The Entropy/Naturalness of Software Documentation

# Abstract

# Introduction

If programmers writing code are repetitive and predictable, I would bet that questions and answers (and to a lesser extent, formal documentation) about code are even more repetitive and predictable because people tend to ask and answer the same questions repeatedly (ie people learn in similar ways). The entropy comparison that I would like to see/do is not between general English and code or even among programs, but between the (informal) documentation of an API and the programs that use the API. Given that there is English as well as code fragments in documentation, I would suspect that documentation would have higher entropy than source code. However, if we use ACER to extract the code elements and ignore the English text, I would suspect that source code would have a higher entropy than code elements in documentation. In effect, I hypothesize that the English between the code elements is obscuring the repetitive and simple nature of the topics covered by the documentation.

## Research Hypotheses

We expect the cross-entropy of the following to be in decreasing order:

1. English text
2. Documentation
3. Source code of API
4. CE found in Docs

This paper is structured as follows. In Section (Background) we discuss …

# Background and Literature

Hindel et al (Hindle2012ICSE)

"Discovering Essential Code Elements in Informal Documentation" RR2013ICSE

“Mining Source Code Repositories at Massive Scale using Language Modeling” Allamanis2013MSR

# Methodology

MitLM and what does, what is ngram, language models,

Extracted code elements are from stackoverflow using Rigby and Robillard’s tool [RB2013ICSE].

# Findings and Discussion

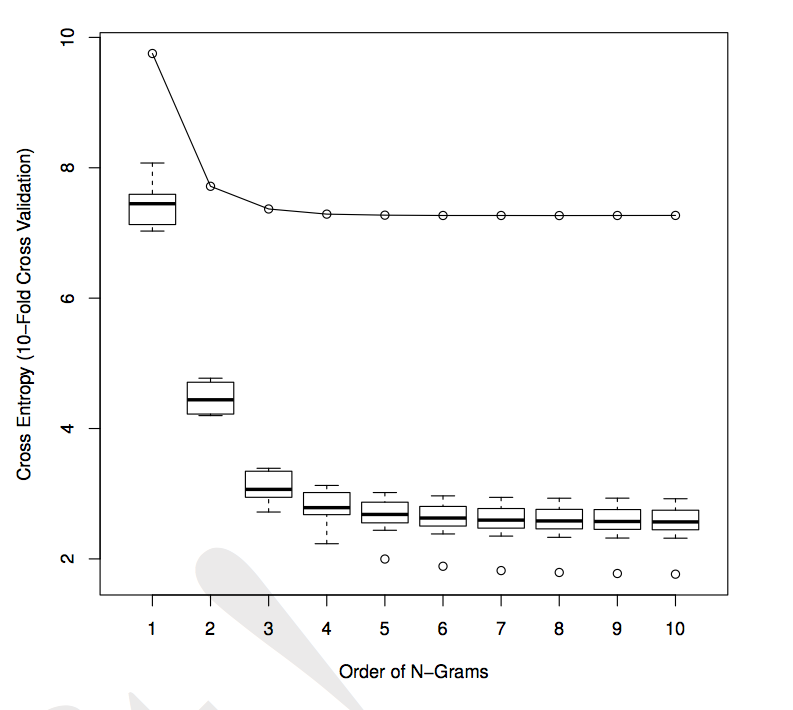


Figure 1 is from Hindle2013ICSE

Plot vs hindle’s plot

Discussion will compare the two plots

# Conclusion

# References

10 References or more:

"On the Naturalness of Software" Hindle2012ICSE