THE INFLUENCE OF AGE ON VOCABULARY ACQUISITION IN ENGLISH AS A FOREIGN LANGUAGE

Tesi doctoral presentada per

Immaculada Miralpeix Pujol

com a requeriment per a l'obtenció del títol de

Doctora en Filologia Anglesa

Programa de Doctorat: *Lingüística Aplicada*
(Bienni 2000-2002)
Departament de Filologia Anglesa i Alemanya

Directors: **Dra. Carme Muñoz Lahoz i Dr. Paul M. Meara**

Universitat de Barcelona

2008

APPENDIX D

D_TOOLS MANUAL

_lognostics

D_Tools v 1.0

Paul Meara
Imma Miralpeix

Centre for Applied Language Studies University of Wales Swansea
and
University of Barcelona

CONTENTS

DISCLAIMER

These programs are provided free of charge to bona fide researchers on the understanding that the authors accept no liability arising out of the use of the programs. We make every attempt to ensure that the programs work in the way we expect them to do, but we cannot guarantee that they will work with your data, and you use them at your own risk.

Feedback on the programs is encouraged and welcome.

1. INTRODUCTION

D_tools v1.0 is designed to compute D values for short texts. D is an index of lexical richness originally proposed by Malvern and Richards (1997,2002). Unlike VOCD (the program created by Malvern and Richards to calculate Ds), the input data required by D_tools consists of simple ASCII files and the data does not have to be coded in CHAT format.

Calculation of D:

D is computed by selecting samples from the text of different token size: 100 samples of 35 tokens, 100 samples of 36 tokens and so on up to 100 samples of 50 tokens. The program then computes and averages TTRs at each point and matches the curve produced by our text with a theoretical curve produced by Malvern and Richards' formula: $TTR= D/N [(1+2N/D)^{1/2} -1]$, the best match between the two curves, which is calculated using a least-square algorithm, is the D value of our text.

Note that this program does not provide all the facilities available in Malvern and Richards' VOCD program, (see Richards and Malvern 2000b). Specifically, D_Tools does not allow you to set switches which determine how the raw data is processed. D_Tools does not automatically lemmatise your text: *go, goes* and *going* will be treated as different types unless you edit the input files and replace all instances of *goes* and *going* by *go.*

2. LOADING THE PROGRAMS

D_tools comprises two programs: D_0 and D_1.
The programs can be separately downloaded from the _lognostics website http://www.swan.ac.uk/cals/calsres/lognostics.htm

The files are provided in .zip format.
To install the programs, download the .zip file, and unizip it into a folder called:
c:\lognostics\d_tools
You can download the files to other locations if you wish, but it is easier to debug a faulty program if you stick to the recommended folder name.

Installing D_tools will save the following files to your computer:

D_0.exe	Imma's D (main program)
D_1.exe	D estimator (main program)
D_Toolsman.doc	This manual

D_Tools does not install any other files on your computer. You can delete D_Tools by sending the entire D_Tools folder to the recycle bin.
Once you have installed the programs, you can run them by following the instructions on the pages that follow.

3. RUNNING THE PROGRAMS

This package contains two programs (D_0 and D_1). Used together, the programs calculate a D value for a set of texts.

3.1 D_0: Imma's D

This program computes a curve of TTR values for a given text.
The program requires as input a complete text in ASCII format. It then calculates a set of 16 Type-Token ratio values, using different text lengths. Each value is based on 100 random samples from the original text. The output from this program is a list of the TTR values for each text.

Notes:
The input text must be in ASCII format, **one word to a line**, with all punctuation, tabs and other formatting information removed. Your data should look like the example below.

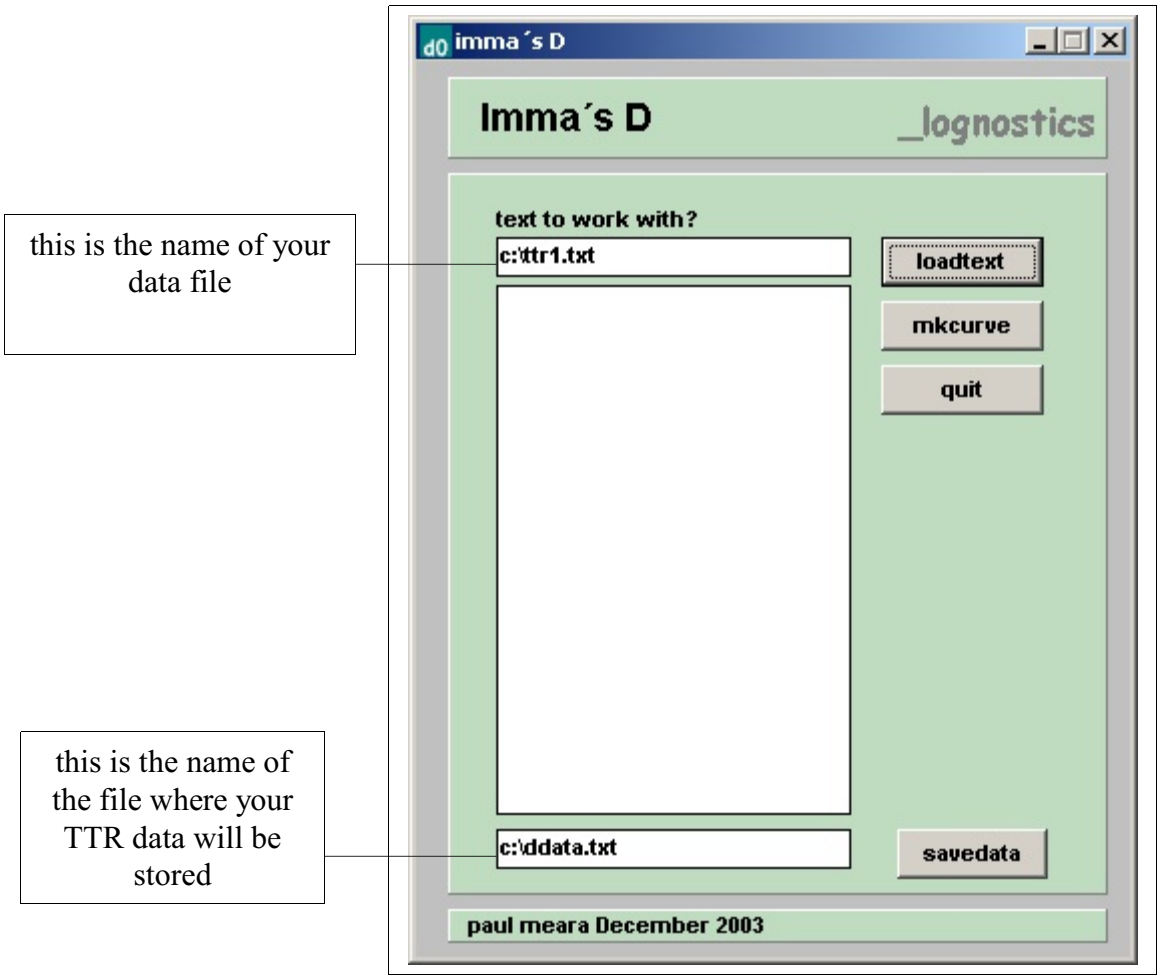
once
upon
a
time
there
were
three
bears
who
lived
in
a
dark
and
lonely
wood
etc...

If your text is in .DOC or another similar format, save it as a .txt file.
If it is inconvenient for you to format your data in this way by hand, then you can use a progam like V_Tools to do it for you. You can download V_Tools from the _lognostics website.
http://www.swan.ac.uk/cals/calsres/lognostics/

To run D_0 open the D_Tools folder, and click on the icon labelled Imma's D. This will open a window like the one shown on the next page.
Type the name of your text in the top box, then click **loadtext.** If your text is less than 50 words, the program will not let you proceed any further.

When the text has been loaded, click **make curve.** This will generate a set of 16 data points that characterise the TTR curve for this text. Click **savedata** to save this curve to another file.
Click **quit** to exit the program.

Imma's D: screenshot



3.2 D_1 D_Estimator.
This program finds out the best fit between a set of 16 TTR values output from D_0 and a theoretical curve described in work by Malvern and Richards. **D_1** works with the output files generated by the D_0 program. These files are simple ASCII files, containing a set of 16 TTR values, one to a line. You can construct these files by hand if you really want to, but this is not recommended.

To run D_1, open the D_Tools folder and click on the D_1 icon. This will open a window like the one shown below.

Type the name of your file in the top box and click **loadfile.**

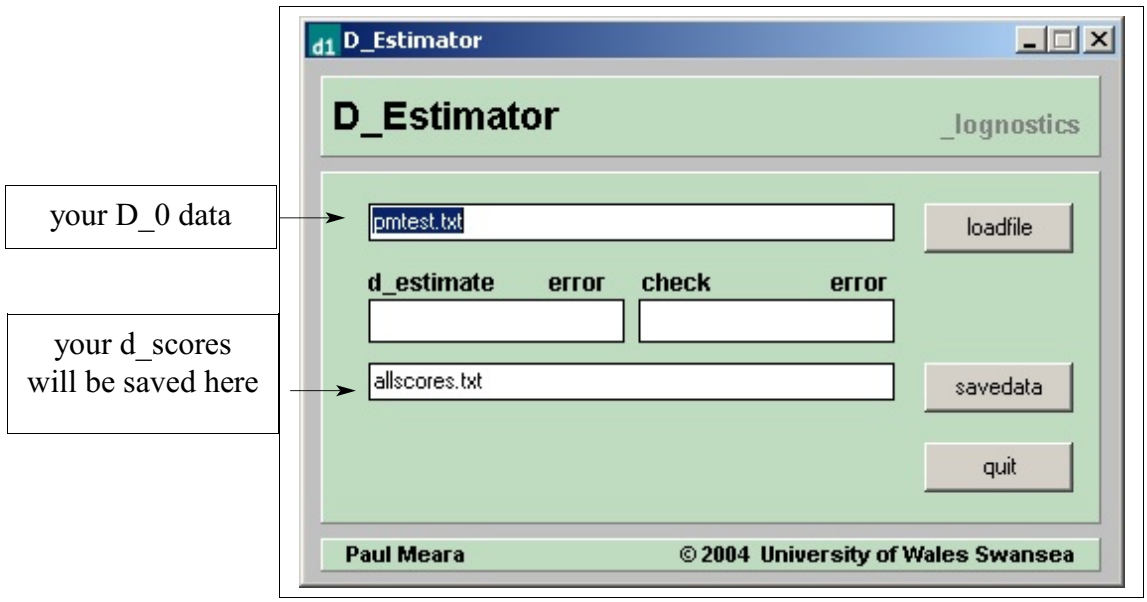
The program will then start automatically looking for the D value that best describes your data and this value will appear in the D_estimate box. This box also shows an **error** figure, which tells you how closely the result produced by D_1 matches your data. This figure should be very small – large error figures usually indicate that your data is unusual in some way.

Check that these two figures (D value and error) also appear in the **check box**. The figures in the D_estimate box and check box will normally be the same. If the D values in the two boxes differ, then this is usually a sign that your data is abnormal in some way.

The D values generated by the program will vary between 0 and 90. Most data produced by L2 speakers will normally lie between these two extremes.

After you have calculated the D values for a set of texts, type in the name of the file you want to keep the results in and click **savedata**, the name of each text together with its D value and error will be saved for you.
Click **quit** to exit the program.

D_Estimator screenshot



4. THEORETICAL BACKGROUND

D has a number of advantages over other measures of lexical richness, it is thought to be better than TTR because:

- It is not a function of the number of words in the data. It uses all the data available and it is not necessary to reduce the texts in a set to the number of tokens found in the shortest text.
- It is more informative because, as opposed to the single value of the TTR, it represents how the TTR varies over a range of token size for each speaker or writer, therefore it is theoretically more valid.
- It is useful to analyse short texts (you need a minimum of 50 tokens).
- It has been shown to discriminate between a wide range of language learners and users.

For more information, see bibliography below.

5. FORMAL EVALUATION OF THE PROGRAM

This program is currently under evaluation. Any comments and suggestions would be welcome.

6. CITATIONS

If you need to cite this program in your own work, our preferred format is:
Meara, P.M. and Miralpeix, I.
D_Tools. Swansea: Lognostics. 2004

Please let us know if you use these programs. It helps us get sponsorship for further development work. You can contact us on:
p.m.meara@swansea.ac.uk
imiralpeix@ub.edu

7. REFERENCES

Durán, P., Malvern, D., Richards, B., & Chipere, N. (2004). Developmental trends in lexical diversity. *Applied Linguistics, 25* (2), 220-242.

Jarvis, S. (2002). Short texts, best-fitting curves and new measures of lexical diversity. *Language Testing, 19* (1), 57-84.

Malvern, D., & Richards, B. (1997). A New Measure of Lexical Diversity. In A. Ryan & A. Wray (Eds.), *Evolving Models of Language* (pp. 58-71). Clevedon: Multilingual Matters.

Malvern, D., & Richards, B. (2002). Investigating accommodation in language proficiency interviews using a new measure of lexical diversity. *Language Testing, 19* (1), 85-104.

McKee, G., Malvern, D. & Richards, B. (2000). Measuring vocabulary diversity using dedicated software. *Literary and Linguistic Computing* (15), 323-337.

Richards, B., & Malvern, D. (2000a). Accommodation in oral interviews between foreign language learners and teachers who are not native speakers. *Studia Linguistica, 54* (2), 260-271.

Richards, B., & Malvern, D. (2000b). Measuring vocabulary diversity in teenage learners of French. In B. E. R. Association (Ed.). Cardiff: Education-line Electronic Database.

APPENDIX E

D_TOOLS VALIDATION

In order to validate the program, we followed three steps, as explained in chapter 5. The first two, where we used 40 tasks, are exemplified with one of the tasks: the composition of the student 1084t3a.

Subject 1084t3a
My name is Ruth. I am a young girl. I am studious and hard-working. I am an active and ambitious person. I like the languages very much, and I would like learn English very well, French and Euroland. I am very fond of dancing and studying music. I play the piano, I began at eight years, and I am continuing with it. I am in this institute nearly five years, and now I am studying HS, and I hope to pass it with good marks. I am nervous because it is difficult. I hope to improve my English in general speaking etc. I want to continue with the dance and the piano, and I also hope to progress and improve with them.

First of all, the Ds of 40 tasks were computed manually, with the help of *Excel* for the longest calculations. In Table 1, the first column (N) is the number of tokens used to compute TTRs (from 35 to 50, according to Malvern and Richards). The next four columns show the values generated by applying Malvern and Richards formula for D: $TTR=D/N\ [(1+2N/D)^{1/2} -1]$. They are represented in the diagram in blue, that is the "theoretical curve". The last columns show the values generated by our composition, that is, the values generated when we apply the formula to our data ("the empirical curve" shown in the diagram in pink). The D value has been adjusted accordingly so that the error is as small as possible. The smaller the error is, the closer the theoretical and the empirical curve are, as it can be seen in Figure 1.

Secondly, 10 values of D were computed using *vocd* and using *D_Tools*. From Table 2 it can be seen that the difference between the highest (33.84) and lowest (33.42) *vocd* D value for this text is 0.42 and the difference between the highest and lowest *D_Tools* value (33.85-33.41) is 0.44. This is a reasonable fluctuation if we take into account that there is always a small error when we compute D, as this index is the result of a curve-fitting process which is hardly ever exact.

Finally, the correlations between the Ds computed with *vocd* and *D_Tools* in different tasks are very high, they range from .997 to 1 ($p \leq .01$) as indicated in chapter 5. The results for the different tasks using both programs can be seen in Table 3.

	Theoretical curve					Empirical curve		
N	**D/N**	**2*(N/D)+1**	**SQRT**	**SQRT-1**	**D**	**TTRempirical**	**D/N*(SQRT-1)**	**Error**
					33.6			
35	0.96000000	3.08333333	1.75594229	0.75594229		0.72514286	0.72570460	0.00000032
36	0.93333333	3.14285714	1.77281052	0.77281052		0.73055556	0.72128982	0.00008585
37	0.90810811	3.20238095	1.78951975	0.78951975		0.71810811	0.71696929	0.00000130
38	0.88421053	3.26190476	1.80607441	0.80607441		0.71631579	0.71273948	0.00001279
39	0.86153846	3.32142857	1.82247869	0.82247869		0.71000000	0.70859702	0.00000197
40	0.84000000	3.38095238	1.83873663	0.83873663		0.69350000	0.70453877	0.00012185
41	0.81951220	3.44047619	1.85485207	0.85485207		0.69463415	0.70056169	0.00003514
42	0.80000000	3.50000000	1.87082869	0.87082869		0.69238095	0.69666295	0.00001834
43	0.78139535	3.55952381	1.88667003	0.88667003		0.68651163	0.69283984	0.00004005
44	0.76363636	3.61904762	1.90237946	0.90237946		0.68545455	0.68908977	0.00001321
45	0.74666667	3.67857143	1.91796023	0.91796023		0.68755556	0.68541030	0.00000460
46	0.73043478	3.73809524	1.93341543	0.93341543		0.67913043	0.68179910	0.00000712
47	0.71489362	3.79761905	1.94874807	0.94874807		0.68276596	0.67825394	0.00002036
48	0.70000000	3.85714286	1.96396101	0.96396101		0.68416667	0.67477271	0.00008825
49	0.68571429	3.91666667	1.97905701	0.97905701		0.67122449	0.67135338	0.00000002
50	0.67200000	3.97619048	1.99403873	0.99403873		0.67280000	0.66799403	0.00002310
							11.12857670	**0.00047**

Table 1. The values that generate the theoretical curve according to Malvern and Richards' formula (on the left) and the values produced by the TTR values in the example text (on the right). In bold, the final D value and the error.

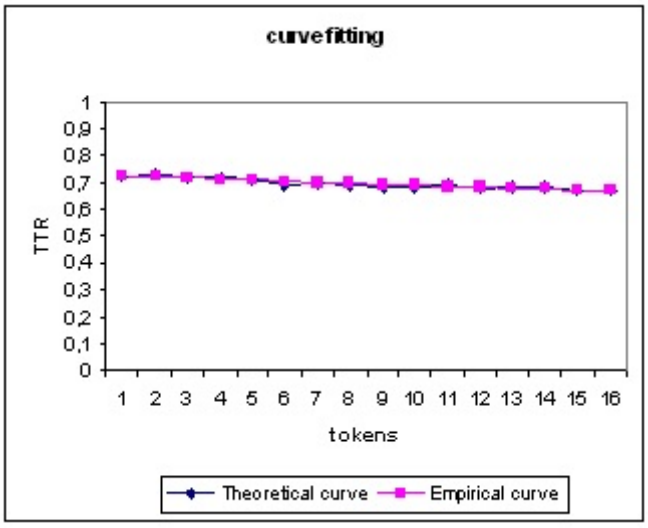


Figure 1. The "theoretical curve" produced by Malvern and Richards' formula and the "empirical curve" produced by the TTRs of the text in the example. As the error is very small, the curves are close together.

Trial	*vocd*	*D_Tools*	D error (*D_Tools*)
1	33.42	33.41	0.000355
2	33.52	33.51	0.000433
3	33.53	33.54	0.000475
4	33.58	33.54	0.000368
5	33.61	33.67	0.000529
6	33.64	33.75	0.000793
7	33.69	33.77	0.000638
8	33.79	33.80	0.000631
9	33.82	33.82	0.000720
10	33.84	33.85	0.000465

Table 2. Ds obtained with the two programs in different trials for the same task are very similar.

	D computed with *D_Tools*				D computed with *vocd*			
Subject Code	D int	D story	D role	D comp	D int	D story	D role	D comp
0150t3a	39.50	16	26.30	28	38.95	16.18	26.10	28.22
0160t3a	30.80	10.90	30.20	37.30	30.77	10.86	30.42	37.06
1084t3a	45.30	15	45.60	33.58	45.68	14.91	45.42	33.75
0208t3a	30.30	20.50	31.50	32.20	29.97	20.44	31.57	32.36
0541t3a	47.90	20.90	44.40	40.70	47.44	21.06	44.25	42.01
0684t3b	51.20	26.90	40.10	55.20	50.74	26.80	40.29	55.22
0625t3b	42.10	24.30	44.30	42.10	42.37	24.62	44.81	41.04
0689t3b	46.80	30.60	45.10	68.50	46.22	30.71	45.20	67.31
0632t3b	47.80	18.30	46.50	46.20	47.33	18.26	46.78	45.37
0753t3b	34.60	29.70	42.40	63.50	34.80	29.41	42.49	63.75
0761t3b	48.40	17.10	25.90	28.90	48.82	17.16	26.08	28.17
0824t3b	53.40	27.40	37	39	53.97	26.98	37.23	36.59
0785t3b	68.80	21.50	48.80	35.40	69.06	21.32	48.74	35.99
05t3av	35.10	16.60	25.90	40.70	35.42	16.66	25.99	40.99
09t3av	45.90	21.60	38.60	45.10	45.85	21.79	38.08	44.51
10t3av	35.60	26.20	30.10	30.70	37.18	26.30	29.96	30.71

Table 3. Ds computed with *D_Tools* and *vocd* for each of the four tasks that the subjects performed.

APPENDIX F

V_SIZE MANUAL

_lognostics

V_Size v. 1.0

Paul Meara
Imma Miralpeix

Centre for Applied Language Studies University of Wales
and
University of Barcelona

CONTENTS

DISCLAIMER

This program is provided free of charge to bona fide researchers on the understanding that the author accept no liability arising out of the use of the programs. We make every attempt to ensure that the program works in the way we expect it to do, but we cannot guarantee that it will work with your data, and you use it at your own risk.

Feedback on the program is encouraged and welcome.

1. INTRODUCTION

Estimating vocabulary size is probably the oldest type of vocabulary research (see for instance Ellegard 1960). V_Size v1.0 is designed to estimate the amount of words known actively by a learner in a particular task. It makes this estimation from the vocabulary profile of a text produced by the learner and the program uses a curve-fitting procedure. The program can also carry out the inverse process, that is, it gives a theoretical profile for a vocabulary of *n* number of words.

Calculation of Vocabulary Size:

When a vocabulary profile is entered into the program, V_Size compares the curve of the actual profile with the curves produced by other theoretical profiles that the program generates. It generates theoretical profiles according to the formula of the Logarithmic Randomisation Function, which takes into account that some words will appear more often than others. Through a process of curve-fitting, the program compares the empirical profile we have entered with the other idealised theoretical profiles to find the "best match". Finally, it computes a theoretical vocabulary size for that particular profile.

2. LOADING THE PROGRAM

V_Size can be downloaded from the _lognostics website:
http://www.swan.ac.uk/cals/calsres/lognostics.htm

The files are provided in .zip format.
To install the program, download the .zip file, and unizip it into a folder called:
c:\lognostics\v_size
You can download the files to other locations if you wish, but it is easier to debug a faulty program if you stick to the recommended folder name.

Installing V_Size will save the following files to your computer:

V_Size.exe	the program
V_Size_Manual.doc	this manual

V_Size does not install any other files on your computer. You can delete V_Size by sending the entire V_Size folder to the recycle bin. Once you have installed the program, you can run it by following the instructions on the pages that follow.

3. RUNNING THE PROGRAM

To run V_Size open the V_Size folder, and click on the icon labelled V_Size. This will open a window like the one shown below (V_size screenshot).

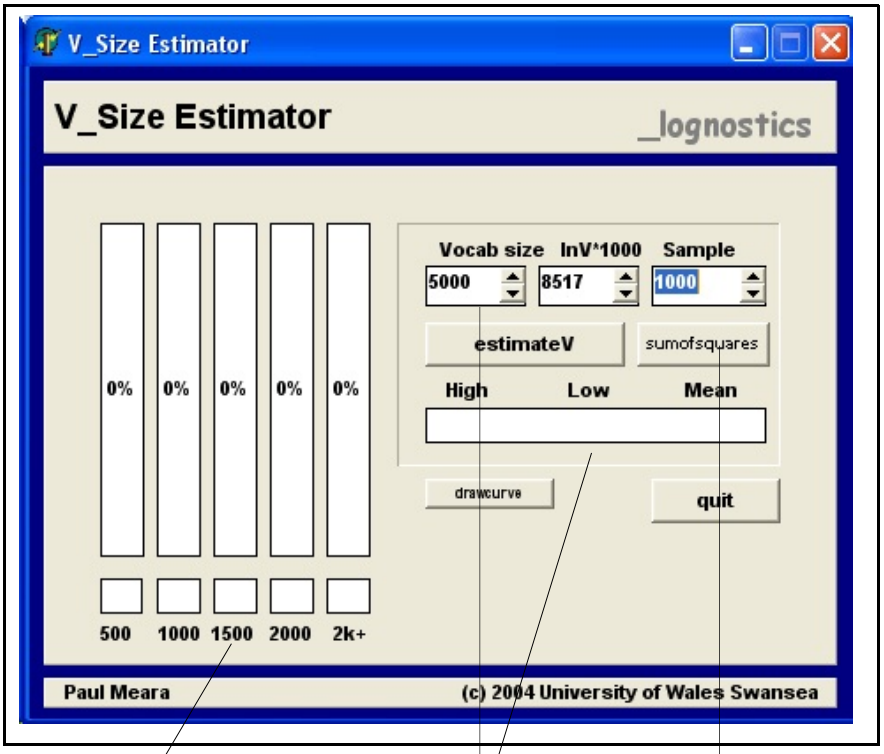
In order to type the vocabulary profile figures into the 5 boxes on the left, you need to obtain first the profile of the text produced by the learner. 5 bands are thus needed to enter the profile into V_Size. Make sure that the band figures add up to 100.

Vocabulary profiles for texts can be calculated by hand following a frequency list (i.e. how many tokens in the text belong to the first 500 words, to the second 500 words and so on...). There are many frequency lists available, although some of the most well-known in English as a Second or Foreign language are Nation's Vocabulary Lists (1996) and the JACET List (2003). There are also some computer tools available to compute profiles:

- *Range* and *VocabProfile*: Both programs were created by Paul Nation and they are available from: http://www.vuw.ac.nz/lals/. They are based on Nation's Vocabulary Lists (1996), which in turn come from A General Service List of English Words by Michael West (Longman, London 1953) and The Academic Word List by Coxhead (1998, 2000). These lists that the program works on can be changed, but only three bands for each profile can be calculated with these two programs. Profiles can also be computed at Tom Cobb's website: http://www.lextutor.ca

- *WordClassifier*, version 2.5 (November 2004). Prepared and updated by the EFL Teacher Training Unit of the Faculty of Arts of the K.U. Leuven (see Goethals 2005). It is downloadable from: http://engels.vvkso-ict.com/engict/wordclassifier2004.zip. It is based on the E.E.T-list (the European English Teaching Vocabulary List) and produces profiles of five bands, although the lists cannot be edited or changed.

V_Size screenshot



Enter into these boxes the profile of your text

The vocabulary estimate will appear here

Error

After entering each band of the profile into its box, click on **estimateV**. Do not change any other figure. The program will then automatically start looking for the Vocabulary Size value that best describes the profile you have entered. This value will appear in the **Vocab size** box as well as in the **High Low Mean** box, the three figures in this box will normally be the same and will coincide with the value given in the **Vocab Size** box. If the values differ, the figure that appears in **Vocab size** will be the best fit value, although this may indicate that your data is abnormal in some way. The figure in **lnV*1000** gives you the logarithm of the vocabulary size you have estimated multiplied by 1,000. The figure in the **Sample** box indicates the number of times you make a sample as part of the estimation process. The default option for this box will always be of 50,000 trials, as it is a sufficiently high value to make an accurate estimation.

The difference found between your profile and the theoretical one will be displayed in the **Sumofsquares** box: the lower the value, the better the fit between the two curves and more precise the estimation will be. This box shows you how closely the result produced by V_Size matches your data. These values should vary between 0 and 10,000. Most data produced by L2 speakers will normally lie between 500 and 10,000.

If you want the program to generate an idealistic profile for a specific vocabulary size, enter the vocabulary size value in the **Vocab size** box, the ln (logarithm) of the value you enter will automatically appear in the **lnV*1000** box. Choose the sample size in the **Sample** box. Click on the **drawcurve** button and the theoretical profile for this vocabulary will appear in the gauges. For instance, the profile for a learner who knows about 5,000 words (**Vocab size**) would be of 73-8-5-3-11 if we make this prediction with a 50,000 **Sample**. You will notice that there is no figure on the **sumofsquares** button in this case, as you are asking just for a theoretical profile, not for a profile that best matches with another. Therefore, we have no error.

Click **quit** to exit the program.

4. THEORETICAL IMPLICATIONS AND WARNINGS

V_Size is an inferential program rather than a descriptive one. Vocabulary estimates may vary from one task or text to another, that is why it is advisable to make more than one estimate per learner. A mean of the estimates for different texts may give us a more reliable value of the learner's productive vocabulary size. It is also worth noticing that vocabulary profiles may be sensitive to the lists on which they are based and probably to the language these lists are in. Finally, we should bear in mind that the logarithmic function may not yet be the best function for an optimal estimation. However, it is one of the very few formulae used in research on vocabulary size that makes a "weighted selection" of words, because it takes into account that some words occur more frequently than others (as observed in Zipf's Law).

For more information, see bibliography below.

5. FORMAL EVALUATION OF THE PROGRAM

This program is currently under evaluation. Any comments and suggestions would be welcome.

6. CITATIONS

If you need to cite this program in your own work, our preferred format is:
Meara, P.M. & Miralpeix, I.
V_Size. Swansea: Lognostics. 2004

Please let us know if you use this program. It helps us get sponsorship for further development work. You can contact us on:
p.m.meara@swansea.ac.uk
imiralpeix@ub.edu

7. REFERENCES

Ellegard, A. 1960. Estimating vocabulary size. *Word, 16*, 219-244.

Goethals, M. 2005. *WordClassifier*. *TESL-EJ* 9 (1) 1-7.

Ishikawa, S., Uemura, T., Kaneda, M., Shimizu, S., Sugimori, N., Tono, Y., et al. 2003. JACET 8000: JACET List of 8000 Basic Words. Tokyo: JACET.

Nation, I. S. P. 1996. Vocabulary Lists. Wellington: Victoria University of Wellington
English Language Institute. Occasional Publication n.17.

Nation, I. S. P. 1995. *VocabProfile*. Wellington.

Zipf, G. K. 1935. The Psycho-Biology of Language: An Introduction to Dynamic Philology. Boston: Houghton Mifflin.

APPENDIX G

PROFILES AND VOCABULARY SIZES GENERATED BY V_SIZE

Voc. size	LnV*1000	Sample	1k	2k	3k	4k	5k
500	6,214	500	100				
		1,000	100				
		3,000	100				
		10,000	100				
		50,000	100				
600	6,396	1,000	98	2			
700	6,551	1,000	96	4			
800	6,684	1,000	94	6			
900	6,802	1,000	92	8			
1,000	6,907	500	89	11			
		1,000	90	10			
		3,000	90	10			
		10,000	90	10			
		50,000	90	10			
1,100	7,003	1,000	88	11	1		
1,200	7,090	1,000	88	10	2		
1,300	7,170	1,000	86	11	3		
1,400	7,244	1,000	86	10	4		
1,500	7,313	500	84	10	6		
		1,000	84	11	5		
		3,000	84	10	6		
		10,000	85	9	6		
		50,000	85	9	6		
1,600	7,377	1,000	83	11	5	1	
1,700	7,438	1,000	84	10	5	1	
1,800	7,495	1,000	83	10	5	2	
1,900	7,549	1,000	82	10	6	2	

Voc. size	LnV*1000	Sample	1k	2k	3k	4k	5k
2,000	7,600	500	82	9	5	4	
		1,000	81	10	6	3	
		3,000	82	9	5	4	
		10,000	82	9	5	4	
		50,000	82	9	5	4	
2,100	7,649	1,000	81	9	6	4	
2,200	7,696	1,000	82	8	6	3	1
2,300	7,740	1,000	80	9	6	3	2
2,400	7,783	1,000	79	9	7	3	2
2,500	7,824	500	79	8	7	3	3
		1,000	79	9	7	3	2
		3,000	78	9	6	4	3
		10,000	79	9	5	4	3
		50,000	79	9	5	4	3
2,600	7,863	1,000	78	9	6	4	3
2,700	7,901	1,000	78	9	6	4	3
2,800	7,937	1,000	77	9	6	4	4
2,900	7,972	1,000	77	9	6	4	4
3,000	8,006	500	76	10	5	4	5
		1,000	78	9	5	4	4
		3,000	78	9	5	3	5
		10,000	77	9	5	4	5
		50,000	77	9	5	4	5
3,100	8,039	1,000	78	8	5	4	5
3,200	8,070	1,000	77	8	5	5	5
3,300	8,101	1,000	78	8	5	4	5
3,400	8,131	1,000	76	8	5	5	6
3,500	8,160	500	75	9	4	5	7
		1,000	76	8	5	5	6
		3,000	76	8	5	4	7
		10,000	75	9	5	4	7

Voc. size	LnV*1000	Sample	1k	2k	3k	4k	5k
		50,000	75	9	5	4	7
3,600	8,188	1,000	76	8	5	4	7
3,700	8,216	1,000	76	8	5	4	7
3,800	8,242	1,000	75	8	5	4	8
3,900	8,268	1,000	75	8	5	4	8
4,000	8,294	500	74	8	5	4	9
		1,000	75	8	5	4	8
		3,000	75	8	5	4	8
		10,000	75	8	5	3	9
		50,000	75	9	5	3	8
4,100	8,318	1,000	75	8	5	4	8
4,200	8,342	1,000	74	8	5	4	9
4,300	8,366	1,000	74	8	5	4	9
4,400	8,389	1,000	73	8	5	4	10
4,500	8,411	500	73	9	5	3	10
		1,000	73	8	5	4	10
		3,000	73	8	5	4	10
		10,000	74	8	5	3	10
		50,000	74	8	5	3	10
4,600	8,433	1,000	74	8	5	3	10
4,700	8,455	1,000	73	8	5	3	11
4,800	8,476	1,000	73	8	5	3	11
4,900	8,496	1,000	73	8	5	3	11
5,000	8,517	500	71	9	5	3	12
		1,000	73	8	5	3	11
		3,000	73	8	5	3	11
		10,000	73	8	5	3	11
		50,000	73	8	5	3	11
5,100	8,536	1,000	74	8	4	3	11
5,200	8,556	1,000	73	8	4	3	12
5,300	8,575	1,000	73	8	4	3	12

Voc. size	LnV*1000	Sample	1k	2k	3k	4k	5k
5,400	8,594	1,000	73	8	4	3	12
5,500	8,612	500	70	9	5	3	13
		1,000	72	8	4	4	12
		3,000	72	8	5	3	12
		10,000	72	8	5	3	12
		50,000	72	8	5	3	12
5,600	8,630	1,000	72	8	5	3	12
5,700	8,648	1,000	71	8	4	4	13
5,800	8,665	1,000	71	8	4	4	13
5,900	8,682	1,000	71	8	4	4	13
6,000	8,699	500	69	9	5	4	13
		1,000	72	7	4	4	13
		3,000	71	8	4	4	13
		10,000	71	8	5	3	13
		50,000	71	8	5	3	13
6,100	8,716	1,000	71	7	5	4	13
6,200	8,732	1,000	71	7	5	4	13
6,300	8,748	1,000	72	7	4	4	13
6,400	8,764	1,000	71	7	4	4	14
6,500	8,779	500	69	8	5	4	14
		1,000	71	7	4	4	14
		3,000	71	7	4	4	14
		10,000	70	8	5	3	14
		50,000	70	8	5	3	14
6,600	8,794	1,000	71	7	4	4	14
6,700	8,809	1,000	71	7	4	4	14
6,800	8,824	1,000	70	8	4	4	14
6,900	8,839	1,000	70	8	4	4	14
7,000	8,853	500	69	8	5	4	14
		1,000	70	8	4	4	14
		3,000	71	7	4	3	15

Voc. size	LnV*1000	Sample	1k	2k	3k	4k	5k
		10,000	71	7	5	3	14
		50,000	70	8	5	3	14
7,100	8,867	1,000	70	8	4	3	15
7,200	8,881	1,000	70	7	4	4	15
7,300	8,895	1,000	71	7	4	3	15
7,400	8,909	1,000	71	7	4	3	15
7,500	8,922	500	68	8	5	3	16
		1,000	71	7	4	3	15
		3,000	71	7	4	3	15
		10,000	70	7	5	3	15
		50,000	69	8	5	3	15
7,600	8,935	1,000	69	8	4	3	16
7,700	8,948	1,000	69	8	4	3	16
7,800	8,961	1,000	70	7	4	3	16
7,900	8,974	1,000	70	7	4	3	16
8,000	8,987	500	68	8	5	3	16
		1,000	70	7	4	3	16
		3,000	70	7	4	3	16
		10,000	69	7	5	3	16
		50,000	68	8	5	3	16
8,100	8,999	1,000	70	7	4	2	16
8,200	9,011	1,000	70	7	4	2	16[69]
8,300	9,024	1,000	70	7	4	3	16
8,400	9,035	1,000	70	7	4	2	17
8,500	9,047	500	68	8	5	2	17
		1,000	69	8	4	2	17
		3,000	69	7	4	3	17
		10,000	69	7	5	3	16

[69] The profiles for the 8,100 and 8,200 vocabulary sizes do not add up to 100 but to 99. This error, which does not affect the results in our study, has already been corrected for the new version of V_Size (v.2.0) downloadable from http://www.lognostics.co.uk/tools/index.htm.

Voc. size	LnV*1000	Sample	1k	2k	3k	4k	5k
		50,000	68	8	5	3	16
8,600	9,059	1,000	69	8	4	2	17
8,700	9,071	1,000	68	8	4	3	17
8,800	9,082	1,000	68	8	4	3	17
8,900	9,093	1,000	68	8	4	3	17
9,000	9,104	500	67	8	5	3	17
		1,000	68	8	4	3	17
		3,000	68	7	5	3	17
		10,000	68	7	5	3	17
		50,000	68	7	5	3	17
9,100	9,116	1,000	68	8	4	3	17
9,200	9,126	1,000	68	7	4	3	18
9,300	9,137	1,000	68	7	4	3	18
9,400	9,148	1,000	68	7	4	3	18
9,500	9,159	500	66	8	4	4	18
		1,000	68	7	4	3	18
		3,000	68	7	4	3	18
		10,000	68	7	5	3	17
		50,000	68	7	5	3	17
9,600	9,169	1,000	67	8	4	3	18
9,700	9,179	1,000	67	8	4	3	18
9,800	9,190	1,000	68	7	4	3	18
9,900	9,200	1,000	67	8	4	3	18
10,000	9,210	500	67	8	4	3	18
		1,000	67	8	4	3	18
		3,000	67	8	4	3	18
		10,000	68	7	4	3	18
		50,000	67	7	5	3	18

Table 1. Theoretical profiles obtained for different vocabulary sizes and samples.

Empirical Profile					Theoretical Size and Profile							
1k	**2k**	**3k**	**4k**	**5k**	**Vocabulary Size**	**LnV*1000**	**Error (Sum of Squares)**	**1k**	**2k**	**3k**	**4k**	**5k**
90	10				1,000	6,907	0	90	10			
80	20				1,300	7,170	152	86	10	4		
70	30				3,300	8,101	554	76	9	5	4	6
60	40				4,200	8,342	1,272	74	9	5	3	9
50	50				4,200	8,342	2,372	74	9	5	3	9
90	5	5			1,400	7,244	32	86	9	5		
80	10	10			1,700	7,438	30	83	9	6	2	
80	15	5			1,700	7,438	50	83	9	6	2	
70	20	10			3,300	8,101	234	76	9	5	4	6
70	15	15			3,300	8,101	224	76	9	5	4	6
70	25	5			3,300	8,101	344	76	9	5	4	6
60	30	10			4,200	8,342	752	74	9	5	3	9
60	25	15			4,200	8,342	642	74	9	5	3	9
90	6	2	2		1,100	7,003	22	89	10	1		
90	6	3	1		1,000	7,003	22	89	10	1		
85	8	5	2		1,800	7,495	2	84	9	5	2	
86	7	5	2		1,800	7,495	8	85	9	5	2	
80	10	5	5		2,200	7,696	4	81	9	5	4	1
80	15	3	2		2,200	7,696	46	81	9	5	4	1
80	8	7	5		2,200	7,696	8	81	9	5	4	1
80	18	1	1		2,200	7,696	108	81	9	5	4	1
90	4	3	2	1	1,100	7,003	46	89	10	1		
90	5	2	2	1	1,100	7,003	32	89	10	1		
90	7	1	1	1	1,100	7,003	12	89	10	1		
85	10	3	1	1	1,300	7,170	4	86	10	4		
85	8	4	2	1	1,800	7,495	4	84	9	5	2	
85	7	6	1	1	1,600	7,377	6	84	9	6	1	
80	15	3	1	1	2,400	7,783	50	80	9	5	4	2
80	10	5	3	2	2,400	7,783	2	80	9	5	4	2

Empirical Profile					Theoretical Size and Profile							
1k	**2k**	**3k**	**4k**	**5k**	**Vocabulary Size**	**LnV*1000**	**Error (Sum of Squares)**	**1k**	**2k**	**3k**	**4k**	**5k**
80	6	5	5	4	2,600	7,863	12	79	9	5	4	3
75	20	3	1	1	2,800	7,937	152	78	9	5	4	4
75	15	5	3	2	3,000	8,600	50	77	9	5	4	5
75	10	7	5	3	3,000	8,600	14	77	9	5	4	5
75	7	6	6	6	3,500	8,160	10	75	9	5	4	7
70	25	3	1	1	3,500	8,160	330	75	9	5	4	7
70	20	5	3	2	3,500	8,160	172	75	9	5	4	7
70	15	7	5	3	3,500	8,160	82	75	9	5	4	7
70	12	7	6	5	3,500	8,160	46	75	9	5	4	7
65	30	3	1	1	3,500	8,160	590	75	9	5	4	7
65	25	5	3	2	3,500	8,160	382	75	9	5	4	7
65	20	7	5	3	3,500	8,160	242	75	9	5	4	7
65	15	10	5	5	4,200	8,342	162	74	9	5	3	9
65	15	10	8	2	3,500	8,160	202	75	9	5	4	7
65	10	9	8	8	7,200	8,881	106	70	8	5	3	14

Table 2. Theoretical or estimate vocabulary sizes and profiles for different empirical profiles.