



Natural Language Processing
Project Phase 1

Human Values Behind Arguments

Shahrzad Azariazad (98521018)

Dr. Sauleh Eetemadi

-

Contents

1 Introduction

This is a classification task based on SemEval 2023 Task 4. Its objective is to identify human values behind arguments. To be more clear, given a text and some value categories, here we want to identify which category it falls in and also is it against or in favor of that category. You can access the project codes from [here](#), and also all the data has uploaded to hugging face accessible via this [link](#)

2 Dataset

Initially there was a dataset of 5393 annotated arguments. There was 20 values which i chose only 3 of them:

- Achievement
- Power: dominance
- Power: resources

After extracting the values which we want to classify, there remains 2188 arguments.

2.1 Data Collection

Using requests in python, I gathered needed data from the dataset provided.

2.2 Data Structure

The annotated corpus in tab-separated value format. Contains the following files for the different dataset splits: arguments-*jsplit*.tsv: Each row corresponds to one argument Argument ID: The unique identifier for the argument Conclusion: Conclusion text of the argument Stance: Stance of the Premise towards the Conclusion; one of "in favor of", "against" Premise: Premise text of the argument labels-*jsplit*.tsv: Each row corresponds to one argument Argument ID: The unique identifier for the argument Other: Each other column corresponds to one value category, with a 1 meaning that the argument resorts to the value category and a 0 that not

2.3 Data Filtering

In this state, I joined the arguments and labels by the Argument ID. I deleted all columns and rows which were related to other values. For each of the remaining values, there is two labels: 'against' and 'in favor of' which I showed by adding 'N' or 'P' at the end of the value name.

2.4 Data Cleaning

For cleaning data, I removed all punctuation marks except dot. The reason for that is it makes the sentence and word tokenizing easier.

2.5 Data Breaking

After cleaning the data, I broke data by its sentence. You can find the label separated files inside data/sentencebroken directory. I also broke data by its words which you can find the label separated files inside data/wordbroken directory.

3 Statistics

These are some basics statistics about the dataset.

3.1 Row Count

Number of rows for each label.

row count
$\text{power}_{\text{dominance}_N}$ 310
$\text{power}_{\text{resources}_P}$ 306
$\text{power}_{\text{resources}_N}$ 319
achievement_P 783
$\text{power}_{\text{dominance}_P}$ 300
achievement_N 729

3.2 Sentence Count

Number of sentence for each label.

sentence count
$\text{power}_{\text{dominance}_N}426$
$\text{power}_{\text{resources}_P}433$
$\text{power}_{\text{resources}_N}412$
achievement_P1222
$\text{power}_{\text{dominance}_P}497$
achievement_N970

3.3 Word Count

Number of words for each label.

word count
$\text{power}_{\text{dominance}_N}6622$
$\text{power}_{\text{resources}_P}7045$
$\text{power}_{\text{resources}_N}6773$
$\text{achievement}_P19222$
$\text{power}_{\text{dominance}_P}7793$
$\text{achievement}_N15397$

3.4 Unique Word Count

Number of unique words for each label.

unique word count
$\text{power}_{\text{dominance}_N}$ 1642
$\text{power}_{\text{resources}_P}$ 1730
$\text{power}_{\text{resources}_N}$ 1535
achievement_P 3465
$\text{power}_{\text{dominance}_P}$ 2066
achievement_N 2782

3.5 Common Unique Word Count

Number of common unique words for each label. Extra explanation: I computed the common words between all labels, then count the number of unique words for each label which are present in common words.

common unique word count
$\text{power}_{\text{dominance}_N}$ 416
$\text{power}_{\text{resources}_P}$ 416
$\text{power}_{\text{resources}_N}$ 416
achievement_P 416
$\text{power}_{\text{dominance}_P}$ 416
achievement_N 416

3.6 Uncommon Unique Word Count

Number of uncommon unique words for each label. Extra explanation: I computed the common words between all labels, then count the number of unique

words for each label which are not present in common words.

uncommon unique word count
$\text{power}_{\text{dominance}_N}1226$
$\text{power}_{\text{resources}_P}1314$
$\text{power}_{\text{resources}_N}1119$
achievement_P3049
$\text{power}_{\text{dominance}_P}1650$
achievement_N2366

3.7 10 Most Frequent Uncommon Words

10 most frequent uncommon words for each label. Extra explanation: I computed the common words between all labels, then count the number of unique words occurrences for each label which are not present in common words, and sort them by number of occurrences to get the 10 most frequent uncommon words.

word 1	word 2	word 3	word 4	word 5	word 6	word 7	word 8	word 9	word 10	word1	word2	word3	word4	w
$\text{power}_{\text{dominance}_N}$	<i>company</i>	<i>voted</i>	<i>deterrent</i>	<i>democracy</i>	<i>kill</i>	<i>judicial</i>	<i>voting</i>	<i>activism</i>	<i>war</i>	<i>threat</i>				
$\text{power}_{\text{resources}_P}$	<i>debt</i>	<i>games</i>	<i>prostitution</i>	<i>expensive</i>	<i>resources</i>	<i>olympic</i>	<i>unfair</i>	<i>loan</i>	<i>strad</i>	<i>students</i>				
$\text{power}_{\text{resources}_N}$	<i>jobs</i>	<i>company</i>	<i>telemarketing</i>	<i>exploration</i>	<i>food</i>	<i>farming</i>	<i>factory</i>	<i>whaling</i>	<i>executives</i>	<i>business</i>				
achievement_P	<i>students</i>	<i>wikipedia</i>	<i>loans</i>	<i>stem</i>	<i>exploration</i>	<i>celle</i>	<i>embryonic</i>	<i>young</i>	<i>learn</i>	<i>cars</i>				
$\text{power}_{\text{dominance}_P}$	<i>voting</i>	<i>compulsory</i>	<i>policy</i>	<i>army</i>	<i>democracy</i>	<i>I</i>	<i>political</i>	<i>vote</i>	<i>multi – party</i>	<i>party</i>				
achievement_N	<i>telemarketing</i>	<i>cloning</i>	<i>farming</i>	<i>jobs</i>	<i>food</i>	<i>cars</i>	<i>factory</i>	<i>child</i>	<i>actors</i>	<i>autonomous</i>				

3.8 Words Histogram

In this words histogram, you see 30 of the most frequent words in all labels.

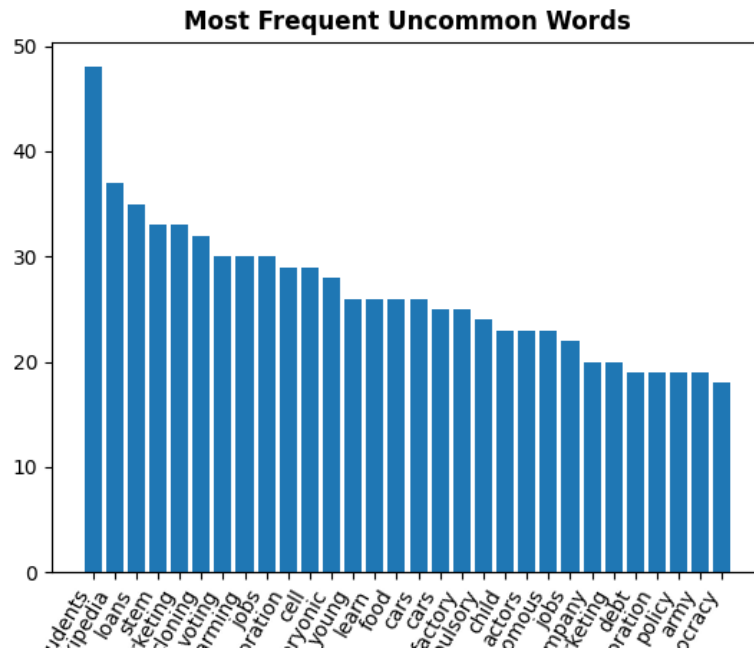


Figure 1: Most Frequent Words

4 Run Script

I wrote a python script which allows any user to run each part of this project separately.