

Article

Humans Outperform Machines at the Bilingual Shannon Game

Marjan Ghazvininejad ^{1,†,‡} and Kevin Knight ^{1,‡}

¹ Information Sciences Institute, University of Southern California; ghazvini@isi.edu

² Information Sciences Institute, University of Southern California; knight@isi.edu

* Correspondence: ghazvini@isi.edu; Tel.: +1-310-822-1511

† Current address: Information Sciences Institute, University of Southern California, 4676 Admiralty Way, Marina del Rey, CA, 90292, USA

‡ These authors contributed equally to this work.

Academic Editor: name

Version September 7, 2016 submitted to Entropy; Typeset by L^AT_EX using class file mdpi.cls

Abstract: We provide an upper bound for the amount of information a human translator adds to an original text, i.e., how many bits of information we need to store a translation, given the original. We do this by creating a Bilingual Shannon Game that elicits character guesses from human subjects, then developing models to estimate the entropy of those guess sequences.

Keywords: Compression, Multilingual, Translation

1. Background

Zoph *et al.* [1] ask the question “How much information does a human translator add to the original?” That is, once a source text has been compressed, how many additional bits are required to encode its human translation? If translation were a deterministic process, the answer would be close to zero. But in reality, we observe an amount of free variation in target texts. We might guess, therefore, that human translators add something like 10% or 20% extra information, as they work.

To get an upper bound on this figure, Zoph *et al.* [1] devise and implement an algorithm to actually compress target English in the presence of source Spanish.¹ The size of their compressed English is 68% of the size of compressed Spanish. This bound seems rather generous. In setting up a common task, Zoph *et al.* [1] encourage researchers to develop improved bilingual compression technology.²

In this paper, we investigate how good such algorithms might get. We do not do this by building better compression algorithms, but by seeing how well human beings can predict the behavior of human translators. Because human beings can predict fairly well, we may imagine that bilingual compression algorithms may one day do as well.

Shannon [2] explores this exactly question for the simpler case of estimating the entropy of free text (not translation). If a human subject were able to write a probability distribution for each subsequent character in a text (given prior context), these distributions could be converted directly into entropy. But it is hard to get these from human subjects. Shannon instead asks a subject to simply guess the next character until she gets it right, and he records how many guesses are needed to correctly identify it. The character sequence thus becomes a *guess sequence*, e.g.:

¹ Zoph *et al.* [1] do not confirm that their bilingual data was indeed translated from Spanish to English, rather than vice-versa, or from a third language. We proceed similarly in this paper, deferring whether one direction is more predictable than another.

² www.isi.edu/natural-language/compression.

	Monolingual	Bilingual
Machine	1.39	0.89
Human	1.25	0.42 (this paper)

Figure 1. Estimates of the entropy of English (in bits per character). Machine results are taken from actual compression algorithms [1], while human results are computed from data elicited by the Shannon Game. The monolingual column is the original case studied by Shannon [2]. The bilingual column represents the number of additional bits needed to store English, given a Spanish source translation.

27 T h e _ b r o k e n _ v
28 2 1 1 1 11 3 2 5 1 1 1 15

29 The subject's identical twin would be able to reconstruct the original text from the guess
30 sequence, so in that sense, it contains the same amount of information.

31 Let c_1, c_2, \dots, c_n represent the character sequence, let g_1, g_2, \dots, g_n represent the guess sequence,
32 and let j range over guess numbers from 1 to 95, the number of printable English characters plus
33 newline. Shannon [2] provides two results.

34 (*Upper Bound*). The entropy of c_1, c_2, \dots, c_n is no greater than the unigram entropy of the guess
35 sequence:

$$36 \quad -\frac{1}{n} \log(\prod_{i=1}^n P(g_i)) = -\frac{1}{n} \sum_{i=1}^n \log(P(g_i)) = -\sum_{j=1}^{95} P(j) \log(P(j))$$

37 This is because this unigram entropy is an upper bound on the entropy of g_1, g_2, \dots, g_n , which
38 equals the entropy of c_1, c_2, \dots, c_n . In human experiments, Shannon obtains an upper bound of 1.3 bits
39 per character (bpc) for English, significantly better than the character n -gram models of his time (e.g.,
40 3.3 bpc for trigram).

41 (*Lower Bound*). The entropy of c_1, c_2, \dots, c_n is no less than:

$$42 \quad \sum_{j=1}^{95} j \cdot [P(j) - P(j+1)] \cdot \log(j)$$

43 with the proof given in his paper. Shannon reported a lower bound of 0.6 bpc.

44 2. Contributions of This Paper

45 Figure 1 gives the context for our work, drawing prior numbers from Zoph *et al.* [1]. By
46 introducing results from a Bilingual Shannon Game, we show that there is significant room for
47 improving bilingual compression algorithms, meaning there is significant unexploited redundancy
48 in translated texts. Our contributions are:

- 49 1. A web-based bilingual Shannon Game tool.
- 50 2. A collection of guess sequences from human subjects, in both monolingual and bilingual
51 conditions.
- 52 3. An analysis of machine guess sequences and their relation to machine compression rates.
- 53 4. An upper bound on the amount of information in human translations. For English given
54 Spanish, we obtain an upper bound of 0.48 bpc, which is tighter than Shannon's method, and
55 significantly better than the current best bilingual compression algorithm (0.89 bpc).

56 *Related Work.* Nevill and Bell [3] describe a parallel-text Shannon Game, but they work in an
57 English-English paraphrasing scenario, with different versions of the Bible. Zoph *et al.* [1] briefly
58 mention a Shannon Game experiment in which human subjects guessed subsequent characters in a
59 human translation. They report a Shannon upper bound for English-given-Spanish guess sequences
60 as 0.51 bpc, but they do not give details, and they do not appear to separate testing sequences from
61 training.

Predict the next character:

Current Source Sentence:

Exageraciones, como la condena a prision del alcalde porque leyo un poema, en lugar de debilitar sus posturas, las refuerzan entre el amplio publico, por supuesto en la medida en que estas posturas son realmente fundamentalistas.

Google Translation of Current Source Sentence:

Exaggerations, as the prison sentence of the mayor because he read a poem, rather than weaken their postures, reinforcing the broad public, of course to the extent that these positions are really fundamentalists.

Guess the Actual Translation of Current Source Sentence:

Excesses such as sen

You've tried:

'a','h',' ','='

You can try:

'e','t','o','t','n','s','y','d','t','c','u','m','w','f','g','y','p','b','v','k','j','x','q','z','E','T','A','O','I','N','S','H','R','D','L','C','U','M','W','F','G','Y','P','B','V','K','J','X','Q','Z',' ',' ','#','\$','%','&','(',')',' ','+',' ','/','0','1','2','3','4','5','6','7','8','9',' ','<','>','?','@','[','\','^','_','{','|','}','~'

Previous Source Sentence:

Se defiende con argumentos.

Previous Target Sentence:

It is defended through reasoning.

Dictionary Part

sent
senior
sentenced
sends
send
sending
sense
sentiment
sensitive
sentence
sentences
sen.
senegal
sensation
sentencing
sentiments
seniors
sensors

Figure 2. Bilingual Shannon Game interface. The human subject reads the Spanish source and guesses the translation, character by character. Additional aids include a static machine translation and a dynamic word completion list.

We note that different text genres studied in the literature yield different results, as some genres are more predictable than others. Different alphabet sizes (e.g., 26 letters versus 95 characters) have a similar effect. Our interest is not to discover an entropy figure that holds across all genres, but rather to study the entropy gap between humans and machines, and between monolingual and bilingual settings.

3. Shannon Game Data Collection

Figure 2 shows our bilingual Shannon Game interface. It displays the current (and previous) source sentence, an automatic Google translation (for assistance only), and the target sentence as guessed so far by the subject. The tool also suggests (for further assistance) word completions in the right panel. Our monolingual Shannon Game is the same, but with source sentences suppressed.

To gather data, we asked 3 individual subjects plus a team of 4 bilingual people to play the bilingual Shannon game. For each subject, we assigned a 3-5 sentence text from the Spanish/English Europarl corpus v7 [4] and asked them to guess the English characters of the text one by one. We gathered a guess sequence with 684 guesses from our team and a guess sequence with 1694 guesses from our individuals (2378 guesses in total). We also asked 3 individuals and a team of 3 people to play the monolingual Shannon game. We gathered a guess sequence with 514 guesses from our team and a guess sequence with 1769 guesses from our individuals (2283 guesses in total).

Figure 3 shows examples of running the monolingual and bilingual Shannon Game on the same sentence.

Monolingual Shannon Game (no source sentence)

```

I t _ i s _ d e f e n d e d _ t h r o u g h _ r e a s o n i n g .
D   w   h i m                               e i f i   a   ,
m   t                                       a   s   -   -
c   o                                       n
                                           c
                                           i
                                           f
                                           m
                                           o
                                           d
                                           p
                                           t

```

Bilingual Shannon Game (source sentence = "Se defiende con argumentos.")

```

I t _ i s _ d e f e n d e d _ t h r o u g h _ r e a s o n i n g .
                                           w   d   .
                                           a

```

Figure 3. Example guess data collected from the Shannon Game, in both monolingual (top) and bilingual (bottom) conditions. The human subject's guesses are shown from bottom up. For example, in the bilingual condition, after seeing '...reason', the subject guessed '.' (wrong), but then correctly guessed 'i' (right).

4. An Estimation Problem

Our overall task is now to estimate the (per-guess) entropy of the guess sequences we collect from human subjects, to bound the entropy of the translator's text. To accomplish this, we build an actual predictor for guess sequences. Shannon [2] and Zoph *et al.* [1] both use a unigram distribution over the guess numbers (in our case 1..95). However, we are free to use more context to obtain a tighter bound.

For example, we may collect 2-gram or 3-gram distributions over our observed guess sequence and use those to estimate entropy. In this case, it becomes important to divide our guess sequences into training and test portions—otherwise, a 10-gram model would be able to memorize large chunks of the guess sequences and deliver an unreasonably low entropy. Shannon [2] applies some ad hoc smoothing to his guess counts before computing unigram entropy, but he does not split his data into test and train to assess the merits of that smoothing.

We set aside 1000 human guesses for testing and use the rest for training—1378 in the bilingual case, and 1283 in the monolingual case. We are now faced with how to do effective modeling with limited training data. But before we turn to that problem, let us first work in a less limited playground, that of *machine guess sequences* rather than human ones. This gives us more data to work with, and furthermore, because we know the machine's actual compression rate, we can measure how tight our upper bound is.

5. Machine Plays the Monolingual Shannon Game

In this section, we force the state-of-the-art text compressor PPMC [5] to play the *monolingual* Shannon Game. This is easy, because at each character token, PPMC builds a context-dependent probability distribution over the 95 possible character types. If we sort this distribution by probability, the machine can guess until it correctly identifies the current character.

We let PPMC warm up on 50m characters, then collect its guesses on the next 100m characters (for training data), plus an additional 1000 characters (our test data). For the text corresponding to this test data, PPMC's actual compression rate is 1.37 bpc.

Table 1. Unigram probabilities of machine and human guesses, in both monolingual and bilingual conditions. Amounts of training data (in characters) are shown in parentheses.

guess #	monolingual		bilingual	
	machine (100m)	human (1283)	machine (100m)	human (1378)
1	0.732	0.744	0.842	0.916
2	0.105	0.086	0.074	0.035
3	0.047	0.047	0.024	0.013
4	0.027	0.030	0.014	0.011
5	0.017	0.020	0.009	0.005
6	0.012	0.012	0.007	0.004
7	0.009	0.008	0.005	0.001
8	0.007	0.007	0.004	0.001
9	0.006	0.005	0.003	0.001
10	0.005	0.004	0.003	0
...
93	7.05E-08	0	0	0
94	7.69E-08	0	0	0
95	1.09E-07	0	0	0

The simplest model of the training guess sequence is a unigram model. Table 1 shows the unigram distribution over 100m characters of training, for both machine and human guess data.³

We also consider two types of context: the g guess numbers preceding the current guess, and the c characters preceding the current guess. For example, if $c = 1$ and the previous character is 'q', then we find the machine able to correctly guess the next character on its first try with probability 0.981, versus 0.732 if we ignore that context. Likewise, having g previous guesses allows us to model "streaks" on the part of the Shannon Game player.

As g and c grow, it becomes necessary to smooth, as test guess sequences begin to contain novel contexts. PPMC itself makes character predictions using $c=8$ and $g=0$, and it smooths with Witten-Bell, backing off to shorter n -gram contexts $c = 1..7$. We also use Witten-Bell, but with a more complex backoff scheme to accommodate the two context streams g and c . If $g \geq c$ we back off to the model with $g - 1$ previous guesses and c previous characters, and if $g < c$ we back off to the model with g previous guesses and $c - 1$ previous characters.

Table 2 shows test-set entropies obtained from differing amounts of training data, and differing amounts of context. We draw several conclusions from this data:

- Character context (c) is generally more valuable than guess context (g).
- With large amounts of training data, modest context ($g=1$, $c=2$) allows us to develop a fairly tight upper bound (1.44 bpc) on PPMC's actual compression rate (1.37 bpc).
- With small amounts of training data, Witten-Bell does not make effective use of context. In fact, adding more context can result in worse test-set entropy!

The last column of Table 2 shows entropies for necessarily-limited human guess data, computed with the same methods used for machine guess data. We see that human guessing is only a bit more predictable than PPMC's. Indeed, PPMC's guesses are fairly good—its massive 8-gram database is a powerful counter to human knowledge of grammar and meaning.

³ These numbers combine data collected from individuals and from teams (Section 3). In the bilingual case, teams outperformed individuals, guessing correctly on the first try 94.3% of the time, versus 90.5% for individuals. In the monolingual case, individuals and teams performed equally well.

Table 2. Entropies of monolingual test guess-sequences (1000 guesses), given varying amounts of context (c = number of previous characters, g = number of previous guess numbers) and different training set size (shown in parentheses). Witten-Bell smoothing is used for backoff to shorter contexts.

c	g	Machine Guessing				Human (1283)
		(100m)	(10m)	(1m)	(1k)	
0	0	1.72	1.72	1.72	1.76	1.68
0	1	1.70	1.70	1.71	1.84	1.75
0	2	1.69	1.69	1.71	2.03	1.92
1	0	1.54	1.54	1.70	1.86	1.74
1	1	1.52	1.52	1.54	2.20	2.18
1	2	1.50	1.52	1.55	2.37	2.32
2	0	1.45	1.51	1.58	2.25	2.20
2	1	1.44	1.53	1.54	2.56	2.58
2	2	1.48	1.56	1.59	2.73	2.70
8	0	1.37				

6. Modeling Human Guess Sequences

How can we make better use of limited training data? Clearly, we do not observe enough instances of a particular context to robustly estimate the probabilities of the 95 possible guess numbers that may follow. Rather than estimating the multinomial directly, we instead opt for a parametric distribution. Our first choice is the *geometric* distribution, with one free parameter p , the chance of a successful guess at any point. For each context in the training data, we fit p to best explain the observations of which guesses follow. This one parameter can be estimated more robustly than the 94 free parameters of a multinomial.

Figure 4 shows that the geometric distribution is a decent fit for our observed guess data, but it does not model the head of the distribution well—the probability of a correct guess on the first try is consistently greater than p .

Therefore, we introduce a smoothing method (“Frequency and Geometric Smoothing”) that only applies geometric modeling to guess numbers greater than i , where data is sparse. For each context, we choose i such that we have seen all guess numbers $1..i$ at least k times each, where

$$k = \min\left(\frac{\text{Number of samples seen in context}}{20}, 4\right)$$

Table 3 (left half) demonstrates the effect of different smoothing methods on estimated entropies for human guess data. The monolingual Witten-Bell smoothing column in this figure is the same of last column of Table 2.

The right half of Table 3 shows the bilingual case. For the machine case, we use the algorithm of Zoph *et al.* [1]. Note that the machine and human subjects both make use of source-sentence context when predicting. However, we do not use source context when modeling guess sequences, only target context.

7. Summary Results

For calculating final entropy bounds, we first divide our guess sequence into 1000 for training data, 100 for development, and remainder for test (1183 for the monolingual case and 1278 for the bilingual case). We use the development set to find the best context model and smoothing model. In all experiments, using $c = 1$ previous characters, $g = 1$ previous guesses, and Frequency and Geometric Smoothing works best.

Table 4 summarizes our results. As shown in the figure, we also computed Shannon lower bounds (see Section 1) on all our guess sequences.

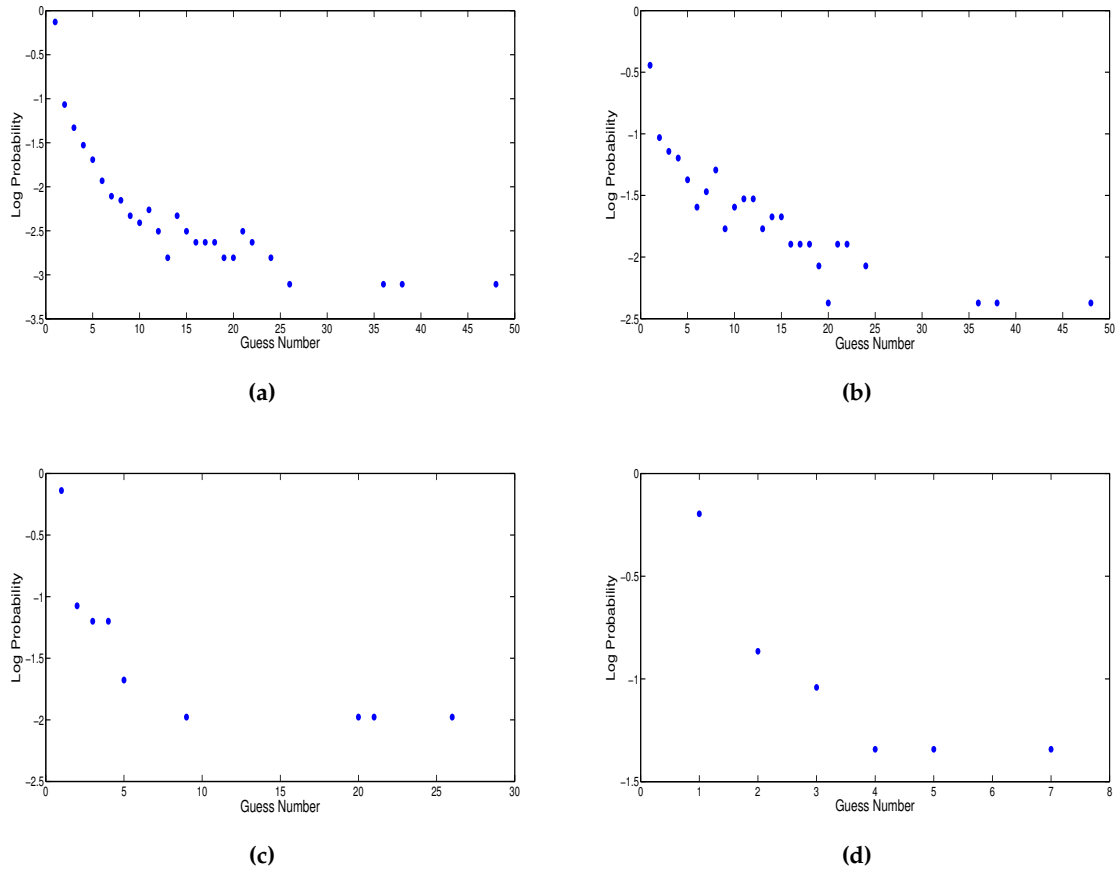


Figure 4. Guess number distributions from human monolingual Shannon Game experiments (training portion). Plot 4a shows all 1238 guesses, while plots 4b, 4c and 4d show guesses made in specific character contexts ' ' (space), 'a' and 'p'. The y-axis (probability of guess number) is given in log scale, so a geometric distribution is represented by a straight line. We observe that the single-parameter geometric distribution is a good fit for either the head or the tail of the curve, but not both.

Table 3. Entropies of human guess-sequences (1000 test-set guesses), given varying amounts of context (c = number of previous characters, g = number of previous guess numbers) and different smoothing methods. Prediction models are trained on a separate sequence of 1283 guesses in the monolingual case, and 1378 guesses in the bilingual case.

c	g	Monolingual Human Guessing			Bilingual Human Guessing		
		Witten-Bell Smoothing	Geometric Smoothing	Frequency and Geo. Smoothing	Witten-Bell Smoothing	Geometric Smoothing	Frequency and Geo. Smoothing
0	0	1.68	2.02	1.62	0.54	0.73	0.67
0	1	1.75	2.06	1.62	0.56	0.72	0.66
0	2	1.92	2.06	1.65	0.61	0.72	0.67
1	0	1.74	1.57	1.50	0.65	0.57	0.48
1	1	2.18	1.55	1.48	0.84	0.56	0.48
1	2	2.32	1.52	1.49	0.93	0.56	0.49
2	0	2.20	1.65	1.60	0.94	0.63	0.63
2	1	2.58	1.57	1.57	1.10	0.63	0.62
2	2	2.70	1.59	1.58	1.18	0.63	0.63

Table 4. Summary of our entropy bounds.

	Guesser	Shannon Upper Bound	Our Improved Upper Bound	Compression Rate	Shannon Lower Bound
Monolingual	Machine	1.76	1.63	1.39	0.63
	Human	1.65	1.47	~ 1.25	0.57
Bilingual	Machine	1.28	1.01	0.89	0.46
	Human	0.54	0.48	~ 0.42	0.21

For the bilingual case of English-given-Spanish, we give a 0.48 bpc upper bound and a 0.21 bpc lower bound. In the case of machine predictors, we find that our upper bound is loose by about 13%, making it reasonable to guess that true translation entropy might be near 0.42 bpc.

8. Conclusion and Future Work

We have presented new bounds for the amount of information contained in a translation, relative to the original text. We conclude:

- Bilingual compression algorithms have plenty of room to improve. There is substantial distance between the 0.95 bpc obtained by [1] and our upper bound of 0.48 and lower bound of 0.21.
- Zoph *et al.* [1] estimate that a translator adds 68% more information on top an original text. This is because their English-given-Spanish bilingual compressor produces a text that is 68% as big as that produced by a monolingual Spanish compressor. Using monolingual and bilingual Shannon Game results, we obtain a revised estimate of $0.42 / 1.25 = 34\%$.⁴

Meanwhile, it should be possible to reduce our 0.48 upper bound by better modeling of guess sequence data, and by use of source-language context. We also conjecture that the bilingual Shannon Game can be used for machine translation evaluation, on the theory that good human translators exhibit more predictable behavior than bad machine translators.

Bibliography

⁴ Here, the denominator is monolingual English entropy, rather than Spanish, but we assume these are close under human-level compression.

1. Zoph, B.; Ghazvininejad, M.; Knight, K. How Much Information Does a Human Translator Add to the Original? Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2015.
2. Shannon, C.E. Prediction and Entropy of Printed English. *Bell System Technical Journal* **1951**, *30*, 50–64.
3. Nevill, C.; Bell, T. Compression of parallel texts. *Information Processing & Management* **1992**, *28*, 781–793.
4. Koehn, P. Europarl: A parallel corpus for statistical machine translation. Proceedings of the Machine Translation Summit X, 2005.
5. Cleary, J.G.; Witten, I. Data compression using adaptive coding and partial string matching. *IEEE Transactions on Communications* **1984**, *32*, 396–402.

© 2016 by the authors. Submitted to *Entropy* for possible open access publication under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>)