CoLAG *N*-Gram Models

Tawa Suleman
LING 72800

23 December 2021

For this project I created n-gram models (specifically, bigram and trigram models) for four languages in the CoLAG Language Domain. In this paper I will first discuss my approaches in creating the language models and evaluating performance. Then, I will discuss the performances of these models, discussing common grammatical errors in each language model, and comparing the performances of *n*-gram models across the languages. Scripts used for data preparation, sentence generation, and sentence evaluation, as well as text files of generated sentences can be found at https://github.com/tsuleman/colag-ngram.

The Languages

Bigram and trigram models were created for four languages in the CoLAG Language Domain. The primary focus in the variation of these languages was the first two CoLAG parameters: Subject Position and Headedness in IP, NegP, VP, PP. The four languages chosen are below:

L1	0001001100011
L2	0111100010000
L3	1010010111001
L4	1110011111001

These models were created using *n*-grams of declarative sentences in the given languages. The syntactic structure of these sentences was not taken into account. So, while a language may have a certain number of sentences, it may have fewer 'unique' sentences based on surface structure. The total number of declarative sentences and the number 'unique' surface structures are provided:

	TOTAL	UNIQUE
L1	360	180
L2	252	252
L3	504	408
L4	504	408

Table 1. Total and Unique sentences in CoLAG languages

The Models

All models are of the OpenGrm-NGram library, using cyclic finite state transducers (FSTs) and Knesser-ney smoothing. The sentences were generated using `ngramrandgen`, with any occurrences of `<epsilon>` removed. More information may be found on the OpenGrm-NGram Library Quick Tour page at https://www.openfst.org/twiki/bin/view/GRM/NGramQuickTour.

For each language, two bigrams and trigrams were created. The first pair of bigram and trigram models did not include beginning- and end-of-sentence markers—`<bos>` and `<eos>`, respectively—, while the second pair included these tokens.

Evaluation and Data

When evaluating the generated sentences, only unique sentences were considered (i.e. if one grammatical sentence was produced twice it would only be counted once).

Initially, the sentences were to be generated in 4 batches—100 sentences, 200 sentences, 1000 sentences and 2000 sentences. However, the largest number of unique sentences possible for any of the languages is 408. This may play a large factor into why the percentage of grammatical sentences produced decreased as the number of sentences generated was increased (but we may also want to question why, as we increase the number of sentences produced, more ungrammatical sentences are produced, even when not all possible grammatical sentences have yet been produced). As I noticed this, I focused on the 100-sentence batches after L2. Also, after L1 I ran 5 trials of each batch and have displayed their averages, rather than recording a single batch data as in L1.

Results

	100 sentences	200 sentences	1,000 sentences	2,000 sentences
L1, with sentence boundary markers				
bigram	16/61 ≈ 26.23%	29/127 ≈ 22.83%	79/497 ≈ 15.90%	104/875 ≈ 11.89%
trigram	35/91 ≈ 38.4%	66/166 ≈ 39.76%	146/466 ≈ 32.74%	165/626 ≈ 26.36%
L1, without sentence boundary markers				
bigram	34/66 ≈ 51.52%	45/130 ≈ 34.62%	69/356 ≈ 19.38%	84/654 ≈ 12.84%
trigram	45/78 ≈ 57.69%	71/129 ≈ 55.04%	112/326 ≈ 34.36%	145/442 ≈ 32.81%

Table 2. L1 results

In both the pair of models with sentence boundary markers and those without, the trigram models outperformed their bigram counterparts.

The most noticeable construction that the L1 bigram models had lower success rate in generating sentences with preposition stranding. While O3 may be fronted, the preposition would sometimes still be followed by another O3, as in sentence (1). However, the bigram including sentence boundary markers was able to successfully generate sentences with preposition stranding, such as sentences (2).

(1) *O3 S Aux Verb O1 P O3
(2) O3 S Never Verb O1 P Adv

	100 sentences	200 sentences
L2, with sentence boundary markers		
bigram	55.27%	45.44%
trigram	74.45%	72.86%
L2, without sentence boundary markers		
bigram	58.34%	54.60%
trigram	74.11%	72.62%

Table 3. L2 results

A common error by L2 bigram models was including a lexical item both with the +WA marker as well as without it, as in sentences such as (3). Including sentence boundary markers did not appear to make a significant difference in either bigram in regards to this. The error also occurred in the trigram generated sentences as well, but less frequently. All models generated sentences with multiple +WA markers (see (4), generated by a trigram model including sentence boundary markers).

(3) *O2[+WA] Adv O3 P O2 Verb Aux
(4) *O3 P[+WA] P[+WA] O1[+WA] S[+WA] O3 P O1 Verb Aux

	100 sentences
L3, with sentence boundary markers	
bigram	33.68%
trigram	59.35%
L3, without sentence boundary markers	
bigram	32.75%
trigram	61.16%

Table 4. L3 results

	100 sentences
L4, with sentence boundary markers	
bigram	37.34%
trigram	65.04%
L4, without sentence boundary markers	
bigram	38.84%
trigram	68.60%

Table 5. L4 results

A challenge L3 and L4 models was that the languages were both Subject Final (with L3 also being Object Final) and also included the +WA topic marking which always occurred in the beginning of the sentence. So, while the models could successfully topicalize subjects, more often than not, they would also occur with finalized subjects, such as sentence (5), generated by a L3 trigram model with boundary markers.

(5) *S[+WA] Verb Not P O3 S

Given the small order of these *n*-grams, this does not come as a surprise. The beginning and end of the sentences are too far apart for the models to be able to take into account that topicalized subjects do not occur with finalized subjects. Many otherwise grammatical sentences rendered ungrammatical, this may be one of the most common grammar errors that reduced the accuracy of the L3 and L4 models.

"Cross-linguistic" Analysis

It appears some languages can be more accurately modeled by *n*-gram models than others. Are there specific parameters that may be playing a part in this? According to the results, L2 models were generally more able to produce grammatical sentences. The parameters unique to L2 in respect to this set of languages are Optional Null Subject and Obligatory Question Inversion. Is it possible that these parameters can influence the language enough to make it more easily represented through *n*-gram models?

Considerations, Concerns, & Comments

While I did not expect *n*-gram models to be the best language models, I was still surprised by the love percentages of grammatical sentences produced.

Another contrast to my expectations was the models including sentence boundary markers not performing better overall. While the markers helped increase accuracy in certain constructions such as preposition stranding, it almost appears that they decreased the accuracy of the models overall. This may be because the models (especially bigrams) would sometimes generate sentences with multiple beginning-of-sentence or end-of-sentence markers.

It would be interesting to see results using different settings for the *n*-gram models and looking more into how well the models perform with respect to each of the CoLAG parameters.