Data Collection

SourceAuthor: Yuan Dang

Instructor: Amir Jafari

Dec 11, 2022

Abstract

Writing is a critical skill for success especially for students. One way to help students improve their writing is via automated feedback tools, which evaluate student writing and provide personalized feedback. There are currently numerous automated writing feedback tools, but they all have limitations. Many often fail to identify writing structures, such as thesis statements and support for claims. One way to improve the feedback tools is to develop a better identification and classification model.

Data Collection

Source

This dataset is from Kaggle competition *Evaluating Student Writing*(https://www.kaggle.com/competitions/feedback-prize-2021/overview).

Overview

The dataset contains argumentative essays written by U.S students in grades 6-12. Essays are automatically segment into discrete discourse elements. The number of samples in this dataset is 144,280. The goal of this project is to identify these elements in student writing. More specifically, this project will classify discourse elements in essays into following categories:

- **Lead** - an introduction that begins with a statistic, a quotation, a description, or some other device to grab the reader's attention and point toward the thesis
- **Position** - an opinion or conclusion on the main question
- **Claim** - a claim that supports the position
- **Counterclaim** - a claim that refutes another claim or gives an opposing reason to the position
- **Rebuttal** - a claim that refutes a counterclaim
- **Evidence** - ideas or examples that support claims, counterclaims, or rebuttals.
- **Concluding Statement** - a concluding statement that restates the claims

Head of Dataset

	discourse_text	discourse_type	label
0	Modern humans today are always on their phone....	Lead	0
1	They are some really bad consequences when stu...	Position	2
2	Some certain areas in the United States ban ph...	Evidence	4
3	When people have phones, they know about certa...	Evidence	4
4	Driving is one of the way how to get around. P...	Claim	3
5	That's why there's a thing that's called no te...	Evidence	4

Figure 1 Dataset Head out of 144,280

Data Preprocessing

Missing Value

	missing value exists
id	False
discourse_id	False
discourse_start	False
discourse_end	False
discourse_text	False
discourse_type	False
discourse_type_num	False
predictionstring	False

Figure 2 Missing Value in Each Column

NLP Text Cleaning

- To Lower Case:
  Convert all alphabet characters to lower case.
- Remove Parenthesis
- Fix contractions:
  Using Python package *contractions* to split contractions(e.g. I'm, it's in to I am, it is).

- Non-ASCII Text:

  Some text contains non-ASCII characters, use Python package *unidecode* to decode them.

  ```
  'it is better to seek\xa0multiple opinions instead of just one. '
  ```

- URL & Not Well-formatted URL

  Some text contains URL and they are not well formatted, use python Regular Expression to extract and remove them.

  ```
  and two seconds, but while texting it increased to three to four seconds, regardless of whether the driver
  was typing or reading a text." (https://www. abc. net. au/science/articles/2011/10/06/3333955. htm). When
  ```

- Consecutive Characters

  Some text contains typo and meaningless repeated characters, use Regular Expression to fix them into correct format.

  ```
  ' you can get a bunch of diffferent opinions'
  ```

- Stop Words

  Remove stop words in text according to NLTK stop words.

- Non-relevant Text

  Since that some text became empty after removing stop words, we find some unfinished text containing all stop words. Remove them from dataset.

	discourse_text	discourse_type	label	text
24893	We should do this	Position	2	
26295	you should	Position	2	
32573	more to do for themselves	Claim	1	
44492	how we will	Claim	1	
61549	where it is	Claim	1	
67927	because its not	Rebuttal	4	
92185	but what about now	Rebuttal	4	
98278	We have to do it	Claim	1	
113822	when it's not	Rebuttal	4	

Figure 3 Non-relevant Text in Dataset

- Lemmatization

  Reformat words by their roots.

- Tokenization
  - Word tokenization
  - Word chunk tokenization: tokenize based on phrases

Visualization

Dataset Distribution

		count
discourse_type	label	
Claim	3	50204
Evidence	4	45702
Position	2	15417
Concluding Statement	6	13505
Lead	0	9305
Counterclaim	5	5817
Rebuttal	1	4334

Figure 4 Number of Samples in Each Category

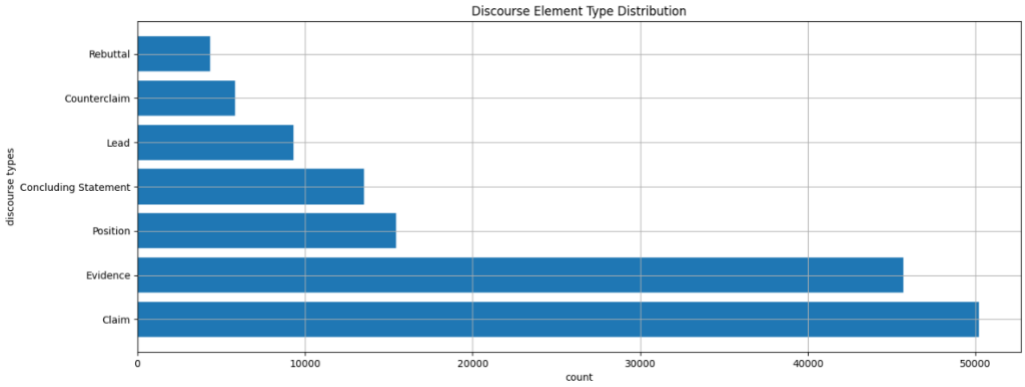

Figure 5 Discourse Type Distribution

The dataset has 7 types of discourse elements, the most two components are Claim and Evidence that have over 40000 samples; the least component is Rebuttal statement, which has less than 5000 samples.

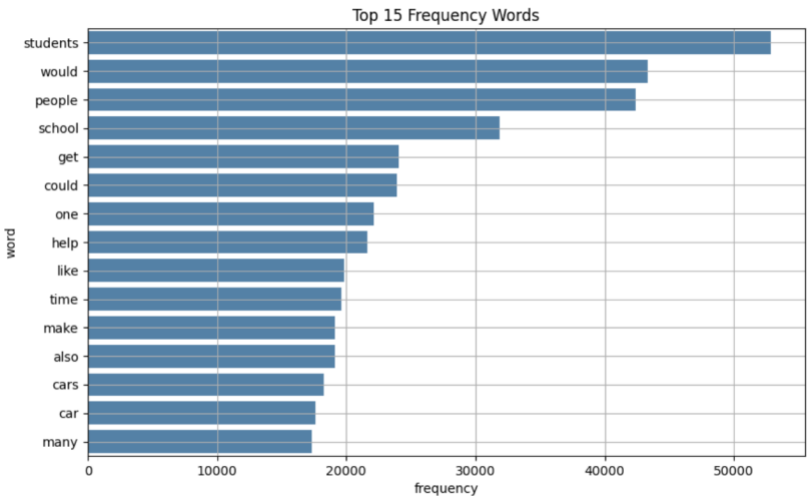

Figure 6 Most Frequent Words in Dataset Text

We can find the most frequency words after word tokenization. Total number of words in dataset is 57,414. The top1 used word in students' essays is "students". Some common nouns are "people", "school", "time", "car".

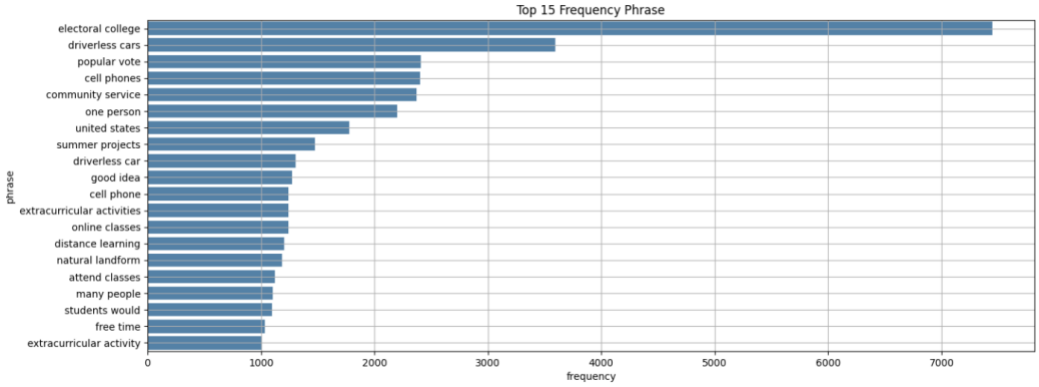

Figure 7 Most Frequent Phrase in Dataset Text

Similarly, after extracting phrases, we find that the total number of phrases in dataset is 328,811, and the top frequency phrases are "electoral college", "driverless cars", "popular vote" .etc. These seem to be the most frequent essay topics in student essays.

Word Cloud

We could use WordCloud to take a better review of frequency of words in each discourse element category. The frequency of word is considered as its importance, and the importance of each word is shown with font size. We can use those keywords to identify the discourse element types.

Note that in every category, the most used words are some nouns that also appears in previous top frequency words among whole dataset.

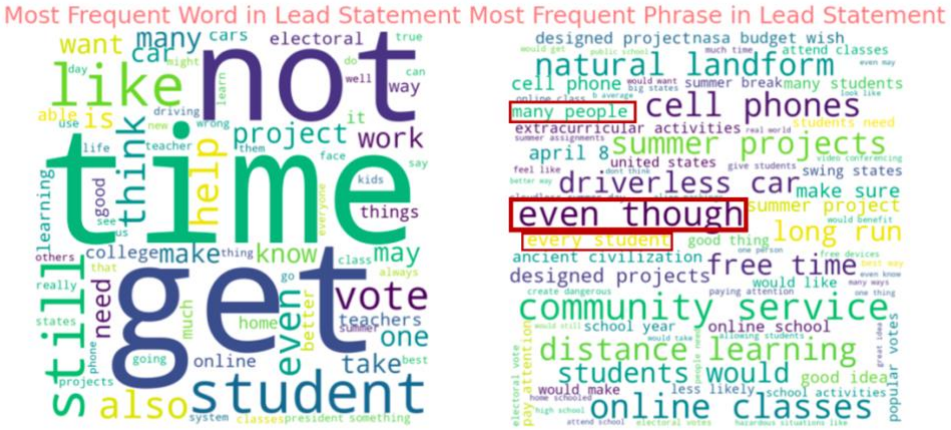

Figure 8 Lead Argumentative Type Word Cloud

In lead statement, nouns are used a lot, and some noun phrases such as “many people”, ‘every student”, “many students” are common in beginning of paragraph. The most common conjunction is “even though”.

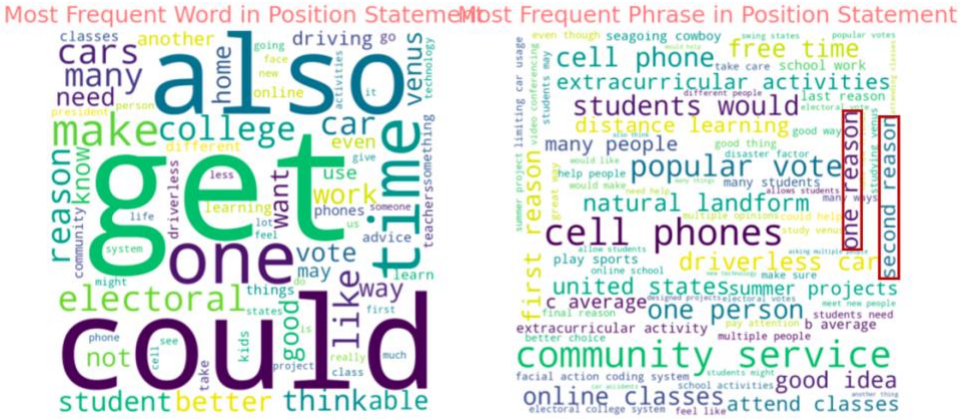

Figure 9 Position Argumentative Type Word Cloud

Position is defined as an opinion on the main question. Some frequently used word phrases here are “one reason”, “second reason”

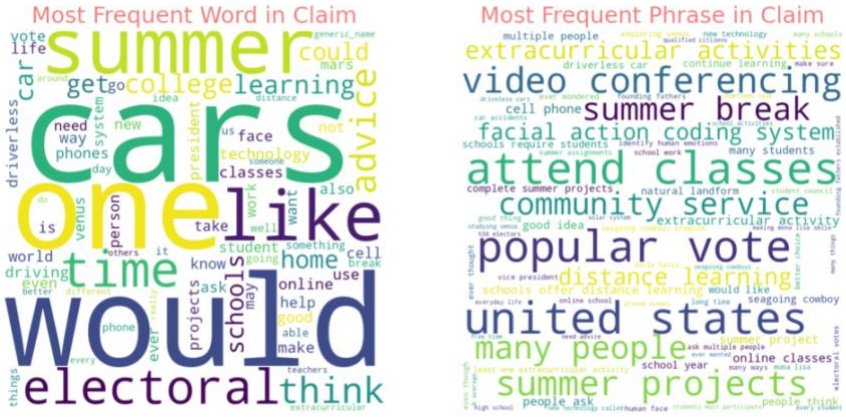

Figure 10 Claim Argumentative Type Word Cloud

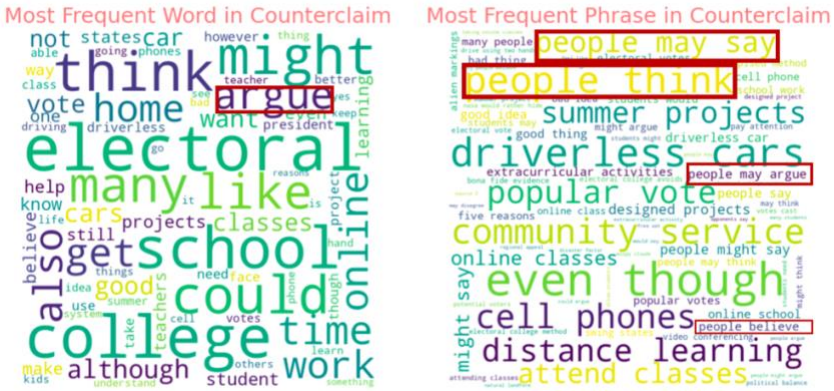

Figure 11 Counterclaim Argumentative Type Word Cloud

Claim supports the main opinion, and counterclaim always gives an opposite opinion. For the most time, counterclaim starts with phrase “people may argue”. The word cloud here shows this obvious attribute of counterclaim that is different than any other elements.

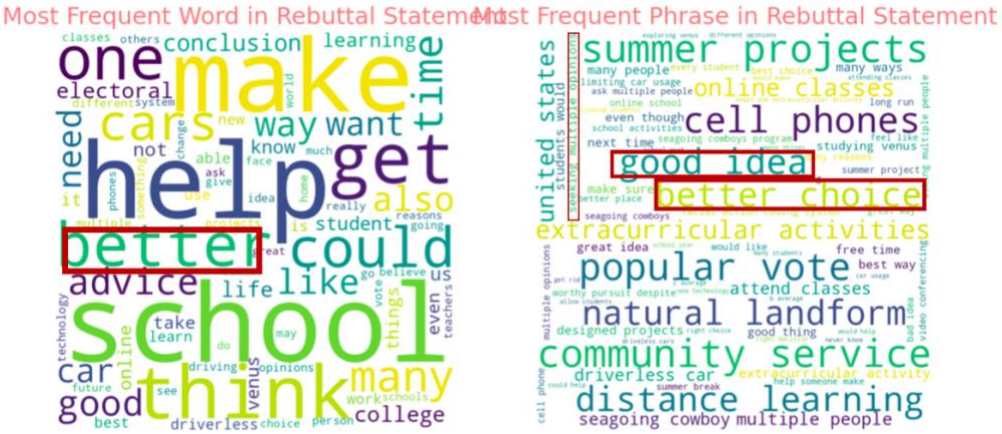

Figure 12 Rebuttal Argumentative Type Word Cloud

Rebuttal elements always follows the counterclaim and refutes the counterclaim in order to support author's main opinion. The most common words in rebuttal statements is "better", "better choice".

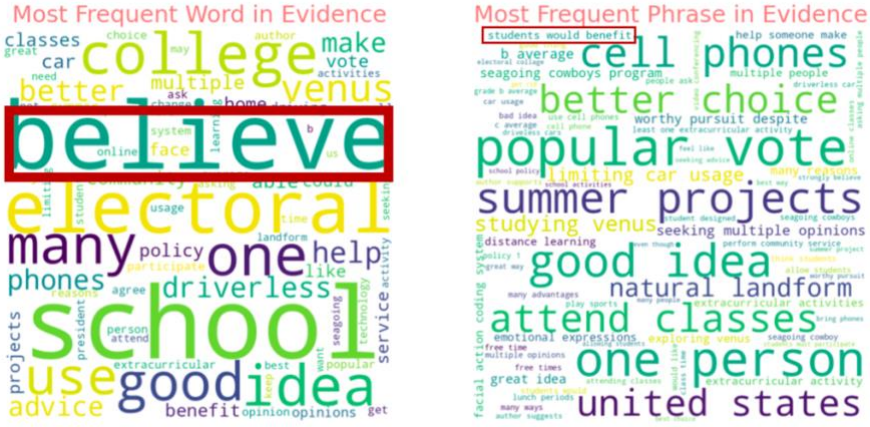

Figure 13 Evidence Argumentative Type Word Cloud

Evidence gives ideas or examples that support any claims, they often use words "believe", "benefit".

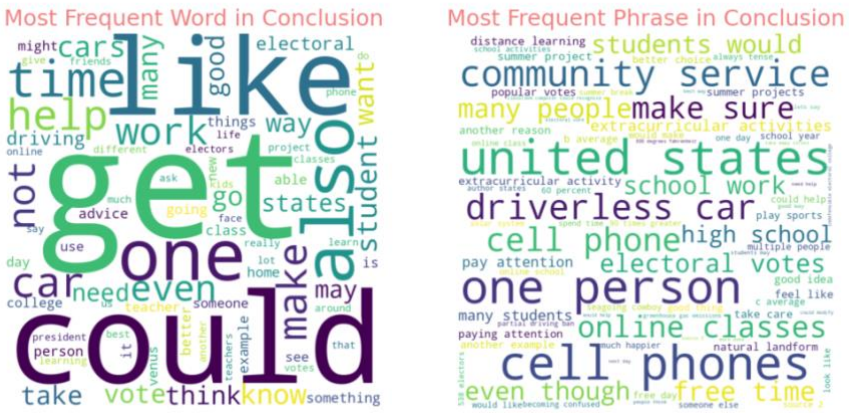

Figure 14 Conclusion Argumentative Type Word Cloud

Concluding statement restates the claims, therefore, the bag of words here are very similar to it is for claim statement.

References

Nikita Saxena, *Extracting Keyphrases from Text: RAKE and Gensim in Python,* https://towardsdatascience.com/extracting-keyphrases-from-text-rake-and-gensim-in-python-eefd0fad582f

Generating Word Cloud in Python, https://www.geeksforgeeks.org/generating-word-cloud-python/#:~:text=For%20generating%20word%20cloud%20in,from%20UCI%20Machine%20Learning%20Repository.