Type-token & Hapax-token Relation: A Combinatorial Model*

Jiří Milička†

Abstract

If we consider type-token relation to be a feature of text and not of language, we can approach a theoretically based and precise description of this relation. Such description will suit the demands of text linguistics better than the empirical laws that are used nowadays. This paper offers a model of the relation based on the combinatorial characterization of the distribution of types in a text. This method is subsequently used to formulate a model of hapax-token relation and the subject is further generalized.

1 Introduction

We scan a text, token by token (e.g. word form), and whenever we identify a new type, we record it in our lexicon list. Thereby we get a function in which each token in the text is assigned to the relevant number of types in the lexicon[1]. The curve which represents this function[2] is an outcome of the laws of probability (since any text is a natural entity), other linguistic laws, and the author's intention (i. e. the semantic content and function of the text), it needs nevertheless to be said that we cannot exclude other factors.

If we want to study the textologicaly-interesting (or generally linguistic) component of the mechanism that plays an influence upon the shaping of this curve, then it is essential to subtract from it the probability function that would simulate this curve if this were to be related to the quantity of data with the same elements as in the text, without however either semantic content or textual structure. Of course, the mere random transposition of tokens would be ineffective since the curve's shape would be affected by random deflections. It would be ideal to arrive at the demanded curve by averaging type-token curves measured for all permutations of tokens in the text under investigation. It is however necessary to describe the *result*

*Published in *Glottotheory. International Journal of Theoretical Linguistics 2/1.* (2009), pp. 99–110. ISSN 1337–7892. DOI:10.1515/glot-2009-0009.

†www.milicka.cz, jiri@milicka.cz

[1]For details see [2], the method of measurement we use describes Wimmer in Ch. 4, Para. iii.

[2]And which resembles a power function, on which Heaps' Law is based.

of such computationally unmanageable efforts through a mathematical model[3]. Using permutations to discover hidden features of a text may resemble the Kabbalah[4] but our approach was strictly non-religious.

Heaps' Law, constructed empirically, sets diametrically differing aims from those we set ourselves. While our task relates to *text* and is limited by its size[5], Heaps' Law attempts to predict a trend on the basis of parameters relating to "language" (or as the case may be, style)[6]. This is why the law is to some extent valid for predicting vocabulary size for large corpora [1], but when applied to small texts, we can notice some obvious features of unsystematic approach in Heaps' method (apart from the different character of the curve, the fact that it fails to respect clearly set requirements that the number of types does not exceed the number of tokens, etc). In addition, parameters are determined on an ad hoc basis, according to previous course of the measured curve, so that the values of these parameters are influenced by the influences of the linguistic components of the curve

We shall endeavour to find an adequate theoretical model and we shall only use empirical data to corroborate the hypothesis. Then we shall apply the same method and approach to model the hapax-token relation (including its generalised version), demonstrating that they have the same nature.

2 The Type-token Relation

We start with input data represented by the absolute frequencies of all types in text and expect that subsequently it will be possible to capture them by Zipf's distribution (or another suitable distribution) subsequently.

2.1 Theoretical Derivation

For each token, we can determine whether it belongs to a given type or not — this is a binary decision. Hence the number of all possibilities, how tokens of a given type can be transposed in a text, is a number of permutations of two groups of same elements with repetitions[7], while the number of elements in the first group is equal to the number of occurrences of the given type in the text, and the second group is the complement of

[3]Thus, not dependent upon the subject to which it is related. Let us imagine for instance, a hunter, who shoots a variety of game in a fenced forest. How do the numbers of animals killed relate to the number of different species killed?

[4]Two stones build two houses, three stones build six houses, four build twenty-four houses, five build one hundred and twenty houses, six build seven hundred and twenty houses and seven build five thousand and forty houses. From thence further go and reckon what the mouth cannot express and the ear cannot hear. (Sepher Yezirah 4:16, translated by Isidor Kalisch, New York 1877).

[5]The moment the hunter shoots the last animal in the fenced forest, the relation loses sense.

[6]Presupposes an unfenced forest.

[7]English terminology is rather confusing, we use term *permutations with repetitions* in its German sense (i. e. strictly speaking *Anordnungen von Objekten aus mehreren Klassen*).

this count in relation to the length of the text (expressed by the number of tokens). Let r_0 be the number of all permutations of tokens belonging to a given type in the whole text, then:

$$r_0 = \frac{d!}{f_i!(d-f_i)!}$$

Where d is the number of all tokens in the text, and f_i is the number of occurrences of type i in the text (i. e. its frequency).

Considering the N^{th} position (the N^{th} token of the text), in how many of these permutations is the given type included in the lexicon? I. e. in how many of these permutations occurred a token of the given type up to the N^{th} position (inclusively)? Let us begin with a complement of this set — we are trying to determine a formula that would express the number of the permutations in which any token of a given type did *not* occur up to position N (inclusively): We are to exclude all possibilities, where the given type occurs up to the N^{th} position; this means that the length of the text is reduced by N, but the number of occurrences of the given type in the text remains constant (the size of the first group of elements does not change, while the second is dependent on N).

$$r_N = \frac{(d-N)!}{f_i! \cdot (d-f_i-N)!}$$

The quotient of the number of possible permutations of the given type in the text after the N^{th} position and the number of all possible permutations expresses the probability that the given type did *not* occur in any position up to the N^{th} position (i. e. the probability that the given type is *not* included in the lexicon considering the N^{th} token in the text).

$$p'_{i;N} = \frac{r_{i;N}}{r_{i;0}} = \frac{\frac{(d-N)!}{f_i!(d-N-f_i)!}}{\frac{d!}{f_i!(d-f_i)!}}$$

The complement of this probability expresses the probability, that the given type is recorded in the lexicon after we scan the N^{th} position[8].

$$p_{i;N} = 1 - \frac{\frac{(d-N)!}{f_i!(d-N-f_i)!}}{\frac{d!}{f_i!(d-f_i)!}}$$

The number of V types for N tokens is sum of these probabilities for all types:

$$V = \sum_{i=1}^{M} p_{i;N}$$

[8]In order to illustratively demonstrate this, let us adduce the following example: The first matrix expresses all possible distributions of tokens of a given type in the text (1 — occurrence; 0 — absence of the type). The rows depict individual possible distributions, the columns indicate the individual positions (i. e. tokens). This illustration is related to a "text" with six tokens, in which chosen type occurs three times.

If the given type in the text occurred in the N^{th} position, then it is included in the lexicon in the following positions too. This is represented by the second matrix. The number of zeros in the N^{th} column of the second matrix expresses the value of r_N; the quotient of the number of *1* symbols in the N^{th} column and the number of all of the rows of the matrix expresses the probability that the given type is recorded in the lexicon in the N^{th} position.

By substitution and a simple transformation we arrive at the final formula:

$$V = M - \sum_{i=1}^{M} \frac{(d-N)!(d-f_i)!)}{d!(d-N-f_i)!} \qquad \text{for } d \geq N + f_i \wedge N; f_i; d \in N$$

- Variables:
  N ... number of tokens
  V ... number of types
- Parameters:
  f_i ... number of occurrences of type i in the text (frequency)
  M ... maximum number of types in the text
  d ... number of tokens in the text, which we have measured for f_i and M;
  $d = \sum_{i=1}^{M} f_i$
- Control variable:
  i... order of type in the lexicon

2.2 Practical Implementation

Taking into consideration the technical demands of factorial computation for large numbers, our algorithm must be based on the following formulation:

$$p_{i;N} = 1 - 1 \cdot (d-N-f_i) : (d-N) \cdot (d-N-f_i+1) : (d-N+1) \cdot \ldots \cdot (d-f_i-1) : (d-1)$$

If we require the whole sequence of increasing numbers of types in the lexicon (to make a graph), the recurrent shape of the equation ensures that the calculation time will be linearly dependent on the number of tokens.

0	0	0	1	1	1		0	0	0	1	1	1
0	0	1	0	1	1		0	0	1	1	1	1
0	0	1	1	0	1		0	0	1	1	1	1
0	0	1	1	1	0		0	0	1	1	1	1
0	1	0	0	1	1		0	1	1	1	1	1
0	1	0	1	0	1		0	1	1	1	1	1
0	1	0	1	1	0		0	1	1	1	1	1
0	1	1	0	0	1		0	1	1	1	1	1
0	1	1	0	1	0		0	1	1	1	1	1
0	1	1	1	0	0		0	1	1	1	1	1
1	0	0	0	1	1	$\longrightarrow$	1	1	1	1	1	1
1	0	0	1	0	1		1	1	1	1	1	1
1	0	0	1	1	0		1	1	1	1	1	1
1	0	1	0	0	1		1	1	1	1	1	1
1	0	1	0	1	0		1	1	1	1	1	1
1	0	1	1	0	0		1	1	1	1	1	1
1	1	0	0	0	1		1	1	1	1	1	1
1	1	0	0	1	0		1	1	1	1	1	1
1	1	0	1	0	1		1	1	1	1	1	1
1	1	1	0	0	0		1	1	1	1	1	1

$$p'_{i;0} = 1$$

$$p'_{i;n} = p'_{i;n-1}\frac{d - f_i - n + 1}{d - n + 1} \qquad \text{for } d > n - 1 \wedge n; f_i; d \in N$$

$$V = \sum_{i=1}^{M} 1 - p'_{i;N}$$

The graphs in the following appendices were calculated using algorithms based on these relations:

Appendix 1 shows, how the curve of our combinatorial model parallels a curve measured on Cooper's novel *The Last of the Mohicans*[9], these curves for a short text we can find in Appendix 2[10]; Appendix 3 compares a curve measured on a part of *The Last of the Mohicans* with randomly transposed tokens with the expected curve for the original text.

In Appendix 4 we can see that it is difficult to determine, whether two type-token curves measured on two language mutations of the same text show any similarities. But if we subtract values combinatorialy modelled from these curves (see appendices 5 and 6), correspondences and differences are evident[11]. The data for these three appendices were gained from a part of the Qur'ān and its Czech translation, to compare highly incoherent texts in two completely different languages as an example of the application of this model.

3 Hapax-token Relation

We derive a model for this relation using the same method as in the case of type-token relation; however, we exclude from the lexicon any type that occurred for a second time in the text[12].

3.1 Theoretical Derivation

If the given type occurs in the given position N for the first time, or if it does not occur in this position and at the same time it had occurred just once in the previous positions, then the given type is a hapax in the N^{th} position.

[9] Graphical words, case and punctuation insensitive, were considered to be tokens.

[10] Two-page Arabic short story *Ḥilmu 'l-'Abd* by Nabīl Na"ūm Ǧorǧī.

[11] Should we proportionally align the two curves that we got by subtraction of the measured and modelled curves, we arrive at a correlation coefficient of about 0.9; so, we can claim that the difference between the measured and modelled curve is, to some extent, language independent and characteristic for each text. Of course we can assume, that this is influenced by the typology of each language (i. e. connection of average distance between a curve and the axis x to syntheticity of language is possible), or potentially by cultural conventions (e. g. Arabic stylistics differs from European ones and tolerates several usages of the same word in a small segment of the text) etc., but these facts are not a subject of this paper.

[12] The first matrix representing all possible distributions of tokens of the given type in the text does not differ from the matrix in footnote 8; The second one marks by symbol *1* the positions where the token occurs as a hapax. As in the first case, this matrix helps us to

Analogically to type-token relation we permute the rest of occurrences of the given type ($f_i - 1$; i. e. the first group of elements) and its complement in the rest of tokens ($(d - N) - (f_i - 1)$; i. e. the second group of elements) N times, because the type could occur for the first time in N positions.

$$V_1 = \sum_{i=1}^{M} \frac{\frac{N(d-N)!}{(f_i-1)!(d-N-f_i+1)!}}{\frac{d!}{f_i!(d-f_i)!}} \qquad \text{for } d \geq N - 1 + f_i \wedge N; f_i; d \in N$$

- Variables:
  N ... number of tokens
  V_1 ... number of hapaxes
- Parameters:
  f_i ... number of occurrences of type i in the text (frequency)
  M ... maximum number of types in the text
  d ... number of tokens in the text, which we have measured for f_i and M;
  $d = \sum_{i=1}^{M} f_i$
- Control variable:
  i... order of type in the lexicon

3.2 Practical Implementation

Again, we look for the least computationally demanding form — the recurrent shape. By simple transformations, we get:

$$V_1 = N \sum_{i=1}^{M} p_{i,N}$$

depict our considerations by the concrete example.

0	0	0	1	1	1		0	0	0	1	0	0
0	0	1	0	1	1		0	0	1	1	0	0
0	0	1	1	0	1		0	0	1	0	0	0
0	0	1	1	1	0		0	0	1	0	0	0
0	1	0	0	1	1		0	1	1	1	0	0
0	1	0	1	0	1		0	1	1	0	0	0
0	1	0	1	1	0		0	1	1	0	0	0
0	1	1	0	0	1		0	1	0	0	0	0
0	1	1	0	1	0		0	1	0	0	0	0
0	1	1	1	0	0	$\longrightarrow$	0	1	0	0	0	0
1	0	0	0	1	1		1	1	1	1	0	0
1	0	0	1	0	1		1	1	1	0	0	0
1	0	0	1	1	0		1	1	1	0	0	0
1	0	1	0	0	1		1	1	0	0	0	0
1	0	1	0	1	0		1	1	0	0	0	0
1	0	1	1	0	0		1	1	0	0	0	0
1	1	0	0	0	1		1	0	0	0	0	0
1	1	0	0	1	0		1	0	0	0	0	0
1	1	0	1	0	1		1	0	0	0	0	0
1	1	1	0	0	0		1	0	0	0	0	0

where:

$$p_{i;1} = \frac{f_i}{d}$$

$$p_{i;n} = p_{i;n-1}\frac{d-n-f_i+2}{d-n+1} \qquad \text{for } d > n-1 \wedge n; f_i; d \in N$$

Appendices 7 and 8 show how the hapax-token model parallels an undisrupted text (*The Last of the Mohicans*) and how disrupted text structure (randomly transposed tokens of a part of this novel). The following two appendices outline the dependencies between type-token and hapax-token relations: Appendix 9 provides measured and modelled curves, while in Appendix 10 are subtractions of these curves (the data gained from a part of the novel *The Invisible* by H. G. Wells).

4 Further Generalisation

Hapax-token is in fact a special case, where a given type occurs once in the text, but we can however ask ourselves, what relation is relevant for types which occurred in N tokens exactly g times. Given the lack of practical applicability, this relation has mostly been neglected, let us adduce its model to put the hapax-token ratio into a wider context.

As in the previous, case we permute rest of occurrences of the given type ($f_i - g$; i. e. the first group of elements) and its complement in the rest of tokens ($(d-N)-(f_i-g)$; i. e. the second group of elements); while for hapaxes these permutations could only occur N times (i. e. how many times the type could occur for the first time), now it can occur so many times, how many times the given type could occur for the g^{th} time, so we permute g occurrences of the given type and its complement in N tokens. After multiplication we arrive at the following formula:

$$V_g = \sum_{i=1}^{M} \frac{\frac{N!}{g!(N-g)!}\frac{(d-N)!}{(f_i-g)!(d-N-f_i+g)!}}{\frac{d!}{f_i!(d-f_i)!}} \qquad \text{for } N \geq g \wedge d \geq N-g+f_i \wedge N; f_i; d; g \in N$$

- Variables:
  N ... number of tokens
  V_g ... number of types that occur g times
- Parameters:
  g ... demanded number of occurrences of the given type
  f_i ... number of occurrences of type i in the text (frequency)
  M ... maximum number of types in the text
  d ... number of tokens in the text, which we have measured for f_i and M;
  $d = \sum_{i=1}^{M} f_i$
- Control variable:
  i... order of type in the lexicon

By simple transformation we again derive a recurrent form:

$$V_g = \sum_{i=1}^{M} p_{i,N}$$

We begin from the g^{th} token:

$$p_{i;g} = \frac{(f_i - g + 1) \cdot (f_i - g + 2) \cdot \ldots \cdot f_i}{(d - g + 1) \cdot (d - g + 2) \cdot \ldots \cdot d}$$

$$p_{i;n} = p_{i;n-1} \frac{n}{n-g} \frac{d - n - f_i + g + 1}{d - n + 1} \qquad \text{for } d > n{-}1 \wedge d > g \wedge n; f_i; d \in N$$

Appendix 11 shows, the functionality of this model for types that occur exactly 5 times in the novel *The Last of the Mohicans*.

5 Conclusion

We have found a mathematical model which simulates type-token relation, based on probability and combinatorics, to simulate the sequence of elements without structure; it gives us a possibility of gaining a new insights to the structure of a text.

An output of this system is the curve modelling the demanded relation, as an input it needs absolute frequencies of all types in the text, so it outlines the connection between type-token relation and Zipf's Law. The derivation method was applied to hapax-token relation and this model was further generalised. The equations we find were converted into algorithms, by means of which the model was tested and the examples are presented in the appendices; we suggested falsification methods, so we can consider the theory to be ready for corroboration.

Further development of these models dwells in reduction of input data using Zipf's or other suitable distribution and in discovering connections, which have remained hidden to the author.

Some applications have been suggested in this paper. By subtraction of the modelled curve from the measured one we acquire the linguistic component of these relations, which can be exploited in the field of text linguistics (text coherence, the semantics of longer parts, suprasentential segmentation), in the science of literature (in studying the repetition of longer elements or how the progress in the lexicon correlates with the meaning of the work, its style, and composition); and finally, it can contribute to the formation of a more profound quantitative insight into language (the fractal nature of text, the similarities between text and non-language structures etc.).

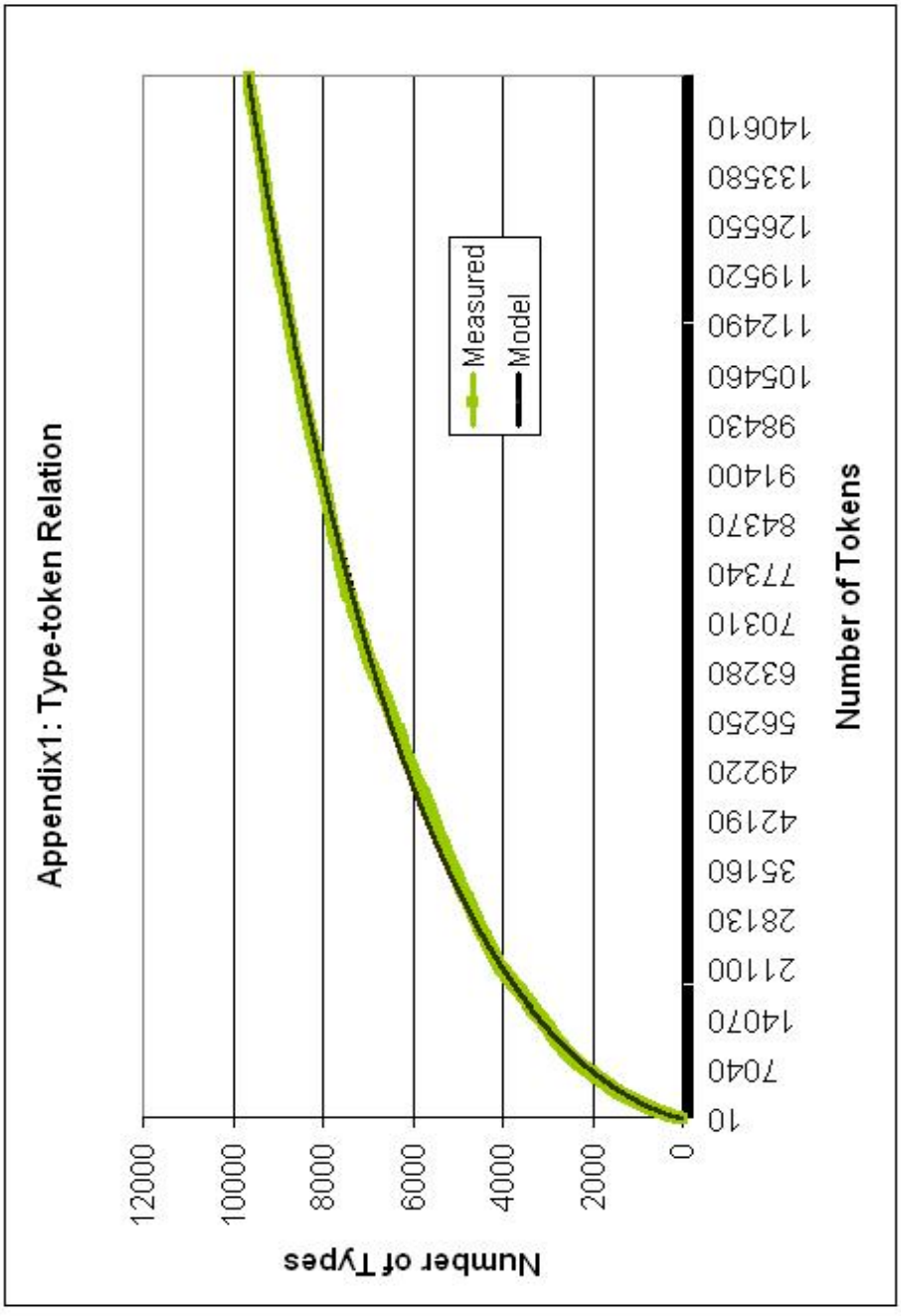
This model was tested for situations where the tokens represented words and character n-grams, but we can however assume that it will work for syllables, morphemes, lemmas etc., and possibly even tokens could represent non-language elements.

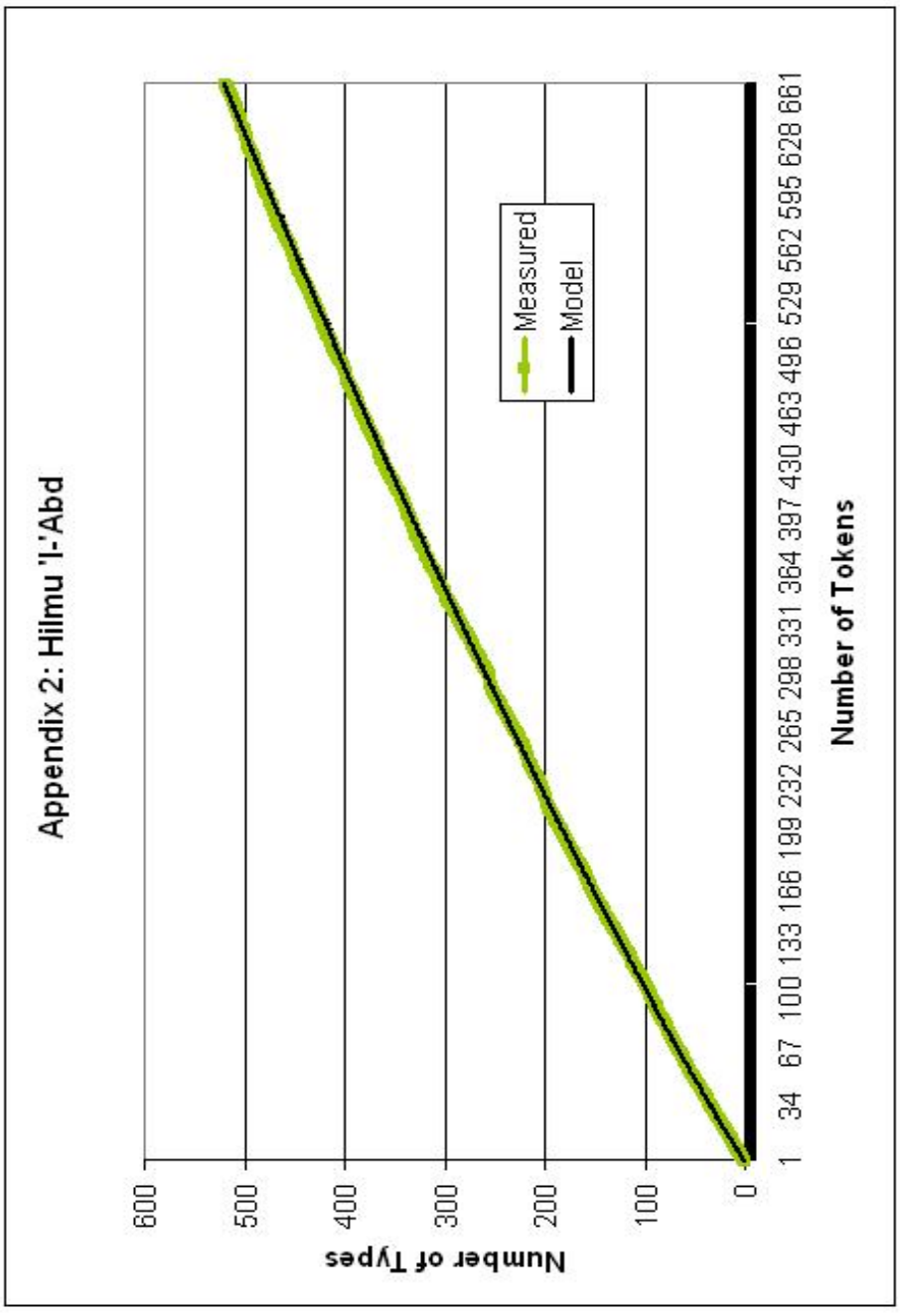
References

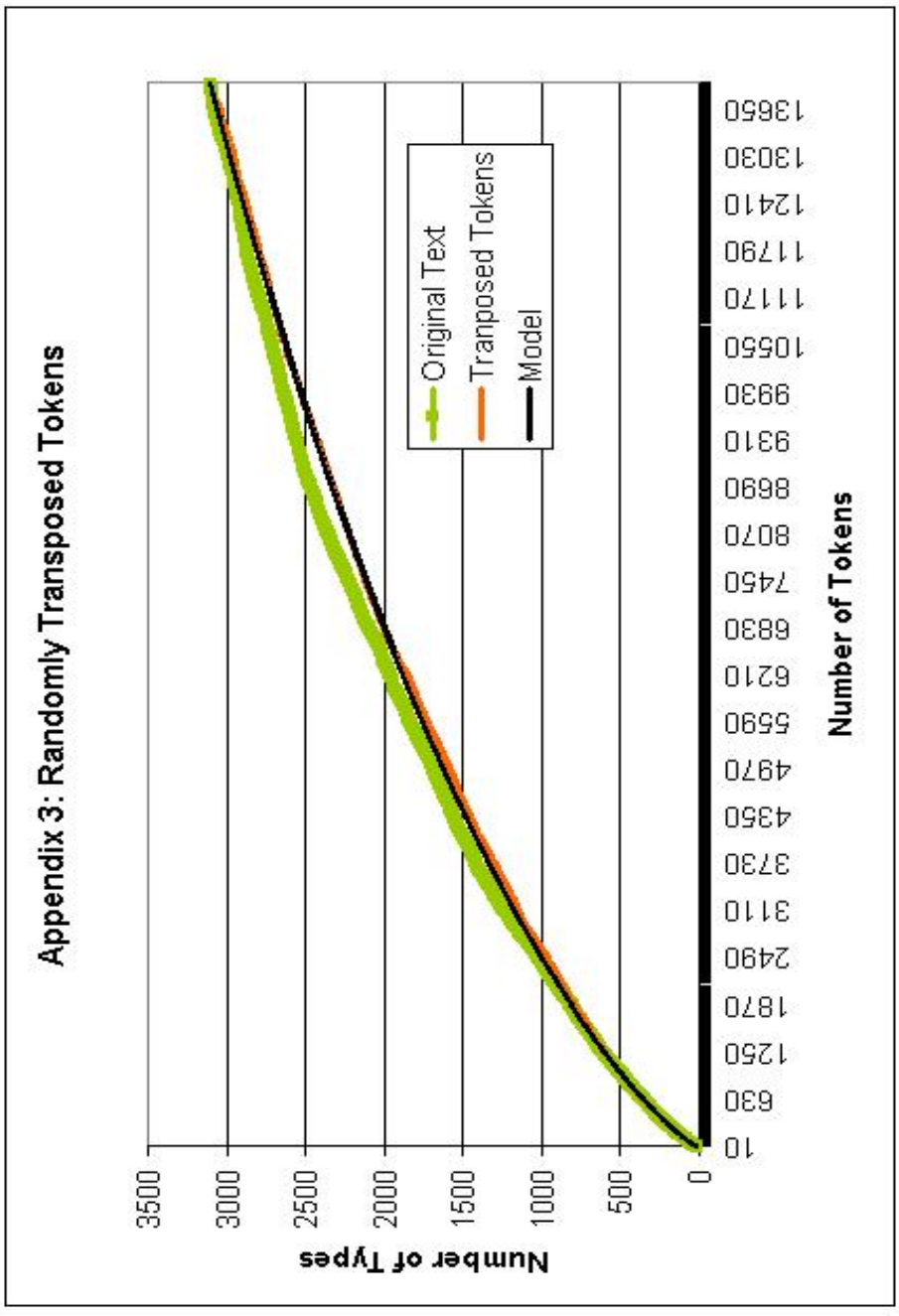
[1] French J., C.: *Modeling Web Data.* In: JCDL '02: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries. Portland 2002, pp. 320–321.

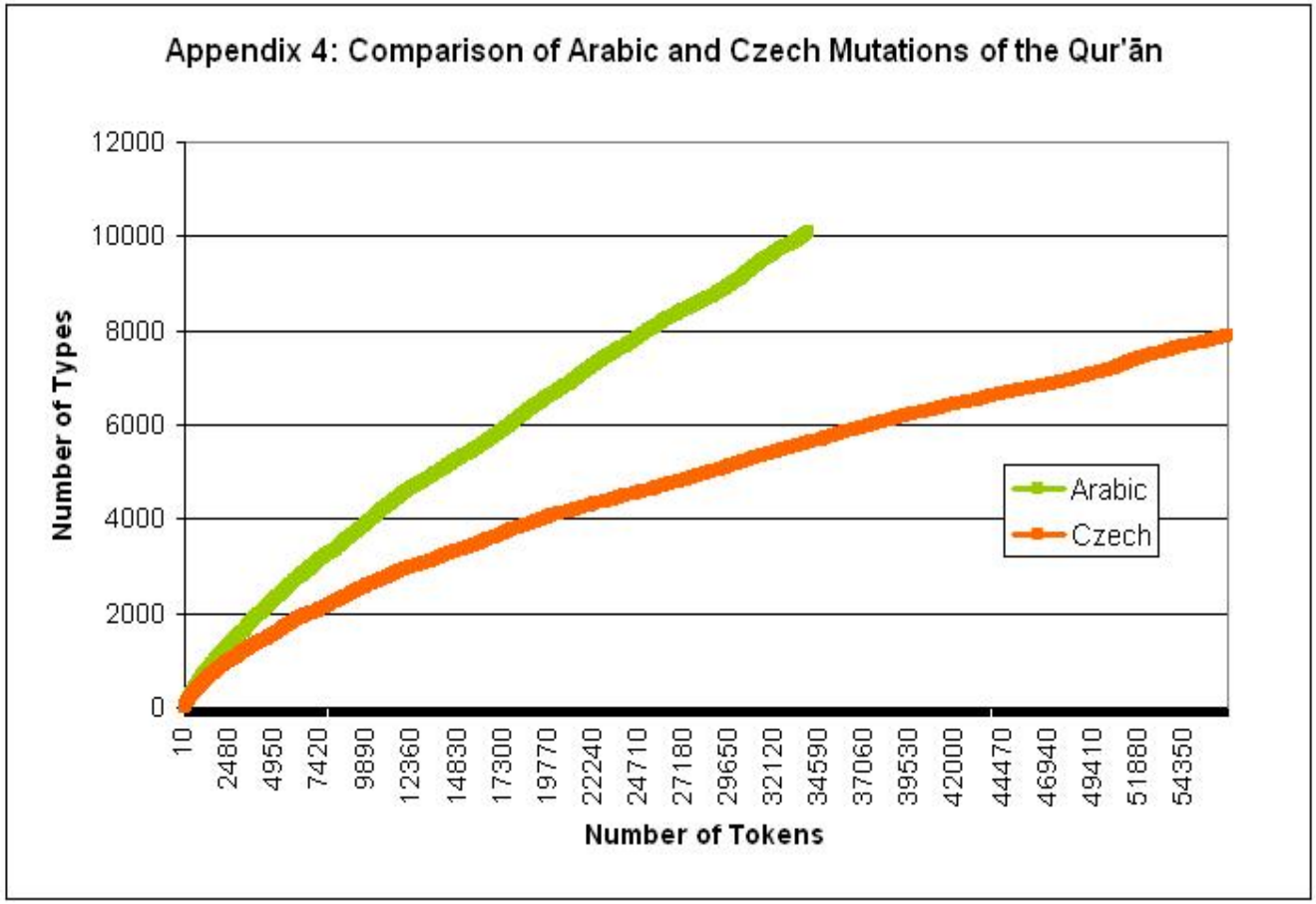
[2] Wimmer G.: The type-token relation. In: R. Köhler (ed.), *Quantitative Linguistics. An International Handbook.* 2005, pp 361–368

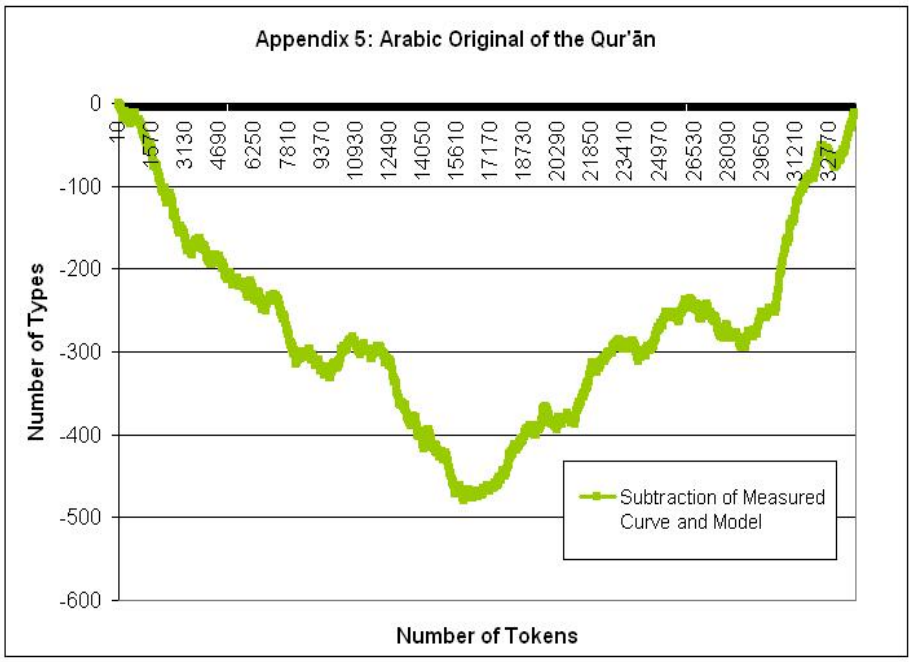
[3] Wyllys, R. E.: *Empirical and Theoretical Bases of Zipf's Law.* In: Library Trends 30(1) no. 53–64. 1981, pp 53–64.

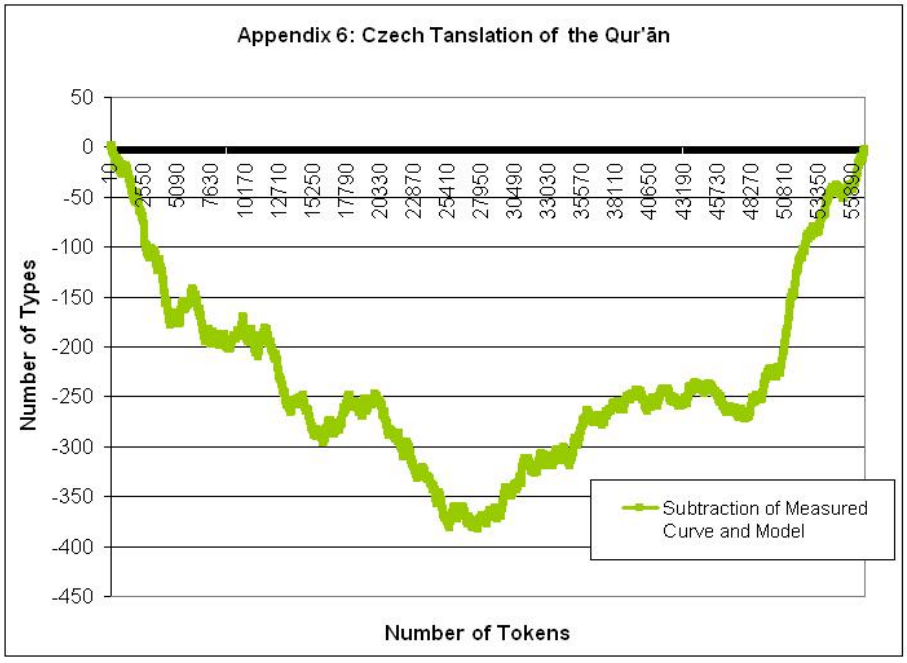

Appendix1: Type-token Relation
Number of Types
12000
10000
8000
6000
4000
2000
0
Measured
Model
10
7040
14070
21100
28130
35160
42190
49220
56250
63280
70310
77340
84370
91400
98430
105460
112490
119520
126550
133580
140610
Number of Tokens

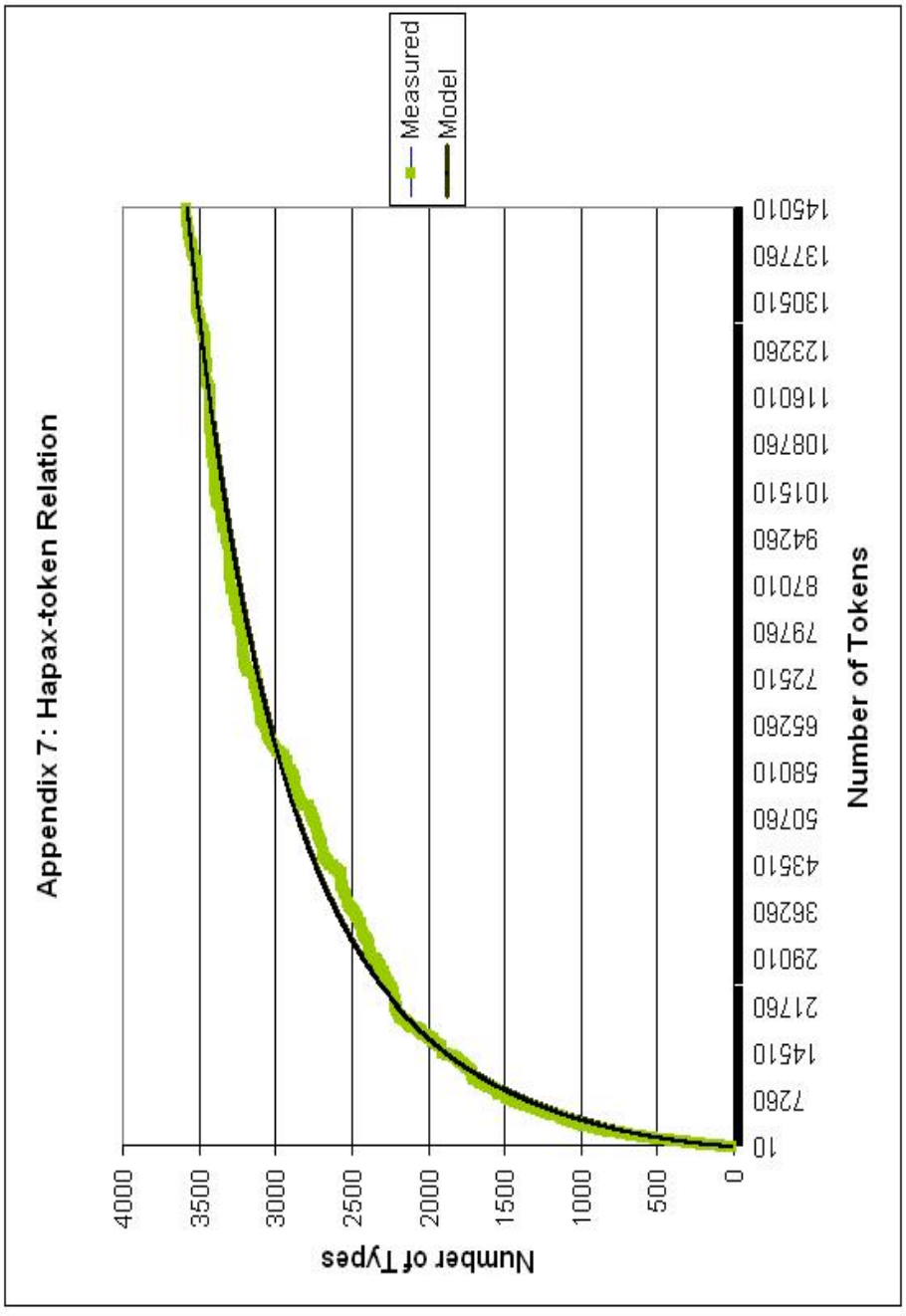

Appendix 2: Hilmu 'l-'Abd
Number of Types
600
500
400
300
200
100
0
1 34 67 100 133 166 199 232 265 298 331 364 397 430 463 496 529 562 595 628 661
Number of Tokens
Measured
Model

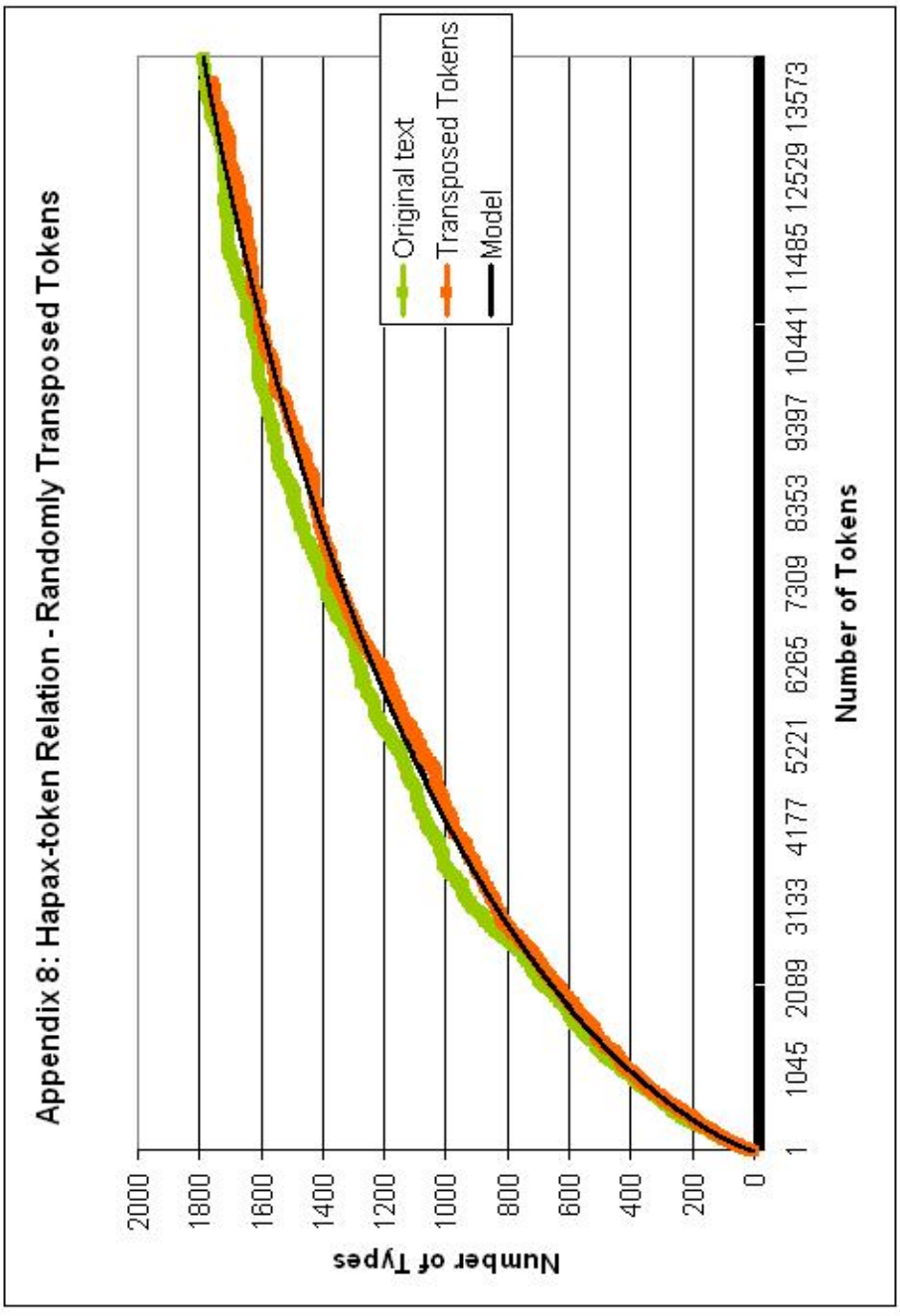

Appendix 3: Randomly Transposed Tokens
Number of Types
3500
3000
2500
2000
1500
1000
500
0
Original Text
Tranposed Tokens
Model
10
630
1250
1870
2490
3110
3730
4350
4970
5590
6210
6830
7450
8070
8690
9310
9930
10550
11170
11790
12410
13030
13650
Number of Tokens

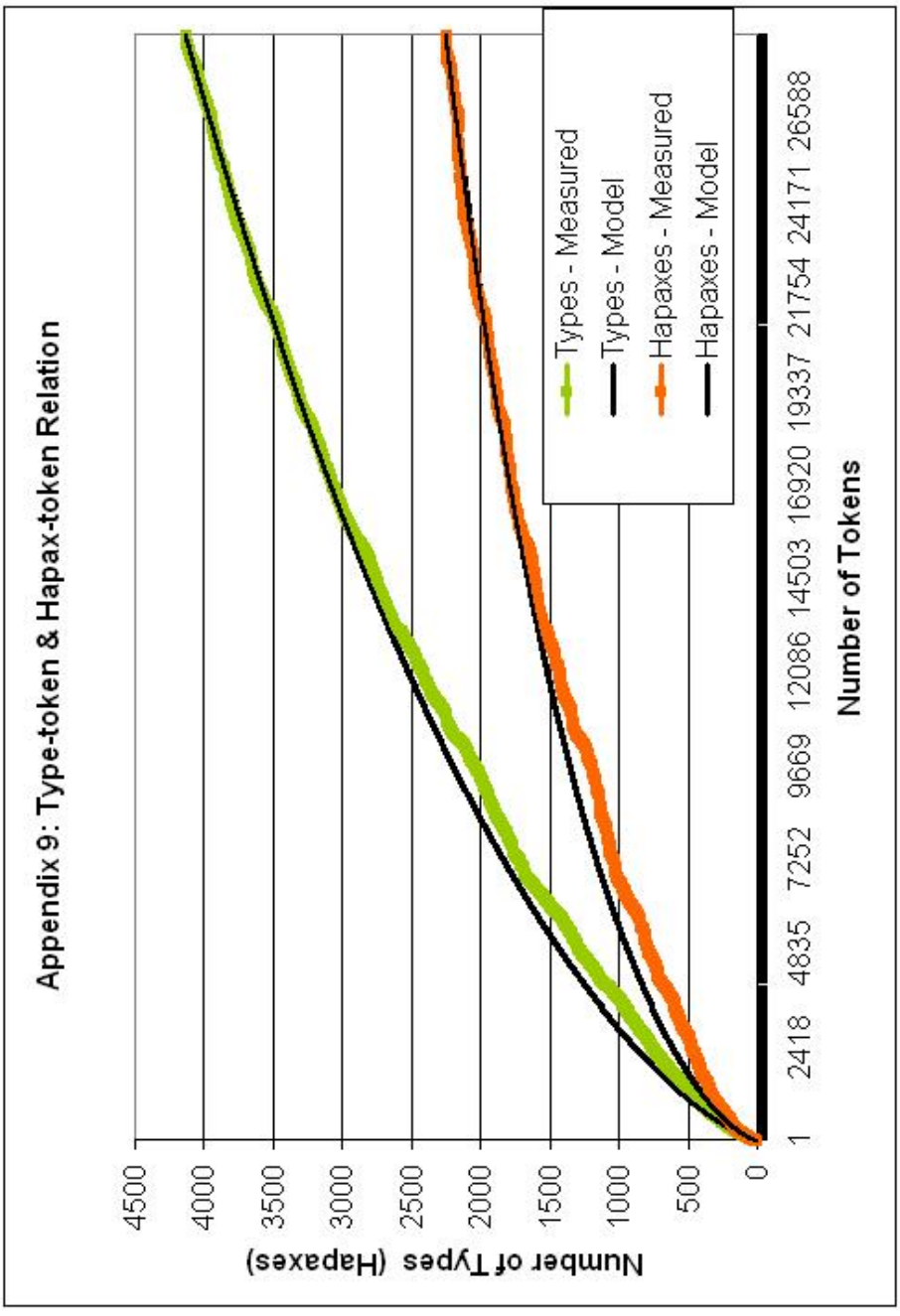

Appendix 4: Comparison of Arabic and Czech Mutations of the Qur'ān
Number of Types
12000
10000
8000
6000
4000
2000
0
10
2480
4950
7420
9890
12360
14830
17300
19770
22240
24710
27180
29650
32120
34590
37060
39530
42000
44470
46940
49410
51880
54350
Number of Tokens
Arabic
Czech

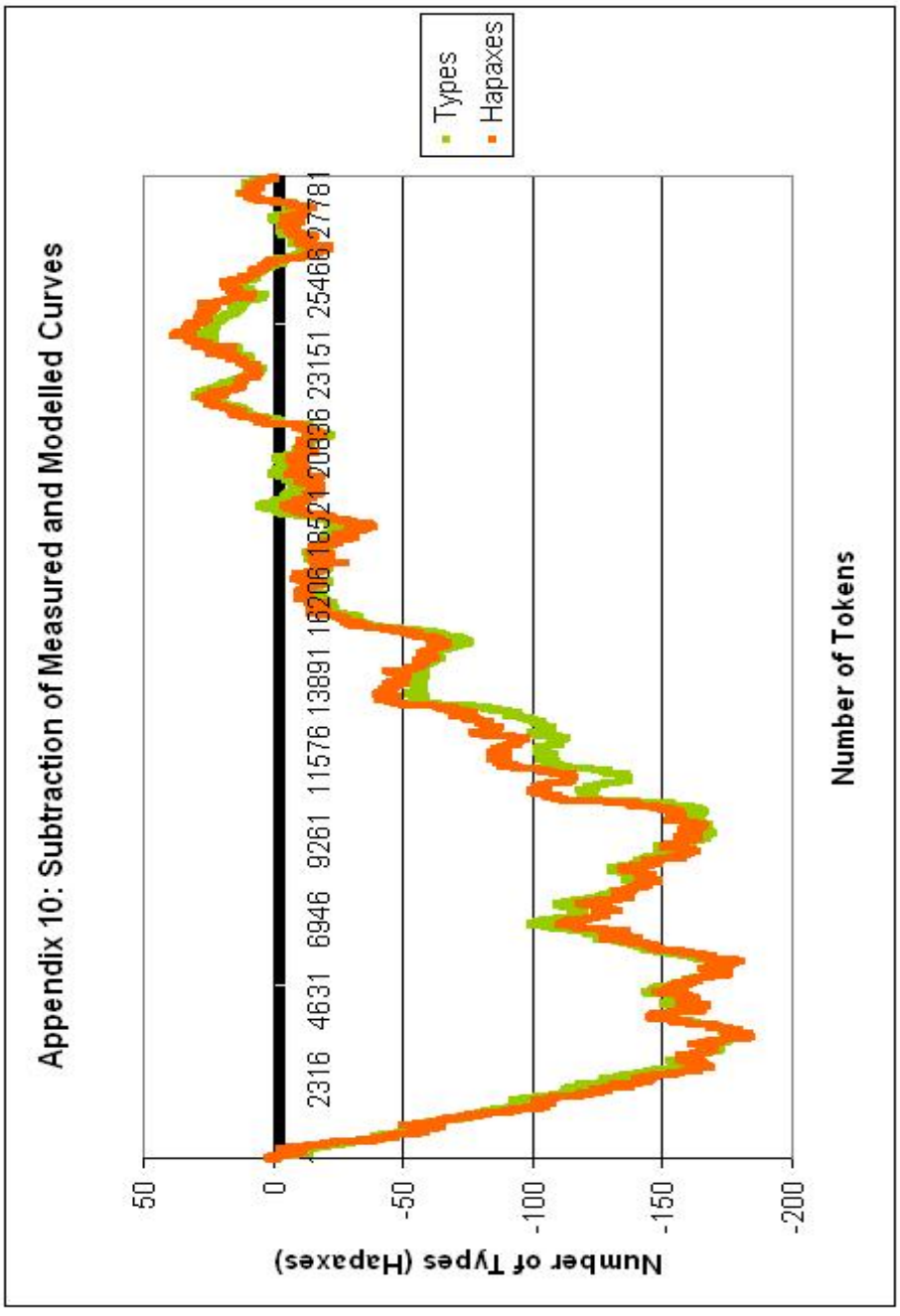

Appendix 5: Arabic Original of the Qur'ān
0
-100
-200
-300
-400
-500
-600
Number of Types
10
1570
3130
4690
6250
7810
9370
10930
12490
14050
15610
17170
18730
20290
21850
23410
24970
26530
28090
29650
31210
32770
Subtraction of Measured Curve and Model
Number of Tokens

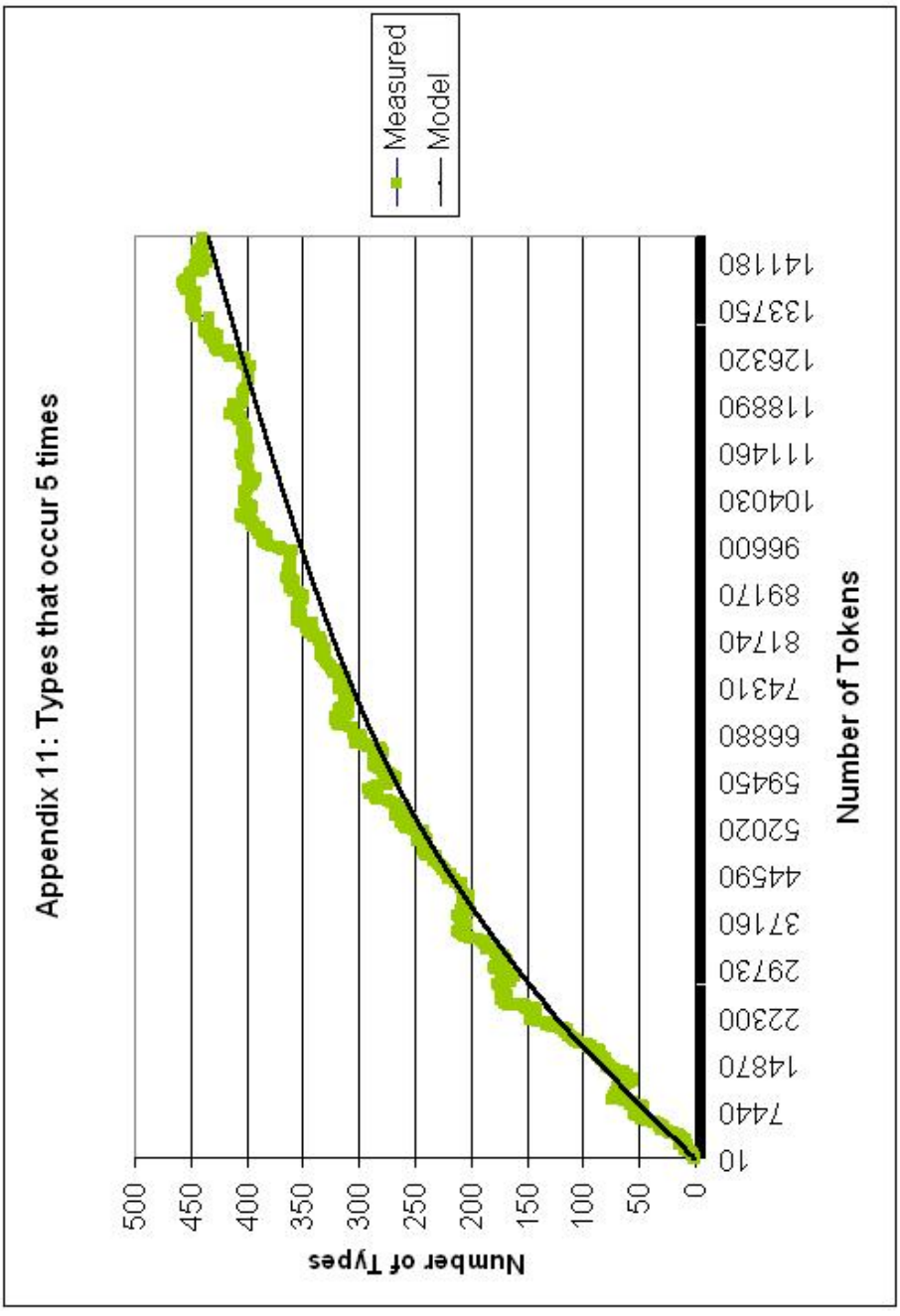

Appendix 6: Czech Tanslation of the Qur'ān
50
0
-50
-100
-150
-200
-250
-300
-350
-400
-450
Number of Types
10
2550
5090
7630
10170
12710
15250
17790
20330
22870
25410
27950
30490
33030
35570
38110
40650
43190
45730
48270
50810
53350
55890
Subtraction of Measured Curve and Model
Number of Tokens


Appendix 7: Hapax-token Relation
Measured
Model
Number of Types
0
500
1000
1500
2000
2500
3000
3500
4000
Number of Tokens
10
7260
14510
21760
29010
36260
43510
50760
58010
65260
72510
79760
87010
94260
101510
108760
116010
123260
130510
137760
145010


Appendix 8: Hapax-token Relation - Randomly Transposed Tokens
Number of Types
2000
1800
1600
1400
1200
1000
800
600
400
200
0
1
1045
2089
3133
4177
5221
6265
7309
8353
9397
10441
11485
12529
13573
Number of Tokens
Original text
Transposed Tokens
Model


Appendix 9: Type-token & Hapax-token Relation
Number of Types (Hapaxes)
0
500
1000
1500
2000
2500
3000
3500
4000
4500
1
2418
4835
7252
9669
12086
14503
16920
19337
21754
24171
26588
Number of Tokens
Types - Measured
Types - Model
Hapaxes - Measured
Hapaxes - Model


Appendix 10: Subtraction of Measured and Modelled Curves
Types
Hapaxes
2316 4631 6946 9261 11576 13891 16206 18521 20836 23151 25466 27781
50
0
-50
-100
-150
-200
Number of Types (Hapaxes)
Number of Tokens


Appendix 11: Types that occur 5 times
Measured
Model
Number of Types
0
50
100
150
200
250
300
350
400
450
500
Number of Tokens
10
7440
14870
22300
29730
37160
44590
52020
59450
66880
74310
81740
89170
96600
104030
111460
118890
126320
133750
141180