Jordan Frimpter JEF180001
Henry Kim HTK180000

CS 4395.001 Human Language Technology
Dr. Mazidi

Chatbot Report: Barry the ACL Librarian (and Baxter the Puppetmaster)

Summary

Our chatbot was inspired by the project where we summarized a paper from the ACL Anthology. Our chatbot is designed to assist someone in researching natural language processing using the ACL anthology. Users can ask the bot about different papers and authors, or simply chat with the bot. The chatbot will do a live lookup for information on authors and papers the user asks about, drawing from the ACL Anthology website.

System Description

The chatbot system is composed of two chatbots: the rules-based Baxter and the more probabilistic Barry, with Barry in a subservient position to Baxter.

Barry

Barry is a chatbot generated using the ChatterBot package and yaml training documents with example conversation flows for different topics. Barry is a retrieval based chatbot that accepts inputs from a user and then finds the training sentence it has that most closely matches user input before outputting the paired response. Barry is responsible for off-topic conversation to give the chatbot a semblance of personality and to fill conversations with substance.

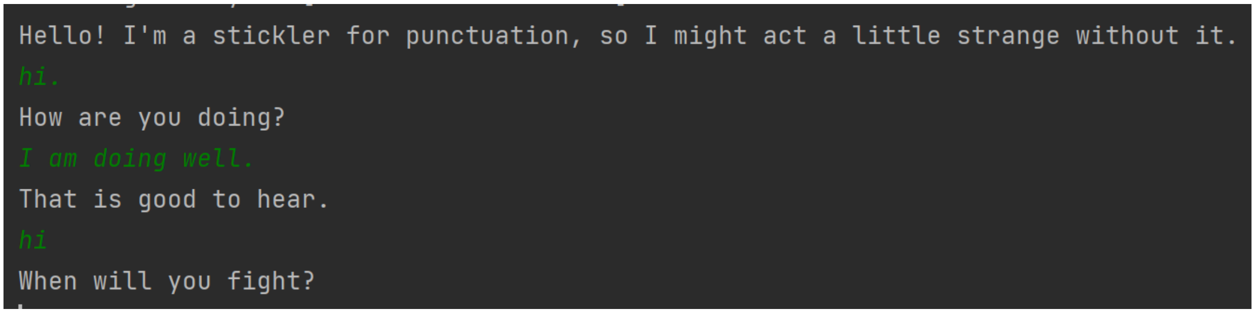
Barry is trained on a series of .yml files, some of which were sourced from the chatterbot.english corpus and some of which were designed entirely for the task of this project. The files in the corpus were cleaned and reengineered to give Barry a smoother and more consistent response style. Yaml files are read in with the following format:

```
- - Conversation Root
 - Line 1
 - Line 2
 - Line 3….
- - New Conversation Root
…
```

The Chatterbot model will train each line as the successor to the line prior, resetting when a new conversation root is reached. A limitation, however, is that the relationship can also be reversed; in some cases the chatbot will respond with a conversation root and not a subsequent line. For this reason, the training files needed to be cleaned to have a consistent pattern of grammar and for more obscure topics to make sense with less context. Lines that were irrelevant to the subject of ACL Anthology retrieval were screened to include only those that added to the personality of the bot without increasing its volatility, those that were not offensive to the target audience, and those that make sense in the context of the bot. For example, lines like “Have a good day” were retained, whereas lines like “Dr. Smith will see you now” were omitted.

In addition, positive statements should be paired with a negative statement that expresses the same idea, or the positive statement will match both a positive and negative expression of the same idea. For example, queries like “Do you sleep?” should have an inverse version of “Do you not sleep?” to prevent the same answer for being given to both queries due to similarity.

The model encounters a severe limitation in regards to punctuation, as demonstrated in the following example (green is user, white is bot):



```
Hello! I'm a stickler for punctuation, so I might act a little strange without it.
hi.
How are you doing?
I am doing well.
That is good to hear.
hi
When will you fight?
```

The model is unable to properly discern some prompt sentences that lack punctuation, but since the model is capable of returning the conversation root phrase as a possible response, this limitation to the model is not easily overcome without rewriting the model’s functionality.

The following is a table of files in the chatterbox.english corpus, along with a summary of their contents and whether we used them to train Barry. Note that all training files used were grammatically cleaned, screened of irrelevant sentences, and filled with negated queries (along with a few additional conversation prompts).

Corpus	**Used in our chatbot**	**Summary**
ai	yes	A series of ai-related questions, specifically to emulate a personality of a person whose identity is that of an ai
botprofile	yes	A series of personal details for an ai
computers	no	A series of factual computer-related statements

emotion	yes	Vague responses describing an ai’s lack of emotion
food	yes	Vague responses regarding the foods an ai eats
gossip	no	Gossip conversation lines
greetings	yes	A series of greeting protocols.
health	no	Extremely short conversation regarding someone feeling unwell
history	no	Short corpus of a few history questions
humor	no	A series of jokes
literature	no	A series of facts about literature
money	no	A series of exchanges involving money
movies	no	A poorly cleaned file of movie references
politics	no	A poorly cleaned file of government facts
psychology	no	A poorly cleaned file of what appears to be a series of insults
science	no	A small file of science facts
sports	No	A very poorly cleaned file discussing sports
trivia	no	A small file of trivia facts

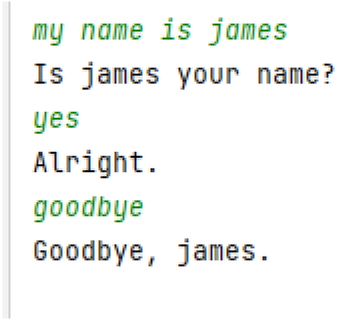
In addition to these files, we included a few more customized training files to give the bot a more task-specific domain of knowledge. Barry’s innate knowledge base for personality is stored in a database called ‘db.sqlite3’. The model needs to be trained only once and will be trained if the ‘db.sqlite3’ file is not found in the same directory as the chatbot scripts.

One drawback of the Chatterbot package is that it is not actively being maintained, so there are many branches that fix different bugs. One bug encountered is that a conversation line in a .yml file consisting of a short token without punctuation will cause a crash. Another drawback is that the model is permitted to select from among the conversation roots for potential response lines, and that among multiple duplicated conversation roots and possible branches of conversation, only one will be chosen and favored.

Initially, we were going to have Barry recognize when the user is attempting a specific function by identifying a conversation root with the user’s input and responding with a special token that could be identified with a switch-like series of statements to perform a function. Barry proved to be too unreliable in accurately tagging the statements, and thus Baxter was introduced to overcome some of Barry’s limitations.

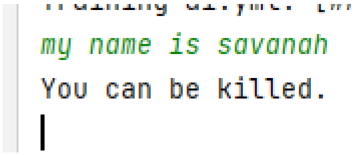
Baxter

Baxter is a rules-based chatbot that identifies user intentions from the presence and distribution of certain key tokens. Baxter was introduced to make user function calling more consistent and reliable. When the chatbot receives user input, it is passed through a function called “seek” that returns an array of token codes and function codes. The different function codes and token codes are interpreted to determine the user intention with the current state of the program in the main chatbot file. Baxter controls the chatbot’s overall output, allowing Barry to control the response only when Baxter detects no attempted function from the user input.



```
my name is james
Is james your name?
yes
Alright.
goodbye
Goodbye, james.
```

Baxter’s interpretation of user inputs. (user is green)



```
my name is savanah
You can be killed.
```

Barry’s interpretation of inputs (unreliable) (user is green)

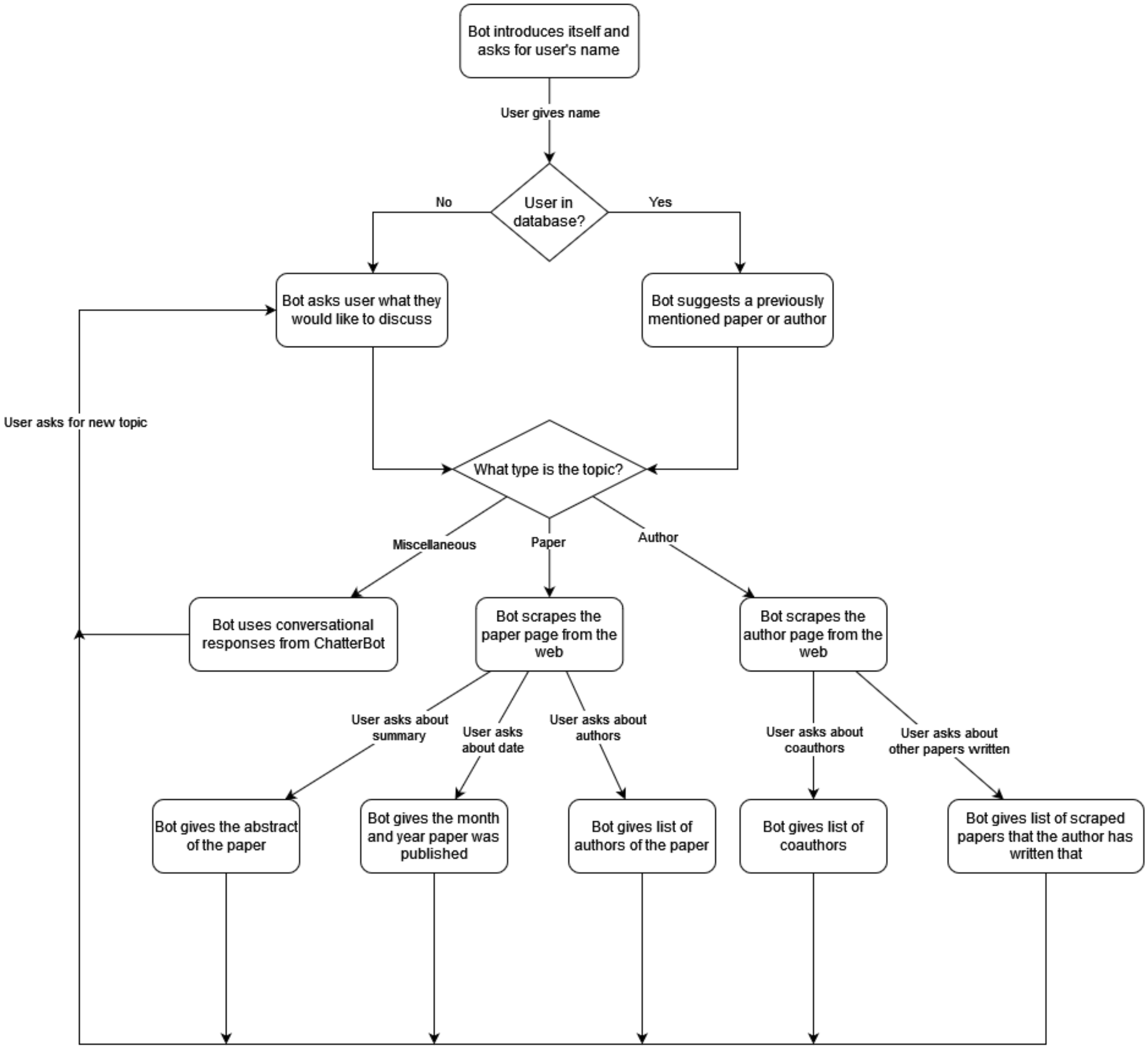
The seek function uses lowercase tokenizing of input to match a series of token codes with the input. The token codes do not use lemmatization and instead each token accounts for several different variations of words to satisfy the user’s intentions. For example, to identify the ‘TOKEN_AUTHOR’ token, the words “author”, “authored”, “authors”, “wrote”, and the presence of the coauthor token can each trigger the author token to be identified.

The seek function then uses combinations of different tokens to identify if the user is trying to perform specific functions, such as provide a name or exit the program. The seek function returns the set of identified functions and tokens as an array of codes, which can then be interpreted by the chatbot main control flow.

The Baxter bot has many helper functions, such as a proper noun filter that uses the nltk tokenization and named entity recognition in spacy to determine what and where a proper noun might be, for the purpose of parsing a name from a sentence. If the conversation partner is informal, this can sometimes fail (for example, lowercase ‘james’ does not always ping as a named entity or a proper noun). For this reason, there is a validation loop for when a user’s input fails to conform to the named entity recognition system’s expectations for reality.

The Baxter bot tracks the conversation state as a topic, a subtopic, and a user. The subtopic can become the main topic if certain prompts are given. A token counting function measures the similarity between user inputs and a list of authors specifically when the user wants to talk about an author when the authors are subtopics; the author with the greatest similarity is returned as the selected author, and no author is returned when there are no authors with a similarity greater than 0 to the user input text. Similarity here is measured in size of token intersection.

Dialog Tree



Bot introduces itself and asks for user's name
User gives name
User in database?
No
Yes
Bot asks user what they would like to discuss
Bot suggests a previously mentioned paper or author
User asks for new topic
What type is the topic?
Miscellaneous
Paper
Author
Bot uses conversational responses from ChatterBot
Bot scrapes the paper page from the web
Bot scrapes the author page from the web
User asks about summary
User asks about date
User asks about authors
User asks about coauthors
User asks about other papers written
Bot gives the abstract of the paper
Bot gives the month and year paper was published
Bot gives list of authors of the paper
Bot gives list of coauthors
Bot gives list of scraped papers that the author has written that

This dialogue tree is not comprehensive, but provides a good abstraction of the function flow.

At almost any point in time, if the user does not give a response that is parsable as a command, conversational responses are used without breaking the ‘topic’. Furthermore, there are many looped ways to switch between topics and at almost any point in time the user can exit the program by triggering an exit token (such as ‘goodbye’). A user also has the option to change their name (and associated lookup profile).

Sample Dialog

Here is some sample dialog focusing on the ACL Anthology functions with a new user:

Hello! I'm Barry, and I can help you with your ACL anthology research. I'm a stickler for punctuation, so I might act a little strange without it.
Can you tell me your name?

My name is Henry.

Is Henry your name?

Yes.

Welcome, Henry. What would you like to talk about? An ACL paper, author, or something else?

I want to find a good ACL paper.

Just to clarify, would you like to talk about authors or papers?

A paper.

Ok, let's talk about a paper. What is the subject or title of the paper you want to talk about?

Dementia

You have said 'Dementia'. Is this correct?

Yes

Let me go look for that...
I found a paper called 'GPT-D: Inducing Dementia-related Linguistic Anomalies by Deliberate Degradation of Artificial Neural Language Models'. I can tell you about its authors, abstract, and publish date.

who are the authors?

Let me go looking...
Here are the authors of the paper 'GPT-D: Inducing Dementia-related Linguistic Anomalies by Deliberate Degradation of Artificial Neural Language Models':

1. Changye Li
2. David Knopman
3. Weizhe Xu
4. Trevor Cohen
5. Serguei Pakhomov

What is the paper about?

Here's the abstract for 'GPT-D: Inducing Dementia-related Linguistic Anomalies by Deliberate Degradation of Artificial Neural Language Models':" Deep learning (DL) techniques involving fine-tuning large numbers of model parameters have delivered impressive performance on the task of discriminating between language produced by cognitively healthy individuals, and … [Response truncated]

When was it published?

The paper 'GPT-D: Inducing Dementia-related Linguistic Anomalies by Deliberate Degradation of Artificial Neural Language Models' was published on May, 2022

let's talk about something else.

We are currently discussing the paper 'GPT-D: Inducing Dementia-related Linguistic Anomalies by Deliberate Degradation of Artificial Neural Language Models'.
Just to clarify, would you like to talk about authors or papers?

I want to discuss an author.

Ok, let's talk about an author. What is the name of the author you want to talk about?

Sergui Pakhomov.

You have said 'Sergui Pakhomov.'. Is this correct?

yes

Let me go look for that...
I found an author named Serguei Pakhomov. I can tell you about their papers and co-authors.

Tell me about the papers they have written.

Let me go looking...
Here are some papers written by Serguei Pakhomov:
1. GPT-D: Inducing Dementia-related Linguistic Anomalies by Deliberate Degradation of Artificial Neural Language Models
2. Everyday Living Artificial Intelligence Hub
3. Conversational Agent for Daily Living Assessment Coaching Demo
4. A Tale of Two Perplexities: Sensitivity of Neural Language Models to Lexical Retrieval Deficits in Dementia of the Alzheimer’s Type
5. What Analogies Reveal about Word Vectors and their Compositionality

Who are the coauthors?

Here are Serguei Pakhomov's coauthors:
1. Bridget McInnes
2. Ted Pedersen
3. Christopher Chute
4. James Buntrock
5. Genevieve B. Melton
[Response truncated]

Thanks

You're very welcome.

goodbye for now

Is it time for you to leave?

yes

Goodbye, Henry.

Here is sample dialogue with a returning user:
Hello! I'm Barry, and I can help you with your ACL anthology research. I'm a stickler for punctuation, so I might act a little strange without it.
Can you tell me your name?

I am Henry.

Is Henry your name?

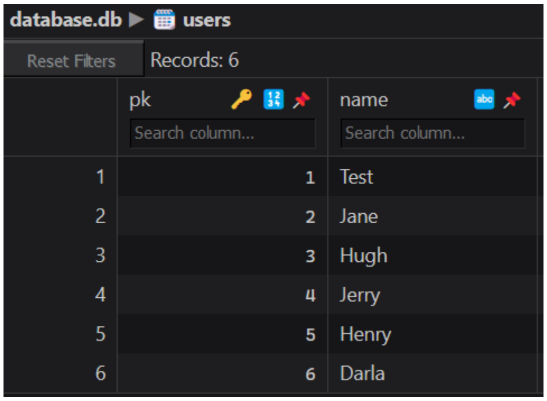
yes.

```
Welcome back, Henry.
Since we last spoke, I've been thinking about the paper 'GPT-D:
Inducing Dementia-related Linguistic Anomalies by Deliberate
Degradation of Artificial Neural Language Models' that I previously
mentioned to you.

Can you remind me when it was published?

The paper 'GPT-D: Inducing Dementia-related Linguistic Anomalies by
Deliberate Degradation of Artificial Neural Language Models' was
published on May, 2022
```

Knowledge Base and User Models

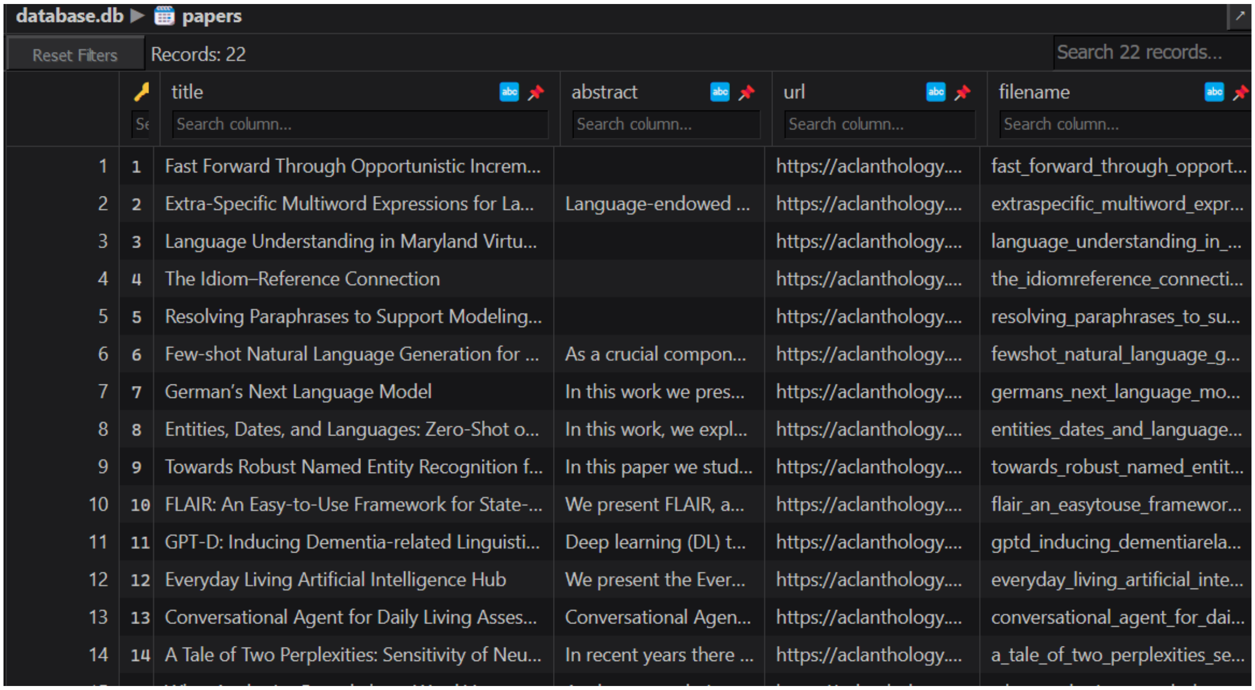
The backend for the knowledge base and user models are combined into a single SQLite database file that records the titles of papers, who wrote the papers, and the volume that the papers were published in. These papers are added to the database and downloaded by the bot as the user asks about them. The database also keeps track of when it gives the user a piece of information so that it can track which user asked about which topic.

There are nine tables in the database: users, papers, authors, volumes, explored_papers, explored_authors, written_by, coauthors, and published_in.



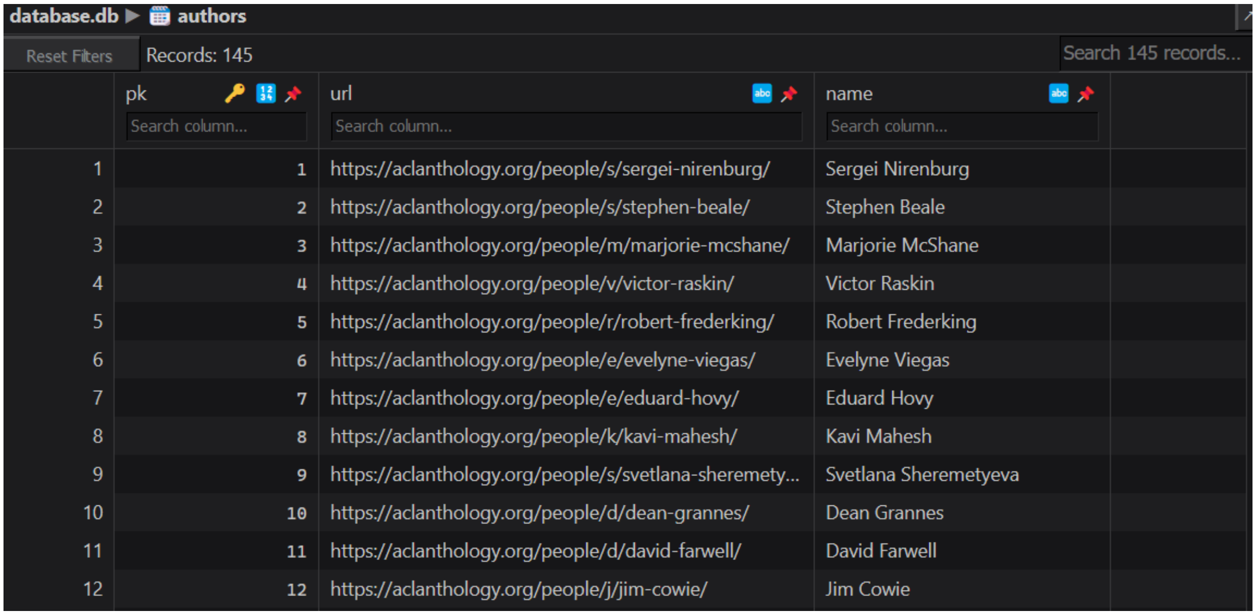
	pk	name
1	1	Test
2	2	Jane
3	3	Hugh
4	4	Jerry
5	5	Henry
6	6	Darla

The users table consists simply of the user’s name and a primary key.



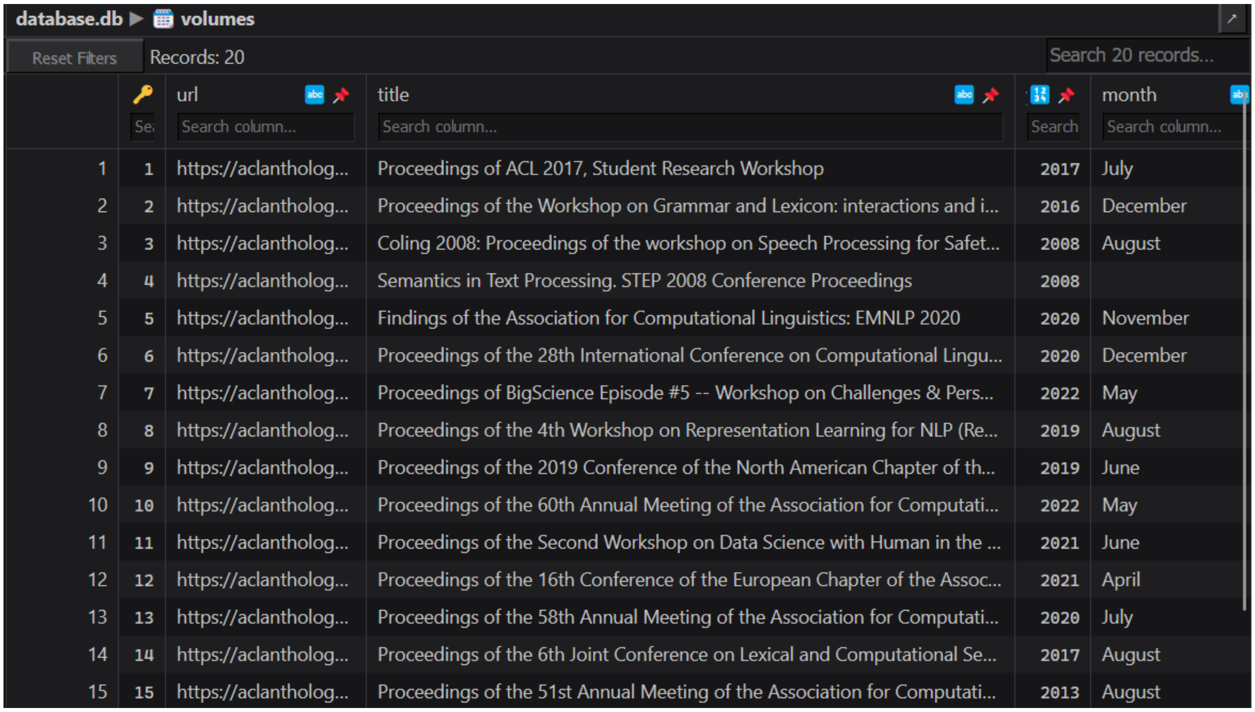
		title	abstract	url	filename
1	1	Fast Forward Through Opportunistic Increm...		https://aclanthology....	fast_forward_through_opport...
2	2	Extra-Specific Multiword Expressions for La...	Language-endowed ...	https://aclanthology....	extraspecific_multiword_expr...
3	3	Language Understanding in Maryland Virtu...		https://aclanthology....	language_understanding_in_...
4	4	The Idiom–Reference Connection		https://aclanthology....	the_idiomreference_connecti...
5	5	Resolving Paraphrases to Support Modeling...		https://aclanthology....	resolving_paraphrases_to_su...
6	6	Few-shot Natural Language Generation for ...	As a crucial compon...	https://aclanthology....	fewshot_natural_language_g...
7	7	German's Next Language Model	In this work we pres...	https://aclanthology....	germans_next_language_mo...
8	8	Entities, Dates, and Languages: Zero-Shot o...	In this work, we expl...	https://aclanthology....	entities_dates_and_language...
9	9	Towards Robust Named Entity Recognition f...	In this paper we stud...	https://aclanthology....	towards_robust_named_entit...
10	10	FLAIR: An Easy-to-Use Framework for State-...	We present FLAIR, a...	https://aclanthology....	flair_an_easytouse_framewor...
11	11	GPT-D: Inducing Dementia-related Linguisti...	Deep learning (DL) t...	https://aclanthology....	gptd_inducing_dementiarela...
12	12	Everyday Living Artificial Intelligence Hub	We present the Ever...	https://aclanthology....	everyday_living_artificial_inte...
13	13	Conversational Agent for Daily Living Asses...	Conversational Agen...	https://aclanthology....	conversational_agent_for_dai...
14	14	A Tale of Two Perplexities: Sensitivity of Neu...	In recent years there ...	https://aclanthology....	a_tale_of_two_perplexities_se...

The papers table consists of the paper's title, the paper's abstract, the file name that the PDF version is stored under, the primary key, and the URL to the page on the ACL Anthology website.



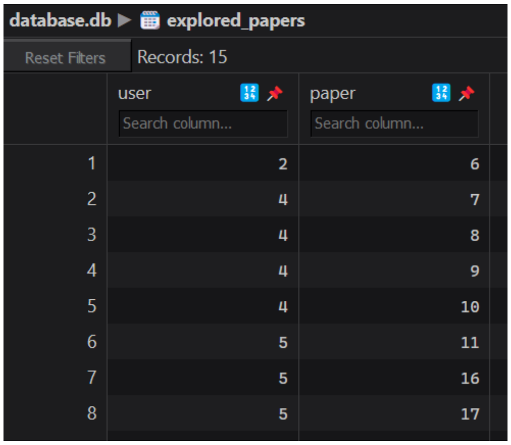
	pk	url	name
1	1	https://aclanthology.org/people/s/sergei-nirenburg/	Sergei Nirenburg
2	2	https://aclanthology.org/people/s/stephen-beale/	Stephen Beale
3	3	https://aclanthology.org/people/m/marjorie-mcshane/	Marjorie McShane
4	4	https://aclanthology.org/people/v/victor-raskin/	Victor Raskin
5	5	https://aclanthology.org/people/r/robert-frederking/	Robert Frederking
6	6	https://aclanthology.org/people/e/evelyne-viegas/	Evelyne Viegas
7	7	https://aclanthology.org/people/e/eduard-hovy/	Eduard Hovy
8	8	https://aclanthology.org/people/k/kavi-mahesh/	Kavi Mahesh
9	9	https://aclanthology.org/people/s/svetlana-sheremety...	Svetlana Sheremetyeva
10	10	https://aclanthology.org/people/d/dean-grannes/	Dean Grannes
11	11	https://aclanthology.org/people/d/david-farwell/	David Farwell
12	12	https://aclanthology.org/people/j/jim-cowie/	Jim Cowie

The authors table consists of the author's name, a primary key, and the URL to the author's page on the ACL Anthology website.



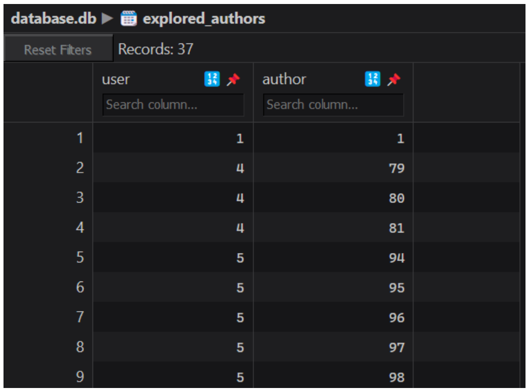
		url	title		month
1	1	https://aclantholog...	Proceedings of ACL 2017, Student Research Workshop	2017	July
2	2	https://aclantholog...	Proceedings of the Workshop on Grammar and Lexicon: interactions and i...	2016	December
3	3	https://aclantholog...	Coling 2008: Proceedings of the workshop on Speech Processing for Safet...	2008	August
4	4	https://aclantholog...	Semantics in Text Processing. STEP 2008 Conference Proceedings	2008	
5	5	https://aclantholog...	Findings of the Association for Computational Linguistics: EMNLP 2020	2020	November
6	6	https://aclantholog...	Proceedings of the 28th International Conference on Computational Lingu...	2020	December
7	7	https://aclantholog...	Proceedings of BigScience Episode #5 -- Workshop on Challenges & Pers...	2022	May
8	8	https://aclantholog...	Proceedings of the 4th Workshop on Representation Learning for NLP (Re...	2019	August
9	9	https://aclantholog...	Proceedings of the 2019 Conference of the North American Chapter of th...	2019	June
10	10	https://aclantholog...	Proceedings of the 60th Annual Meeting of the Association for Computati...	2022	May
11	11	https://aclantholog...	Proceedings of the Second Workshop on Data Science with Human in the ...	2021	June
12	12	https://aclantholog...	Proceedings of the 16th Conference of the European Chapter of the Assoc...	2021	April
13	13	https://aclantholog...	Proceedings of the 58th Annual Meeting of the Association for Computati...	2020	July
14	14	https://aclantholog...	Proceedings of the 6th Joint Conference on Lexical and Computational Se...	2017	August
15	15	https://aclantholog...	Proceedings of the 51st Annual Meeting of the Association for Computati...	2013	August

The volumes table consists of the volume's title, the month the volume was published, the year the volume was published, a primary key, and the URL to the volume's page on the ACL Anthology website.



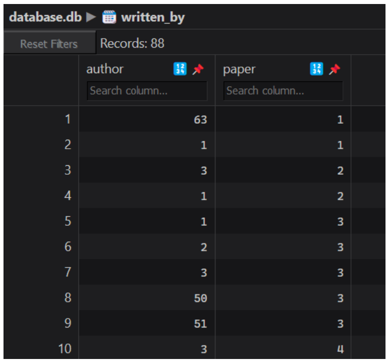
	user	paper
1	2	6
2	4	7
3	4	8
4	4	9
5	4	10
6	5	11
7	5	16
8	5	17

The explored_papers table tracks the many-to-many relation of users exploring a specific paper. The table stores the primary key of a user and the primary key of a paper they explored.



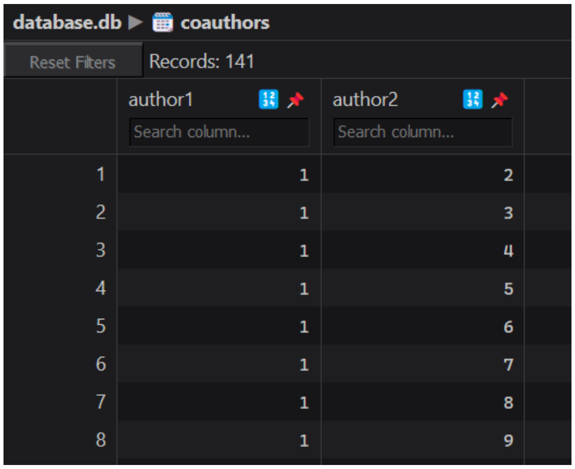
	user	author
1	1	1
2	4	79
3	4	80
4	4	81
5	5	94
6	5	95
7	5	96
8	5	97
9	5	98

The explored_authors table tracks the many-to-many relation of users exploring a specific author. The table stores the primary key of a user and the primary key of an author they explored.



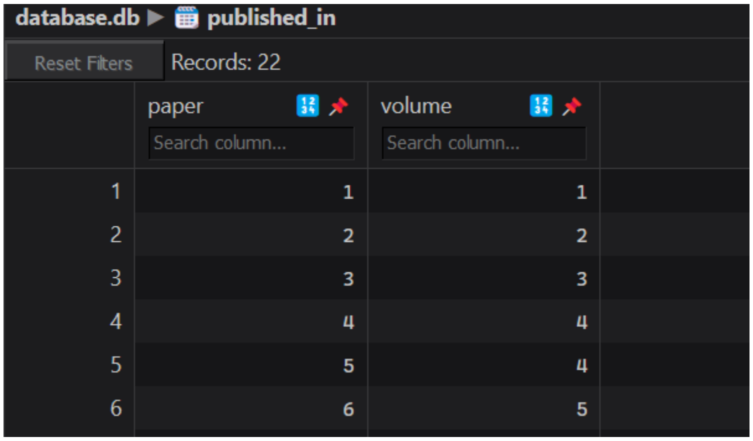
	author	paper
1	63	1
2	1	1
3	3	2
4	1	2
5	1	3
6	2	3
7	3	3
8	50	3
9	51	3
10	3	4

The written_by table tracks the many-to-many relation of paper authorship. The table stores the primary key of a paper and the primary key of an author that wrote it.



	author1	author2
1	1	2
2	1	3
3	1	4
4	1	5
5	1	6
6	1	7
7	1	8
8	1	9

The coauthors table tracks the many-to-many relation of coauthorship. The table stores the primary key of an author and the primary key of an author that is listed as a coauthor on the ACL Anthology website. Although in reality this relationship is symmetrical, our table tracks each side of the relationship separately and only adds the coauthors of an author when that author is explored by the user. For example if Mary and John have written a paper together but the user has only talked about Mary with the bot, then the database will know John is a coauthor of Mary but not the other way around. This ends up being something of a moot point since if the user asks about Mary and is told John is her coauthor, the user can then ask about John and John's webpage will be scraped, adding Mary as a coauthor to John.



	paper	volume
1	1	1
2	2	2
3	3	3
4	4	4
5	5	4
6	6	5

The published_in table tracks the many-to-one relation of a paper being published in a specific volume. The table stores the primary key of a paper and the primary key of the volume. This lets the user find the date the paper was published as well.

Evaluation

Strengths

The chatbot can assist in research by tracking papers similarly to the ACL search engine. What makes the chatbot more useful is that it also downloads the papers the users are interested in so they can reference them in their own research later on. The chatbot can provide the user with the download location for papers as well so the user does not need to go hunting through a folder of .pdf files.

The other strength of our system design is the combination of probabilistic and rules-based approaches. Human language is hard to predict, can be ambiguous, and has a lot of variance; however, it also follows grammatical structures. When the user asks questions or says things outside of the planned domain of the ACL Anthology, we use Barry to handle conversation in a probabilistic way. When the user asks about an author or paper, we use Baxter to keep the conversation structured and track an internal model of the subject. The rules-based approach structure works well for discussing papers and authors because we can get data from a source that has already organized and categorized the information such as which authors wrote which paper.

In future development, the chatbot could be expanded to other features that assist in writing papers, such as searching for keywords across downloaded .pdf files if a user wants to read more about a specific topic or if a user wants to remember what paper a specific idea came from.

Web Scraping Limitations

There are some limitations to our approach. First, we heavily rely on the structure of the HTML and organization of the ACL Anthology website to scrape structured data. For example, we assume that title text has the h2 tag and that author pages have the word ‘people’ in their URLs. Additionally, not all papers have their abstracts on the website.

We use the ACL Anthology search page instead of web-crawling the entire website ourselves; however, this approach comes with some disadvantages. This page is a front end for a Google Custom Search, so it relies on Google’s indexing. While this generally provides better results than we could find on our own, ACL notes that it does not work for pages that have not yet been indexed by Google. For example, when searching for the author David Knopman our bot was unable to retrieve results since his page had not been indexed for the custom search at the time.

Furthermore, the search results require the use of JavaScript to render, so we use the requests-html Python library to run JavaScript on the HTML. If too many requests are made to the search page in a short amount of time, Google may detect your IP address as a bot and stop providing results. This can be overcome by manually opening the ACL Anthology in a web browser and completing the Captcha.

Chatbot Limitations

The rules-based approach is limited in that human creativity is a powerful tool and not all the possible structures for command requests will be accurately captured by a rules-based approach, only the most common request forms will be.

On the other hand, the probabilistic retrieval approach can fail easily when sentences are similarly structured but feature radically different topics. The Chatterbot package itself is limited and can be expanded. It does not disregard punctuation as thoroughly as it could, and it could be limited to prevent the chatbot from prompting the user with a conversation root to increase the accuracy of its responses. It could be further expanded to rank similarity from lemmas and the similarity of the semantics of words.

How to Run

1. Make sure the following files and folders are in the same directory:

- `chatterbot-corpus-master-edit` folder
    - Contains the edited chatterbot corpus files
    - (separately stored due to a different distribution license)
- `new-corpus` folder
    - Contains the task-specific chatterbot files
- `jef180001_htk180000_backend.py` script
- `jef180001_htk180000_baxter.py` script
- `jef180001_htk180000_baxter_conversation.py` script
- `jef180001_htk180000_chatbot.py` script
- `jef180001_htk180000_database.py` script
- `jef180001_htk180000_webscraping.py` script

2. Download and install all the libraries in the requirements.txt file

```
python -m pip install -r requirements.txt --no-deps
```

3. Run the jef180001_htk180000_chatbot.py script

```
python -m jef180001_htk180000_chatbot.py
```

Notes
- We highly recommend installing these libraries in a new virtual environment
- The --no-deps argument for pip installation is needed because the ChatterBot library does not recognize newer versions of PyYAML
- The first time you search something, the bot may need to download Chromium to access JavaScript functionality.