Assignment 1: PFL067 Statistical NLP

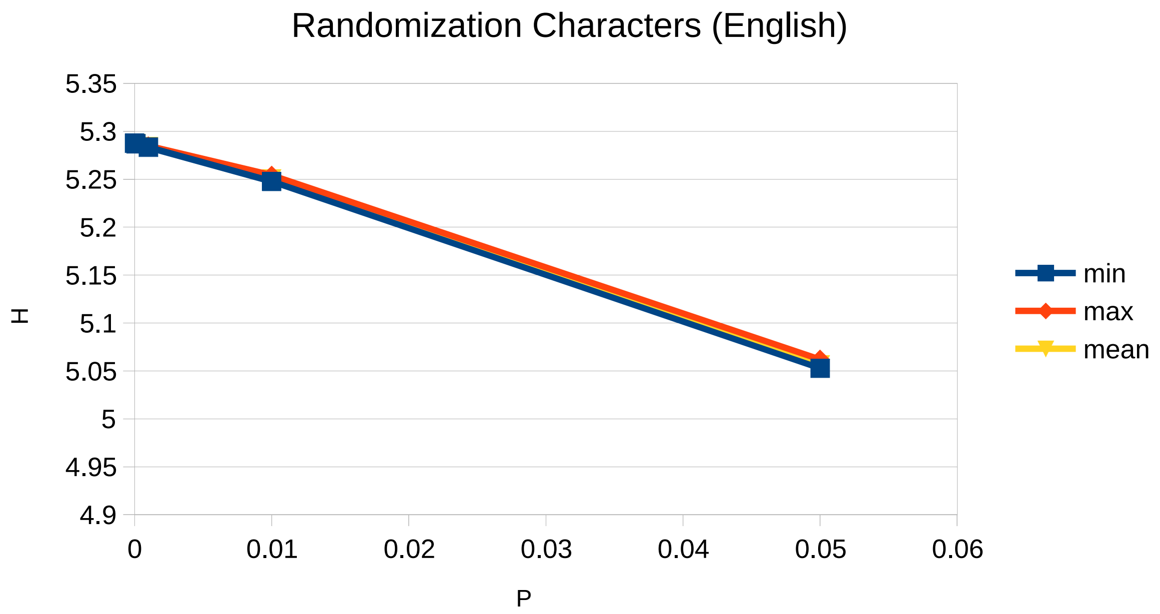
Eric Lief
28 February 2017

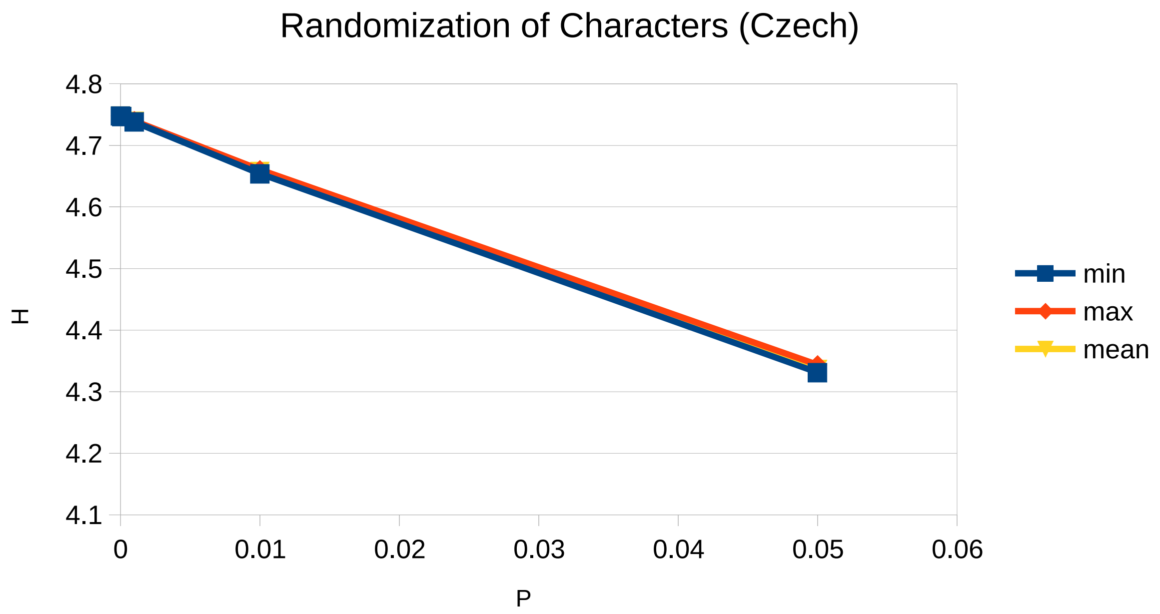
Conclusions

Task 1: Entropy of English and Czech Texts

Randomization of characters

In this task, characters from the alphabets (symbol sets, including punctuation) were randomly mapped to another member of the character inventory of the language, with a likelihood of 0,05, 0,01, 0,001, 0,0001, 0,00001. The resulting plots are given below:





Assignment 1: PFL067 Statistical NLP

Eric Lief
28 February 2017

Randomization of characters (English)

Conditional Entropy (H)			
Likelihood	**min**	**max**	**mean**
0,05	5,05275336	5,0619835009	5,0562534884
0,01	5,247646526	5,2540882527	5,2502720811
0,001	5,283318834	5,2846748259	5,2839441659
0,0001	5,287023237	5,2873870939	5,2871992632
1E-05	5.2872719513	5.28747978719	5.287426283
0	5,287493686	5,2874936863	5,2874936863

Randomization of characters (Czech)

	Conditional Entropy (H)		
Likelihood	**min**	**max**	**mean**
0,05	4,3309110093	4,3434614312	4,3359293952
0,01	4,6537825974	4,6603344594	4,6574242111
0,001	4,7380670489	4,7396662837	4,7388945103
0,0001	4,7466129744	4,7470544267	4,7468909931
1E-05	4.74765680610	4.74780109967	4.747726337
0	4,7478399379	4,7478399379	4,7478399379

Table 1: Randomization of characters

From the graphs above (see data in Table 1), it is clear that randomizing ("messing up") characters lowers entropy of the texts. The lowest entropy corresponds to the messing up with the highest probability (i.e. .05%)

In Table 2 below, we can see some characteristics of the two languages. Although the word count is almost the same in each language, the number of characters per word is slightly higher in Czech (4.63) than in English (4.40), which implies that Czech has longer words. Note that in this project, in accordance with the instructions, punctuation was considered as word. Although the entropy in Czech is lower, the entropy per word is about the same.

	EN	**CZ**
words	221098	222412
chars	972917	1030631
chars/word	4.4003880632	4.6338821646
|V|	9608	42827
H	5.28749	4.7478185909
H/word	2.39146894E-05	0.000021347

Table 2: Text Statistics

English		**Czech**	
char	**freq**	**char**	**freq**
e	0.128673875	o	0.073966337
t	0.086736073	e	0.07126605
a	0.076539931	a	0.060347496
i	0.069745929	n	0.059604262
o	0.069054195	t	0.047659152
n	0.068869184	s	0.042347843
s	0.0659388211	i	0.041570649
r	0.060572485	v	0.0391148723
h	0.04899493	l	0.037617731
l	0.04048444	r	0.036929803
d	0.036021572	d	0.032554814
c	0.033318361	k	0.031980408
f	0.026307486	p	0.027904264
u	0.024598193	u	0.027763574
m	0.023880763	m	0.027251266
p	0.017908002	í	0.026587595
g	0.017349887	c	0.022199992
y	0.01599417	á	0.020573804
b	0.015974641	h	0.020263314
w	0.015301408	z	0.01778037

Table 3: Most frequent letters

We can explain this as follows.

Examination of the most frequent characters in each language, also shows that both languages are quite similar (Table 3) below. However, the total character set for Czech from which random characters were drawn in the experiment was about 1.5 times larger (n = 117) than English (n = 74). This resulted in a slightly more steeped drop in entropy for Czech because more new singleton (haplax) words and bigrams were created.

The randomization of characters in the texts produces many new and unique words, which occur only once in our text (*hapax legomena*). The frequency of these unique words can be seen below.

Hapax Legomena (average)				
	English		Czech	
Likelihood	**N**	**% increase**	**N**	**% increase**
0	3811		26315	
0.1	55023.1	1443.796904	84010.9	319.2509975
0.05	34393.6	902.4822881	60463.1	229.766673
0.01	11641	305.4578851	34367.9	130.6019381
0.001	4721.8	123.899239	27153.5	103.1863956
0.0001	3902.5	102.4009446	26405.2	100.3427703

Table 3: Hapax Legomena/unique words (average)

The occurrence of many unique words will in turn produce many novel bigrams. Suppose that we messup up the first word of a bigram, *i;* now the following word *j* now only occurs in one context (namely after word *i*). The conditional probability of this modified bigram would now have a value of 1, and its logarithm would be zero or close to zero, making the product of joint and conditional probabilities zero as well, regardless of the joint probability of these two words *i* and *j*.

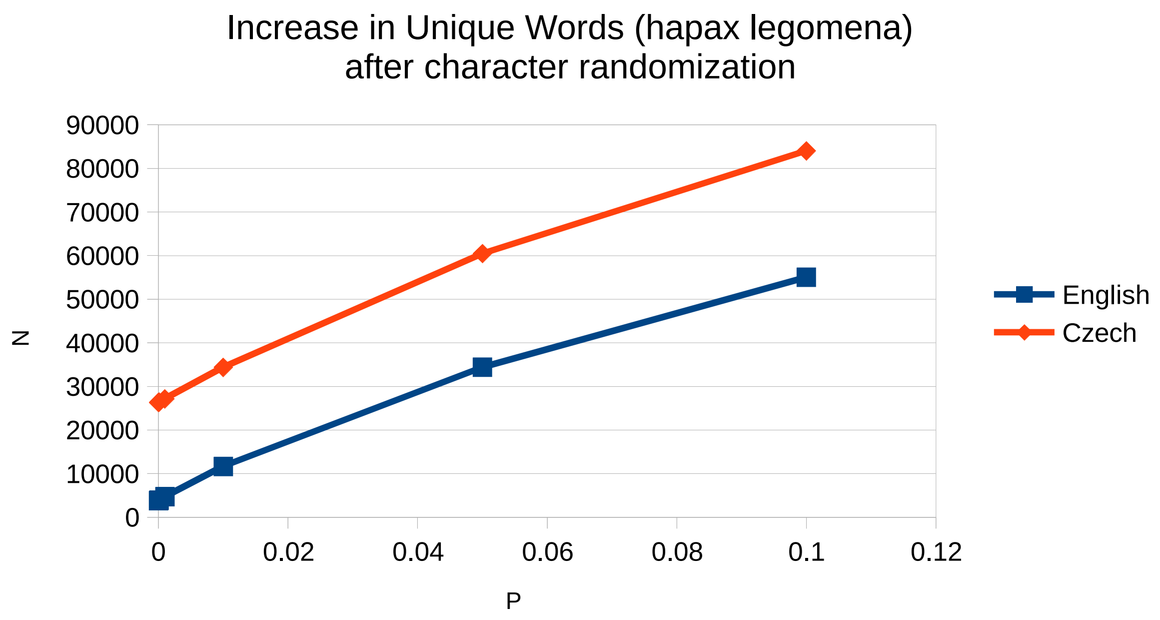
Now consider what happens in the preceding bigram when we mess up the word, now *j*. Assuming we created a new word, we'd now have a conditional probability *1/x*, where *x* is the count of *i*. If this count is 1, then again we have zero, but if it occurs a lot, taking the log will give us a large value. However, this large value will be lowerd substantially when multiplied by the join probability of this new bigram, i.e. 1/(|T| + 1). Thus, in both cases, the product of joint and conditional probabilities will thus be reduced due to this huge increase in bigrams. Summing over the text for conditional entropy (equation below), we find an inverse relationship between the decrease in entropy and the percentage (likelihood) of character randomization.

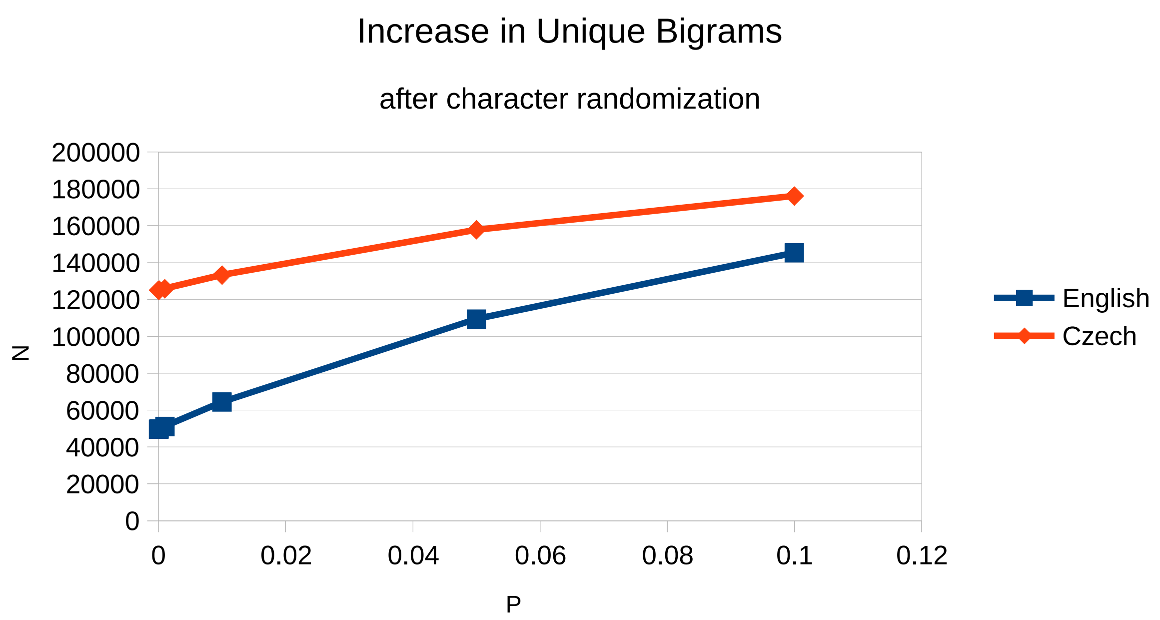
$$
\begin{aligned}
H(j|i) &= -\sum\sum p(i, j) \log_2(p(j|i)) \\
&= -\sum\sum p(\textit{very small}) \log_2(\text{close to } 1) \\
&= -\sum\sum p(\textit{very small})(\textit{close to zero}) \\
&= \textit{low}
\end{aligned}
$$

This observation is borne out when we compare the plots of new words (hapax) and new bigrams after each randomization (mean numbers). The steep climb in unique words and bigrams in English, as will be seen below, is a product of the mainly periphrastic nature of English grammar (the expression of morphological characters mainly through the use of lexical words, as opposed to the highly inflective morphology of Czech, in which, even before randomization, there is a large amount of unique bigrams, i.e. 125,000 Czech bigrams next to about 50,000 for English.

Assignment 1: PFL067 Statistical NLP

Eric Lief
28 February 2017



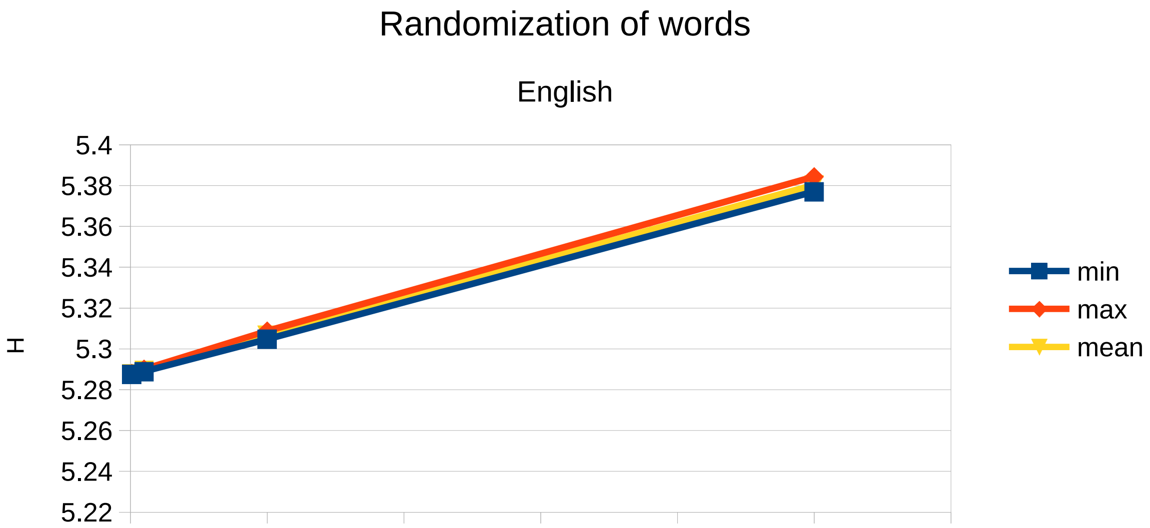


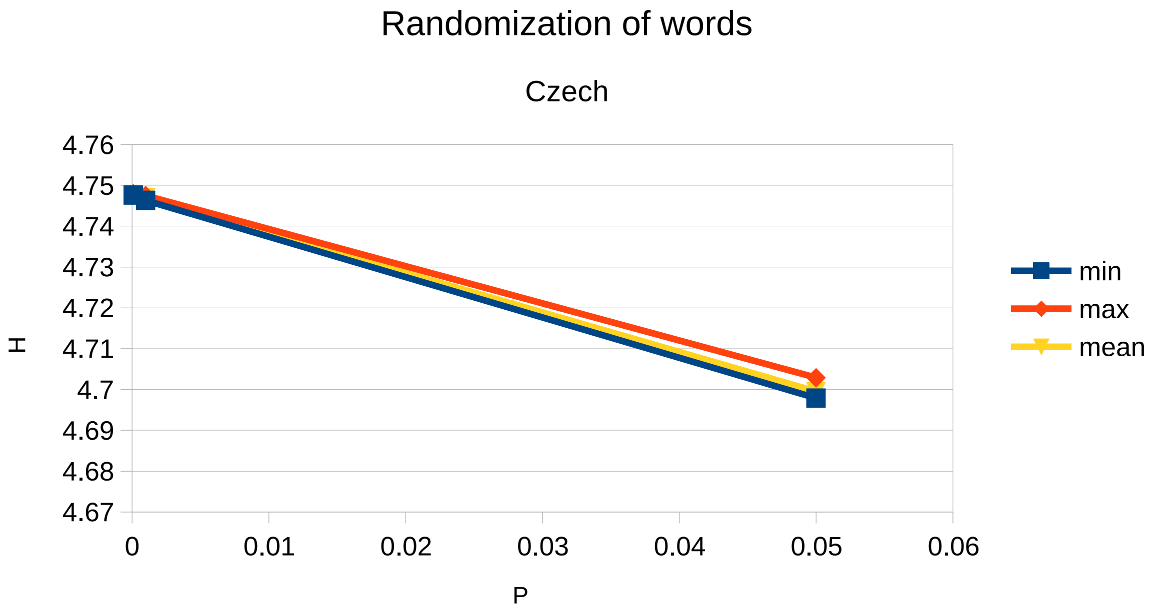
Randomization of words

Above we observed that the randomization (messing up) of characters lowered entropy in both languages, due to the generation of many new bigrams, which in turn lowered conditional entropy to an increasing degree with greater likelihood. The data below show that randomizing words, however, produces different effects in these two languages.

	Conditional Entropy (H)				Conditional Entropy (H)		
Likelihood	**min**	**max**	**mean**	**Likelihood**	**min**	**max**	**mean**
0,05	5,3770036645	5,3843319897	5,3803335074	0,05	4,697876333	4,7028708591	4,6994661857
0,01	5,304697288	5,3086785599	5,3066922972	0,01	4.7367210504	4.7407426847	4.7392793913
0,001	5,2888897915	5,2900354146	5,2894929198	0,001	4,746304713	4,747480085	4,7469491764
0,0001	5,2874916326	5,2878665305	5,2877149179	0,0001	4,7475798984	4,7479300734	4,7477482515
1E-05	5.2874406395	5.2875602640	5.2874909572	1E-05	4.7477650550	4.7478599708	4.7478128422
0	5,2874697717	5,2874697717	5,2874697717	0	4,7478185909	4,7478185909	4,7478185909

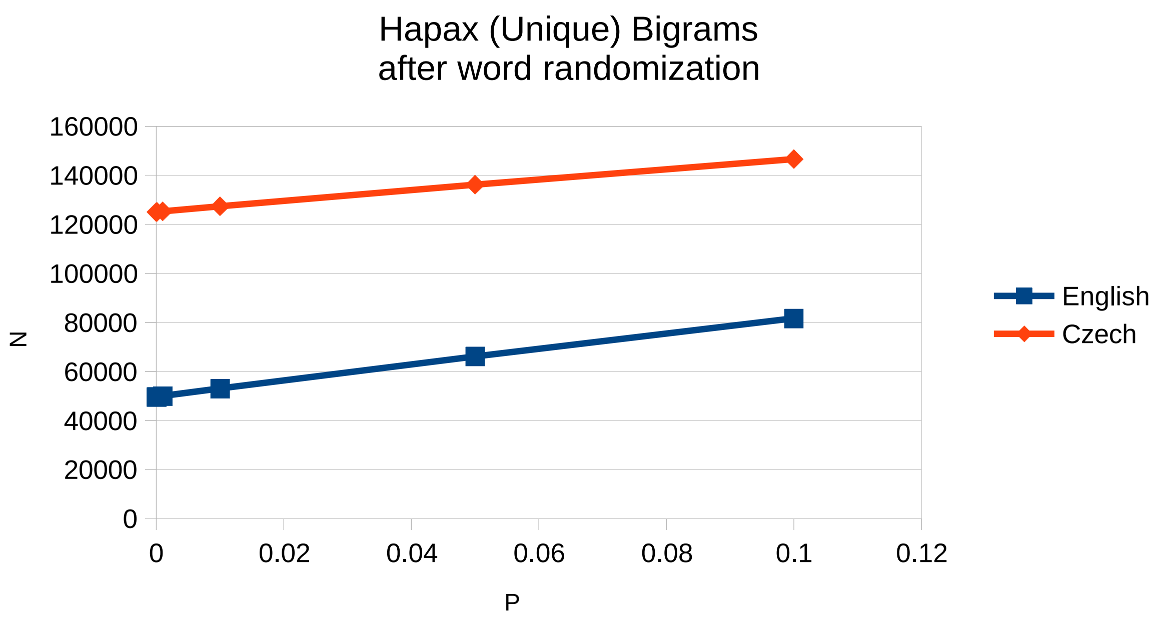
Table 4: Randomization of Words





As can be seen in the graphs and in Table 4 above, entropy increases in English, but decreases in Czech.

Outline of differences between English and Czech



What is the reason for the different behavior seen above with respect to word randomization (messing up)? Recall that the text sizes are almost the same. Examination of the most common words in both languages is also very similar.

English			Czech		
word	**count**	**freq**	**word**	**count**	**freq**
,	14721	0.066581335	,	13788	0.061993058
the	13299	0.0601497978	.	12931	0.058139849
of	9368	0.0423703516	a	4486	0.020169775
.	5645	0.0255316647	v	4043	0.018177976
and	5537	0.0250431935	:	3434	0.015439814
in	4761	0.0215334377	se	3378	0.015188029
to	4548	0.020570064	na	2646	0.0118968401
a	3132	0.0141656641	-	2549	0.0114607126
that	2637	0.0119268379	"	2506	0.0112673777
;	2151	0.0097287176	)	1761	0.007917738
have	2084	0.0094256845	(	1748	0.007859288
be	2072	0.00937141	že	1696	0.007625488
as	2056	0.0092990439	je	1457	0.006550906
is	2032	0.0091904947	o	1431	0.006434005
species	1779	0.0080462058	s	1162	0.005224538
which	1762	0.0079693168	z	1060	0.00476593
by	1703	0.0077024668	do	987	0.00443771
are	1621	0.0073315905	i	985	0.004428718
or	1607	0.0072682702	to	970	0.004361275
for	1346	0.0060877982	1	881	0.0039611172

Table 5: Most Frequent Words

	English		Czech	
Likelihood	**N**	**% increase**	**N**	**% increase**
0,1	81580,3	164,46646372	146601	117,27141829
0,05	66130,8	133,32016209	136178,1	108,9337653
0,01	53024,4	106,89756668	127322,1	101,84953204
0,001	49951	100,70157047	125243,1	100,18646508
0,0001	49635,5	100,06552023	125031,4	100,01711863
1E-05	49606	100,00604802	125013	100,00239981
0	49603		125010	

Table 6: Unique Bigrams (Hapax)

Furthermore, the most frequent words of both languages are also quite similar. So no explanation from these facts alone. Consider the bigram counts.

Likelihood	**English**	**Czech**
|V|	9608	42827
|V|^2	92313664	1834151929
Total Bigrams	73249	147139
Unique Bigrams	49603	125010

Note the marked difference in vocabulary sizes. When we are drawing a random word in the messup experiment, we have a much greater probability of selecting the same word more than once in English. This could lead to a lesser likelihood of producing as many unique bigrams, but more significant would be the effect of inserting the same word in random positions in the text. Clearly this would lead to greater unpredictability (entropy). On the other hand, the larger vocabulary from which we select a word would decrease the likelihood that we create as much surprise, thus decreasing the entropy for the same reasons we saw in character randomization.

Pen and Paper exercise

If we concatenate two texts sharing no vocabulary items, which have the same second-order (bigram) condional entropy *E,* what will be the resulting entropy?

For this exercise, as in the project, I assume a language model which formed bigrams from trigrams, such that the bigram set (distribution) of each text in effect would have a starting symbol <s> and ending symbol </s>. Thus, the number of bigrams for each text is the text size |T| +1:

Bigram size

|T| + 1

Let us assume that we have vocabularies $|V_1|$ and $|V_2|$. Since there are no items in common in the two texts, the conditional probability of a given bigram P(j|i) will not change. The joint probability P(i,j), however, depends on the size of the text (i.e. number of bigrams), which must be scaled because the total number of bigrams in one lower in the concatenated text (see explanation below):

$P'_1(i,j) = |N_1| / (|N_1| + |N_2| - 1)\ P_1(i,j)$
$P'_2(i,j) = |N_2| / (|N_1| + |N_2| - 1)\ P'_2(i,j)$

Summing up the product of the new probabilities P' and the conditional probabilities of both texts will give us the new entropy. We, however, need to account for what happens on the boundary of both texts. Consider two hypothetical small original texts extracted from Dr. Seuss's *Green Eggs and Ham*, which I have translated literally and quite poorly, with no syntactic changes, with care not to introduce any common vocabulary (Note that I chose distinct punctuation in the Spanish translation here also for this purpose). This is given below, where N is the bigram count.

T_1	<s>	I	am	Sam	.	Sam	I	am	.	</s>	**N = 9**
T_2	<s>	Yo	soy	Samuel	,	Samuel	yo	soy	,	</s>	**N = 9**

When concatenated, removing the internal </s>, we get:

<s>	I	am	Sam	.	Sam	I	am	.	**Yo**	soy
Samuel	,	Samuel	yo	soy	,	</s>				**N = 17**

Note that the bigram count is one less, since we have in effect deleted the final bigram of T_1, ('.' , </s>) and the initial bigram of T_2, (<s>, Yo)., both highlighted in red above. We, however, have also added a new bigram containing the penultimate element of T_1 ('.'), which occurs right before the final symbol </s>, and the element which occcurs after the start symbol <s> in T_2, 'Yo'--that is, the bigram above highlighted in yellow ('.', Yo).

However, the value of the product of joint and conditional log probability of the new bigram ('.' , Yo), as it is unique, will be equal to the last one in T_1, ('.' , </s>), which is also unique.

P'('.' , </s>) log P(</s> | '.') = P'('.' , Yo) log P(Yo | '.') = $\frac{1}{17}\log(\frac{1}{2})$

In fact, we can generalize for any two texts, with a pre-</s> element *a* and post-<s> element:

P'(*a* , </s>) log P'(</s> | *a*) = P'(*a, b*) log P'(*b* | *a*) = $\frac{1}{N'}\log(\frac{1}{c(a)})$

where c(a) is the unigram count of the second to last word of T_1, above '.'.

The upshot is that since we will include this in the sum of the partial entropy of T_1, we do not need to add it. We can further note that the initial bigrams of both texts containing the start symbol, i.e. (<s>, I) and (<s>, Yo) above were also unique and had a value of zero. Therefore when we deleted these in the concatenated text, there was no change in entropy.

P(<s>, I) log P(I | <s>) = P(<s>, Yo) log P(Yo | <s>) = 0,

since the logarithm of 1 is zero.

Taking this into account, the new entropy appears not to change. Thus is corroborated below:

$$H_{T1.T2} = -\sum_{i,j \in T1} P'_1(i,j)\log (P'_1(j|i)) - \sum_{i,j \in T2} P'_2(i,j)\log (P'_2(j|i))$$

$$= -\sum_{i,j \in T1} \frac{N1}{N1+N2-1} P_1(i,j)\log (P'_1(j|i)) - \sum_{i,j \in T2} \frac{N2}{N1+N2-1} P_2(i,j)\log (P'_2(j|i)$$

$$= \frac{N1}{N1+N2-1} [-\sum_{i,j \in T1} P_1(i,j)\log (P'_1(j|i))] + \frac{N2}{N1+N2-1} [-\sum_{i,j \in T2} P_2(i,j)\log (P'_2(j|i)]$$

$$= \frac{N1}{N1+N2-1} E + \frac{N2}{N1+N2-1} E$$

$$= \frac{N1+N2}{N1+N2-1} E$$

If we plug the values from the example above, noting that the initial entropy E of both texts is .667, we get the following:

$$H_{T1.T2} = -\sum_{i,j \in T1} P'_1(i,j)\log (P'_1(j|i)) - \sum_{i,j \in T2} P'_2(i,j)\log (P'_2(j|i))$$

$$= \frac{N1+N2}{N1+N2-1} E = \frac{9+9}{9+9-1} (.667) = \frac{18}{17} (.667) = .706$$

Therefore, the entropy is slightly larger, which will always be the case if we handle the symbols the same, since the denominator is smaller. We should note that the example texts above were of identical size and counts, as they were both translations of the same source. However, as the new entropy is in fact weighted (see equation above), this should not matter.

To sum up, when we mark the text which start and end tags, and then remove the internal end symbol, we find that the entropy of concatenating two texts which share no common vocabulary (or punctuation) is slightly larger.

Task 2: Cross-Entropy and Language Modeling

For this task, I created a Smoothed Language Model (SmoothedLM), which calculated the lambda smoothing parameters. As mentioned in the previous discussion, the beginning and end of texts was handled by the insertion of <s> <s> and </s> </s> for trigrams, from which bigrams and unigrams were constructed.

The lambdas and cross-entropy calculated with them are given in the table below for each language.

	English	**Czech**
λ_0	0.098488106	0.2528754
λ_1	0.264127071	0.442825626
λ_2	0.50775492	0.242910993
λ_3	0.129629903	0.0613879811
∑	1	1
H	7.562179947	10.39619816

Table 7: Smoothing Lambdas and Cross-Entropy

The task asks us to demostrate what happens when the lambdas are computed on the training data, rather than the heldout data. Below, the results are given, with λ_3 in both languages converging to 1.

	English	Czech
λ_0	1.53E-24	1.43E-35
λ_1	2.00E-12	1.66E-17
λ_2	2.34E-05	2.41E-05
λ_3	0.999976563	0.9999759118
∑	1	1
H	25.52946623	53.17624655

Table 8: Smoothing with training data

Assignment 1: PFL067 Statistical NLP

Eric Lief
28 February 2017

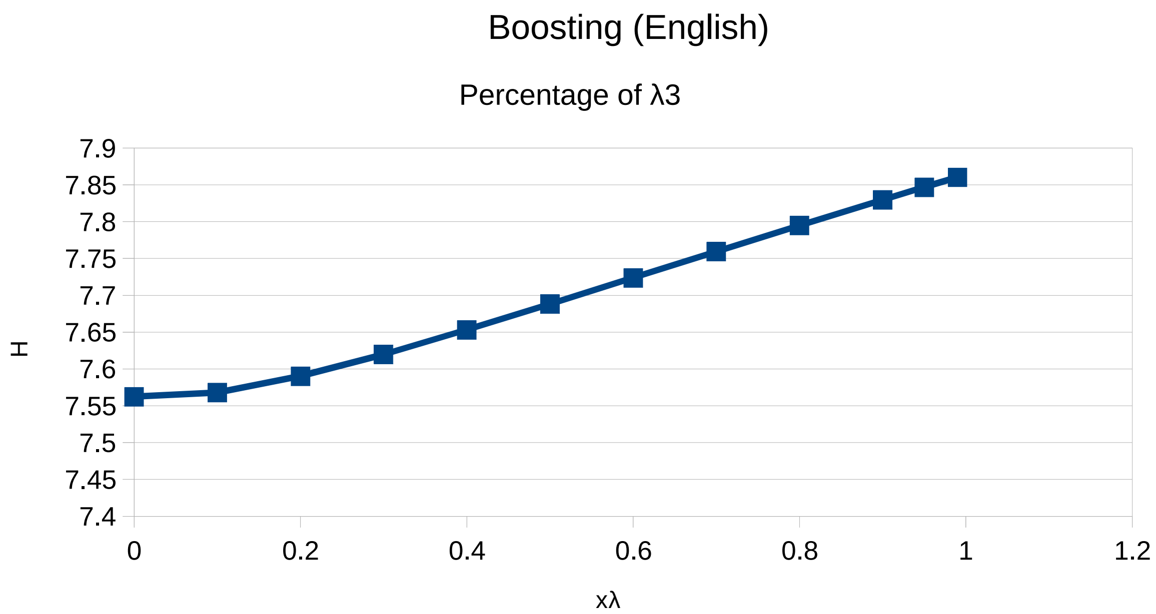
English results

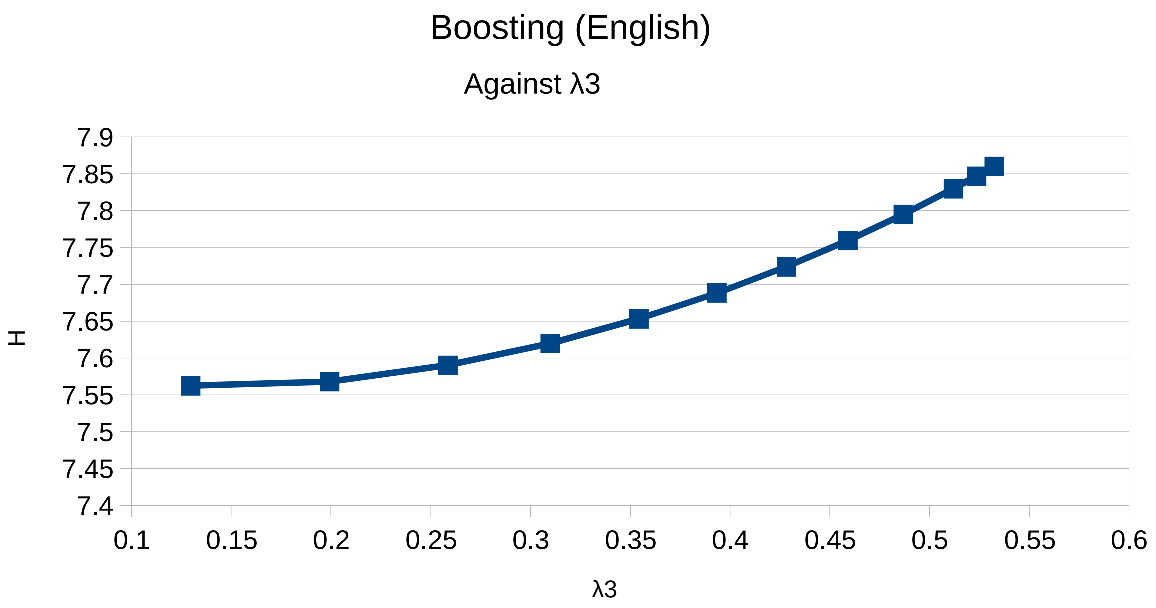
The coverage graph for English is 75.82, cf Czech 65.18 (below).

When we boost the data by adding x% to the value, we find an increase in cross-entropy of the test set from base of 7.562 to 7.860. Below I plot this against both the percentage increase and actual new lambda values.

		λ_3 Boosting											
		0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.95	0.99
EN	λ_0	0.098488106	0.090602349	0.083885773	0.078096301	0.073054371	0.068623977	0.064700223	0.061200902	0.058060682	0.055226983	0.0539113887	0.052903198
	λ_1	0.264127071	0.242978913	0.224966286	0.209439982	0.195918448	0.184036945	0.173514153	0.164129615	0.1557081188	0.148108658	0.144580475	0.141876693
	λ_2	0.50775492	0.467099938	0.432472665	0.402625073	0.376631427	0.353790558	0.333561662	0.315520931	0.299331542	0.284722424	0.277939884	0.272742163
	λ_3	0.129629903	0.1993188	0.258675277	0.309838644	0.354395755	0.39354852	0.428223962	0.459148552	0.486899658	0.5119419342	0.523568252	0.532477946
	sum	1	1	1	1	1	1	1	1	1	1	1	1
	H	7.562179947	7.567938653	7.590073635	7.619703678	7.652994653	7.688005179	7.723664225	7.759351083	7.794696653	7.829480106	7.846617877	7.860195819

Table 9: Boosting in English



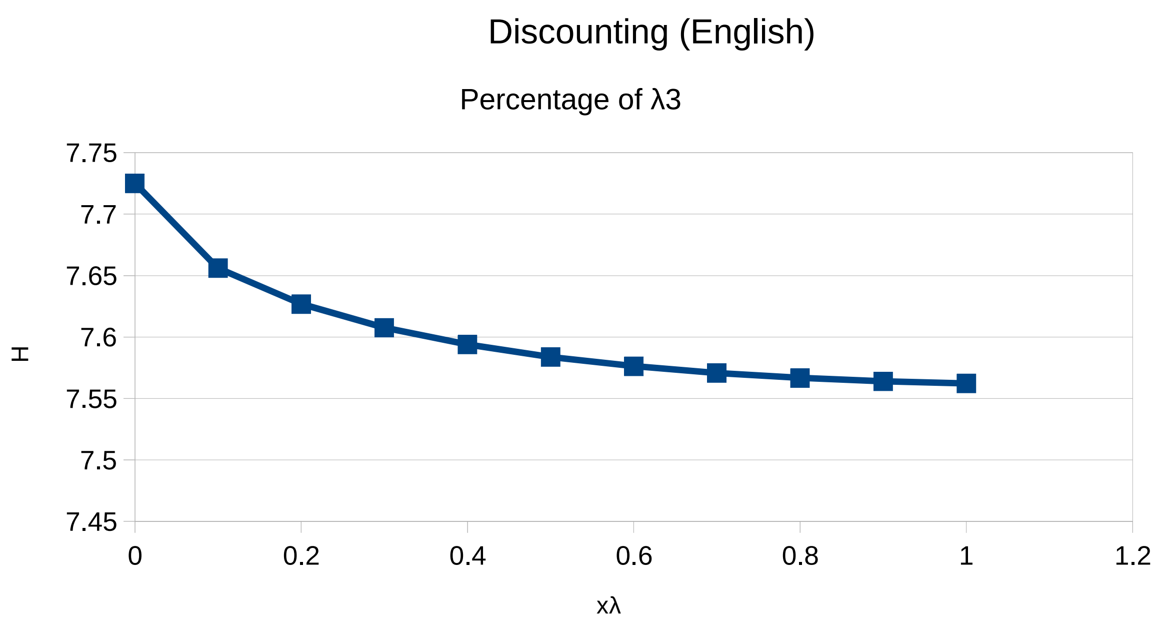


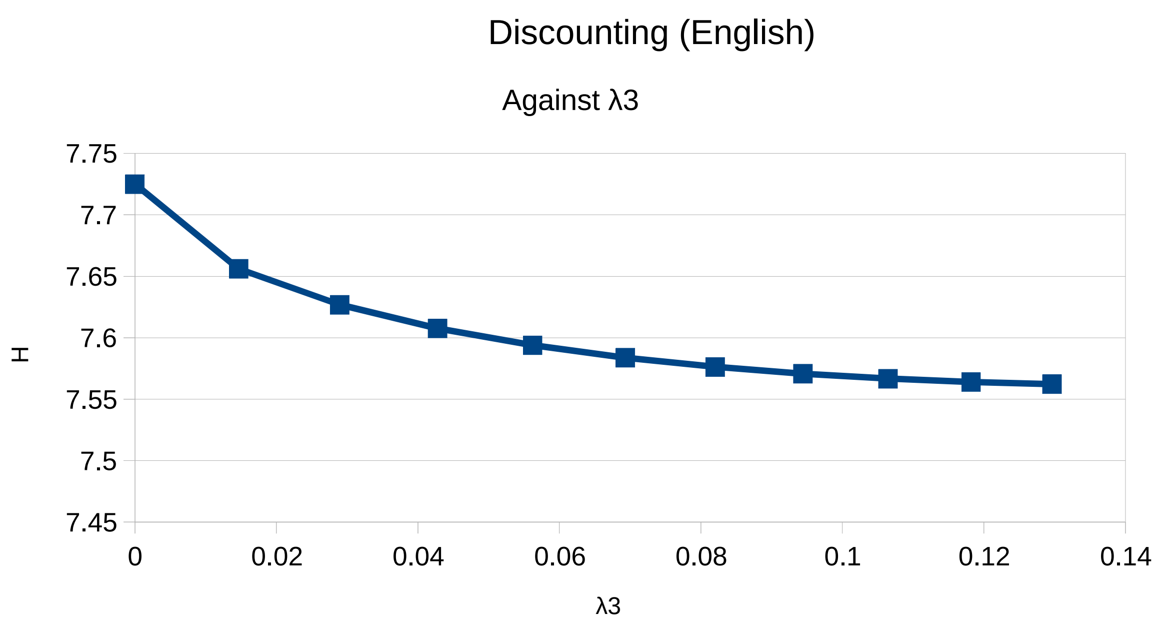
Eric Lief
28 February 2017

When we discount the data by reducing by x%, we also find an increase in cross-entropy of the test set from base of 7.562 to 7.725.

		λ_3 Discounting										
		1	**0.9**	**0.8**	**0.7**	**0.6**	**0.5**	**0.4**	**0.3**	**0.2**	**0.1**	**0**
EN	λ_0	0.098488106	0.099781574	0.1011094685	0.102473183	0.103874187	0.10531403	0.106794351	0.1083168811	0.109883451	0.1114960005	0.1131565832
	λ_1	0.264127071	0.267595914	0.2711570845	0.274814317	0.278571552	0.282432949	0.286402899	0.290486046	0.2946873	0.2990118618	0.303465241
	λ_2	0.50775492	0.514423385	0.521269339	0.528299962	0.535522829	0.542945934	0.550577722	0.5584271168	0.566503561	0.57481705	0.583378176
	λ_3	0.129629903	0.1181991265	0.106464108	0.094412538	0.082031432	0.069307087	0.056225028	0.042769956	0.028925688	0.014675087	0
	sum	1	1	1	1	1	1	1	1	1	1	1
	H	7.562179947	7.563902633	7.566660661	7.570671735	7.576230232	7.583751751	7.593858653	7.607565978	7.626769667	7.656044714	7.7250111994

Table 10: Discounting: English





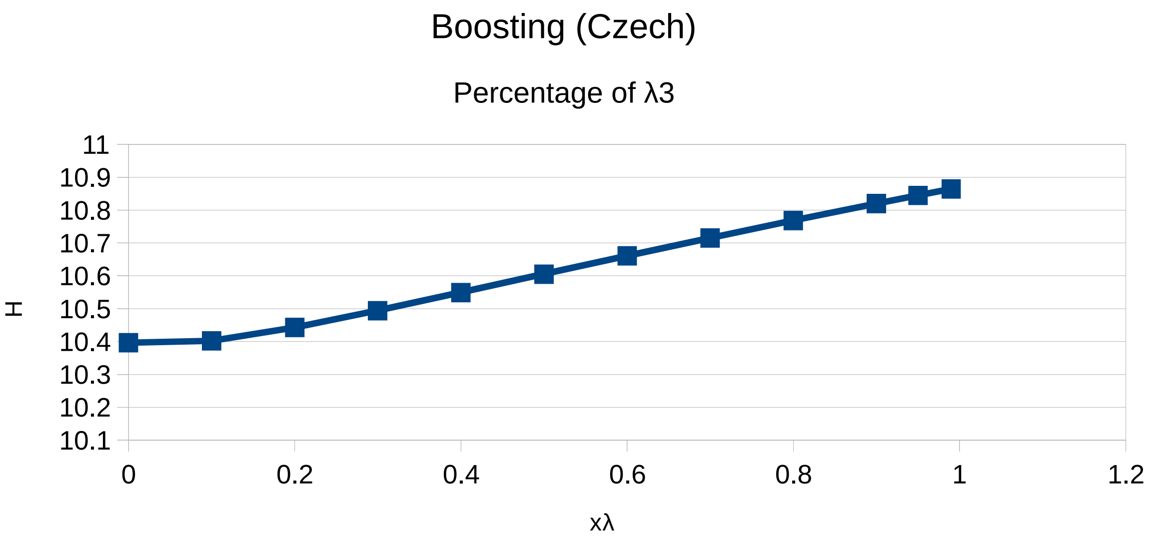
Eric Lief
28 February 2017

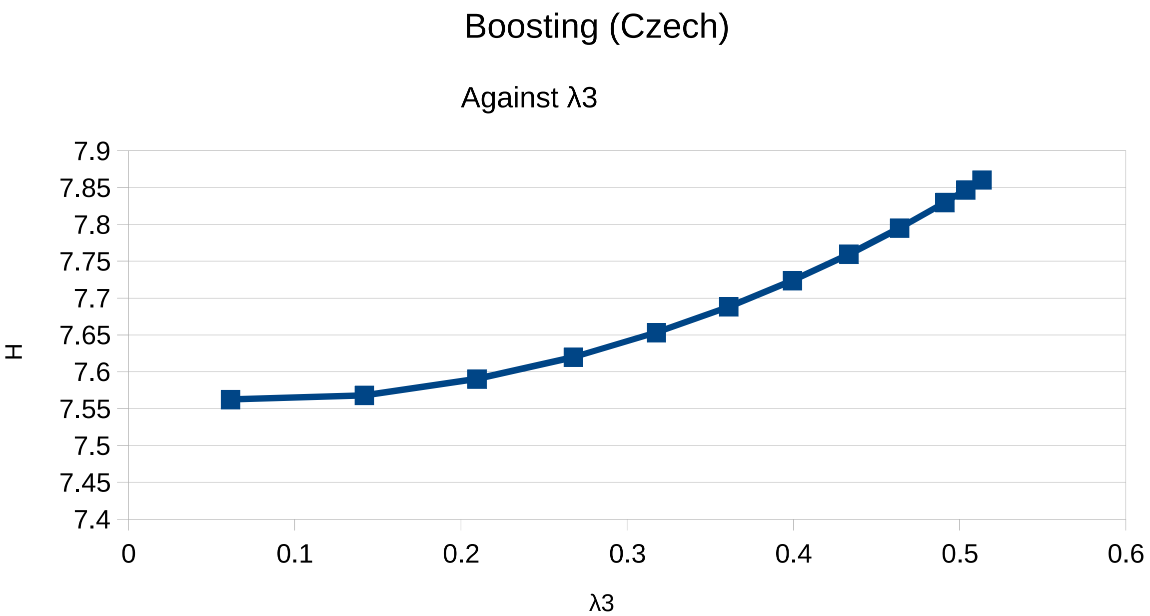
Czech results

The coverage graph for Czech is 65.18, cf. English 75.82. Likewise, we observe increase in cross-entropy for both boosting and discounting.

		λ_3 Boosting											
		0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.95	0.99
CZ	λ_0	0.2528754	0.2311768621	0.21290783	0.19731479	0.183849907	0.172105333	0.1617711774	0.152607763	0.144426809	0.1370783511	0.133677583	0.131076097
	λ_1	0.442825626	0.404827985	0.372835963	0.345530033	0.321950851	0.301384206	0.283287432	0.267240816	0.252914645	0.24004631	0.234091016	0.229535393
	λ_2	0.242910993	0.222067473	0.204518322	0.189539716	0.176605409	0.165323623	0.155396679	0.146594344	0.138735755	0.131676859	0.128410096	0.1259111196
	λ_3	0.0613879811	0.14192768	0.209737885	0.26761546	0.317593833	0.3611868373	0.3995447115	0.433557076	0.463922791	0.4911984804	0.503821304	0.51347739
	sum	1	1	1	1	1	1	1	1	1	1	1	1
	H	10.39619816	10.40175586	10.44238561	10.49382383	10.54900731	10.60505799	10.66063607	10.71508427	10.76808255	10.81948884	10.84457866	10.86435546

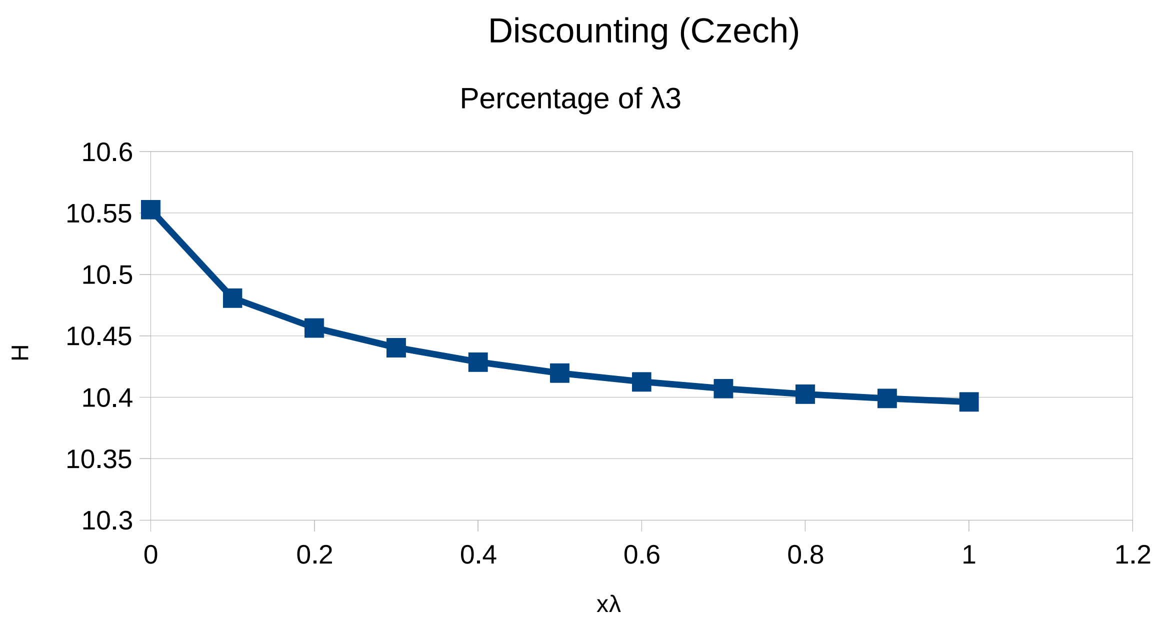
Table 11: Boosting: Czech

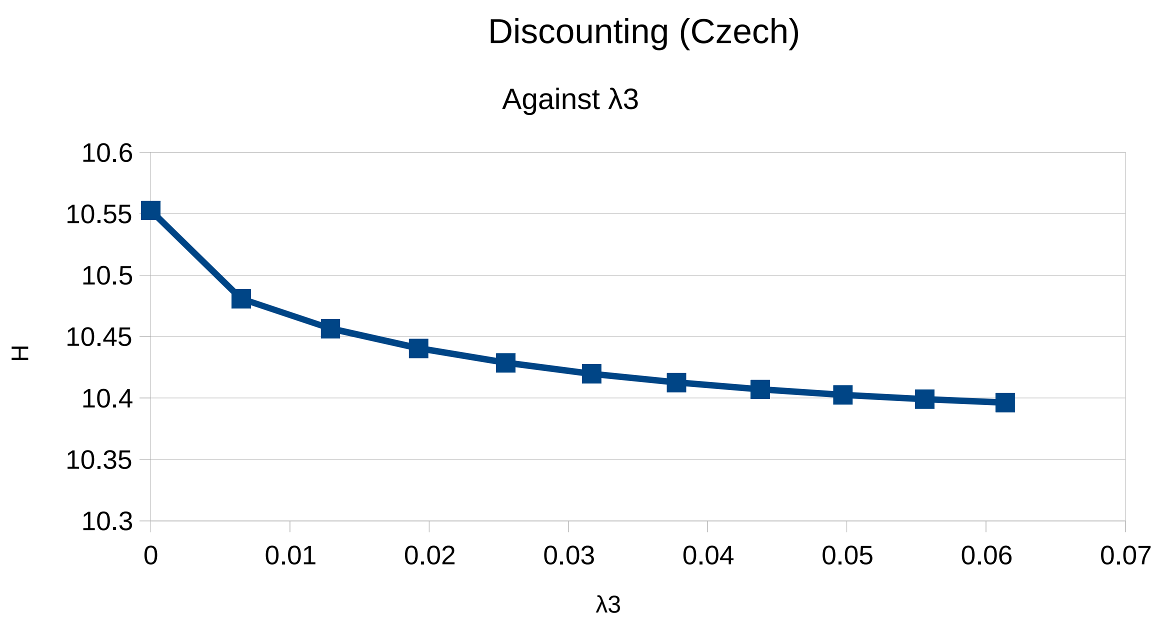




		1	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0
		λ_3 Discounting										
CZ	λ_0	0.2528754	0.25443734	0.256018694	0.257619829	0.2592411157	0.260882939	0.26254569	0.264229773	0.2659356	0.267663595	0.269414194
	λ_1	0.442825626	0.445560834	0.448330041	0.4511338853	0.453973021	0.4568481176	0.459759864	0.462708964	0.465696143	0.4687221411	0.471787721
	λ_2	0.242910993	0.2444113851	0.245930427	0.24746847	0.2490258711	0.250602999	0.252200231	0.253817953	0.255456563	0.2571164677	0.258798085
	λ_3	0.0613879811	0.055590441	0.049720837	0.043777816	0.037759993	0.031665945	0.025494215	0.01924331	0.0129116945	0.006497796	0
	sum	1	1	1	1	1	1	1	1	1	1	1
	H	10.39619816	10.39896566	10.40250042	10.40696086	10.41256423	10.41962272	10.42861442	10.44034265	10.4563725	10.48075742	10.55265134

Table 12: Discounting Czech





Assignment 1: PFL067 Statistical NLP

Eric Lief
28 February 2017

Although both languages behave similarly, recall the difference in coverage graphs (Czech is 65.18, vs. English 75.82). This indicates that those bigrams in the test data which have not been seen in the training data will require smoothing to a greater extent in Czech than in English. Unfortunately, this is not entirely apparent from the plots.