ABA Framework and RBAM

A. Desbiaux, V. Cuzin-Rambaud, B. Vadurel

DynCo - M2 IA - 19/10/2024
University Lyon 1 Claude Bernard

I. ABA Framework

Link to our website: https://aba-generator.onrender.com/

/!\
It may take a few minutes to load the website as the server shuts down when there are no activity for a while.

If you encounter any error *(for example 502)* when trying to access the website, please contact us by email, we'll fix it as soon as possible.

benjamin.vadurel@etu.univ-lyon1.fr
arthur.desbiaux@etu.univ-lyon1.fr
valentin.cuzin-rambaud@etu.univ-lyon1.fr
/!\

A. Creating Arguments and Attacks

To generate arguments, the algorithms are cut in 3 parts.

- Generating arguments from assumption. Basically, if x is a literal and x is an assumption, we create this argument: x → x.
- Generating arguments from rules where premises of the rule is an assumption or is empty.
- Generating argument from the arguments, based on rules:

```
| WHILE can generate more arguments from untreated rules REPEAT:
  | FOR premises IN untreated_rule :
      | retain all assumption
      | FOR all non-premises :
          | search if an argument deducts our premise
          | retrieve the premises of the arguments
          | generate a new argument when all premise from one rule were
seen, and if we have used an premise from another argument
```

Algorithm 1: generate Arguments from arguments

B. Conversion to non-circular and atomic ABA framework

For this part, we have to define some function. We need to know if an ABA framework is circular or atomic. We want to convert a circular ABA framework to a non-circular, so we can convert every framework to an atomic one.

C. Generate ABA+ normal, reverse attacks

In this part we've generated normal attacks and reverse attacks following the definition in the course.

D. Website

Some examples are already loaded (you can switch from one to another by clicking the "generate" button and choosing one) they all come from the courses. Otherwise, you can use a text file with the syntax given in the assignment (you can check directly the syntax on the website).

When the framework is loaded, it will display all the information about it. After you can compute the arguments, attacks and ABA+ attacks.

Finally, you can check if your framework is atomic or circular and automatically convert it into a non-circular or atomic one. When you have converted it, you can compute again all the arguments, attack, and ABA+ attack with your new framework.

When pressing a generate button, it may be possible that you have no attack, in that case the table will show *"No data available"*

II. Relation Based Argument Mining (RBAM)

Using the code contained in the GIT repository provided with the assignment, we extracted the 250 discussions having the highest number of participation. In order to get these, we used the participant filter.

A. Kialo data analysis and exploration

First, we scrapped the topics of the most active debate and their associated tags. We now have a data frame with 2 columns : kialoUrlId and tags.

- *kialoUrlId* : A string representing the url of the debate that will be later used to scrap the discussions
- *tags* : An array of strings containing the category of the debate (for example : Education, Law, Religion).

kialoUrlId	tags
are-purity-pledges-harmful-29355	[Purity, Sex, Virginity, Feminism, Women]
should-american-football-be-banned-10143	[Sports, USA, Football, American_Football]
should-there-be-one-world-state-30339	[States, Nations, Culture, Economics, Politics]

Figure 1: Sample from the debate topic table

We analyzed the table to find out what was the most trending debate topic, and we observed that it was Politics, Ethics and Religion.

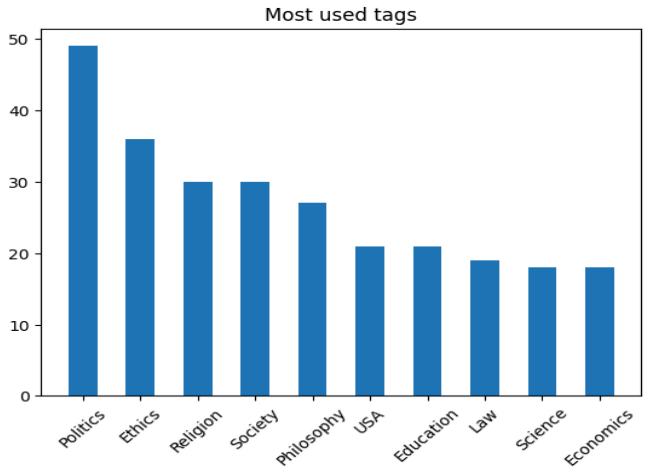

Figure 2: Bar chart showing the number of time each tag are used in a debate

B. Discussion data preprocessing

Using the URL list, we scrapped the debate discussions from the Kialo. A debate is represented by:

- Discussion name
- Discussion tree where:
  - The initial question is the root
  - Several theses as the node child of the root
  - The child node of the thesis are arguments, each arguments relation to the thesis is described by “Pro” or “Con” (translated if the debate is not in English)
  - Each argument can have other arguments such as node child attacking or supporting it.

Every node in the tree is described by an identifier composed of digits separated by dots. The more digits there are in the identifier, the deeper the arguments are in the tree.

- List of sources used by the arguments

Discussion Name:
What is the best writing advice?
1. Question:
What is the best writing advice?
[...]
1.5. Thesis:
Prioritize working on the characters in your story.
[...]
1.5.3. Pro:
Your story needs an active protagonist.
1.5.3.1. Con:
While true in many cases, several well-known books have very passive leading characters.
1.5.3.1.1. Pro:
The Lord of the Rings is one famous example.
[...]
Sources:
[1] https://thewritepractice.com/protagonist/
[2] https://www.boredpanda.com/male-authors-writing-about-women/

Data 1: Extract from the scrapped debate “What is the best writing advice ?”

For this assignment, we only exploited the arguments contained in the discussion tree. Discussion names, questions, thesis, and sources will be ignored.

In order to exploit these debate text file, we have to parse them to produce a dataset containing 3 columns:

- arg1: string representing an argument, this argument is supported or attacked the argument in the arg2 column
- arg2: string representing an argument, this argument supports or attacks the arguments in the arg1 column
- labels: the labeling representing the relation between the two arguments. It can take two numeric values:
  - 0: attack
  - 1: support

arg1	arg2	labels
Enables cohesive national policies	Economical benefits to different regions could be spread more evenly.	1
Enables cohesive national policies	National policies in such a big country is difficult & complex to handle	0
Could create new unstable states	Unsuitability in one region could spread to other regions as well.	1

Data 2: Extract from the dataset we generated

C. Argument referencing other argument

In the debate text file, we observe that some arguments are references to other arguments or discussion using their identifiers.

1.1.2.2.1. Con:
-> See 1.1.1.1.

Data 3: Example of an argument referencing another one.

These arguments are not suitable for training a classification model as it does not contain information itself, so we decided not to add them in the dataset. We could have kept them and search for the referenced arguments, but this type of argument represents only a small portion of the dataset.

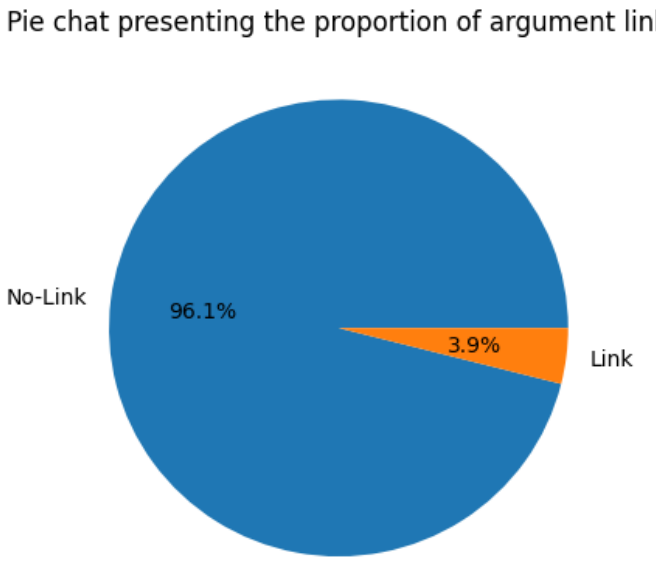

Figure 3: Arguments with links is a minority

D. Non-english debates

We observe that the scrapped debates are using different languages, the majority of them are in English, but we can also find debates in French, Italian or German for example.

As we'll experiment with Roberta and Bert-Multilingual, we'll produce a dataset only in English and another one using any language.

Every debate usually has the same structure apart from the French's one which are translating keywords ("Pro", "Con", "Thesis", etc…).

To verify the language of debate, we used the python library LangID. The predictions are not perfectly accurate, but it will do just fine for the project. We could improve the language detection by using the Google Translate API which is a paid option.

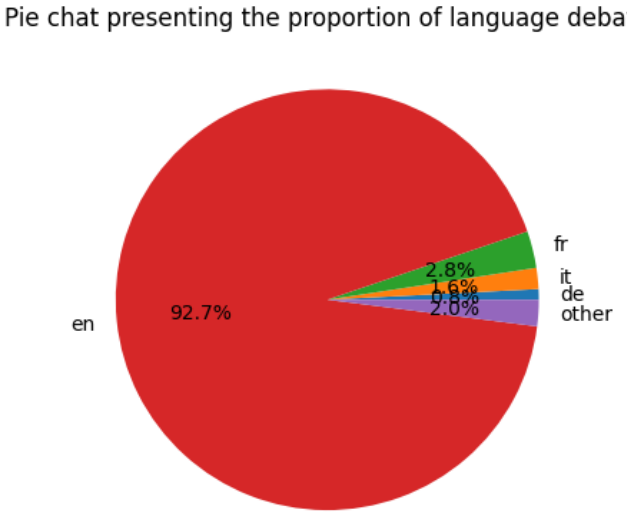

Figure 4: the large majority of English debates, but other languages too.

E. Source reference

We observe that some arguments contain references to sources, generally taking the form : [*identifier of the source*]. As we don't consider sources for the classification, we removed these references from the arguments.

F. Train dataset

To prepare the train, we imported the generated data we prepared before and split the dataframe into a train and a split dataset. The split took 20% of the data for the train dataset, and both train and test dataset are shuffled.

G. Classification models

Parameter	Value
Batch size	16
Number of epoch	5
Learning rate	4e-5
Seed	"0"

Figure 5: Parameter list used for the model training

We decided to test two models for the classification. Firstly, Roberta because we observed in literature that it was generally offering the best performances. Secondly, we used Bert-Multilingual to observe the impact of having multiple languages in the classification.

	Accuracy	F1-score	AUC	TP	TN	FP	FN
Roberta-base	0.813	0.813	0.89	3884	4253	927	935
Bert-Multilingual	0.698	0.698	0.772	3441	3700	1547	1538

Figure 6: Score obtained with the experimented models

We observe that Roberta has better performances than Bert-Multilingual. The differences in performance can be explained by the fact that there's a small proportion of non-English text in the debates, and the model is not trained enough on these texts.

H. Comparison to other approaches in the literature

1. RBAM using Large Language Model

Instead of using Bert-like models, some approaches are using LLM to do RBAM. This paper [1], uses the Mistral LLM and performs better than our approach. In their approach, they built a balanced dataset depending on the debates tags and their relation tag. In average, Mistral 7B-16 bits offers an 75,49% Macro-F1 score and the fine-tuned version of it has 91.59% F1-score

In another paper [2], uses the Lama LLM which is also better than our approach, in particular with Llama70B-4bit with a F1-score of 0.75 . This paper also compared their approach (LLM-based) to the RoBERTa baseline and outperform it.

2. RBAM using graph-based natural language inference

In this approach [3], they didn't balance the dataset depending on the labelling. They offer a comparison between several models and Graph-based models. With S-Bert, they get a 79% accuracy in comparison with GraphNLI: Root-seeking Graph Walk + Weighted Avg reach 82.87%.

3. RBAM using Dialogue Act Tags

In a recent paper [4], they present ArguNet, based on BERT and improved by DASHNet, a model to extract tags from an argument. ArguNet reach 82.1% of accuracy on the kialo Dataset.

Here is a summary of all the studied approaches:

	Accuracy	**F1-Score**
Our approaches		
Roberta	0.813	0.813
Bert-Multilingual	0.698	0.698
Compared approaches		
S-Bert	0.798	-
ArguNet	0.810	-
Llama70B-4bit	0.773	0.75
Mistral LLM 7B-16bits	-	0.75
Mistral LLM Fine-tuned	-	0.91
GraphNLI: Root-seeking Graph Walk + Weighted Avg	0.828	-

I. Indirect support and attack prediction

To test our model on indirect supports and attacks, we generated a few more datasets. Each dataset contains arguments separated by 1, 2, 3, … arguments.

The issue we faced is that we need to compute the relation between two arguments, depending on the arguments between them. We represent the relation with a boolean which equals 0 if it's an attack and 1 if it's a support. To compute the relation between the arguments we used the following algorithm in pseudo-code :

```
| relation = 1 if highest_level_arg.label = ‘pro’ else 0
| FOR arg IN (highest_level_arg+1 : lowest_level_arg) :
   | IF arg.label == ‘Con’ :
      | relation = not relation
```

We generated the dataset for path length from 2 to 7 where we started to have mild results (approx. 50%) with the classification.

Path length	Number of row
2	49035
3	36533
4	21426
5	11400
6	6278
7	3831

Figure 7: number of rows in each dataset.

Path length	Accuracy	F1-score	AUC	TP	TN	FP	FN
2	0.660	0.660	0.66	16076	16304	8026	8628
3	0.617	0.617	0.66	11231	11336	7211	6754
4	0.573	0.573	0.61	6299	5982	4446	4698
5	0.575	0.574	0.60	3154	3401	2366	2478
6	0.539	0.538	0.55	1619	1765	1325	1568
7	0.521	0.519	0.52	869	1129	842	990

Figure 8: evaluation of dataset.

As we can see, based on the accuracy, the prediction for path length > 5 is not good. The number of true positives decreased rapidly, for a path length between 2 and 5, the AUC stayed above 0.60.
We can conclude that the better predictions are for a path length lower than 5.

III. References

[1] : Assisted debate builder with large language models, Elliott Faugier, Fréderic Armetta, Angela Bonifati, Bruno Yun, May 2024.
[2] : Can Large Language Models perform Relation-based Argument Mining? , Deniz Gorur, Antonio Rago, Francesca Toni, February 2024
[3] : GraphNLI: A Graph-based Natural Language Inference Model for Polarity Prediction in Online Debates, Vibhor Agarwal
[4] : Exploiting Dialogue Acts and Context to Identify Argumentative Relations in Online Debates, Stefano Mezza, Wayne Wobcke and Alan Blair, August 2024