

ProtAnt (Windows, MacOS, Linux)

Build 1.3.1

Laurence Anthony, Ph.D.

Center for English Language Education in Science and Engineering, School of Science and Engineering, Waseda University, 3-4-1 Okubo, Shinjuku-ku, Tokyo 169-8555, Japan. August 2, 2023

Introduction

ProtAnt is a freeware prototypical text detection tool developed in collaboration with Paul Baker of Lancaster University, UK. ProtAnt takes a corpus of texts (UTF-8 encoded) and compares them either individually or as a whole against a reference corpus (UTF-8 encoded) or list of 'key' words (UTF-8 encoded) to find characteristic features in the target files. Then, ProtAnt looks at each individual target file and

Add Target File(2) Add Target Directory Clear File Key Tokers Normikey Types Normikey Types Normikey Types Add Target 5 1.bt 7.bt 2.bt 6.b 47.2373 64.1762 444 10 2.bt 3.bt 2.b 8.bt 2.b 6.c 47.2373 64.1762 444 10 2.bt 3.bt 2.b 8.bt 2.b 6.c 45.548 6.c 42.55 42.5 42.5 42.5 42.5 42.5 42.5 42.5 42.5 42.5 42.5 42.5 5.5	arget Corpus 🐵 Raw 🕐 List 📋 Treat files separately	Resu	ults										Clea
1. ht 7. kt 2. i 67 47.237 64.1762 4.44 0.0 3. ht 3. ht 2.0 6.1 42.547 42.547 42.547 4.54 5.16 5.1	Add Target File(s) Add Target Directory Clear		File	КеуТур	ies Key	Tokens		NormKeyTyp	es No	ormKeyToken	s AllTy	pes /	AllTokens
2.44 38.44 4.54 55.45 2.64 3 0.64 0.	l.bt	1	7.txt		21	67		47.29	73	64,176	2	444	1044
Shift 2 8.tx 20 0.12 0.464 0.12 Shift 0.20 0.46 0.132 0.432	2.bd	<u> </u>											
ht 3 4.40 20 44 3223 38.437 4.64 11 ht 5.5 3.0 2.2 9.0 4.1.355 7.7.503 5.7.50 1.1.8 5.7.50 5.7.50 5.7.50 5.7.50 5.7.50	bt	2	8.txt		20	62		46.94	84	6	2	426	1000
Add Add <td>txt</td> <td>3</td> <td>4 tvt</td> <td></td> <td>20</td> <td>45</td> <td></td> <td>41 32</td> <td>73</td> <td>38.428</td> <td>7</td> <td>484</td> <td>1171</td>	txt	3	4 tvt		20	45		41 32	73	38.428	7	484	1171
Add deference Directory Key word list() Add deference Directory Class Sol 22 90 41.085 77.033 5.54 1.12 1 5 1.3 1.17 1.32 37.287 1.11.84 4.65 1.12 1.11 5 1.3 1.17 1.32 37.287 47.813 5.43 1.22 1.11 7.11 1.18 8 33.142 47.813 5.43 1.22 1.11 7.11 1.18 5.41 6.90 41.079 5.21 1.21 1.11 7.11 2.12 1.14 4.00 0.7238 4.007 Key	txt	ŕ.	-					44.04		301120			
5 13 17 122 37.207 111.84 4.65 11 14 5 9.0 18 38 33.142 47.0133 5.12 5.2	DR F	4	5.txt		22	90		41.19	85	74.503	3	534	1208
3 1.2 1.1 1.12 3.1.20 1.1.1.64	ut het	-	42					27.00					
Add Reference Circles Mathematical Regionance (RRB) Three Model (P-value) 0.05 - Borference (Circles) 1 100 -	bit .	2	13		1/	132		37.28	07	111.80	4	400	1180
Alt	Ltat	6	9.txt		18	58		33.14	92	47.815	3	543	1213
List 7 2.ct 14 0 27.343 41.070 5.12 9 List File Kay Kay3 Kay3 </td <td>.bt</td> <td></td>	.bt												
List File Kay Kay <td< td=""><td>2.txt</td><td>7</td><td>2.txt</td><td></td><td>14</td><td>40</td><td></td><td>27.34</td><td>38</td><td>41.407</td><td>9</td><td>512</td><td>966</td></td<>	2.txt	7	2.txt		14	40		27.34	38	41.407	9	512	966
Lit To	bt		File	Key 1	Key 2	Ve		Koul	Key 5	Key 6	Key 7	Key 8	Key
Add Reference Corpus Key word list(a) 1 7.74 destabilising brackit bracket bashar bloomberg benjamin etter def Reference Corpus Key word list(a) 2 8.64 destabilising bracket reference Corpus key word list(a) bashar bloomberg benjamin etter data Reference Directory Clear Reav List 3 4 data destabilising brothehood heabolat reference Corpus key word key	.txt		THE	Ney 1	Ney 2	Ne	y 5	Ney 4	Key 5	Key U	Ney /	Ney 0	Key
Indeference Sin(s) Key word list(s) Key word list(s) both model of calculate regular gengan genda gengan gengan	.bt +	1	7.txt	destabilising	jihadist	brothe	rhood	repugnant	assad	bashar	bloomberg	benjamin	extrem
Add Reference Directory Clear Raw List Add.Reference Directory Clear Raw List ADLts ADLts backs backs backs ADLts ADLts backs backs backs backs ADLts ADLts backs backs backs backs backs ADLts ADLts backs	Reference Corpus Key word list(s)	2	8.txt	destabilising	brotherhood	radical	lised	repugnant	yemen	chilcot	assad	libya	bashar
AD1.bt AD3.bt Stat indext index indext indext	Add Reference File(s) Add Reference Directory Clear @ Raw 🔿 List	3	4.txt	destabilising	brotherhood	hezbol	llah	repugnant	assad	libya	bashar	bloomberg	extremi
A02.0x A23.0x	A01.txt												
AD3.bt. AD3.bt. AD3.bt. S 13 melvyn corole bahrain gfn ievis harvey ofcom broadcasta harves AD3.bt. AD3.bt. AD3.bt. S 13 melvyn corole bahrain gfn ievis harvey ofcom broadcasta harves backat AD3.bt. babrain S 3 melvyn corole bahrain gfn ievis harvey ofcom broadcasta harves babrain ba	A02.txt	4	5.00	jinadist	brothernood	nezpoi	lian	banrain	yemen	assad	пруа	qa	islam
Add.ts. Add.ts. Add.ts. Add.ts. Bathar bioomberg externism putin utraine islam Add.ts. Add.ts. Add.ts. Add.ts. Bathar bioomberg externism putin utraine islam Add.ts. Add.ts. Add.ts. Islam Freq Keyweist Effect Islam	A03.txt	5	13	melvyn	carole	bahrai	n	qfh	levis	harvey	ofcom	broadcasts	harassr
A05.bt A07.b	A04.bit												
Keyword Freq Keywess Effect A85.bt 1 batten 17 130.038 1523.18 Certer Statistic Log Likelihood (4-term) • Threshold (p-value) 0.05 + Borferroni • other 1 batten 17 130.038 1523.18 Effect Stat Measure Ratio of relative frequencies (B8P) • Threshold (p-value) 0.05 + Borferroni • other 2 batten 17 130.038 1523.18 im Freq. per 1000 words 0 3 jeffrey 12 9.178.88 1075.19 sem DF (Regros) bocdedphijthmoopgrituwsyzdBCDEFGHUKLMNOPQRSTUW/2/VZ 5 faran 9 68.404 866.391 5 faran 9 68.404 866.391 5 faran 9 68.404 866.391	A05.txt	6	9.txt	brotherho	radicalised	assad		bashar	bloomberg	extremism	putin	ukraine	islam
MJ. M. MJ. M. MJ. M. ADD. M. ADD. M. MJ. M. MJ. M. ADD. M. ADD. M. MJ. M. MJ. M. ADD. M. ADD. M. Keyword Free Keyword ADD. M. Science Keyword Free Keyword Free Science	A06.txt	-			··· ·· ·		_						
ADD xx Keyword Freq Keyness Effect Keyness Statistic Log Lakelihood (4-term) Threshold (p-value) 0.05 + Borferencii 1 batton 17 1300.38 1523.18 Ffect Size Messure Ratio of relative frequencies (BSP) Threshold (n-value) 0.05 + Borferencii 2 babolit 14 107.088 1523.18 se Improve case (Change to lowercase) 4 tule 9 668.404 806.391 sn Def (Regot) abcdedphijtemnoportuwwyzd8CDEFGHUKLMNOPQRSTUWZZYZ 5 far.m 9 68.404 866.391	AU7.bd	•											
Regress Statistic Log Likelihood (4-term) Threshold (p-value) 0.05 + Bonfernoni 1 1 batten 17 130.08 1523.18 iffect Size Messure Batio of relative frequencies (RSf) Threshold (p-value) 0.05 + Bonfernoni 1 batten 17 130.08 1523.18 iffect Size Messure Batio of relative frequencies (RSf) Threshold (n-nk) 10 2 babolat 14 107.088 1254.39 is joffrey 12 9.1788 107.159 4 value 9 68.8404 866.391 enen Def (Reget) abcdefghigHmnopgrstuxwyszABCDEFGHUKLIKNOPQRSTUW/Z/YZ 5 faraf 9 68.8404 866.391	409.txt -		Keyword		Freq Ke	eyness	E	ffect					
Flatic of relative frequencies (RRF) * Threshold (nnk) 100 2 babolat 14 107.08 1254.39 Im Freq. per 1000 words C 3 joffrey 12 91.788 1075.19 se Ignore case (Change to lowercase) 4 tule 9 66.804 866.391 se Ordefghijdminoportuwwyzd8CDEFGHUKLMN0PQRSTUW/2/VZ 5 farah 9 68.804 866.391	Keyness Statistic Log Likelihood (4-term) Threshold (p-value) 0.05 + Bonferroni	1	batten		17 1	30.038	152	23.18					
mm Freq. per 1000 words	effect Size Measure Ratio of relative frequencies (RRF) Threshold (rank) 100	2	babolat		14 10	07.088	125	i4.39					
Imm Freq. per 1000 words Imm Freq. per 1000 words Imm Freq. Per 1000 words Imm Freq. Per 1000 words Imm Freq. Per 1000 words Imm Freq. Per 1000 words Imm Freq. Per 1000 words Imm Freq. Per 1000 words Pe 1000 words Pe 1000 words		3	joffrey		12 9:	1.7888	107	5.19					
is Image: Second s	rm Freq. per 1000 words 🔄												
xen Def (Rege) abcdefghijklmnopqrstuvwsyzABCDEFGHUKLMNOPQRSTUVWZXYZ 5 farah 9 68.8404 806.391 6 kundinani 9 68.8404 806.391	se 📝 Ignore case (Change to lowercase)	4	tulle		9 6	5,8404	806	5.391					
6 kundhani 9 68,8404 806,391		5	farah		9 6	3.8404	806	5.391					
	ten Def (Regex) abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWZXYZ												

counts how many of these characteristic features are in them. The target files are then ranked in terms of their prototypicality by the number of characteristic features they contain. *ProtAnt* runs on any computer running Microsoft Windows (tested on Windows 10), Apple MacOS (Intel/Silicon) (tested on OSX 10.15 Catalina and OSX Ventura), or Linux (tested on Linux Mint 17). *ProtAnt* is developed in *Python* and *Qt* using the *PyInstaller* compiler to generate executables for the different operating systems.

Getting Started

🂐 Windows - Installer

Double click the *ProtAnt.exe* file and follow the instructions to install the application into your Programs folder. You can delete the .exe file when you are finished. You can start the application via the Start Menu.

🥙 Windows - Portable

Unzip the *ProtAnt.zip* file into a folder of your choice. In the *ProtAnt* folder, double click the *ProtAnt.exe* file to launch the program.

Macintosh OS X

Double click the *ProtAnt.dmg* file to create a ProtAnt disk image on your desktop. Open the disk image and drag and drop the ProtAnt app onto the Applications folder (or into another location if you desire). You can then launch the app by double clicking on the icon in the Applications folder or the Launchpad.

Å Linux

Decompress the *ProtAnt.tar.gz* file into a folder of your choice. In the *ProtAnt* folder, double click the *ProtAnt.sh* file to launch the software. On the command line, type ./ProtAnt.sh to launch the software.

Finding prototypical texts using a reference corpus

Step 1: Select the type of "Target Corpus" files you are going to use (raw files or word lists)

Step 2: Select if you want keywords to be generated by comparing all target file together with the reference files ("Treat files separately" - unchecked option) or each target file separately with the reference files (("Treat files separately" - checked option)

Step 3: Select the target corpus files that you want to analyze. You can do this in three ways:

- 1) Click on the File->Open Target File(s) menu option or the "Add Target File(s)" button below the "Target Corpus" label and select the files you want to analyze;
- 2) Click on the File->Open Target Dir menu option or the "Add Target Directory" button below the "Target Corpus" label and select a directory of files you want to analyze;
- 3) Drag and drop files directly onto the *ProtAnt* application.

Step 4: Select the reference corpus files that you want to analyze. You can do this in three ways:

- 1) Click on the File->Open Reference File(s) menu option or the "Add Reference File(s)" button below the "Reference Corpus" label and select the files you want to add;
- 2) Click on the File->Open Reference Dir menu option or the "Add Corpus Directory" button below the "Reference Corpus" label and select a directory of files you want to add;
- 3) Drag and drop files directly onto the *ProtAnt* application.

Step 5: Choose the keyness statistic, keyness threshold value (p-value), effect size measure, and effect size threshold (the cutoff rank) you would like to use to generate keywords for the target corpus. Note that a keyness threshold of 0 means all (positive) keywords will considered, and an effect size threshold of -1 means that all effect values will be shown. A positive keyword means that it is relatively more frequent in the target corpus than the reference corpus. Negative keywords (relatively more frequent in the reference corpus than the target corpus) are not considered.

Step 6: Choose the normalization function for frequencies displayed in the results window.

Step 7: Choose to ignore case in the target corpus (change all words to lowercase)

Step 8: Decide the ordering or keywords (in the bottom right pane), which will also determine the ordering of keywords shown per file in the middle pane.

Step 10: Decide a suitable token definition based on regular expression (regex) syntax.

Step 9: Click "Start" to begin the analysis.

Finding prototypical texts using 'key' word lists

Step 1: Select the type of target corpus files you are going to use (raw files or word lists)

Step 2: Select the target corpus files that you want to analyze. You can do this in three ways (see Step 3 above): **Step 3:** Select the 'key' word list files that you want to analyze. You can do this in three ways:

- 1) Click on the File->Open Keywords File(s) menu option or the "Add Keywords File(s)" button below the "Key word list(s)" label and select the files you want to add;
- 2) Click on the File->Open Keywords Dir menu option or the "Add Keywords Directory" button below the "Key word list(s)" label and select a directory of files you want to add;
- 3) Drag and drop files directly onto the *ProtAnt* application.
- Step 4: Choose the normalization function for frequencies displayed in the results window.
- Step 5: Choose to ignore case in the target corpus (change all words to lowercase)
- Step 6: Decide a suitable token definition based on regular expression (regex) syntax.
- Step 7: Click "Start" to begin the analysis.

- Note 1: If you click on the File->Close Target Files menu option, the File->Close Reference Files menu option, or the File->Close Keywords Files menu option, the files will removed from the relevant list.
- Note 2: If you click on the "Clear" button below the "Target Corpus" label "Key word list(s)" label, or the "Clear" button below the "Reference Corpus" label, or the "Clear" button below the "Key word list(s)" label, the files will removed from the relevant list.
- Note 3: The results of the prototypical text detection are shown in the top right window. The keywords appearing in each (ranked) corpus file are shown in the middle right window. The complete list of keywords is shown in the bottom right window. All columns can be sorted in either ascending or descending order by clicking on the column headers.

Note 4: The prototypical text detection analysis can be stopped at any time by clicking the "Stop" button.

Note 5: If you are using target corpus or reference corpus word lists, they should be formatted as RANK, FREQ, TYPE separated by tabs. Any line beginning with # will be ignored.

Note 6: If you are using 'key' word files, they should be formatted with each 'key' word on a new line.

Additional Features

The following shortcuts are available as is standard on the operating system:

Windows

CTRL-A	⇒ Select All
CTRL-C	🖙 Сору
CTRL-V	⇒ Paste

Macintosh

Command-A	⇒ Select All
Command-C	🖙 Сору
Command-V	⇔ Paste

CITING/REFERENCING *ProtAnt*

Use the following method to cite/reference *ProtAnt* according to the APA style guide:

Anthony, L. and Baker, P. (YEAR OF RELEASE). *ProtAnt* (Version VERSION NUMBER) [Computer Software]. Tokyo, Japan: Waseda University. Available from http://www.laurenceanthony.net/

For example if you download *ProtAnt 1.2.0*, which was released in 2016, you would cite/reference it as follows: Anthony, L. and Baker, P. (2016). *ProtAnt* (Version 1.2.0) [Computer Software]. Tokyo, Japan: Waseda University. Available from http:// www.laurenceanthony.net/

Note that the APA instructions are not entirely clear about citing software, and it is debatable whether or not the "Available from ..." statement is needed. See here for more details: http://owl.english.purdue.edu/owl/resource/560/10/

KNOWN ISSUES

None at present.

Copyright 2023 Laurence Anthony. All rights reserved.