中文信息熵

王铭泽 ZY2203115

wmz20000729@buaa.edu.cn

1 Introduction

Information entropy is a fundamental concept in information theory that describes the uncertainty of possible events in an information source. In the 1940s, Claude Shannon borrowed the concept from thermodynamics and defined the average amount of information in information after removing redundancy as "information entropy," providing a mathematical expression for calculating it. The introduction of information entropy resolved the problem of quantifying and measuring information.

$$H(x) = E[-\log(P(x))] = -\sum_{x \in X} P(x) \log(P(x)) \tag{1}$$

In general, the greater the uncertainty of an event, the greater its information entropy. Conversely, if the uncertainty is lower, the information entropy will also be lower. The more uncertain an event is, the greater the amount of information we obtain when we receive relevant information. Therefore, information entropy is positively correlated with the value of the information. Conversely, for events that are relatively certain, the value of the information is lower, and the information entropy is relatively smaller.

A language model (LM) is a model of natural language, and from the perspective of machine learning, it can also be regarded as a model of the probability distribution of sentences. Simply put, human language has contextual relationships, and what is said later in a sentence is inevitably related to what was said before. The language model utilizes this point to learn human language abilities.

Given a sentence sequence S = w1, w2, ..., wN, we first want to know if it conforms to the rules of language, so we use a joint probability P(w1, w2, ..., wN) to measure the probability that it conforms to the rules. According to the chain rule, this probability can be expressed as:

$$\begin{aligned} P(S) = P(w_0, w_1, \ldots, w_n) &= P(w_0)\, P(w_1; w_0)\, P(w_2 \mid w_0 w_1) \ldots P(w_n \mid w_0 w_1 \ldots w_{n-1}) \\ &= \prod_i P(w_i \mid w_0 w_1 \ldots w_{i-1}) \end{aligned} \tag{2}$$

Therefore, a language model is essentially a series of conditional probability multiplications. However, when the length of the sentence sequence is too long, that is, when n is too large, it is difficult to calculate such a long sequence of conditional probabilities. Russian mathematician Markov proposed a hypothesis that the probability of any word w_i appearing is only related to the previous word w_{i-1}, and this assumption is called the Markov assumption. In this case, we have:

$$\begin{aligned} P_2(S) &= P(w_0)\, P(w_1 \mid w_0)\, P(w_2 \mid w_1) \ldots P(w_n \mid w_{n-1}) \\ &= \prod_i P(w_i \mid w_{i-1}) \end{aligned} \tag{3}$$

This type of model is also known as a 2-gram model. Similarly, there are also 1-gram models, 3-gram models, and n-gram models.The unigram (1-gram) model assumes that each word is only related to itself,

while the trigram (3-gram) model assumes that each word is related to its preceding two words. Usually, the unigram model is used less frequently because it completely ignores the contextual relationship between words. The bigram, trigram, and even higher-order models are more commonly used.

We can model a given Chinese text using unigram, bigram, and trigram language models, and calculate their information entropy using a similar formula as equation (1). The information entropy of the three language models is defined as follows:

$$H_1(w) = -\sum_i P(w_i) \log(P(w_i)) \tag{4}$$

$$H_2(w) = -\sum_i P(w_{i-1}w_i) \log(P(w_i \mid w_{i-1})) \tag{5}$$

$$H_3(w) = -\sum_i P(w_{i-2}w_{i-1}w_i) \log(P(w_i \mid w_{i-2}w_{i-1})) \tag{6}$$

2 Method

Firstly, before formally processing the text, we need to do some data preprocessing. The article contains line breaks and a small amount of English letters, which need to be removed. In some state-of-the-art language models, punctuation marks are also considered, but here, for the convenience of word segmentation and processing, all punctuation marks are removed, leaving only pure text. Regular expressions are used for text filtering.

Then, before calculating the information entropy based on word frequency, we need to segment the text. Therefore, the Jieba segmentation tool is used, and stop words are removed after segmentation. For the case of calculating information entropy based on characters, no deletion processing is done.

Counting word frequency is relatively simple. The author uses a dictionary in Python to store all combinations of characters (words), and continuously counts the frequency of each combination through a sliding window.

For the 1-gram model, only the frequency of each character (word) needs to be counted; for the 2-gram model, the frequency of adjacent two characters (words) also needs to be counted; and for the 3-gram model, the frequency of adjacent three characters (words) is also included.

It is worth noting that when counting, the order of each combination must also be considered. Different orders are treated as different combinations.

Based on equations (6-8) and the word frequency statistics in Section 2.2, we can easily calculate the information entropy of each novel under the 1-gram/2-gram/3-gram model.

For the information entropy of the entire corpus, we add up the number of occurrences of all character (word) combinations in each novel, and then use the sum to calculate the overall information entropy.

3 Experiment

According to the content described in section two, the entropy is calculated based on characters and words respectively, and the results are in table 1.

The unit of information entropy is bits per word. Obviously, when calculating the information entropy of Chinese characters, the entropy of the 1-gram model is the highest, presumably because the 1-gram model completely ignores the context, and there is no pattern, thus the uncertainty is very high.

表 1: The entropy calculated based on characters

书名	一元模型	二元模型	三元模型
白马啸西风	8.91	4.58	1.61
碧血剑	9.45	5.86	2.34
飞狐外传	9.31	5.75	2.40
连城诀	9.17	5.39	2.15
鹿鼎记	9.29	5.68	2.94
三十三剑客图	9.67	4.83	0.98
射雕英雄传	9.44	6.06	2.75
神雕侠侣	9.37	6.06	2.84
书剑恩仇录	9.46	5.78	2.39
天龙八部	9.40	6.12	2.94
侠客行	9.15	5.59	2.36
笑傲江湖	9.20	5.89	2.87
雪山飞狐	9.20	5.15	1.75
倚天屠龙记	9.39	6.02	2.80
鸳鸯刀	9.04	4.21	1.11
越女剑	8.83	3.64	0.90
全文	9.53	6.72	3.95

The entropy of the 2-gram and 3-gram models has decreased, indicating that the context relationship may determine the appearance of some characters, which increases the level of certainty. Furthermore, the longer the considered context, the higher this certainty.

It is worth noting that when considering the frequency of all characters in the corpus together, there is a slight increase in the overall entropy. It is inferred that the variability in word usage, especially some nouns, between different articles has resulted in significant changes in word frequency and an increase in uncertainty.

The unit of information entropy is bits per word. Obviously, when calculating the information entropy of Chinese characters based on a character-by-character approach, the entropy of the unigram model is the highest. This is probably because the unigram model completely ignores context and has no regularity, leading to high uncertainty.

The entropy of the bigram and trigram models decreases, indicating that the context relationship may determine the appearance of some characters, thus increasing certainty. Furthermore, the longer the considered context, the higher the certainty.

It is worth noting that when considering the frequency of characters across all texts and calculating the total entropy, there is a certain increase in entropy. This is probably due to the fact that different articles have different writing styles and word usage, especially with different nouns, leading to significant changes in word frequency and an increase in uncertainty.

When calculating the information entropy based on words, similar results to the previous section were obtained, with the trigram model having the lowest entropy and the unigram model having the highest entropy. However, it is worth noting that when counting words, the entropy of the unigram model is higher than that of the character-based approach. For the bigram model, the entropy is almost the same, but when looking at each novel individually, the entropy based on words is smaller. By the trigram model, the entropy based on words is smaller in all aspects.

表 2: The entropy calculated based on words

书名	一元模型	二元模型	三元模型
白马啸西风	11.16	2.89	0.35
碧血剑	12.92	3.93	0.42
飞狐外传	12.66	4.02	0.45
连城诀	12.24	3.55	0.35
鹿鼎记	12.67	4.93	0.81
三十三剑客图	12.57	1.75	0.08
射雕英雄传	13.06	4.56	0.52
神雕侠侣	12.87	4.73	0.62
书剑恩仇录	12.75	4.12	0.48
天龙八部	13.07	4.80	0.64
侠客行	12.32	3.96	0.49
笑傲江湖	12.56	4.81	0.77
雪山飞狐	12.08	3.04	0.28
倚天屠龙记	12.93	4.65	0.62
鸳鸯刀	11.12	2.13	0.23
越女剑	10.48	1.72	0.23
全文	13.65	6.50	1.15

In my personal understanding, the unigram model is only related to word (character) frequency. As there are more possible combinations of characters that can form words, the uncertainty is higher, resulting in higher entropy. In the bigram model, there is some regularity in word combinations, resulting in a decrease in entropy. By the trigram model, this regularity is even more evident, leading to a further decrease in entropy.