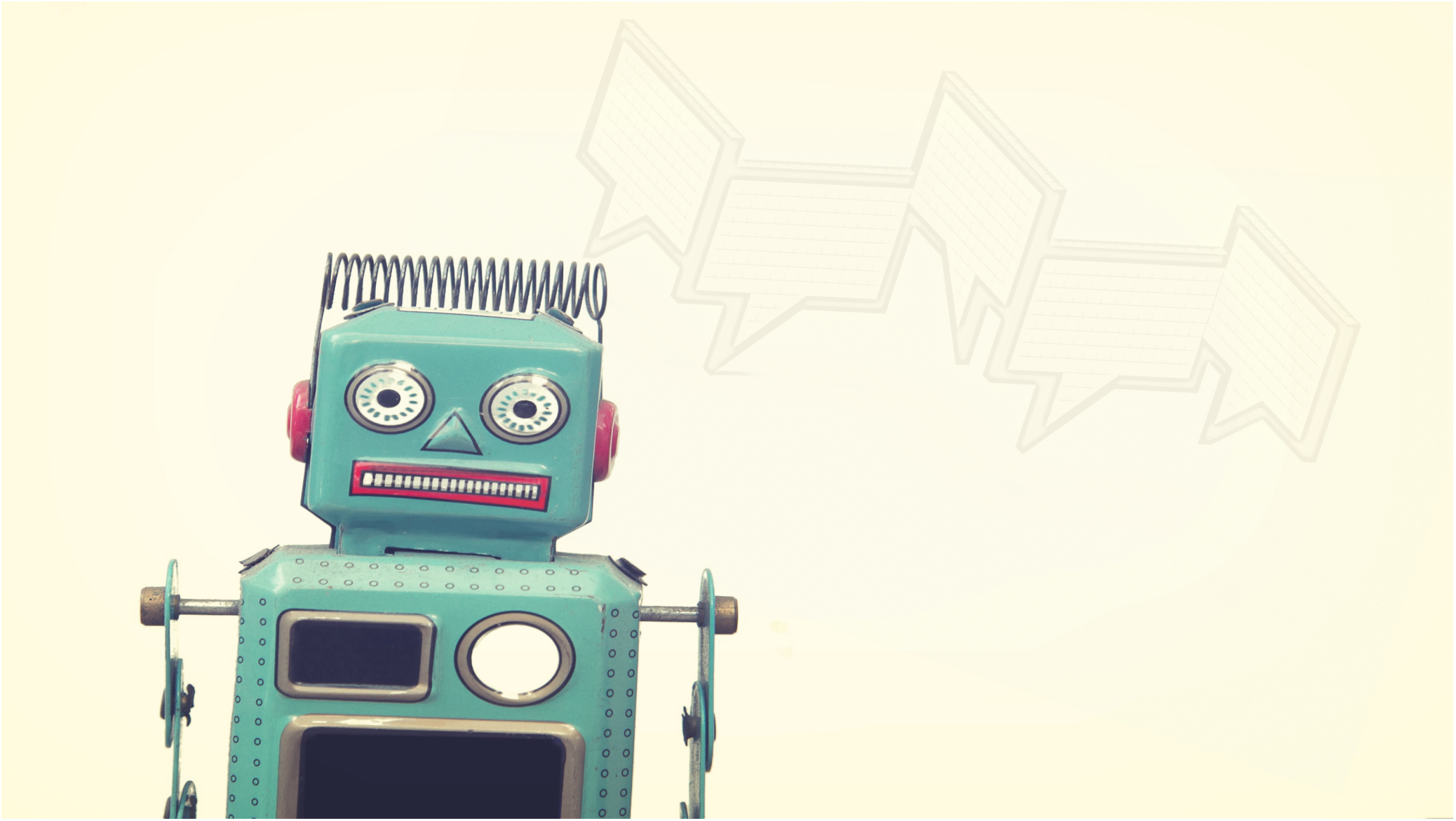
Often, applicants establish this by submitting documents such as reports from national and international human rights and governmental organizations and news articles describing the types of harm the applicant suffered or fears suffering in the future in their country of origin.

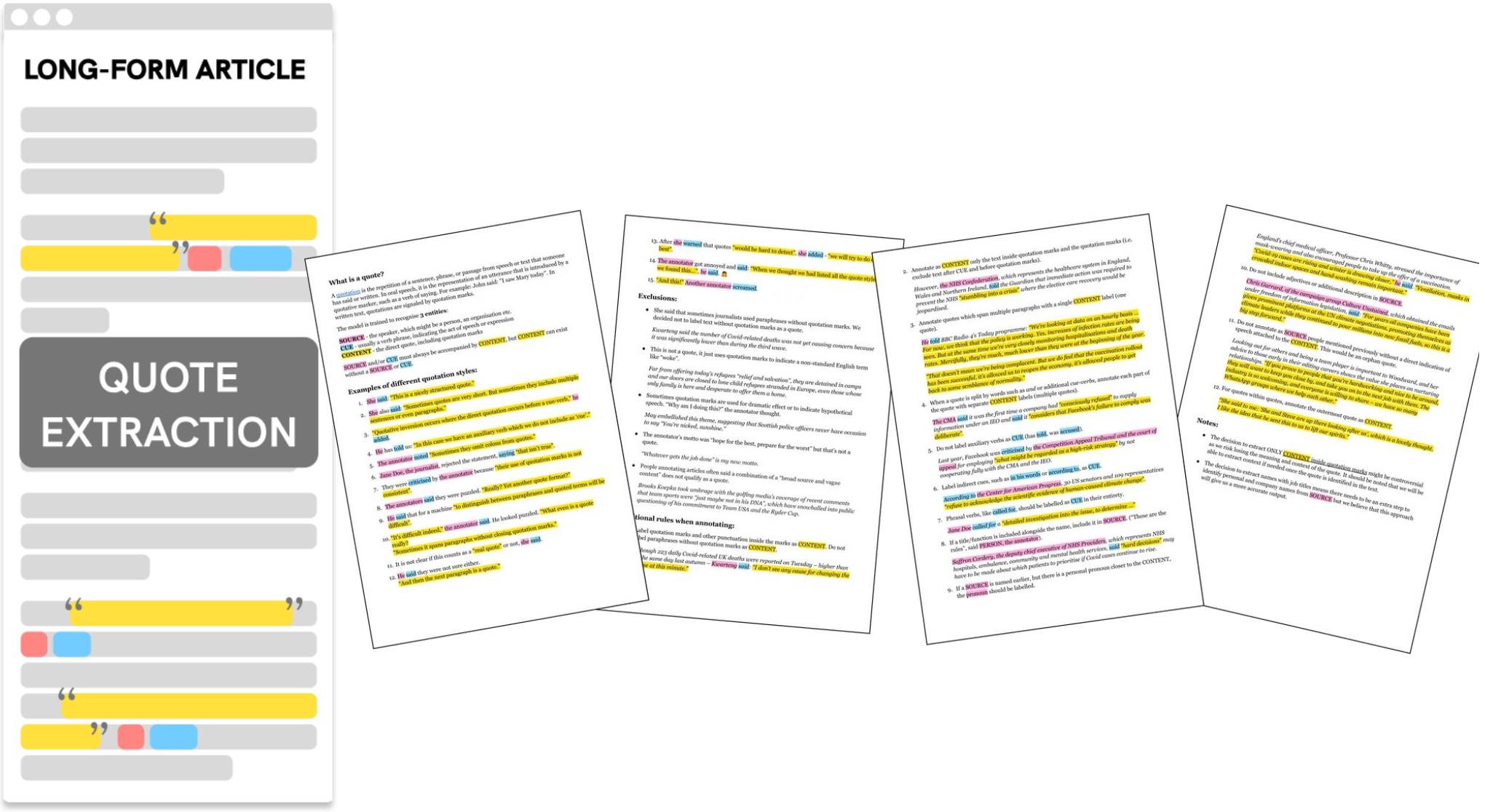
This evidence is generally referred to as "country conditions."

Country conditions packets typically consist of an index (highlighting important quotations from specific reports and articles) followed by printouts of the full documents.

The average reader will read 200 pages in **5.6 hours**






LONG-FORM ARTICLE
QUOTE EXTRACTION
What is a quote?
Examples of different quotation styles:
Exclusions:
Notes:

Let's stop reinventing the wheel.

someecards
user card