Language Model Exploration: N-Grams across Genres and Time

Melinda Tang
University of Chicago
melindatang@uchicago.edu

Vivian Wu
University of Chicago
vivianwu@uchicago.edu

Cooper Powell
University of Chicago
cooperpowell@uchicago.edu

Luc Saintfort
University of Chicago
lsaintfort@uchicago.edu

Abstract

This paper aims to examine just how well a language model generalizes across corpora of various genres and time periods, and how the choice of training set affects the generalizability of the model. Specifically, bi-gram and tri-gram models were first trained on corpora from one of five different genres: news, nonfiction, texts, tweets, and fiction, and tested on each of those genres. Additional bi-gram and tri-gram models were then trained on corpora from one of three different time periods for two different genres: novels and poems, and tested on each of those time periods.

In the end, the results demonstrated that the model does generalize well across corpora of various genres of time periods, to the degree that the test corpus derived from our baseline training corpus actually had the highest perplexity among all tested corpora, but it does much better when trained on a corpus with a more general, simple range of vocabulary. This indicates that, whenever possible, we should choose the most representative corpus to train our model, but the model will still do fairly well if trained on other corpora.

1 Introduction

Language models have become heavily prevalent in what seems like just a few years. From speech recognition used by various virtual assistants to text auto complete and correction, what once lived exclusively in the labs of researchers has now been imbued into our daily lives.

However, despite the vast advancements the field has made, from Neural Network Language Models to BERT, the issue of limited resources has continually restricted the efficacy of language models. A good model requires training on a corpus representative of its intended task and extensive training to become well honed for that task. Unfortunately, this consumes a lot of resources, limiting the amount of language models that can actually be created. Thus, it is very important to ensure that a language model is highly generalizable for its given task.

N-gram models are simple and easy to use. When building statistical language models, we can significantly reduce the difficulty of mapping associations between numerous discrete variables, or unique words, by taking advantage of word order derived from n-grams. Using the probabilities of occurrences of specific words in sequence can improve the predictions of subsequent words for NLP tasks. Additionally, N-gram models make up the basis for much of the advancement in the field of language modeling, so we could make the assumption that any task-specific generalization issues found N-grams would be magnified in more complex models. Thus, it makes sense to use N-gram models for our study.

Our general aim was to examine our language model's ability to generalize over various genres and time periods, to assess a language model's longevity in accordance to its trained task. To do this, we used perplexity as a measurement to evaluate the performance of our language model. Perplexity in NLP is a probability-based metric that is used as a way to capture the degree of "uncertainty" a model has in predicting (assigning probabilities to) the next word. A lower perplexity is associated with a higher accuracy in the model's prediction. To put it simply, a researcher would pick a model that is least "perplexed" when presented with a never-before-seen data set.

Our experiment was split into two phases. In the first phase, bi-gram and tri-gram models were trained on one of five contemporary genre-specific corpora including news, nonfiction (Wikipedia), fiction (Harry Potter), texts, and tweets, and tested on all five genres. In the second phase, bi-gram and tri-gram models were trained on one of three

different time periods in one of two genres including novels and poems, and tested on all three time periods.

To provide an overview of our approach, we first trained our model on smoothed bi-grams and tri-grams produced from a specific corpus. Next, we generated a testing corpus for each genre to be run on the trained model, and calculated the perplexities associated with that testing corpus by the trained bi-gram and tri-gram models. We extended this methodology to use each genre as a baseline for training the model, and use the non-baseline genres and its own isolated test corpus to test the baseline-trained model. For example, sentences from news articles would be used to train the model and provide a baseline, while the other categories (nonfiction, tweets, texts, fiction) and its own isolated test corpus would be tested for perplexity values by the bi-gram and tri-gram news-trained models.

We ultimately found that perplexity values are much higher on models trained on corpora with a more specific, complex range of vocabulary, indicating that those are less generalizable. This was reflected in our findings in both sections of testing. In the comparison of written texts, we found that news and wikipedia-corpus trained models demonstrated the highest perplexity scores. For fictional works (novels and poems) over different time periods, we found that the models trained on the oldest novel and the middle-aged poem had the highest perplexity values.

2 Related Work

N-gram models have everyday applications: smart assistants, such as Alexa and Siri, text and sentiment analysis on social media, and translation devices, including Google Translate. The field and industry of natural language processing is so vast that it is easier to situate the model used in this paper in the broader context of the field, rather than focusing on specific papers. Existing capabilities and uses of n-gram models have become commonplace in our day to day lives, furthering the need to look at contemporary language models that accomplish similar but further reaching tasks to simple n-gram models.

BERT (Bidirectional Encoder Representations from Transformers) is a multidirectional language model that was published by Devlin et al. in 2019. This model looks at the context to both the right and left of a token during the training phase, fusing them together, rather than looking from say just right-to-left. The BERT model also used a new approach: pre-training. Pre-training with a massive dataset acts as a skeleton upon which task specific fine tuning layers can be applied, giving the model the power to complete a wide array of tasks to a higher degree of accuracy. This Transfer Learning approach is now commonplace in NLP applications.

ELMo (Embeddings from Language Models) is a recent groundbreaking model by AllenNLP. ELMo models complex word characteristics and tackles the issue of polysemy. ELMo maps words onto vector points, but takes into account the entire input sentence. This allows it to embed different words to different vectors depending on the context. For example, given the sentences "I read the book yesterday" and "Can you read the book?", ELMo can correctly differentiate between the two readings by putting "read" at two different vectors. By using context-dependent embeddings, ELMo is a powerful tool used from sentiment analysis to question-answering.

The next major work in the field is the incredibly powerful GPT-2 unidirectional model by OpenAI. While BERT is a powerful tool, the unidirectional nature of GPT-2 allows it to do things that BERT was not meant to: text generation. It focuses on important words that occurred earlier in the text and uses those to predict the following words. GPT-2 is able to generate mostly cogent stories from little input prompts. The full model was not published, as the creators deemed that it could be misused to generate fake news articles or spread disinformation. The model was trained on 8 million websites and has 1.5 billion parameters, ten times as many as its predecessor, GPT-1.

The model used in this paper uses similar basic predictive concepts as the more advanced ones described above. However, it did not use any Transfer Learning techniques and was not pre-trained on massive corpora like is becoming standard. Instead, our model uses small training and testing sets to see whether a model trained on fairly small corpora could still generalize well to other corpora.

3 Model and Methods

We opted to work with bi-gram and tri-gram language models in our implementation, cho-

sen under the assumption that any task-specific generalization issues found in the simplest language model would be magnified in more complex modes. Our implementation drew heavily from the n-gram model described in Jurafsky and Martin Ch.3 and consisted of two phases: preprocessing and evaluation.

In the preprocessing stage, we cleaned each of our corpora to make them as structurally similar as possible. We started by removing any corpus-specific tags or phrases. This included the ham and spam tags within the texts corpus, the starting ¡p¿ and ending ¡/p¿ tags for the tweets corpus, and chapter headings and page endings for the fiction corpora. We then removed punctuation, capitalization, and any additional newlines from each corpus. Finally, we split each corpus into a list of corpus-specific lines. For the news, nonfiction, texts, tweets, and poetry corpora, this entailed simply splitting the corpus along newlines. For the fiction corpora from Project Gutenberg, this meant splitting the corpus by periods.

After cleaning up the corpora, for each experiment we chose a baseline corpus to train on. We then split that corpus in a 2:1 ratio, utilizing the first of the corpus lines as a training set, and reserving the last of the corpus lines as a testing set, unless the remaining number of lines was greater than any of the other corpora being utilized as testing sets. Since there was some variance between the number of lines in each corpus, we restricted the training set to be the minimum of the corpus line count and the minimum number of lines in any corpus. This enabled each trained model to be tested on the same number of lines.

Moving into the evaluation phase, we started by training our chosen bi-gram or tri-gram model on the isolated training set from the preprocessing phase. This consisted of calculating the maximum likelihood conditional probability of the next word in an n-gram model as the frequency of the n-gram over the frequency of the n-grams leading n-1-grams, using Equation 3.11 from Jurafsky and Martin Ch.3 for the bi-gram model:

$$P(W_n|W_{n-1}) = \frac{C(w_{n-1}w_n)}{C(w_{n-1)}}$$

and adapting it for the tri-gram model:

$$P(W_n|W_{n-2}W_{n-1}) = \frac{C(w_{n-2}w_{n-1}w_n)}{C(w_{n-2}w_{n-1})}$$

During this phase, we added special start and end symbols $< s >$ and $< /s >$ to the first and last $n - 1$ words for our chosen bi-gram or tri-gram model to provide the n-gram context of the first and last $n-1$ words. Additionally, we utilized add-1 smoothing in our models, to ensure that bi-grams or tri-grams found in the testing sets which appear in the vocabulary of the training set, but in a different context, are not automatically given zero probability. This consisted of adding one to numerator and the count V of unique leading $n-1$ grams to the denominator of the above equations, using Equation 3.20 from Jurafsky and Martin Ch. 3:

$$P_{Laplace}(w_i) = \frac{c_i + 1}{N_V}$$

This solved the potential problem of an infinite perplexity calculation.

Following the training phase, we then tested our trained bi-gram and tri-gram model on each of the isolated testing sets from the preprocessing phase. This consisted of first calculating the cross-entropy of each testing set using Equation 3.51 from Jurafsky and Martin Ch. 3:

$$H(W) = -\frac{1}{N}logP(w_1w_2...w_n)$$

Using the result of that calculation, we were then able to calculate the perplexity of our model on each test corpus using Equation 3.52 from Jurafsky and Martin Ch.3:

$$Perplexity(W) = 2^{H(W}$$

Here perplexity, the geometric mean of the inverse test set probability, acts as our measure of how well our model has generalized on a given test corpus, where lower perplexity scores indicate better generalization.

4 Experiments

To better understand the generalization of our models across corpora of different genres and time periods, our experiments were focused on providing clarity, starting from the datasets we elected to use.

To determine similarities and differences between different genres of corpora, we decided to compare a wide range of data, from informal, colloquial settings like SMS messages and tweets, to more formal registers like news articles, nonfiction articles, and popular fiction.

To collect data from each of these broad categories, many different avenues were pursued. The SMS messages were collected from a Spam Analysis database of 15K text messages submitted from National University of Singapore students and a UK web forum. The tweets were collected from the American National Corpus MASC database, which includes 24K tweets. The nonfiction (Wikipedia) corpus and news articles were found from a paper by Goldhahn et al., each containing 10K words taken from 2016. Finally, the popular fiction dataset is taken directly from J.K. Rowling's Harry Potter and the Philosopher's Stone. Using these datasets, we opted to train multiple language models, one on each genre, and compare each of those models against tests on every genre. Since we wanted to include a test on the baseline genre while maintaining a uniform test set size across tests, each model began by separating the baseline dataset into a training and testing set, with testing holding $\frac{1}{3}$ of the data and training holding $\frac{2}{3}$. With the length of this testing set, we then took equally sized sets out of the different genres for testing.

To examine the similarities and differences in corpora taken across time, we decided to examine novels and poems from three different time periods. For both of these genres, the datasets were found using the Gutenberg Project. The specific novels used are Le Morte D'Arthur, by Sir Thomas Mallory (1485), Pride and Prejudice by Jane Austen (1813), and Harry Potter and the Philosopher's Stone by J.K. Rowling (1998). The specific poems used are the complete collection of Shakespeare's Sonnets (1590), Leaves of Grass by Walt Whitman (1855), and New Hampshire by Robert Frost (1923). These datasets were tested against each other in the same way as with the genre comparison, creating a model trained on each time period within a genre and then testing it on all the other time periods.

In both experiments, we received perplexity scores, an inverse value denoting how closely the test set conforms to the data the model was trained on—the smaller the perplexity, the more similar the two datasets—from the model. With this in mind, we ran through our experiments and received the results outlined in Tables 1 through 6.

Baseline	**News**	**Nonfiction**	**Tweets**	**Texts**	**Fiction**
News	41.18	23.95	3.98	5.66	5.68
Nonfiction	19.54	38.65	2.90	3.82	4.02
Tweets	15.04	9.21	6.15	7.45	5.40
Texts	19.96	11.14	6.07	75.38	7.65
Fiction	35.59	35.59	4.93	11.44	57.14

Table 1: Bi-gram Model, Genre Comparison

Baseline	**News**	**Nonfiction**	**Tweets**	**Texts**	**Fiction**
News	2.28	2.01	1.13	1.25	1.39
Nonfiction	2.02	2.27	1.05	1.11	1.24
Tweets	1.82	1.38	1.59	1.57	1.42
Texts	1.85	1.32	1.28	11.40	1.48
Fiction	2.52	1.84	1.21	1.53	4.91

Table 2: Tri-gram Model, Genre Comparison

Baseline	**Old (∼1500)**	**Middle (∼1820)**	**Young (∼1990)**
Old	95.06	14.43	9.32
Middle	13.42	79.75	12.04
Young	8.10	12.03	57.14

Table 3: Bi-gram Model, Novel Timeline Comparison

Baseline	**Old (∼1500)**	**Middle (∼1820)**	**Young (∼1990)**
Old	19.01	1.81	1.44
Middle	2.04	5.41	1.58
Young	1.63	1.80	4.91

Table 4: Tri-gram Model, Novel Timeline Comparison

Baseline	**Old (∼1590)**	**Middle (∼1850)**	**Young (∼1920)**
Old	12.67	25.38	5.04
Middle	5.40	51.44	6.60
Young	5.71	38.12	16.91

Table 5: Bi-gram Model, Poem Timeline Comparison

Baseline	**Old (∼1590)**	**Middle (∼1850)**	**Young (∼1920)**
Old	1.72	1.95	1.25
Middle	1.30	3.76	1.28
Young	1.21	2.35	1.90

Table 6: Tri-gram Model, Poem Timeline Comparison

5 Results

The overall results are reported in Tables 1 through 6. Perplexity values produced across the five models trained and tested on contemporary genres can be found in Tables 1-2, perplexity values produced across the three models trained and tested on novels across different time periods can be found in Tables 3-4, and perplexity values produced across the three models trained and tested on works of poetry from different time periods can be found in Tables 5-6.

In the contemporary genre section, the most noteworthy result was a tendency in all genres for the test corpus drawn from the baseline corpus to perform the worst, most obvious in the case of texts and fiction. More broadly, we found models trained on typically more formal registers, such as the news, nonfiction, and fiction corpuses, had a harder time generalizing and being useful across multiple genres, whereas the texts and tweets, which are more informal, performed very well across the majority of datasets.

In the time comparison section, the novels still exhibited the problem of the test corpus drawn from the baseline corpus performing the worst on its own model, but it was much less pronounced in the middle and young novels. More generally, the middle and young models did well at generalizing across all time periods. Even more interesting, though, is the lack of this same pattern in the poetry section. While the test corpus drawn from the baseline corpus still performs the worst, the difference in perplexity scores is much smaller, implying that there may be a much greater ability to generalize across time periods in poetry than in other genres. The young and old corpus trained models works appear best for performing that generalization.

As an aside, our models based on tri-gram probabilities displayed lower perplexity values across all sections, which is to be expected as tri-grams include all the information that bi-grams present.

6 Discussion

In the end, our results demonstrated that the model does generalize well across corpora of various genres and time periods. Our results also demonstrated that our original prediction held true: that perplexity values were expected to be much higher on models trained on corpora with a more specific range of vocabulary. This makes sense as the model trained on sentences from highly specific subjects and genres would face lower probability values for strings of words from other subjects.

Our most confounding finding was that the perplexities were the highest in the same subject that the model was trained on. We conjecture that this is due to an overly distinctive set of vocabulary used for training the model in each instance. For example, news and nonfiction Wikipedia articles perform relatively poorly when tested on themselves because the sentences pulled from individual articles vary by subject. We find this problem to be most pronounced in texts and fiction, which can both be explained by the problem of novel words due to subject specificity.

Within the problem of novel words, the two datasets have different specific contexts. Tweets and text messages are more likely to have misspellings, incorrect contractions, and negligent usage of homonyms. To provide some instances, "you" can be spelled as "u" and the incorrect usage of "your/you're" and "there/their/they're" are more common in informal text speak. Additionally, a model trained on formal writing genres and tested on Tweets or text messages would perform poorly as both genres can include informal abbreviations such as "idk" and "lol". While humans understand and use these unique words interchangeably in text, the model recognizes them as distinct lexical items, and finding a large amount of disparate spellings throughout the texts could throw off the model that would, in other written situations, be catching them as the same word.

In the case of fiction, novel words are also constantly being found, and are equally meaningless in any context besides the sentence found there, to the eyes of the model. For instance, with Harry Potter and the Philosopher's Stone, the test set performs poorly when tested on common genre-trained models because of its magical terminology. So, despite being properly trained on general written English, the test would still encounter many novel words in the training set that it would not find in, say, a Wikipedia article. This also may be affected in some way by our method of choosing training and testing sets, as they are chosen from contiguous sections of text, which could cause the test set to have content very dissimilar to that of the training set, thus making novel words more likely.

7 Conclusion

Through our examination of an N-Gram Language Model trained on a variety of contemporary and historical genres, we were able to determine the relationships in speech between those genres and extract interesting conclusions, including a tendency for more colloquial registers of language to generalize better, and signs pointing to poetry as a relatively unchanged written form through history. However, these are observations any person can make in their own experience of language, and what is much more compelling are the implications this has for the future of language models and their viability in the study of human language. In our model, we found a tendency to perform worse on the same dataset with which the model was trained. This finding is counterintuitive to the understanding of language models, and very well may be the result of errors in our implementation, but could also be caused by some of the issues outlined in discussion.

With these areas of possible weakness in the model as it is now, the next steps in our study of N-Gram Language Models could be around trying to counteract those forces on the same dataset test discrepancy, such as randomizing the lines present in the training and testing sets, or providing larger sets that give a broader overview of the various novel spellings and their contexts. Additionally, having a more standardized preprocessing procedure to guarantee a similarity in the data provided to the model would help with ensuring the model is providing perplexity values that are truly caused by the similarity between the datasets used. However, finding that our model performs well in generalizing across many different genres certainly helps to solidify the viability of N-Gram Language Models in analysing human language in the interim.

8 References

Spam Dataset:
Almeida, T.A., Gómez Hidalgo, J.M., Silva, T.P. 2013. Towards SMS Spam Filtering: Results under a New Dataset. International Journal of Information Security Science (IJISS), 2(1), 1-18. http://www.dt.fee.unicamp.br/ tiago/smsspamcollection/

Tweets:
Passonneau, R., Baker, C., Fellbaum, C., Ide, N. 2012. The MASC Word Sense Sentence Corpus. Proceedings of the Eighth Language Resources and Evaluation Conference, Istanbul. http://www.anc.org/data/masc/corpus/browse-masc-data/

Fiction Corpora:
Rowling, J.K. 1999. Harry Potter and the Sorcerer's Stone. Scholastic. New York, NY

Project Gutenberg. (n.d.). Retrieved December 13, 2020, from www.gutenberg.org.

News and Wikipedia Corpuses:
D. Goldhahn, T. Eckart U. Quasthoff. 2012. Building Large Monolingual Dictionaries at the Leipzig Corpora Collection: From 100 to 200 Languages. Proceedings of the 8th International Language Resources and Evaluation (LREC'12)

Other:
Horev, Rani. 2018. "BERT Explained: State of the Art Language Model for NLP." Medium, Towards Data Science. towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270.

Vig, Jesse. 2019. "OpenAI GPT-2: Understanding Language Generation through Visualization." Medium, Towards Data Science. towardsdatascience.com/openai-gpt-2-understanding-language-generation-through-visualization-8252f683b2f8

Joshi, Prateek. 2020. What Is ELMo: ELMo For Text Classification in Python. Analytics Vidhya. www.analyticsvidhya.com/blog/2019/03/learn-to-use-elmo-to-extract-features-from-text/?utm$_source = blog$.

Jurafsky Martin Ch. 3: Daniel Jurafsky and James H. Martin. 2009. Speech and Language Processing (2nd Edition). Prentice-Hall, Inc., USA.