CRANFIELD UNIVERSITY
Msc Computational And Software Techniques In Engineering

Master of Science

Academic Year 2022 - 2023

SIMÉON FEREZ

Noisy Entropy Estimation For Various Languages Using Social Media Data And Programming Languages

Supervisor: Dr. Jun Li
Associate Supervisor: Dr. Barnes Stuart
August 2023

This thesis is submitted in partial fulfilment of the requirements for the degree of Computational Intelligence for Data Analytics

ABSTRACT

Language, both written and spoken, exhibits inherent complexities that reflect the unpredictability and intricate patterns of human communication. This thesis embarks on a comprehensive exploration of entropy in language. Through a multifaceted methodological approach, this research navigated noisy entropy across a wide array of domains, including natural languages, social media, contemporary events, news outlets, individual communication, and programming languages.

Key highlights of the study encompass the rigorous investigation of entropy within the literature, demonstrating a more realistic and chaotic environment by considering a broader character set. On Twitter, it revealed a profound consistency in character-level entropy with literature, shedding light on the chaotic nature of individual word usage and the logical structure within the noise. The inclusion of elements such as punctuation, emojis, and accents further enriched the understanding of entropy in online communication.

Through time-series analysis during significant global events such as the COVID-19 pandemic and the Ukraine War, the research identified convergences in unigram entropy, possibly reflecting societal adaptation and the unifying influence of global media. The inclusion of programming languages further uncovered the duality of surface-level complexity and underlying predictable structure, highlighting the creativity and universality inherent in programming.

The study, underpinned by robust statistical analyses and various entropy estimators, laid the foundation for future explorations and applications in the realm of noisy entropy. The insights developed have substantial potential implications for areas such as biased detector development, uncertainty analysis, fact-checking technology, and understanding human emotional discourse. The work transforms abstract understandings into quantifiable measurements, acting as a pivotal step towards new theories, applications, and understandings the interplay of structure and chaos in language.

Keywords: Noisy Entropy, Complexity, Social Media, Programming Languages

ACKNOWLEDGEMENTS

I would like to begin by expressing my gratitude to Cranfield University for granting me the opportunity to undertake this intellectually stimulating project. The support, resources and academic environment provided by the university have played a role in my academic growth leading to the successful completion of this research endeavour.

I would also like to extend my appreciation to Dr. Irene Moulitsas, who has served as my course director. Her guidance and unwavering commitment, to excellence have profoundly influenced my journey.

A special acknowledgement goes out to Dr. Jun Li, my supervisor, whose wisdom, knowledge and passion for research have been invaluable. His continuous feedback, encouragement and meticulous attention to detail have guided me through the aspects of this research project. Shaped it into what it is today.

I am genuinely grateful for the guidance provided by Dr. Barnes Stuart, my associate supervisor. His expertise and availability have been instrumental in shaping the direction of my work.

Furthermore, I would like to express thanks to all the lecturers who have contributed significantly during my Master's degree program, at Cranfield University. Their wisdom, guidance and unwavering dedication have enriched my experience in ways.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

LM	Language Modelling
PPM	Prediction by Partial Matching
AI	Artificial Intelligence
API	Application Programming Interface
GPU	Graphics Processing Unit
LLM	Large Language Model

1 INTRODUCTION

Note: For the purpose of this report, the entity formerly known as "Twitter" will continue to be referred to as "Twitter".

1.1 Project Introduction

Much like physical laws that govern the natural world, entropy in language serves as a fundamental principle that dictates the unpredictability and complexity inherent in various forms of communication.

Information is the heart of the digital area. Information is in every aspect of our daily life from influencing decisions and shaping societal conversations, to driving technological innovations. The amount of information generated and consumed has grown exponentially over the last decade. The principal factors are the rapid advancement in information, communications technology and Artificial Intelligence. This expansion has highlighted the intricate need to understand, quantify and process this information in an efficient and accurate manner.

This Master's Thesis delve into one of the pillars of information processing: the challenges of noisy information that habit in social media conversations and programming languages.

1.2 Background and Context

The digital age is characterized by an exponential increase in the generation and consumption of information, that shapes an information-centric world. The fundamental field leading this transition is information theory, a domain introduced by Claude Shannon in the mid-20th century. Shannon's work lays the foundation for a quantitative and mathematical approach to information. He introduced concepts such as information entropy, which provides a measure of the uncertainty in a set of outcomes (Shannon, 1948). Since its conception and introduction information theory has led to numerous innovations and become particularly influential in the field of computer science, machine learning and data science.

The processing and understanding of language, a unique and complex form of information, has gained momentum with the recent advent of computation linguistics and specific high-performing GPU kernels. The new computational power and technology are leading innovations in language modelling and Artificial Intelligence. The application of information theory principles to linguistic data, as examined by Pierce (John R. Pierce, 1980) provide a powerful and rich toolset for handling the diversity and ambiguity of languages. Furthermore, the relevance of statistics and predictions to understanding the human language has been emphasized by researchers such as Feder, Merkhav and Gutman (Feder et al., 1992).

Over the years, significant advancements have been made in applying these principles to language processing. Zipf's word frequency law, which applies the frequency of a word's occurrence in natural language to its rank, is an example of an early instance of the application of information theory in language processing (Piantadosi, 2014; Zipf, 1936; Zipf G, 1949). More recently computational linguistics has seen the interaction of sophisticated information theory concepts, aiding in tasks such as semantic analysis, machine translation and sentiment analysis.

Despite these advances, real-world language, spoken language and SMS-writing often present unique challenges, particularly in the face of noisy information. Noisy language data are composed of irregularities, informal language use, errors, and other abnormalities, which are increasingly present in source as social media as demonstrated by Jacob Eisenstein (Eisenstein, 2013). Baldwin & all (Baldwin et al., 2013; Han & Baldwin, 2011) highlight the characteristics and differences of noisy datasets on the internet. They show that such data disrupts traditional methods of information pre- and post-processing, complicating task as natural language processing and language modelling by predictions.

Entropy, a concept from information theory represents the amount of uncertainty, or randomness in a set of data (Shannon, 1948). In the context of language information, quantifying entropy can help us understand the predictability of language patterns and the amount of information carried by a particular message

(Cover et al., 2006). However, noisy data can affect the accuracy and method of entropy estimations, consequently, the rise of noisy data necessitates novel methodologies to address these challenges.

In the recent entropy estimations methods, we find compression algorithms. At their core, data compression algorithms reduce the amount of data needed to represent information, by minimizing redundancy and maximizing predictability (Salomon & Motta, 2010). They operate under the information theory exploiting the concept of entropy for efficient data representation (Cover et al., 2006). Compression algorithms such as (Lempel-Ziv) LZ77, LZ78, and Prediction by Partial Matching (PPM) have been applied to estimate entropy, presenting a novel approach to the analysis of language information.

Nonetheless, the performance of these algorithms varies and their effectiveness is still a subject of ongoing research. Furthermore, the increasing presence of noisy data and the choice of enlarging the data acquisition to train larger AI models introduce a new dimension to this challenge. Thus, while progress has been made in the realm of information theory and language processing, significant gaps persist, particularly concerning entropy estimation, especially in noisy environments.

Entropy is still the subject of recent development in machine learning. In a recently published work, researchers offered a novel and lightweight method to estimate text entropy and classify text data (Jiang et al., 2023). This work employs compression algorithms for entropy estimation and combines a common compression tool, gzip, with a k-nearest-neighbour classifier for text classification. The gzip-based method has shown surprisingly competitive results compared to more complex deep neural network classifiers.

1.3 Statement Problem

The problems that this research seeks to address relate to the effective estimation of entropy in the context of noisy language data. As established, language data extracted from sources such as social media platforms, web

pages, and text messages often carry noise in the form of errors, irregularities, and informalities, disrupting traditional information processing methods. With the rise of artificial intelligence and machine learning, noisy data present significant hurdles in achieving efficient and accurate language processing and modelling.

Although entropy estimation offers a mathematical approach to quantify uncertainty and randomness in data, existing studies and methodologies have mainly focused on a limited alphabet or vocabulary with heavy pre-processing methodology. Furthermore, there is a lack of comprehensive studies that focus on quantifying entropy in noisy language data, particularly from social media and computer language sources. This situation underscores a significant gap in the literature and establishes the need to develop effective methodologies for noisy entropy estimation and noise analysis in these data types.

1.4 Research Objectives

This research aims to contribute to the field of information theory and language processing by focusing on noisy entropy estimation in language data derived from social media and computer languages, areas that have been underrepresented in previous studies. The specific objectives of this research include:

To quantify the noisy entropy in language data extracted from social media and computer languages. The research aims to fill the gap in the literature by focusing on noisy entropy estimation in these often-overlooked sources of data.

To understand and analyse the nature of the noise present in these data types. This involves identifying the type of noise, its origin, and how it impacts the overall data structure.

To critically analyze the impact of noise on entropy estimation in language data. This analysis will consider different forms and levels of noise and their impact on the entropy estimation.

To review and compare the performance of various compression algorithms, in estimating entropy in language data. The research will seek to understand the

strengths and weaknesses of each algorithm and its suitability to different scenarios.

To validate the developed methodology using real-world noisy language datasets. This step will enable an evaluation of the effectiveness of the methodology in a practical setting and allow for possible refinements.

To evaluate the effects of different preprocessing techniques on the noise level and entropy estimation. The research will consider various factors such as the nature and level of noise and the type of preprocessing techniques used.

To develop a methodology to effectively reduce noise in the data used in AI or machine learning pipelines. This methodology aims to enhance the accuracy and efficiency of subsequent language processing tasks.

By fulfilling these objectives, this research aims to enhance our understanding of noisy entropy and provide valuable insights into its impact on language processing tasks. It also aims to guide the development of more effective pre-processing techniques to handle noise in language data, potentially improving the performance of AI models and machine learning pipelines.

1.5 Significance of the Study

This research holds significant potential for both theoretical and practical implications. From a theoretical perspective, it contributes to the existing knowledge base by exploring and expanding upon the concept of entropy estimation in noisy language data. By focusing on this niche, this study helps to enrich the understanding of entropy's applicability in these contexts and enhances the theoretical framework of noisy entropy estimation.

Practically, the findings of this study are of value for numerous applications, particularly those reliant on effective and efficient language processing. As noisy language data is a common occurrence in real-world contexts, especially on platforms such as social media, a refined understanding of entropy in these environments can substantially improve the quality of language processing,

sentiment analysis, machine translation, and other similar tasks. Further, by delving into the specifics of where the noise manifests in the data, this study can guide the development of robust preprocessing techniques specifically tailored to handle noise in these contexts, thereby improving efficiency and accuracy in downstream machine learning or AI applications.

Moreover, the study can benefit the area of data compression by providing insights into the relationship between noisy language data and compression algorithms. As compression algorithms are central to handling large-scale data, advancements in this area can have a far-reaching impact on the fields of computer science and data engineering.

1.6 Thesis Structure

This thesis is systematically structured to offer an in-depth exploration of entropy estimation in noisy language data.

LITERATURE REVIEW contains a literature review that highlights the gaps this research aims to fill. It covers information theory, entropy estimation, and the current state of research in language processing in the context of social media and computer languages.

METHODOLOGY delves into the methodology, discussing the techniques used for data collection, entropy estimation, and noise analysis, in addition to the statistical and analytical techniques employed.

ANALYSIS covers data exploration, where the dataset is analyzed to gain deeper understanding and insights.

RESULTS presents and describes the results, detailing the entropy calculations and the comparative effectiveness of different entropy estimation techniques.

DISCUSSION, the discussion section, interprets the findings considering the research objectives. It ties the results back to the broader theoretical and practical implications, contributing to the existing knowledge base.

FUTURE WORKS outlines potential applications, future work, and improvements based on the research findings, offering a direction for subsequent research in the field.

The final chapter, CONCLUSION, concludes the thesis, summarizing the key findings, their implications, the limitations of the study, and provides suggestions for future research in this area.

2 LITERATURE REVIEW

The literature review in this paper offers an examination of the intersection of information theory and language processing, with a particular focus on entropy estimation in the context of noisy language data. It begins with the theoretical foundations, where the basic principles of information and communication are outlined, followed by a historical overview of the field, and the role of information theory in language processing. The next section delves into information theory, discussing the foundations, the evolution, and the role of entropy. Following this, the review explores language information, discussing its quantification, the handling of noisy language data, and challenges involved in processing language information. The concept of information entropy is then thoroughly examined, including its calculation, estimation, applications, and limitations. This paves the way for the review to explore the use of compression algorithms in entropy estimation. The literature review concludes with an overview of current uses, limitations, and future directions in the estimation of language entropy.

2.1 Theoretical Foundations

2.1.1 Basic Principles of Information and Communication

The foundational principles of information and communication can be traced back to the work of Claude Shannon, who transformed our understanding of these domains. Shannon articulated the concept of information as a quantifiable entity, introducing a measure known as entropy which quantifies the uncertainty or randomness of information by calculating the average rate at which information is produced by a stochastic source of data (Shannon, 1948). This development in information theory provided a scientific method for understanding, measuring, and quantifying information. At the same time, key distinctions have been made between human and machine communication, as observed by Dretske (Dretske, 1983). He identified that while humans use knowledge and context to make sense of information, machines are essentially symbol manipulators, relying solely on input and pre-defined algorithms. This distinction in the context of this thesis

underscores the challenge of accurately estimating entropy in complex and noisy human language data, where noise can be extraneous words, misspellings, or non-standard language use.

2.1.2 Historical Overview

The early theories of communication and information, notably those of Wiener (Wiener, 1948), laid the groundwork for our modern understanding of these fields. However, it was the birth of modern information theory by Shannon that truly revolutionized the field. Shannon's work paved the way for the understanding and application of information theory in various sectors, including telecommunications, data compression, and even linguistics. With the advent of the computer age, the role of information theory has evolved, with Turing expanding the scope of the theory by introducing the concept of a universal machine that could simulate the logic of any computer algorithm (Turing, 1936). More recently, quantum information theory (Nielsen & Chuang, 2010) has emerged, adding another layer of complexity to the field by considering quantum systems. Each of these stages of evolution in information theory has been driven by the need to understand and manage the increasing complexity and volume of data. This need has been further amplified in the age of 'Big Data' and sophisticated machine-learning techniques (Mayer-Schönberger & Cukier, 2013).

2.1.3 Role of information Theory in Language Processing

The application of information theory to language processing has long been recognized, with Feder, Merhav, & Gutman stressing the relevance of information theory for understanding human language processing (Feder et al., 1992). Zipf's word frequency law, which states that the frequency of any word is inversely proportional to its rank in the frequency table, marked a significant shift in how researchers approached linguistic data and is one of the earliest applications of information theory in language processing (Piantadosi, 2014). Modern applications in computational linguistics (Daniel Jurafsky & James H. Martin, 2023) have further extended the scope of information theory in language processing.

The advent of deep learning and NLP applications, such as Transformers and GPT-3 (T. B. Brown et al., 2020), have brought new challenges and opportunities. A majority of tokens in these large and diverse training datasets, often used for these models, come from CommonCrawl (Zhao et al., 2023), containing significant amounts of noisy language data. They also often contain a significant amount of noisy language data, such as non-standard abbreviations, emojis, or code-switching events common in online communication (Eisenstein, 2013). Consequently, understanding and estimating entropy in such noisy data becomes crucial for improving the accuracy and effectiveness of these models.

This aims to address the challenge of estimating entropy in noisy language data derived from social media and computer languages seeks to build on the foundational principles of information and communication. This approach could provide valuable insights for the development of more effective preprocessing techniques and improve the performance of AI models and machine learning pipelines.

2.2 Information Theory

2.2.1 Foundations of Information Theory

Information Theory, pioneered by Claude E. Shannon, revolutionized the scientific understanding of how information can be quantified, stored, and transmitted. In his groundbreaking work, "A Mathematical Theory of Communication", Shannon established the principle that information is a measure of one's freedom of choice when selecting a message. In other words, the more possible messages there are, the more information is required to choose one.

Shannon's theory was novel in that it turned information into a quantifiable entity, which was a significant departure from prior qualitative interpretations of information. It allowed for a standardized method of measuring information based on probability and statistics, thereby providing the foundation for the subsequent development of digital communication and data storage systems. (Shannon, 1948)

The concept of entropy is central to Shannon's Information Theory. As a measure of uncertainty inherent in a set of probabilities, entropy quantifies the expected value of the information contained in a message. The higher the entropy, the more uncertain or surprising the information is, and thus the more information the message carries. This conceptualization allowed for a quantifiable way of assessing and comparing information sources in terms of their predictability and informational richness.

In practical terms, for a coin toss, where the probability of heads or tails is equal, the entropy is high because the outcome is uncertain. But for a biased coin, where the chance of one outcome is higher, the entropy is lower. Entropy thus quantifies the expected value of the information contained in a message.

This was a breakthrough because it provided a statistical basis for understanding information and established that information could be processed and managed in much the same way as any physical entity. Moreover, the definition of information in terms of entropy introduced a way to quantify the efficiency of a communication channel and the maximum rate at which information can be reliably transmitted, known as the channel capacity. (Shannon, 1948)

Furthermore, Shannon's theory introduced the idea of a binary digit or 'bit' as the fundamental unit of information. He showed that any analog message can be converted into (encoded as) a string of bits, transmitted over a digital communication channel with greater efficiency and error correction capabilities, and then reconverted back into (decoded as) the original message. This digital revolution brought by Shannon's Information Theory has far-reaching implications, underpinning the operation of modern computers and the internet.

2.2.2 Shannon's Information Theory

While Shannon's theory of information was initially developed to improve telecommunication efficiency, its implications transcend this original application, providing a framework to understand data transfer in complex systems (Cover et al., 2006). A core aspect of Shannon's theory is the concept of entropy, a measure of the uncertainty or randomness of information. However, this is more

than just a statistical metric. When framed in terms of entropy, the process of communication, whether in data transmission or linguistic expression, can be viewed as an effort to reduce uncertainty or, in other words, to minimize entropy.

Redundancy, another key concept in Shannon's Information Theory, measures the portion of a message that isn't unique or, in other words, the degree of predictability within the information. This is calculated using a formula that compares the actual entropy of a message (**H(X**)) to the maximum possible entropy (**log |X|**), where **|X|** is the size of the alphabet. A concept essential in understanding language and communication systems' robustness (Cover et al., 2006).

2.2.3 Evolution and Extensions of Information Theory

The concepts and principles established by Shannon have been continuously expanded upon. A significant extension is the Kullback-Leibler divergence (or relative entropy), a measure of the dissimilarity between two probability distributions. It provides a method to quantify the inefficiency of assuming a distribution Q when the true distribution is P (Kullback & Leibler, 1951).

MacKay emphasizes the increasing relevance of information theory in data science and machine learning, for example, in developing decision tree algorithms, where entropy measures can inform optimal branching decisions. Furthermore, information theoretic concepts are employed for feature selection, clustering, and regularization in machine learning models (Mackay, 1995).

A novel frontier in this area is the Quantum Information Theory, which introduces quantum bits (qubits) instead of traditional bits, providing new methods to handle information. Unlike classical bits, which can only exist in a state of 0 or 1, quantum bits or 'qubits' can exist in multiple states at once due to a property known as superposition. This allows quantum systems to process a higher volume of information, revolutionizing the way we handle and transmit data. While classical entropy measures such as Shannon entropy remain relevant, novel entropy measures like Von Neumann entropy come into play in this context (Nielsen & Chuang, 2010).

The attempt is to understand the nuances of entropy estimation better and develop more accurate and efficient methods, contributing to the ongoing development of information theory and its applications in language processing.

2.3 Language Information

2.3.1 Linguistic Theory in Information Processing

The cornerstone of our understanding of linguistic theory is Noam Chomsky's work on transformational-generative grammar, which postulates that all human languages share a common, innate syntactic structure. Key to his theory are the concepts of 'deep structure' (the underlying syntax) and 'surface structure' (the actual spoken or written output), implying that different languages may have different surface structures but share similar deep structures (Chomsky, 1957). Chomsky’s universal grammar theory provided a foundation for analyzing language data and for the construction of computational models capable of understanding and generating human language.

John R. Pierce extended Shannon's principles of information theory to language data, emphasizing the probabilities of certain linguistic structures over others (John R. Pierce, 1980). Chomsky’s universal grammar theory established the foundation for analysing language data but lead the development of computational models that can understand and generate human language.

2.3.2 Quantifying Information in Language

Quantifying information in language has been a significant focus of research. Shannon, in his pioneering work, proposed methods for estimating information content in human language by considering the probabilities of sequences of characters or words, essentially laying the ground for N-gram models in computational linguistics (Shannon, 1948, 1951).

This approach was extended by Piantadosi, who compared quantities of information across different languages, finding remarkable consistency in the rate of information transmission (Piantadosi, 2014; Piantadosi et al., 2011). This

discovery underpins many modern statistical language models, which attempt to predict the likelihood of a word or a sentence based on previous context (Goodman, 2001; Rosenfeld, 2000).

2.3.3 Noisy Language Information

Language data is often noisy, especially in the context of social media or informal communication. The irregularities, use of slang, abbreviations, typographical errors, and other types of non-standard usage contribute to this noise (Baldwin et al., 2013; Eisenstein, 2013).

The literature offers various techniques for handling noisy language data. Preprocessing steps like text normalization and error correction, as well as developing robust language models that can handle such anomalies, have been a focus of research (Beaufort et al., 2010; Han & Baldwin, 2011).

However, the presence of noise adds a layer of complexity to the calculation of information entropy. The irregularities introduced by noise increase the unpredictability of language data, leading to a broader distribution of probabilities and, consequently, higher entropy (Cover et al., 2006). It presents challenges in accurately estimating the probabilities of linguistic structures, which is critical to the process of quantifying information in language.

2.3.4 Challenges in Language Information Processing

There are inherent difficulties in language information processing, including issues of ambiguity, complexity, and diversity of languages (Daniel Jurafsky & James H. Martin, 2023). These challenges are compounded in the context of noisy and social media language data, which often includes features like code-switching, regional dialects, and emergent language trends (Eisenstein, 2013).

Additionally, there is the problem of handling non-standard language forms such as slang, abbreviations, and code-switching (the use of multiple languages within a single conversation). These language features, which often defy traditional grammatical rules, are common in noisy data and present unique challenges that require specific methodologies for effective processing (Baldwin et al., 2013).

These challenges underlie the need for continuous advancements in language information processing. In line with the research objective, understanding these issues will help in designing robust computational models capable of handling the complexity and diversity of language information, especially in noisy environments.

2.4 Information Entropy

2.4.1 Concept and Calculation of Information Entropy

The concept of information entropy, also known as Shannon entropy after its developer Claude Shannon, builds upon the idea of information content (Shannon, 1948). Information content, **I(x)**, of an event **x** is defined by the formula **I(x)** = **-log P(x)**, where **P(x)** is the probability of the event. This measures the amount of 'news' or 'surprise' that the event's occurrence brings about, with less probable events delivering more news (Cover et al., 2006).

Information entropy, then, is the expected value of the information content across all possible events. Mathematically, entropy **H(X)** is represented as **H(X)** = **-** $\sum$ **P(x) log P(x)**, effectively summing the information content of each event, weighted by its probability. The logarithm base is often set as 2, which makes the resulting entropy units as bits (Cover et al., 2006).

This measure provides a quantitative understanding of the amount of uncertainty, randomness, or disorder within a set of probabilities. It represents the average amount of information that the occurrence of an event would embody, thereby enabling a more structured understanding of information transfer (Shannon, 1948).

However, while the concept of entropy and its calculation appear straightforward, obtaining accurate probability distributions for real-world data to accurately calculate information content and, subsequently, entropy, is not always feasible or straightforward, introducing challenges in its implementation (Cover et al., 2006).

2.4.2 Entropy estimation

The most straightforward way to estimate entropy is the plug-in method, where the entropy is calculated directly from the observed probabilities. However, this method can lead to biased estimates, especially for smaller datasets or when the data doesn't cover the entire probability space (Silva, 2018).

Owing to the need for accurate probability distributions, various techniques and methodologies have been proposed to estimate entropy. A notable among these are the nonparametric methods, which do not make any assumptions about the underlying probability distribution and rely on direct observations from the data, and have proven particularly effective for entropy estimation (Paninski, 2003).

A landmark study by (Benveniste et al., 1998) demonstrated nonparametric entropy estimation on English text, quantifying its inherent unpredictability and thus, extending its utility in the field of natural language processing. The role of entropy estimation in language is significant, offering a quantitative means to examine the complexity of linguistic structures, and to analyze and compare the information content of diverse languages.

2.4.3 Noisy entropy estimation

Estimating entropy in the presence of noise is particularly crucial in fields such as machine learning and signal processing, where data is often collected from real-world, noisy environments.

Noise within a dataset can significantly distort entropy estimates, as it introduces additional randomness that broadens the probability distribution, which in turn can lead to a higher estimate of entropy (van Erven Peter Harremoës, 2007). This challenge led researchers to develop methods for estimating entropy in the presence of noise. Techniques like the one developed by Nemenman et al. attempt to correct the distortion by reducing the impact of noise and compensating for it in the estimation process, which further enhances the application of entropy in real-world, noisy data scenarios (Nemenman et al., 2002).

2.4.4 Applications of Information Entropy

Information entropy has broad applications in multiple fields. For instance, computational linguistics is employed in data compression algorithms to represent data using fewer bits, thereby optimizing storage or transmission needs (Manning & Schiitze, 1999).

In language modelling, for example, entropy is used to evaluate how well a model predicts a sequence of words. Lower entropy means the model is more accurate in its predictions. Similarly, in text summarization, entropy measures can be used to identify the most informative sentences to include in the summary, making the summary more representative of the original text.

The principle of information entropy has also been applied innovatively in areas like text summarization and topic modelling. Hashimoto et al. for instance, used entropy measures to detect topics, providing a more systematic approach to active learning in systematic reviews (Hashimoto et al., 2016).

2.4.5 Limitations of Information Entropy

Despite the wide array of applications and its foundational role in understanding information, information entropy does have certain limitations. E. T. Jaynes highlighted the challenges associated with interpreting entropy, pointing out that it quantifies uncertainty, not the complexity inherent to the data (Rosenkrantz, 1989).

Additionally, entropy's applicability is limited when dealing with real-world data, particularly noisy, incomplete, or high-dimensional data, due to the inherent challenges in accurately estimating the necessary probability distributions. This research aims to address these challenges by exploring methodologies for effectively applying and interpreting information entropy in the context of noisy language data.

2.5 Compression Algorithms for Entropy Estimation

2.5.1 Introduction to Compression Algorithms

Data compression algorithms, fundamental to efficiently storing and transmitting data, aim to reduce the bit representation of the source data without significantly distorting the information they carry. Compression algorithms achieve their goal by identifying and eliminating redundancy in the data. This is done by replacing repeated occurrences of data with references to a single copy, thus reducing the overall size of the data (Salomon & Motta, 2010).

These algorithms can be broadly categorised into two classes: lossless and lossy. While lossless compression algorithms ensure that original data can be perfectly reconstructed from the compressed data, lossy compression allows for some information loss in exchange for higher compression ratios.

Shannon's source coding theorem is central to the intersection of information theory and data compression. It proposes that the entropy of a data source sets a lower limit on the average length of a compressed version of the data, which implies that the efficiency of a compression scheme cannot surpass the entropy of the data source (Shannon, 1948). Therefore, entropy serves as a theoretical benchmark for the best achievable compression rate (Cover et al., 2006).

2.5.2 Use of Compression Algorithms in Entropy Estimation

In the quest to estimate entropy, compression algorithms have served as practical tools due to their innate association with the concept of information density. Pioneering work by Ziv and Lempel (1977; 1978) laid the groundwork for a wide variety of lossless compression algorithms, including the well-known LZ77 and LZ78. LZ77 and LZ78, are dictionary-based compression methods that replace repeated occurrences of data with references to a single copy. These algorithms build a dictionary of data segments during the compression process, which are then referenced in the output instead of the original data. These algorithms have been used to approximate the entropy of the data being compressed by associating high compressibility with low entropy and vice versa. (Ziv & Lempel, 1977, 1978)

Several researchers have since leveraged these ideas to create more nuanced entropy estimation methods. For instance, Gao et al. suggested an entropy rate estimator utilising LZ78 compression on data partitioned with a sliding window, which yielded improved performance on non-stationary data sources (Gao et al., 2008). Non-stationary data sources refer to data sets wherein statistical properties change over time, adding an additional layer of complexity to entropy estimation.

On another front, Teahan and Cleary presented a PPM entropy estimator using a variable n-gram model and Markov chains (Teahan & Cleary, 1996). An n-gram model is a type of probabilistic language model used for predicting the next item (e.g., word, character, etc.) in a sequence based on the history of previous items. A variable n-gram model can adjust the 'n' to consider different lengths of history, providing flexibility to adapt to the data at hand. Markov chains are mathematical models that describe a sequence of possible events in which the probability of each event depends only on the state attained in the previous event. They are often used in probabilistic models, like those used in compression algorithms or entropy estimation. By incorporating the context of the data, this approach yields a promising avenue for better entropy estimation, especially for inherently context-sensitive data like natural language.

2.5.3 Comparison of Compression Algorithms

Differing compression algorithms come with varying degrees of effectiveness when applied to entropy estimation. Bell, Witten, and Cleary's comparative study (1990) across various compression algorithms (such as Huffman coding, arithmetic coding, LZ77, and LZ78) indicated the presence of trade-offs in computational complexity and entropy estimation accuracy (Bell et al., 1989).

When it comes to entropy estimation, distinct advantages and drawbacks are associated with specific compression algorithms. Witten, Neal, and Cleary demonstrated that while LZ algorithms are laudable for their simplicity and low computational overhead, they may fall short in terms of precision when estimating entropy, particularly for more complex data sources. Thus, careful selection of the compression algorithm, considering the data characteristics and the specific

needs of the entropy estimation task, becomes of paramount importance. For instance, if computational efficiency is a priority, LZ algorithms may suffice. However, for tasks demanding higher precision in entropy estimation, more complex methods like the PPM-based approach might be more suitable. PPM, or Prediction by Partial Matching, is an adaptive statistical data compression technique that operates by maintaining a dynamic model of the data it has seen so far, and using this model to predict what will come next. The prediction is then used to select the most efficient way to represent the upcoming data. In the context of entropy estimation, it can provide a more precise estimation, especially when used with more complex data sets (Witten et al., 1987).

2.6 Previous estimation of Language Entropy

2.6.1 Current Use Cases of Entropy Estimation and Language Modelling

Language entropy estimation has been employed in a myriad of contexts over the past decades, with different approaches yielding diverse estimates.

Shannon's pioneering human prediction experiments, where subjects predicted successive characters in English text, resulted in entropy estimates ranging from 1.3 to 4.2 bits per character (Shannon, 1951). This foundational work proved instrumental in establishing the use of entropy in language modelling.

Cover and King took a gambling approach to the task, leading to slightly lower entropy estimates between 1.25 and 1.35 bits per character. This approach models the prediction task as a betting game, where optimal betting strategy equates to the minimization of prediction entropy. The primary aim of their study was to align the theoretical concepts of information theory with the practical aspects of English language prediction (Cover & King, 1978).

The application of data compression algorithms to entropy estimation has been significantly influential in the field. Notably, Teahan and Cleary applied Prediction by Partial Matching (PPM) compression techniques to language modelling and

yielded an entropy estimate of 1.46 bits per character (Teahan & Cleary, 1996). Similarly, Schürmann and Grassberger utilized compression algorithms to achieve entropy estimates ranging from 1.25 to 1.7 bits per character (Schürmann & Grassberger, 1996).

Building upon these techniques, Kontoyiannis utilized the string-matching technique to estimate entropy, obtaining a value of 1.77 bits per character (Benveniste et al., 1998). Meanwhile, Brown et al. exploited the Trigram model, is a type of n-gram model where sequences of three words are used to predict the next word in a text, to estimate entropy, resulting in a value of 1.75 bits per character (P. E. Brown et al., 1992).

Turning to word-level entropy estimation, Montemurro and Pury, and Bentz et al. (2017) employed entropy estimation techniques on unigrams, yielding entropy values of 6-7 bits per word and 7-9 bits per word, respectively (Bentz et al., 2017; Bentz & Alikaniotis, 2016; Montemurro & Zanette, 2011). Bentz et al. furthered their research by applying data compression techniques which yielded a lower entropy value of 4-5 bits per word.

Bentz et al. employed advanced methodologies to estimate both unigram entropies and the entropy rate per word across an extensive range of texts and languages. The authors explained unigram entropy as the average information content of words, assuming independence from co-text. In contrast, the entropy rate, assuming dependence on an adequately long preceding co-text, is seen as the average information content of words under certain conditions.

Through their analysis, Bentz et al. observed that unigram entropies across languages displayed a unimodal distribution with a mean of approximately nine bits/word, and a standard deviation of approximately 1 bit/word. Entropy rates had a lower mean, hovering around 6 bits/word, with a similar standard deviation of approximately 1 bit/word. The authors proposed that these results demonstrate the presence of robust pressures keeping the majority of languages within a relatively narrow entropy range, particularly evident in the difference between unigram entropy and entropy rate.

Moreover, Bentz et al. identified a strong positive linear relationship between unigram entropies and entropy rates. The authors formulated a simple linear model to predict the entropy rate of a text from the unigram entropy of the same text. This relationship implies that the reduction in uncertainty by co-textual information is approximately linear across languages worldwide.

The researchers also emphasized that higher or lower word entropy doesn't necessarily equate to better or worse communication systems. In real-world scenarios, multiple contextual cues contribute to deriving meaning, many of which may not be present in written language. Nonetheless, in information-theoretic models, the entropy of a symbol inventory represents the upper bound on mutual information between symbols and meanings. Hence, the entropy of words can serve as the upper limit of expressivity in an information-theoretic sense.

Finally, Bentz et al. argued that word entropies across global languages reflect a trade-off between word learnability and word expressivity. Therefore, understanding and modelling the differences and similarities in the information words can carry is a critical undertaking in language sciences. This expansive understanding of entropy estimation illuminates the complexity and diversity of the methods utilized, underscoring the thesis's primary objective – further exploration and refinement in estimating language entropy.

2.6.2 Identified Limitations and Gaps in Current Approaches

A significant limitation of these methods is that they might not fully capture the complexity of natural languages, especially when applied to noisy data. This underlines the need for robust entropy estimators that can handle noise effectively.

The range of entropy estimates varies considerably across different methods, indicating the absence of a universal approach to entropy estimation in language data.

The extant methods for estimating language entropy, while providing pivotal insights into the domain, exhibit certain limitations and gaps that warrant attention.

A common limitation in many approaches is the reliance on heavy preprocessing of language data. While preprocessing aids in reducing complexity, it also potentially discards relevant information, undermining the completeness of the entropy estimate(Feldman & Sanger, 2007). Discarding information can impact the reliability of the entropy estimate as it might remove data that could have been informative in predicting the next word or character.

Many of the methods operate under constraints of limited vocabulary size, thereby inadvertently oversimplifying the richness of the language (P. E. Brown et al., 1992; Shannon, 1951). Moreover, the alphabet size is often confined to 27 characters, corresponding to the English alphabet plus a space. This narrowly defined alphabet size does not encompass the vast variety of characters present in other languages or even in complex forms of English data such as emojis, special characters, and punctuation. The limited alphabet size could result in oversimplified entropy estimates, failing to capture the complexity and diversity of real-world language data.

A significant constraint pertains to the limited ability of current methods to capture the full complexity of natural languages, especially in noisy data. As Eisenstein and Baldwin et al. highlighted, the distinct characteristics of noisy language data introduce unique challenges for language modelling and entropy estimation. This underscores the need for robust entropy estimators that can effectively handle noisy data, a theme that this thesis intends to explore (Baldwin et al., 2013; Eisenstein, 2013; Han & Baldwin, 2011).

Furthermore, the wide range of entropy estimates resulting from different methods indicates a lack of a universally applicable approach to entropy estimation in language data (Bentz et al., 2017; Montemurro & Zanette, 2011; Shannon, 1951). These discrepancies underscore the complexity of the task and the sensitivity of entropy estimates to the methods and models applied.

Addressing these limitations forms an integral part of the current state of the art: to develop a robust, comprehensive, and universally applicable approach for entropy estimation in language data, particularly in the context of noisy data. A universally applicable approach would be a method that is equally effective for estimating language entropy across diverse types of data and is not overly influenced by specific characteristics of the data.

2.6.3 Future Directions in Language Entropy Estimation

The field of language entropy estimation is a vibrant, evolving domain brimming with opportunities for further exploration and advancement.

One prominent direction is the development of entropy estimation techniques that can handle complex and noisy language datasets. Current techniques exhibit limitations when dealing with such data, leaving room for the creation of more robust and versatile methods (Cover et al., 2006).

The integration of advanced machine learning and artificial intelligence methods in entropy estimation represents another promising trajectory. MacKay highlights the potential of these methods in enhancing the accuracy and reliability of entropy estimates. These technologies, characterized by their ability to learn from and make decisions based on data, offer opportunities to build more adaptive and efficient entropy estimation models that can better handle the complexity and diversity of language data (Mackay, 1995).

Quantum information theory also presents an intriguing frontier for language entropy estimation. Nielsen and Chuang discuss the relevance of quantum theory in understanding information and entropy, opening potential avenues for applying these principles to language data. Quantum information theory offers a different lens through which to view entropy, possibly providing novel insights into its estimation in language (Nielsen & Chuang, 2010). In essence, quantum information theory could introduce new concepts of entropy that go beyond classical probabilistic notions, offering novel insights into its estimation in language.

Given the dynamic and evolving nature of languages, considering linguistic diversity and language evolution in entropy estimation could yield a more comprehensive and accurate understanding of language entropy. Eisenstein emphasizes the need to account for these factors in linguistic analysis and entropy estimation. Language diversity reflects the variations in syntactic, semantic, and pragmatic aspects of different languages, while language evolution captures how languages change over time (Bentz et al., 2016; Eisenstein, 2013). Both these factors could introduce additional complexity into language data, which could impact entropy estimates.

Finally, the importance of interdisciplinary approaches in language entropy estimation cannot be overstated. Integrating insights from various fields such as linguistics, information theory, computer science, and data science could enable a more holistic understanding of language entropy (Daniel Jurafsky & James H. Martin, 2023). This approach aligns with the thesis's objective of developing a method that offers robust entropy estimates and integrates knowledge across these diverse domains to contribute to a comprehensive understanding of language information.

Each of these areas underscores the dynamic nature of entropy estimation and the room for innovation in this field. In this context, the present thesis aims to contribute to these advancements, specifically focusing on improving entropy estimation in noisy language data, a challenging yet crucial domain in our increasingly data-driven world.

3 METHODOLOGY

3.1 Data Collection

The process of data collection serves as a fundamental component in the research project. The ability to accurately estimate entropy, a measure of unpredictability or randomness, heavily depends on the nature and quality of the data that is collected. This research aims to draw comparisons between spoken and programming languages, thus necessitating a systematic approach to data collection, encompassing a variety of languages, and structured around key representative sources.

3.1.1 Language Selection

The choice of languages to be analysed is guided by the intention to capture a representative mix of global communication systems. The chosen spoken languages for this research are English, Spanish, and French. These languages were selected based on their widespread usage across the world, which offers a broad and diverse range of data for analysis. English is an extensively used global lingua franca, Spanish has vast native speakers, especially in the Americas, and French is prevalent in various parts of Europe and Africa (Crystal, 2003). The inherent differences in syntax, structure, and vocabulary among languages can potentially impact their respective entropy values, making these aspects crucial in our comparative analysis.

According to Alshaabi et al. English is the most used language on Twitter, making it an obvious choice for the research. Its widespread usage on the platform ensures the availability of a large and diverse dataset, thus enhancing the robustness of our entropy estimation.

Japanese is the second most-used language on Twitter. However, it was not included in the research study due to its unique non-Latin character set, which would present challenges in the entropy estimation methodology. The use of non-Latin characters in Japanese would necessitate a distinct analytical approach that diverges from the research's systemic comparison scope. A separate study

analyzing Japanese or other non-Latin character language and its entropy could provide intriguing insights and could be an interesting avenue for future research.

Spanish, being the third most-used language on Twitter, is included in the research study. It contributes a significant volume of data, while also providing a contrast to English in terms of language structure, idiomatic usage, and regional variances.

French, despite being less represented globally and not being among the top languages used on Twitter, is included in the study. It offers a different grammatical structure and linguistic idiosyncrasies compared to English and Spanish. Despite the lower volume of data compared to English and Spanish, it serves as an important part of the analysis by adding a different dimension to the research (Alshaabi et al., 2021). French is also the native language of the researcher making it an interesting analysis with additional background information and interests.

In the domain of programming languages, the focus is on C++, Java, and Python. These languages were chosen due to their popularity and extensive usage in various application domains, ranging from system software to web applications and data analysis (TIOBE, 2023). Moreover, these languages each offer unique syntactic and structural characteristics. C++ is a language with a high degree of flexibility and complexity, Java promotes object-oriented programming paradigms, and Python is known for its simplicity and readability.

3.1.2 Data Size and Noise

The research aims to collect a large dataset, measured in gigabytes (Gb), for each language to ensure a systematic and comprehensive analysis. The goal is to capture a broad spectrum of language use cases, which will reduce bias and improve the reliability of the entropy estimates. The larger the dataset for each language, the more reliable the entropy estimates are likely to be, as larger datasets can better represent the full range of linguistic use cases.

It's crucial to acknowledge the inherent 'noise' in the collected data. Noise here refers to outliers, anomalies, or irrelevant information in the data. In social media

data, noise could be typos, slang, or non-standard abbreviations. In programming languages, noise could arise from coding errors or unconventional usage of language syntax. Despite the potential challenges it could present in data preprocessing and analysis, noise is an important aspect of 'real-world' language usage and will be considered as an integral part of our data, rather than something to be eliminated.

3.1.3 Social media dataset

3.1.3.1 Twitter

Choosing a suitable platform for the data collection process is crucial to the research project. In this study, Twitter has been chosen as the primary data source due to several reasons. Twitter is a social networking site where users tweet short messages; its data is easily accessible and presents a large variety of user-generated content. Automated content and bot usage are also inherent and present in the collected data, especially in the more recent data collection.

This large and diverse user base enables the collection of a broad range of language samples. Twitter data not only encompasses the linguistic trends of millions of users globally, but it also reflects the dynamism of various socioeconomic, age, and professional groups. Furthermore, Twitter's constraint—tweets limited to a specific number of characters—promotes a unique language style that often includes abbreviations, acronyms, and inventive syntax. This noise is a critical component of natural language that the research aims to capture and analyze, as it could significantly influence entropy calculations.

Twitter's history and its role as a communication tool are also pertinent to this study. Since its creation in 2006, Twitter has grown exponentially and significantly influenced global society and businesses. It has become an essential tool for political discourse, a platform for breaking news, and even a medium for official statements by government entities (Tufekci, 2017). These aspects of Twitter contribute to the richness and relevance of the data it generates, further justifying its choice as a primary data source.

Given the focus on data collection from Twitter, it is essential to acknowledge the ethical implications of this approach. This research strictly adheres to Twitter's policy on data collection, respects user privacy, and only collects publicly available data. In all instances, the data was used responsibly, maintaining anonymization and privacy standards, in line with the guidelines of responsible research.

Twitter's TOS explicitly permits the collection of public data by approved methods, which are the methods we used in the research. Importantly, Twitter's Developer Agreement and Policy stipulates that data collected from Twitter cannot be used to derive or infer potentially sensitive attributes about Twitter users.

This research respects these provisions by ensuring that the collected data does not violate users' privacy. All collected tweets are anonymized and any personally identifiable information is strictly excluded from the dataset. Instead, the collected data is identified solely by the tweet ID in order to maintain user anonymity. This approach ensures compliance with Twitter's TOS and promotes the ethical use of social media data in academic research.

As an additional ethical measure, this research project has undergone review and received approval from the institutional review board to ensure that all procedures are in line with best practices for academic research.

3.1.3.2 Objectives

The data collection objectives align directly with the broader research goal: estimating entropy across distinct languages. A primary aim is to gather a diverse and voluminous dataset covering the three selected languages – English, Spanish, and French. This diversity not only enriches the collected data but also ensures the accurate representation of each language's unique structural attributes, thus providing a reliable entropy estimation.

In this research, attention is specifically devoted to these three languages. Each language, chosen based on criteria detailed earlier in section 3.1.1, introduces distinct linguistic characteristics and usage patterns.

Refining the methodology further, a three-tiered data collection strategy is employed, with each tier catering to specific research needs:

Firstly, Sample Streaming provides a panoramic perspective on global language structures by capturing 1% of all real-time tweets on Twitter. This method serves as an essential reference for understanding language structure and vocabulary on the platform, offering a broad view of language usage given its non-targeted nature.

Secondly, the Targeted Accounts/Topics strategy brings a focused approach into play. By collecting tweets from specific accounts like news outlets, renowned personalities, and government entities, or from topics such as the Ukraine war, this approach facilitates a deep dive into contextual variations in language use and the consequent impact on entropy.

Finally, the COVID-19 Timeline method adds a temporal dimension. By assembling COVID-19-related tweets over several years, this approach enables an exploration of the time effect on this specific global event's language use. It provides a unique opportunity to study the entropy trend on a globally significant topic, uncovering how language evolves during a crisis and its effects on entropy.

Each of these data collection methods offers unique insights into language entropy, providing different lenses to examine language usage on Twitter.

3.1.3.3 Structure

The research employs a multiple approach to data acquisition, primarily focusing on Twitter as the primary data source. The Python-based structure of the code is specifically designed to extract, manipulate and store data for subsequent analysis. This methodology consists of distinct stages, including data scraping, streaming, hydration and dehydration of tweets, as well as specialized procedures for collecting Covid-19 related data.

To begin, the data scraping functionality serves as the center of the data acquisition procedure. Using Python as the base language, this functionality is

applied by executing the file *run-scraping.py*. Several arguments can be passed into this script to configure the search type, environment, date range, and more.

In addition to data scraping, a real-time data-gathering approach known as streaming is also implemented. The *sample-stream.py* script carries out this task, providing access to a continuous stream of current tweets on Twitter, approximating 1% of the total volume. The parameters for language and the maximum number of iterations for each language are adjustable.

Specific to the Covid-19 pandemic, data is collected from a dedicated Covid-19 Github repository. The *scrape-covid-github.py* script retrieves this data and rehydrates the tweets.

3.1.3.4 Web Scraping

Web scraping, in essence, is a method used to extract data from websites when no API is available, or when the provided API proves insufficient for the required data needs. Web scraping is a used approach for digital data collection in contemporary research. It involves programmatically interacting with websites or applications to extract large volumes of information that can then be used in various types of analysis. However, web scraping must be conducted ethically, ensuring respect for data usage boundaries, privacy norms, and protection of sensitive data. These considerations guided the development of the web scraping strategy.

3.1.3.4.1 Twitter as Data Source

The nature of Twitter data makes it a valuable resource for research. The diverse user base provides a wealth of information expressed through tweets, retweets, and likes. This research will primarily focus on tweets, given their direct representation of user expression and sentiment. Given the unique characteristics and structure of this social media platform, it was imperative to develop a web scraping procedure to harvest the necessary data efficiently and effectively.

Over time, Twitter has seen considerable transformations, most notably after being acquired by Elon Musk. This change disrupted the operational capacity of

many libraries previously used for web scraping on the platform. The front end had many changes, and the API became more restricted to the users on the platform. This change occurs due to the increased number of companies collecting user information and data on Twitter to train Large Scale Artificial Intelligence Models.

The decision to implement a scraping strategy, as opposed to using Twitter's API, primarily stemmed from the recent changes imposed by the platform. Twitter has aggressively rate-limited its API usage, making data requests significantly more restrictive. Further, accessing historical data older than seven days through the API is prohibited without special permissions. Consequently, scraping emerged as a crucial strategy as it empowers the research to acquire vast quantities of data from historical sources, almost without limitations. This approach ensures comprehensive coverage of the data, vital for the in-depth language entropy analysis conducted in this study.

3.1.3.4.2 Python and Selenium

Python was selected as the primary language for data collection, with Selenium as the main scraping library. Selenium is an open-source web testing suite that permits automated interaction with browser actions. The library's ability to interact dynamically with webpage elements, imitating human browsing behavior, makes it perfect for extracting data from dynamic web platforms like Twitter. Selenium's ability to handle JavaScript, a dynamic content generator, is particularly beneficial given Twitter's heavily JavaScript-dependent website structure.

3.1.3.4.3 Library Development

Given the platform's updates, a custom approach was necessary for efficient data scraping as the libraries were mostly unavailable. This entailed extensive analysis of former libraries' code, modifying and adapting it to create a new toolset compatible with the updated Twitter. Libraries such as Sweet and SMMT were analyzed and used to create an entirely new and custom library that I will use to collect the data. The library is now available on GitHub along with several Medium articles to explain its behaviors and functionalities. In this iterative process,

constant testing was conducted to ensure the emerging library was appropriately equipped for the updated Twitter interface.

3.1.3.4.4 Data Collection Strategy

Twitter's updated interface poses a unique challenge: it requires users to be logged into an account to browse tweets. An account login functionality was incorporated into the scraping bot to accommodate this. Now capable of acting as a logged-in user. The bot is using advanced search to gather tweets based on specific criteria. This allowed for highly refined data collection, using specific keywords, account tags, hashtags, time frames, and language filters to focus the data collected. The result was targeted scraping from specified users and keywords, within designated timeframes, and in particular languages.

The scraping bot was programmed to meticulously scan each tweet card on the platform, extracting important data such as the username, tweet text (including characters, numbers, punctuation, emoticons, emojis, etc.), and the unique author username. To access the author id from the username, open-source website APIs were utilized (https://tweeterid.com/), requesting to gather the user id from the username. This step was indispensable considering Twitter's current structure and Terms of Service. The chosen data points for scraping - the username, tweet text, and unique author username - were chosen for their relevance to the research. Each element provides unique insights into user behavior and language use, contributing significantly to the overall analysis.

Nevertheless, Twitter's advanced search functionality limits each search to retrieve only 100 tweets. To maximize data collection over an extended period, the search interval was set to a single day, which is the minimum unit used on advanced search. This approach enabled the collection of the maximum amount of data within Twitter's imposed limits.

3.1.3.4.5 Targeted Strategy

3.1.3.4.5.1 Personalities

These personalities have been chosen as they represent key sectors of influence and diversity. Through the lens of language entropy, their tweets provide insights into the variability and information richness in different sectors.

- **@elonmusk (start: 2010-01-01):** As an iconic figure in technological innovation, Musk's tweets are often characterized by diverse topics, technical terminology, and creative language use. This makes his discourse an excellent source for studying language entropy in the context of technological discourse.
- **@billgates (start: 2009-01-01):** Gates' tweets, focused on philanthropy and entrepreneurship, contain a unique blend of business language, technical terms, and humanitarian discourse. This makes them valuable for exploring language entropy within the intersection of these domains.
- **@barackobama (start: 2007-01-01):** Obama's tweets encompass American political discourse, social issues, and international relations. His discourse's complexity and broad scope provide rich insights into language entropy in political and social contexts.
- **@emmanuelmacron (start: 2013-01-01):** Macron's tweets offer a snapshot of the language entropy in European political and social discourse. His tweets, usually in French, contribute to a multilingual analysis, further enriching the entropy quantification.
- **@sanchezcastejon (start: 2009-01-01):** Sanchez's discourse, in Spanish, allows us to explore language entropy in another language and cultural context, adding depth and diversity to the overall entropy analysis.

3.1.3.4.5.2 News Outlets

News outlets provide an understanding of how language entropy manifests in news reporting across different languages and regional contexts. The data were collected since their creation (2007-2009)

- **English Media Outlets:** The variety of topics and writing styles across these outlets **(@BBCNews**, **@CNN**, **@nytimes**, **@guardian**, **@Reuters**)

serves as a rich source for analyzing language entropy in English-language news reporting.

- **French Media Outlets:** By analyzing French news outlets (@lemondefr, @**Le_Figaro**, @**libe**, @**FRANCE24**, @**lexpress**), we can study the entropy in French news discourse, contributing to a comparative analysis of language entropy across different languages.
- **Spanish Media Outlets:** Studying Spanish outlets **(@el_pais**, @**elmundoes**, @**lavanguardia**, @**abc_es**, @**elconfidencial**) allows us to quantify language entropy in Spanish news discourse, deepening our understanding of how entropy varies across languages and cultural contexts.

3.1.3.4.5.3 Russo-Ukraine War Analysis

This aspect of the research aims to understand how language entropy manifests in crisis-related discourse and how it evolves over time.

- **Keywords Analysis:** By targeting specific keywords related to the Ukraine conflict using Chen and Ferrara research, it can focus our analysis on the most relevant discourse (Chen & Ferrara, 2022). This strategy also allows to track how the entropy associated with these keywords changes over time, reflecting shifts in the conflict and related discourse.

Given the nature of the conflict and its broad geopolitical implications, it was crucial to target and track specific keywords for a comprehensive analysis. The keywords selected were integral to the war discourse and provided a robust foundation for assessing the evolution of language entropy. These include:

- Geographic and Administrative Regions: "**Donbas**", "**Donetsk**", "**Kiev**", "**Luhansk**", "**Minsk**", "**Moscow**". These terms are central to the conflict, representing the key regions involved. Tracking these keywords reveals the complexity and variability of discussions surrounding these areas.
- State Bodies and Alliances: "**KGB**", "**FSB**", "**NATO**". References to these entities give insights into the global and political aspects of the discourse.

- Key Personalities and Countries: "**Putin**", "**Russia**", "**Zelensky**", "**Ukraine**". These keywords shed light on the role of major players in the conflict and the narrative surrounding them.
- Historical and Cultural References: "**Soviet**", "**Ukrainian**". These terms provide a historical and cultural context, enriching the analysis of language entropy.

3.1.3.4.6 Exception and Bug Handling

A crucial aspect of the bot's development was the rigorous exception and bug-handling mechanisms put in place. Provisions were made to deal with potential issues including connection problems, frontend interaction bugs, and timeouts. This comprehensive error handling made the bot highly autonomous, capable of running uninterrupted for extended periods without human intervention or debugging. Additional random delays were also put in place to avoid bot detection by the platform and avoid Twitter to disable accounts suspicious of non-human behaviours. In addition, anti-limit rate delay and checks are available in case of rate limitation from the front end.

3.1.3.4.7 Data Storage

The collected data was stored as CSV files, arranged as 'tweet_id, user_id, timestamp, text'. This straightforward format ensured that the scraped data was ready for subsequent analysis, minimizing data pre-processing requirements. The choice of CSV format for storing the scraped data was made for its ease of use and compatibility with most data analysis tools. It also ensures that the data can be easily shared and reviewed by other researchers.

3.1.3.4.8 Ethical Consideration

Respecting Twitter's terms of service and legal considerations regarding data privacy and protection was a fundamental principle during the development of the scraping process. The extensive technical work in creating a comprehensive scraping bot was conducted ethically, ensuring respect for data usage boundaries, privacy norms, and sensitive data protection. Adherence to these

principles was not just a legal necessity but a moral obligation, recognizing the rights of the users whose data was being collected.

3.1.3.5 Twitter API

API, an acronym for Application Programming Interface, is a computing interface that facilitates interactions between different software components. It defines the types of requests that can be made, how to make these requests, the data formats that should be used, and the conventions to follow. In essence, an API is a contract between different software components, detailing how software components should interact.

The Twitter API serves as a medium for applications, services, and systems to communicate with Twitter's platform. The API allows developers to access features of the platform that are essential for data collection and automation tasks, including posting tweets, reading user profile information, and even live-streaming tweets.

3.1.3.5.1 Technology Used

Python was chosen as the primary programming language due to its vast library support for API integrations. Libraries like Tweepy and Twarc, designed specifically for Twitter API interaction, were employed, along with the requests library to facilitate HTTP requests.

Twarc is a Python library and command-line tool for archiving tweets. Its usefulness stems from its ability to respect Twitter's rules for data redistribution, allowing for ethical and compliant data collection.

3.1.3.5.2 API utilisation

The Twitter API was used in two principal ways: streaming and hydrating tweets.

- **Streaming**: The Twitter API's streaming service provides a live feed of public tweets. Using specific criteria, this real-time feed can be filtered to capture only tweets that match specified keywords, languages, or from particular users. This service proved particularly useful for collecting data related to recent events or trending topics.

- **Hydrating Tweets**: Twitter's Terms of Service restrict the full distribution of tweet datasets, allowing only the sharing of tweet IDs. Hydrating these IDs refers to the process of retrieving the complete information of a tweet using its ID, ensuring compliance with Twitter's guidelines. Hydration is achieved using the Twitter API, making it possible to reconstruct complete tweet data from shared tweet ID datasets.

3.1.3.5.3 Sample Streaming

During the process of data collection, the streaming functionality of Twitter API was used to access real-time tweets. To limit the volume of data to a manageable level, the sampling feature was utilized, allowing for the collection of approximately 1% of all current tweets, a data volume generally representative of the broader Twitter activity.

The research by Jürgen Pfeffer et al. offers critical insights into the temporal reliability and coverage of data collected through Twitter's Academic API. They found that the sampling strategy provided an accurate estimation of the overall sentiment and tweets across the platform. The authors also found that one out of 16 sent tweets is not available any more after 24 hours (Urgen Pfeffer et al., 2023). This could be relevant to the entropy estimation as it provides insights into the temporal dynamics of social media. It also highlighted that this data collection method will reflect the instant tweets without this natural filtering of tweet deletion by the users. This decay should be considered an integral part of the entropy estimation process, reflecting the ephemeral nature of content in social media platforms like Twitter.

After establishing a connection to the API's endpoint, the result was filtered based on the target languages: English, French, Spanish, German, and Italian. However, due to a lesser volume of tweets in German and Italian, these datasets were not used for data analysis and entropy estimation.

The streamed data was then saved as CSV files, organized with the following fields: 'tweet_id', 'user_id', 'timestamp', 'text', 'lang'. This consistent formatting

enabled efficient processing and analysis in the subsequent stages of the research.

3.1.3.5.4 Covid-19 Dataset

In addition to the data collected through sample streaming, this research made use of a large-scale dataset specifically related to COVID-19 Twitter discussions. This dataset, compiled by Banda et al. was made available for open scientific research (Banda et al., 2020).

The dataset, which was stored in a publicly accessible repository on GitHub, was initially scraped and downloaded in a "dehydrated" form. This is in adherence to Twitter's policy where only tweet IDs can be publicly shared. Thus, the first step involved "hydrating" the tweet IDs, which meant submitting requests to the Twitter API to retrieve the full metadata associated with each tweet ID. The data retrieved included parameters such as 'author_id', 'text', 'language', and 'date', among others.

Upon hydration, the tweets underwent a filtering process similar to the one used in the sample streaming method. In this case, the tweets were filtered based on language, specifically focusing on English, French, and Spanish tweets. The filtered data was subsequently saved in a CSV format, comprising the following fields: 'tweet_id', 'user_id', 'timestamp', 'text', and 'lang'.

A crucial aspect of the data collection process was to ensure the capture of temporal dynamics in the data. To this end, the hydration, filtering, and saving process was iterated for each day since the onset of the pandemic in early 2020. This method demonstrated a chronological collection of COVID-19-specific tweets, providing a temporal perspective on the evolution of language and topics over the course of the pandemic.

3.1.3.5.5 Limitations and Exception Handling

Despite the access to real-time and historical Twitter data, the API poses several limitations that were navigated during this research. Key among these are rate limits and changes in access policies under the recent acquisition by Elon Musk.

Rate limiting is the restriction on APIs that limits the number of requests a user can make within a certain timeframe. Twitter imposes these limits to maintain the reliability of its platform and prevent abuse. This was a significant constraint during the data collection process, as intensive requests were made to hydrate the tweet IDs and retrieve tweet data. To address this limitation, a programmatic strategy was implemented wherein once the rate limit was reached, the process would be paused and resumed once the window was reset, thereby ensuring continuous data collection without violating the API's policies.

Additionally, the recent acquisition of Twitter by Elon Musk led to significant changes in API access. Most notably, Twitter ended its free API services in mid-June, which limited the amount of data collected.

Beyond the API limitations, the research employed exception handling to manage potential disruptions during the data collection process. Disconnection handling was implemented to account for any abrupt disconnections during the data streaming process, ensuring that any such event would not result in data loss. Additionally, timeout handling was built into the system to manage scenarios where a request to the API was not responded to within an expected timeframe.

3.1.3.6 Data Collection Environment

Data collection was conducted within a secure and controlled environment, ensuring data privacy and integrity throughout the process. Scripts were executed on a local server. This local setup minimized the potential for exposure to data leaks and security threats often associated with cloud-based platforms, thereby adhering to a strong commitment to the privacy of the individuals and the data involved in this research.

The server utilized for data collection was a repurposed old laptop running Ubuntu Server. Moreover, a secure shell (SSH) server was installed on this laptop, enabling secure remote command execution and data transfer within the local network.

To utilise resources efficiently and manage the vast amount of data, multiple scripts and API accounts were run in parallel (Figure 1). Four separate scripts

and Twitter accounts were deployed simultaneously across different accounts for the scraping process, operating 24/7 for a month. This setup was designed to maximize data collection within the constraints of rate limits and efficiently cover the target individuals, topics and news outlets.

With the cessation of Twitter's free API plan in mid-June, the real-time streaming collection was limited to two weeks using a single premium API account.

The collection of the COVID-19 dataset involved two parallel API accounts, both scraping and hydrating tweets. This operation spanned one and a half months, running 24/7, and gather 2 years of historical data.

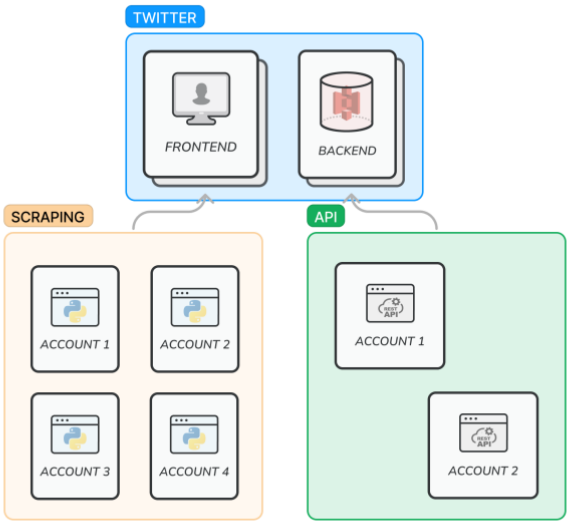

Figure 1: Data Collection Environment

3.1.4 Programming language dataset

The initial data collection strategy was to scrape public repositories from GitHub, focusing on three dominant programming languages: Python, C++, and Java, leveraging GitHub's API for efficient data gathering. However, there were significant challenges to this approach. One notable issue was licensing restrictions; not all repositories have a license that permits scraping or data collection. Furthermore, the time and computational resources required to

develop a sophisticated and performant scraping bot to detect repository licensing and determine its scrapeability were considerable.

Thus, a strategic shift in the data collection approach was executed, opting instead for the CodeNet dataset, a comprehensive, multi-purpose coded dataset by IBM. CodeNet provides an extensive collection of 14 million code samples, approximately 500 million lines of code, spanning across 50 programming languages, which are significantly more than initially targeted (Python, C++, and Java). This dataset is available under the more permissive CC-BY-SA license, alleviating the licensing concerns inherent in the GitHub scraping approach.

The CodeNet dataset is developed for the purpose of advancing AI's understanding and generation of code. It contains problem texts, solutions, and metadata, which makes it a rich resource for a variety of research tasks in the domain of code representation, code completion, code translation, and many more (project-codenet et al., 2021).

For this thesis, approximately 5GB of data per target language were used, encompassing a diverse set of coding problems and solutions. Utilizing the CodeNet dataset ensures access to high-quality, diverse, and large-scale data, necessary for entropy estimation purpose.

3.2 Pre-Processing

Pre-processing is a crucial part of this research, given the noisy nature of the data. The aim here is not to entirely eliminate this noise but rather to format the data to reduce bias during entropy estimation. This objective necessitates a light pre-processing approach rather than an extensive cleanse.

To achieve this balance between noise retention and bias minimization, certain pre-processing steps are applied. These measures are aimed at preserving the core characteristics of the data to ensure its structural and semantic integrity remains intact for analyses. Decisions on data elements to retain or discard are

guided by their relevance to the study objectives and their potential impact on entropy estimation.

3.2.1 Workflow

The pre-processing phase in this research is multifaceted and extends beyond just data cleansing. The workflow comprises several crucial steps designed to optimize the dataset for the study's objectives. First, cleaning and tokenization is carried out, which involves breaking down complex data structures into simpler, manageable units, known as tokens

Next, data management tasks are performed. This involves the creation of sub-dataframes, a strategy that enhances data accessibility and improves computational efficiency. This step also encompasses data labelling, an essential task in supervised learning techniques, and in this case, useful for categorizing and organizing data according to its source and language.

Subsequently, vocabulary files are generated. These are instrumental in maintaining a record of the unique tokens present in the dataset. Metadata is also created and managed, containing supplementary information about the dataset, including details about its collection, structure, and various attributes.

3.2.2 Natural Language Cleaning

One of the fundamental steps in pre-processing is data cleaning. This process is particularly pertinent in this study, where the aim is to eliminate all non-natural characters or tokens, and any elements specifically related to Twitter. The goal here is to minimize any potential bias in the dataset caused by these elements.

To accomplish this, the cleaning process offers several options. For instance, URLs and Twitter-specific URLs within the tweets can be eliminated, as they do not contribute to the semantic context of the message. Similarly, emojis and emoticons can also be excluded, as they might introduce a non-textual form of communication that could affect the entropy estimation.

Further, 'RT' tags denoting a retweet, and the accompanying username can also be removed. Such elements do not add to the original content of the message

and can create additional noise. Punctuation marks, another potential source of disruption, could also be eradicated.

Moreover, accents can be removed to standardize the words and to avoid the misinterpretation of identical words with different accents as distinct entities. Extra spaces can be eliminated to ensure a consistent data format, and all text can be converted to lowercase to avoid distinguishing identical words based on their case.

It's noteworthy that each of these options can profoundly affect the data's noise level. Therefore, a quantifiable evaluation of each cleaning method's effect is essential. This enables us to discern the impact of each preprocessing step on the entropy estimation.

Therefore, in the following stages, various combinations of these preprocessing methods will be applied to the entropy estimation. This allows for an assessment of the impact of elements like punctuation, accents, and emojis on the entropy in social media data, ultimately contributing to a more precise and nuanced understanding of noise in the data. It will help the research to quantify the noise and entropy that each of these preprocessing steps add or remove from the original message.

3.2.3 Programming Language cleaning

The preprocessing approach for programming languages diverges slightly from that applied to natural languages. The focus here is on identifying critical elements such as variables, functions, numbers, strings, and comments within the code. Once these tokens are identified, they are replaced with custom tokens such as #VAR#, #NUM#, #STR#, and so forth. This step facilitates a more structured analysis of the code, reducing the complexity of different languages to common, comparable tokens.

Much like with the natural language cleaning process, the aim is to quantify the impact of each token type on entropy estimation. To do this, combinations of different preprocessing techniques will be applied in an iterative process. This

allows for the assessment of how each token type's presence or absence affects the resulting entropy measures.

The process of identifying tokens is facilitated by a bespoke coding solution, which relies on keyword detection specific to each of the languages being studied: Python, Java, and C++. Tokens are identified based on their association with these keywords. This technique was selected as an initial method for token detection, due to time constraints in the project timeline.

However, it is worth acknowledging that there are potential improvements to this method. In particular, using an Abstract Syntax Tree (AST) parser would likely yield a higher token detection rate. AST parsers are more adept at interpreting the structure of source code, leading to more precise token identification.

The performance and accuracy of the current token detection methodology will be evaluated later in the analysis section, providing an opportunity to consider such enhancements for future work.

3.2.4 Tokenization

Following the cleaning stage, the next critical phase is tokenization. The purpose of tokenization is to break down the text into smaller pieces, known as tokens. For this process, the Python Library NLTK was employed, specifically leveraging its TweetTokenizer class. This class is particularly adept at handling the idiosyncrasies of tweet text, such as emoticons, hashtags, and mentions. NLTK (Natural Language Toolkit) is a leading platform for building Python programs to work with human language data, providing easy-to-use interfaces to over 50 corpora and lexical resources (Bird et al., 2009).

Tokens in this context are predominantly defined as words. However, the option is also available to consider characters as tokens. This character-level tokenization will be explored during the entropy estimation analysis. Furthermore, there's an option to generate n-gram tokens, which consider sequences of n words together. N-grams are a way of capturing language structure, such as the context or word order, to aid in the language modelling process (P. E. Brown et al., 1992).

The tokenization process produces two primary outputs: a text file containing the tokens and a text file with the vocabulary, i.e., the set of unique tokens. This process is equally applied to the programming language data. However, rather than using NLTK's TweetTokenizer, the built-in tokenizer from CodeNet was used to better handle the specifics of programming language structure.

3.2.5 Structure

For the purpose of efficiently handling various preprocessing steps, numerous scripts and files were made available in the project:

1. **Clean-code.py**: This script cleans programming language data. It accepts the input file, the language of the file, and the type of tokens to be replaced by identifiers as arguments.
2. **Clean-tweets.py**: This script handles the cleaning of tweet data. The input file and the type of preprocessing methods to be applied on the file are taken as arguments.
3. **Combine.py**: This script is used to combine multiple csv files into a single one.
4. **Combine-metadata.py**: This script is used to integrate metadata files with corresponding csv files.
5. **Combine-voca.py**: The function of this script is to merge several vocabulary files into one.
6. **Filter-language.py**: This script filters data based on language. It takes as an argument the csv file with a language column and identifies rows corresponding to one or several specific languages.
7. **Generate-metadata.py**: This script is employed to generate the metadata of a csv file.
8. **Generate-sub-df.py**: This script is used to split a large csv file into smaller ones based on a specific column given as an argument. This can be useful for segmenting data by date or language labels.

9. **Labelled-data.py**: This script adds a class column to the csv file, effectively labeling the data.

10. **Tokenize-code.py**: This script tokenizes programming language data. It accepts the size of the n-gram and a flag indicating if character-level tokenization should be used as arguments.

11. **Tokenize-tweets.py**: Similar to **Tokenize-code.py**, this script tokenizes tweet data. The arguments are the n-gram size and a flag for character-level tokenization.

12. **Tokenize-text.py**: This script tokenizes text files. It accepts the n-gram size and a flag for character-level tokenization as arguments.

These scripts allow for a comprehensive and systematic approach to the preprocessing phase, enabling fine control over how the data is cleaned, segmented, and tokenized.

3.3 Natural Language Processing

Natural Language Processing (NLP) is a subfield of linguistics, computer science, and artificial intelligence that focuses on the interaction between computers and humans in natural language. It enables computers to interpret, understand, and derive meaningful insights from human language in a valuable manner. Applications of NLP include machine translation, sentiment analysis, speech recognition, and information extraction, among others (Bird et al., 2009).

3.3.1 Objectives

One of the main objectives in this research is to effectively label the Twitter dataset with attributes such as emotion, sentiment, topic, hate speech detection, and so forth. This will not only add rich metadata to the dataset but also offer multiple angles from which to view and analyze the entropy estimations. It could reveal interesting patterns or behaviour in the noise of data when viewed through the lens of these various labels.

The labelled data can be visualized as a tree structure, with the original data forming the root, and various labels (e.g., emotion, sentiment) forming the branches and sub-branches. This hierarchical structure of data can be used to drill down and analyze the entropy at different levels.

For instance, one would expect the entropy to decrease further down the tree as the language becomes more specific and patterns more redundant, due to the narrowing of the scope with each subcategory. Thus, this classification and its effect on entropy present an intriguing aspect of noise behaviour to explore. Observing the entropy progression through this structure can offer valuable insights into the nature of noise in social media text data and its inherent properties, further fulfilling the thesis objectives.

3.3.2 Technology

For the Natural Language Processing (NLP) portion of this research, Python was chosen as the primary programming language due to its extensive support for NLP tasks and its vast range of libraries, making it a top choice for text analysis. Additionally, TweetNLP, a specialized toolset for social media and web text, was employed to enhance the effectiveness of NLP tasks on tweets (Camacho-Collados et al., 2022).

TweetNLP is a project that provides a set of tools to tackle tokenization and part-of-speech (POS) tagging specifically adapted for Twitter and other social media texts. The purpose is to deal with the linguistic irregularities found in such texts, which are not common in more formal texts and can present unique challenges for text analysis.

For this research, the NLP model targeted several labels for each tweet. These included emotions such as anger, anticipation, disgust, fear, joy, love, optimism, pessimism, sadness, surprise, trust; hate speech; irony; named entities (person, location, event, corporation, product); offensive content; sentiment (positive, neutral, negative), and topic (including a broad range from arts & culture to sports and technology, among others).

To classify these labels, pre-trained models from the Hugging Face library were used. Hugging Face is a state-of-the-art provider of transformer models for NLP tasks, offering a vast selection of pre-trained models which can be fine-tuned to specific tasks. Their models are deep learning classifiers that have been trained on extensive text corpora and can deliver high-quality predictions for a variety of NLP tasks.

For efficiency, tweets were processed in batches, with the pre-trained model classifying the labels for each tweet in the batch. This batch processing methodology enhances computational efficiency and expedites the overall NLP process.

3.3.3 Environment

The computational requirements for deep learning tasks such as the one outlined in this research are significant. To handle this, Kaggle's GPU environment was leveraged, a cloud-based platform that offers access to powerful computing resources for machine learning and data science tasks.

Kaggle provides access to NVIDIA Tesla T4 GPUs, which are designed specifically for machine learning and data analysis tasks. For this study, two T4 GPUs were utilized concurrently.

Running the deep learning detection models on millions of tweets, with such substantial computational power, allowed for swift processing times. Nevertheless, due to the vast amount of data, the entire process still took approximately one and a half weeks to complete.

3.4 Challenges of Big Data

The nature and volume of the data handled in this study qualify it as a Big Data challenge, defined by the four 'Vs': Volume, Velocity, Variety, and Veracity (Smaya, 2022). The dataset in question for this study falls under this category, consisting of several gigabytes of CSV files containing millions of tweets. Approximately 300 million tweets, equivalent to about 50-80 GB, were collected during the course of this study.

Two primary obstacles arise in managing Big Data - time and space complexity.

The time complexity issue stems from the processing required to analyze a massive volume of data. Handling vast datasets requires substantial computational resources and can cause processing tasks to take considerably longer. This is especially true when using advanced analytics techniques, such as machine learning algorithms, which can be computationally intensive.

The space complexity challenge, on the other hand, pertains to the storage requirements of handling such massive volumes of data. For instance, in this study, the entire dataset was too large to fit into the computer's random access memory (RAM), which can impede many common data processing and analysis tasks. Thus, efficiently managing memory usage becomes a crucial aspect of working with Big Data.

3.4.1 Parallel Computing

To address time efficiency, it employed parallel computing when possible. Parallel computing is a type of computation in which many calculations or processes are carried out simultaneously, leveraging the power of multiple processing elements to solve a problem more quickly than with a single processor. However, while parallel computing is an effective way to speed up the processing time, it was not extensively used in this study because many parallel computing paradigms involve an increase in space complexity. As the primarily constrain was space complexity in this study, the application of parallel computing was limited.

3.4.2 Chunk Processing

In order to handle the issue of space complexity and RAM restrictions, chunk processing was used for most of the pre-processing and NLP computation processes. Chunk processing is a data processing method where a large dataset is divided into smaller, more manageable parts, or 'chunks', and these chunks are processed independently. This technique allows for the processing of large datasets that would otherwise not fit into memory all at once, making it a necessary step for the analysis and management of the data. Chunk processing

was employed in all the computing tasks that didn’t require computing the whole dataset at once.

3.5 Entropy Estimators

Entropy estimation is a cornerstone in the analysis of linguistic systems. It aims to quantify the unpredictability or randomness in a given text, reflecting its structural or statistical complexity. By understanding the entropy of a system, one gains insights into the inherent patterns and intricacies within a language. While various methods exist to calculate entropy, they all strive to capture the essence of uncertainty and surprise in the context of language. These estimators translate the abstract concept of information into a concrete, measurable value, providing a common ground to compare different linguistic constructs.

3.5.1 Tokens Type

In this study, the focus is on word tokens, recognizing that characters also offer valuable perspectives. The decision to analyze word tokens lay from their capacity to represent a more cognitive level of linguistic structure. Words are vital carriers of meaning and are natural units in language processing. However, character tokens have also been considered, acknowledging that this has been a large focus in the literature. Characters provide a more granular insight into the language's fundamental structure, revealing patterns and correlations that might be obscured at the word level.

The principle of vocabulary in a corpus refers to the set of distinct words or characters present in the text. This set can be defined either broadly, encompassing all observed tokens, or constrained to a specific subset representing the core lexicon. The vocabulary's richness and diversity are indicative of a language's complexity and expressive power. In considering the vocabulary of a corpus, we inevitably encounter the dynamic interplay between the finite set of observed tokens and the theoretically infinite potential of language.

Considering both word and character tokens allows to capture different facets of language structure. While words provide a window into the semantically rich interactions that underpin meaning construction, characters illuminate the fundamental building blocks from which words and, consequently, meanings are formed.

3.5.2 Entropy Estimation

3.5.2.1 Conditions

Entropy estimation in natural language is not without its challenges, and certain conditions must be met to ensure reliable and meaningful results. The principles of ergodicity and stationarity are foundational, requiring that the statistical properties of text are consistent across different segments and that the text's properties are representative of the language.

A unique challenge arises with the infinite productive potential of language at the word level. Unlike characters, where the set is fixed and finite, words can be theoretically infinite. This poses problems for conventional entropy estimation, leading to what is often referred to as the "infinite problem" of language.

Additionally, the challenge of context and short-long range correlation between words adds complexity to the entropy estimation process. Words are not isolated units in natural language; they interact and correlate with each other over various distances. This introduces dependencies that must be understood and accounted for in the estimation process.

3.5.3 Objectives

One of the primary objectives of this research is to investigate and understand the nuances of entropy estimation in natural language in a noisy environment. As the topic of entropy estimation has evolved, various methodologies and strategies have been developed, each aiming to capture distinct elements and properties of the language data. In line with this, our exploration encompasses a variety of techniques, including plug-in estimators, Hrate, and Prediction by Partial Matching (PPM).

3.5.3.1 Strategy and Selection of Estimators

The choice is to investigate plug-in estimators, Hrate, and PPM based on their prevalence in the literature and their capacity to offer diverse perspectives on entropy estimation. For instance, plug-in estimators, which are grounded on direct calculation from observed frequencies, offer a straightforward and intuitive approach to entropy estimation. Hrate, on the other hand, emphasizes the rate of entropy change, providing insights into the dynamic nature of the information contained in language. PPM, a more sophisticated approach, relies on adaptive modelling techniques and is particularly potent when dealing with long contexts or sequences, making it an invaluable tool for capturing the intricate dependencies in language.

3.5.3.2 Estimators' Bias and Differences

In entropy estimation, different estimators possess inherent biases. This is not a drawback per se but rather a reflection of the estimator's underlying assumptions and design principles. These biases are, in fact, beneficial as they allow for different aspects of the data to be illuminated, providing a more comprehensive understanding of its structure and complexity.

Furthermore, the variance in results across estimators is attributed to the fact that they target different facets of the data. While some might excel in capturing short-term dependencies, others could be more attuned to long-range correlations or specific structural intricacies. This distinction is vital for interpreting and contextualizing the results.

The methodology is structured to:

1. **Work at the individual token level:** Here, the emphasis is on estimating the entropy of single tokens, without consideration for their context. This approach provides a foundational understanding, illuminating the basic unpredictability inherent in the language data with plug-in estimators.
2. **Work with a growing context window:** This strategy progressively incorporates more context, extending the window of tokens considered in the estimation. Such an approach reveals how entropy values evolve with

increasing contextual information, highlighting the role of short to medium-range dependencies, with the entropy rate estimator.

3. **Work with a dynamic and variable context window:** This approach adapts the context window based on the data's inherent structure and the specific sequence in question. By allowing for such flexibility, this strategy captures the non-uniform dependencies across different sections of the data, providing a nuanced understanding of the language's entropy landscape, with the PPM estimators.

3.5.4 Plug in Estimators

The plug-in estimator approach to entropy estimation has become a fundamental technique due to its simplicity and widespread applicability. This method relies on direct calculations from observed frequencies. Within the framework of plug-in estimators, Shannon entropy plays a central role.

3.5.4.1 Shannon entropy

Shannon entropy is named after Claude Shannon, who introduced this concept in his work on information theory in 1948. It is used to quantify the uncertainty associated with the outcome of a random.

Shannon entropy, denoted by $H(X)$, is defined for a discrete random variable X with probability distribution $p(x)$:

$$\boldsymbol{H(X) = -\sum_{x \in X} p(x) \log p(x)}$$

where X is the set of possible outcomes, and $p(x)$ is the probability of occurrence of outcome x. The logarithm can be taken to any base, but in information theory, it is common to use base 2, resulting in a measure of entropy in bits.

3.5.4.2 Bias

Estimating the probability distribution $p(x)$ forms the central challenge in entropy calculation, especially considering the non-finite paradigm of language. This non-finite nature results in the complex task of properly estimating the underlying probabilities, giving rise to the need for a multitude of plug-in estimators. Each of

these estimators is designed to tackle specific aspects of this challenge and inherently carries certain biases.

Corpus: "In the beginning God created [.....] The grace of our Lord Jesus Christ be with you all. Amen." (The Bible)

Vocabulary: "Humanity" is not present in the corpus.

Counter: The counter contains various counts such as 4, 3, 56, [...], 13, 0, 54, 5, which correspond to different token frequencies.

Now, consider different plug-in estimator methods to handle the word "Humanity":

- **Maximum Likelihood (ML):** "Humanity" is effectively invisible in the estimation process since it's not present in the corpus.
- **Miller–Madow (MM):** Although "Humanity" is present, frequencies may be distorted due to the bias correction in MM (Miller & Miller, 1955).
- **Chao–Shen (CS):** "Humanity" may be overestimated because of the bias reduction method applied (Ao, 2003).
- **James–Stein (JS):** "Humanity" may be underestimated because of the tendency of the JS estimator to shrink the estimates (James & Stein, 1961).
- **Nemenman–Shafee–Bialek (NSB):** This estimator handles "Humanity" by taking into account prior knowledge, allowing for a more accurate estimation (Nemenman et al., 2002).
- **Dirichlet–Laplace (DL):** The DL estimator handles "Humanity" based on parameters and the use of a Dirichlet prior, providing a balanced approach.

These examples illustrate the varied ways different estimators handle the complex challenge of estimating probabilities in the non-finite language paradigm. The biases inherent in each estimator need to be carefully considered based on the specific requirements and nature of the data being analyzed.

3.5.4.3 Methodology

This sub-section provides an overview of the various entropy estimators that were employed in the analysis. The objective here is to obtain a reliable and unbiased estimation of entropy, which plays a crucial role in understanding the underlying stochastic processes in the data.

ML (Maximum Likelihood) Estimation

Equation 1: Maximum Likelihood Probability Estimation

$$\hat{p}(w_i)_{ML} = \frac{f_i}{\sum_{j=1}^{V} f_j}$$

The Maximum Likelihood (ML) estimation forms the foundation of the entropy estimation approach. It operates on the principle of using frequency counts to compute the probability of words in a given text and is best applied when the number of tokens is much bigger than the size of the vocabulary. However, the ML method can be unreliable for small texts, where the ratio of word types to word tokens is significant.

MM (Miller–Madow) Estimator

Equation 2: Miller–Madow Estimator Entropy

$$\widehat{H}^{MM} = \widehat{H}^{ML} + \frac{M_{>0} - 1}{2N}$$

The Miller–Madow (MM) estimator is employed to rectify the bias in the ML estimated entropy. By adding a correction term to the ML estimate, as, the MM estimator proves valuable in counterbalancing the underestimation bias in small text sizes.

Chao–Shen Estimator

Equation 3: Chao–Shen Estimator Probability

$$\hat{p}(w_i)^{GT} = \left(1 - \frac{m_1}{N}\right)\hat{p}(w_i)^{ML}$$

Equation 4: Chao–Shen Estimator Entropy

$$\widehat{H}^{CS} = -\sum_{i=1}^{V} \frac{\hat{p}(w_i)^{GT} \log_2(\hat{p}(w_i)^{GT})}{1-(1-\hat{p}(w_i)^{GT})^N}$$

The Chao–Shen estimator serves to overcome the overestimation of probabilities, by first estimating the so-called sample coverage. By integrating the Good-Turing estimated probability, the methodology addresses the challenges associated with vocabulary representation in the sample.

Shrinkage Estimator

Equation 5: Shrinkage Estimator Probability

$$\hat{p}(w_i)^{shrink} = \lambda \hat{p}(w_i)^{target} + (1-\lambda)\hat{p}(w_i)^{ML}$$

The James–Stein shrinkage estimator, provides a flexible approach to entropy estimation by incorporating the shrinkage intensity λ and shrinkage target. The method's effectiveness lies in its ability to control the estimated probability's bias.

Dirichlet Bayesian Estimator

Equation 6: Dirichlet Bayesian Estimator Probability

$$\hat{p}(w_i)^{Bayes} = \frac{f_i + a_i}{N + A}$$

The Bayesian approach brings in the application of priors through the Dirichlet distribution, allowing for various entropy estimates based on the selected priors:

- **Laplace:** By applying a uniform Laplace prior of $a = 1$, the estimate helps flatten the distribution of frequency counts, reducing bias towards short-tailed distributions.
- **Jeffrey:** The Jeffrey's prior with $a = 1/2$ aids in achieving a particular Bayesian entropy estimate.
- **SG:** Utilizing the Schürmann and Grassberger's prior with $a = 1 / \text{length}(y)$ results in another form of entropy estimate.

- **Minimax:** Employing the minimax prior of *a* = sum(*y*) / length(*y*) produces the minimax entropy estimate.

NSB (Nemenman–Shafee–Bialek) Estimator

The NSB estimator represents the most recent and least biased method based on the Bayesian framework. Instead of relying on specific Dirichlet priors, it forms priors as weighted sums of different Dirichlet priors, resulting in entropy estimates that are robust across various sample sizes.

3.5.4.4 Technology

To implement a wide variety of entropy estimators, the following research use:

R Package Entropy: comprehensive package provides functionalities for computing different entropy estimators including ML, MM, Chao Shen, Shrinkage, and various Dirichlet Bayesian Estimators (Hausser et al., 2022).

Python Implementation of NSB: A specialized Python implementation of the Nemenman Shafee Bialek estimator is used.

3.5.5 Prediction by Partial Matching

3.5.5.1 Methodology

Prediction by Partial Matching (PPM) is a statistical technique employing the principles of Markov models to predict upcoming symbols based on the historical context within a sequence (Teahan & Cleary, 1996).

PPM builds an adaptive Markov model, continually adjusting the estimated probabilities of each symbol as the model encounters new data. Its utilization of variable-length n-grams enables the method to dynamically select the most relevant context for predictions.

- **Model Construction**: PPM begins by initializing a tree-like structure that will hold counts of observed n-grams. This flexible structure allows for efficient updating and querying as new data is introduced.

- **Probability Estimation**: Utilizing the context from preceding symbols, PPM computes the likelihood of each possible next symbol. It leverages a blend of historical data and back-off to lower-order models when the specific context has not been observed.
- **Adaptation**: As the model processes new data, it adapts its probability estimations accordingly. This dynamic nature ensures that the model remains relevant, even as underlying data patterns shift.
- **N-gram Selection**: The use of variable-length n-grams allows the model to balance specificity and generality in its predictions. It can choose longer n-grams when the context is robust, and shorter n-grams when less specific context is needed. This decision-making process is guided by Bayesian criteria or other statistical tests (Begleiter, El-Yaniv, & Yona, 2004).
- **Application to NLP**: PPM's capabilities are extended to various NLP tasks in this research, with meticulous consideration of the nature of the language data and the objectives of the specific analysis.

The choice of PPM for this research is grounded in its proven theoretical underpinnings and practical utility in the field of computational linguistics. Its adaptable nature aligns with the dynamic and complex characteristics of natural language.

3.5.5.2 Technology

For the implementation of PPM, this research made use of the PPM R package developed by pmcharrison. This particular package offers a rich set of tools and functions tailored for the nuanced needs of PPM modelling, including decay or not models (Harrison, 2021).

3.5.6 Entropy Rate

The Entropy Rate, denoted by $h(T)$, is a critical measure that provides a lens into the growth of entropy as word tokens accumulate. The concept of entropy rate has its roots in information theory and helps in understanding the unpredictability

within a text corpus. It's a critical measure that provides insights into the statistical dependencies between tokens in a sequence of text.

3.5.6.1 Methodology

The Entropy Rate is a significant measure that quantifies the average growth of word entropy as more word tokens are collected in a given text. It's crucial in understanding the unpredictability and complexity of the data.

Instead of calculating entropy with ever-increasing block sizes, the entropy rate focuses on a specific feature of the entropy growth curve. In essence, it reflects the average information content of a token, conditioned on all preceding tokens (Bentz & Alikaniotis, 2016). This allows for the accommodation of all statistical dependencies between tokens, which is particularly significant in natural languages where words are not independently distributed.

One efficient way to approximate the entropy rate of a text involves finding the longest match-length for each word token, indicating patterns in the text. The method works by looking for the longest matching string for any given word token and finding the redundancy within the token string, which is inversely related to unpredictability or choice.

The formula to calculate the entropy rate, based on this approach, is given in Equation 7.

Equation 7: Entropy Rate based on match-lenghts

$$\hat{h}(T) = \frac{1}{N}\sum_{i=2}^{N}\frac{\log_2 i}{L_i}$$

Here, *h*^(*T*) is the approximation of the entropy rate for the text *T*, *N* is the overall number of tokens, *i* is the position in the string, and *Li* is the longest match-length for the token at position *i* (Bentz & Alikaniotis, 2016).

Consider the text from a 2018 tweet by Elon Musk:

"why₁ falcon₂ heavy₃ &₄ starman₅?₆ life₇ cannot₈ just₉ be₁₀
about₁₁ solving₁₂ one₁₃ sad₁₄ problem₁₅ after₁₆ another₁₇.₁₈

there$_{19}$ *need*$_{20}$ *to*$_{21}$ *be*$_{22}$ *things*$_{23}$ *that*$_{24}$ *inspire*$_{25}$ *you*$_{26,27}$ *that*$_{28}$
make$_{29}$ *you*$_{30}$ *glad*$_{31}$ *to*$_{32}$ *wake*$_{33}$ *up*$_{34}$ *in*$_{35}$ *the*$_{36}$ *morning*$_{37}$ *and*$_{38}$
be$_{39}$ *part*$_{40}$ *of*$_{41}$ *humanity*$_{42.43}$ *that*$_{44}$ *is*$_{45}$ *why*$_{46}$ *we*$_{47}$ *did*$_{48}$ *it*$_{49.50}$
we$_{51}$ *did*$_{52}$ *for*$_{53}$ *you*$_{54.55}$*"*

To illustrate the method for estimating the entropy rate, let's calculate the match-length *L* for the tokens "be" and "problem."

For the word token "be":

- The first occurrence of "be" at $i = 10$ has no previous match, so $L10 = 0 + 1 = 1$.
- The second occurrence of "be" at $i = 22$ matches with the previous occurrence at $i = 10$, so $L22 = (22 - 10) + 1 = 13$.
- The third occurrence of "be" at $i = 39$ matches with the previous occurrence at $i = 22$, so $L39 = (39 - 22) + 1 = 18$.

For the word token "problem":

- There is not previous occurrence of "problem" at $i = 15$, so $L15 = 0 + 1 = 1$.

The calculation of match-length is based on finding the longest matching token string in the preceding token string. The match-length for each word token reflects the redundancy in the token string and can be utilized to approximate the entropy rate using the previously mentioned equation.

3.5.6.2 Technology

In this study, the **Hrate** package for R by dimalik on Github was employed. This package incorporates the principles outlined above and facilitates the seamless computation of the entropy rate.

3.6 Uncertainty Analysis

Understanding and quantifying uncertainty is important in the entropy estimation of textual data. As entropy can significantly vary based on assumptions and

different estimators used, the uncertainty analysis aims to establish a more robust understanding of the entropy estimate.

3.6.1 Objectives

The primary objectives of uncertainty analysis in the context of this research are:

1. **Quantification of Uncertainty**: To provide a comprehensive quantification of uncertainty related to entropy estimation, considering the assumptions and various estimators used.
2. **Enhanced Reliability**: To ensure that the entropy estimation is not merely a byproduct of certain biases or assumptions, and to increase the confidence and reliability of the analysis.
3. **Informative Insights**: To provide insights that may guide future methodological decisions and aid in the interpretation of the results.

3.6.2 Bootstrap Analysis

Bootstrap Analysis is a statistical method that provides a means for uncertainty estimation in complex models where theoretical computations might be challenging.

Bootstrap analysis is conducted with replacement. In this resampling method, a sample is drawn randomly from the original dataset, and then it is placed back into the dataset before the next sample is drawn. This allows for the possibility that the same data point can be sampled more than once, creating different combinations and variations in the resampled datasets. By resampling with replacement, bootstrap analysis can generate a rich variety of samples, which helps to approximate the underlying distribution of the data and provides insights into the statistical properties of the estimator.

This method allows for the calculation of standard errors, confidence intervals, and other statistical measures. In the context of entropy estimation, it is employed with plug-in estimators as a non-parametric approach for estimating uncertainty.

The analysis is conducted over 100 iterations for each estimation to adequately represent the underlying distribution, and it computes 95% confidence intervals and standard deviation to delineate the probable range for the true entropy value. Furthermore, stratified sampling is employed over the length of the tweets, ensuring the same proportion of long and short texts as in the initial sample. This method is crucial since entropy can widely vary based on the corpus length, thus ensuring that the sampling does not introduce bias.

3.6.3 Sensitivity Analysis

The Sensitivity Analysis focuses on understanding the effect of different preprocessing decisions on entropy estimation, aligning with the preprocessing steps explained earlier in the methodology. The analysis revolves around quantifying the differences in entropy estimations based on various preprocessing choices such as keeping or removing punctuation, numbers, special characters, and so on.

The analysis also aims to assign specific values to each preprocessing step, creating a nuanced understanding of how these choices impact the final entropy estimation. Moreover, it systematically analyzes and quantifies the entropy related to each type of "noise," such as special characters or punctuation, thereby providing a methodical way to comprehend their influence on the entropy estimation.

3.7 Workflow

The workflow of the research, depicted in Figure 2, outlines the interconnected and sequential processes that form the methodology of the study. The graph starts with the data extraction, executed through various means such as web scraping and the utilization of APIs, providing the raw data essential for the analysis.

This is followed by a series of carefully structured stages, including preprocessing paired with sensitivity analysis to refine and evaluate the data, NLP labeling and

tokenization for segmenting the data into meaningful components, and entropy analysis with stratified bootstrap uncertainty to estimate and assess the entropy values. The workflow concludes with the final result analysis and graphical interpretation, synthesizing and visualizing the insights derived from the research. Each stage builds upon the previous one, ensuring a comprehensive and methodologically sound exploration of the subject matter.

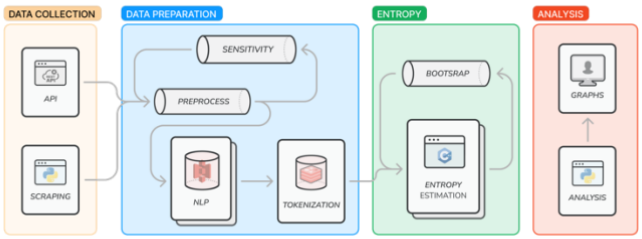

Figure 2: Workflow Diagram

3.8 Challenges

3.8.1 Data Retrieval Challenges

The research faced significant constraints in data retrieval due to the unavailability of an Academic API, with only access to a free API (Academic API is reserved for post-graduate research projects). This presented several specific challenges:

- **Lack of Access to Historical Data**: Without access to the Academic version, retrieving older data was not feasible, that is the reason why the research leveraged scraping methods.
- **Limited Scraping Capabilities**: With the advanced search only revealing a hundred tweets per day, the quantity was insufficient to recreate a comprehensive dataset over time or conduct time series estimations. The comparison with larger publicly available datasets, such as the COVID-19 public found dataset which containing hundreds of thousands of tweets daily, further emphasized this limitation.

- **Reliance on Public Repositories**: To overcome these limitations, it was essential to identify public repositories with tweet IDs, allowing for rehydration via the API. This need led to a focus on the COVID-19 dataset for time series analysis, rather than the more limited datasets obtained through scraping or streaming.

3.8.2 Vocabulary and Estimation Challenges

Attempts to combine vocabularies and utilize a common vocabulary for estimation were caught with difficulties:

- **Size and Accuracy Issues**: The large size of the combined vocabulary led to inaccurate estimations, with substantial noise. Over- or under-estimation was common, obscuring the true patterns within the data.
- **Rapid Vocabulary Recreation**: An unexpected discovery was that despite the noise, the vocabulary could recreate itself relatively quickly on new datasets, suggesting some underlying stability in the lexical structures.

3.8.3 Computational Constraints

Certain methodological approaches were rendered impractical due to computational constraints:

- **Limitations with PPM Model**: The use of the PPM model became unfeasible when dealing with vocabularies exceeding 50,000 tokens. The Markov chain became too computationally intensive, leading to excessively long computation times. This issue was exacerbated by the libraries being written in R and C, potentially restricting optimization.

3.9 Experimental Evaluation

This section detailed the methodology adopted to estimate entropy across various scenarios, ensuring a meticulous evaluation that aligns with the research objectives.

3.9.1 Entropy Estimation of Literature Books

The estimation of entropy within literature serves as a fundamental step in the analysis. This involves an investigation of five classic books: "Alice in Wonderland," "The Bible," "Les Misérables," "The Great Gatsby," and "Romeo and Juliet." The objective is to gather entropy values, inclusive of noise factors such as punctuation, numbers, and special characters, and propose a new quantifiable value for literature entropy with noise. Both character and word tokens will be considered in this estimation.

It is imperative to note that these evaluations will not be reflective of the entirety of the English language's complexity, as the selection of only five books does not capture the breadth of the language's diversity. However, it does create a foundational basis for comparison with other datasets, including social media data and programming languages, as will be elaborated upon later.

3.9.2 Entropy Estimations of English, French, and Spanish using Twitter’s Streams

The use of Twitter's streaming sample dataset offers an extensive scope to estimate the entropy of English, French, and Spanish. This approach allows for the exploration of a broad variety of vocabulary, encompassing noisy data, and thereby presenting a more realistic representation of contemporary language usage. It also proposes a quantification of the up-to-date spoken language, as Twitter is the closest representation of spoken language on the internet, The quantification will be undertaken at both the character and word token levels, employing various pre-processing techniques (with and without punctuation, emoji, special characters) to reveal the nuanced effect of these factors on entropy estimation.

3.9.3 Entropy Variation using NLP Clusters

Analysing the variation of entropy within NLP clusters, such as emotion, topic, and sentiment, enables a deeper understanding of the underlying structure in text data. By assessing entropy differentials between clusters, insights into the

linguistic dynamics and thematic relationships within the textual corpus can be extracted.

3.9.4 Entropy Variation of Targeted Users

This subsection aims to quantify entropy variations among specific user categories, such as news outlets and public personalities. By focusing on these target users, it is anticipated that unique characteristics and linguistic patterns related to different communication paradigms will be unearthed.

3.9.5 Time-bound Entropy using COVID-19 Tweets

A novel aspect of the study will be the quantification of entropy over time, specifically focused on tweets related to COVID-19. This will provide an opportunity to observe how entropy evolves in real-time discourse, reflecting societal responses to an ongoing global phenomenon.

3.9.6 Programming Language Entropy Quantification

Finally, the study extends to the realm of programming languages using Python, Java and C++ where entropy will be quantified using various pre-processing techniques (with and without variables, numbers, strings). This analysis aims to discern the unique features of programming languages and quantify the specific effect of different tokens on entropy.

4 ANALYSIS

In the following section, an initial data analysis will be conducted across the various datasets. This analysis aims to provide a more profound understanding of the internal structures and inherent properties of the data collected. The basic token unit is in word except when specified explicitly.

4.1 Literature Books

4.1.1 Word Tokens

The analysis of word tokens across various literature books, as displayed in Table 1, offers insights into the complexity and diversity of vocabulary within these texts. A key metric of interest is the Type-Token Ratio (TTR), as it provides information regarding the lexical richness of the text, which is directly related to entropy calculations.

Alice in Wonderland exhibits a higher TTR at 8.24%, indicating a greater variety of word usage. Conversely, The Bible, with a lower TTR of 1.86%, signifies more repetitive word choice. This lower TTR, coupled with a high number of tokens, could imply that the vocabulary becomes reused at a certain point, potentially influencing entropy estimations (see Figure 3) with the number of tokens only used once around 40-45%.

Table 1: Literature Books Tokens Analysis (Word)

Book	Nbr. Of Token	Vocab. Size	TTR (%)
Alice in Wonderland	33 626	2 772	8.24
The Bible	949 327	17 694	1.86
Les Miserables	689 369	24 936	3.61
Romeo and Juliette	37 996	4 112	10.82
The Great Gatsby	64 812	6 310	9.73

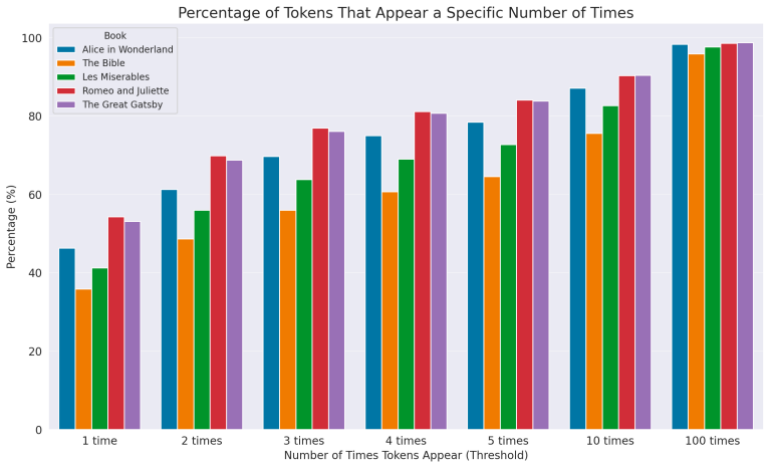

Figure 3: Literature Books Tokens Appearances (Word)

In Figure 3, the token appearance graph further illustrates this trend, with Romeo and Juliette and The Great Gatsby showing similarities in token appearances, perhaps indicative of parallel writing styles or themes.

Les Miserables presents an interesting case where punctuation and common words are among the most frequent tokens (Table 2**,** Figure 4). This observation may influence entropy estimations when considering noise, such as punctuation, and emphasizes the importance of preprocessing decisions in entropy analysis.

Table 2: Les Miserables Most Frequent Tokens (Word)

Token	Count
,	48 760
The	41 034
.	29 971
Of	19 933
And	14 930
A	14 521
To	13 726

Figure 4: Les Misérables WordCloud (Word)

4.1.2 Character Tokens

Character token analysis, as presented in Table 3, adds another layer of understanding. Les Miserables displays the most diverse character set, potentially reflecting a richer syntactic structure or various specialized terms. This diversity might have implications for entropy measurements when considering different character representations or noise levels.

The frequent character tokens in The Bible (Table 4) further illustrate the text's syntactic structures, with spaces and common lowercase letters dominating the counts. Considering only the letter character the analysis match with the most common English character from the overall literature.

Table 3: Literature Books Tokens Analysis (Character)

Book	Nbr. Of Token	Vocab. Size
Alice in Wonderland	140 628	44
The Bible	4 250 846	52
Les Miserables	3 175 878	83
Romeo and Juliette	155 658	63
The Great Gatsby	282 455	65

Table 4: The Bible Most Frequent Tokens (Character)

Token	Count
(space)	750 880
e	412 160
t	317 696
h	282 657
a	275 701
o	243 143
n	225 026
i	193 926

4.2 Twitter Streams

The Twitter streams dataset offers insights into the digital interactions and expressions of users. Analyzing this dataset is crucial in understanding language entropy in a noisy environment, such as social media.

4.2.1 General Analytics

4.2.1.1 Number of Tweets and users

The dataset spans from May 29 to June 13, encompassing 19,919,907 tweets from 8,977,811 users. The distribution of tweets per user is highly skewed, with most users tweeting only a few times (a median of 1) and a maximum user tweeting 3,075 times (raising the mean to 2.21) as shown in Table 5 and Figure 5. This distribution might indicate the potential presence of bots or commercial accounts, which can have implications for the estimation of entropy by possibly introducing non-human-like patterns of language usage. A significant number of users, 207,608, tweeted more than 10 times, while 879 users exceeded 100 tweets.

Table 5: Twitter Streams General Analytics

Number of Tweet	19 919 907
Start Date	29 May
End Date	13 June
Number of User	8 977 811

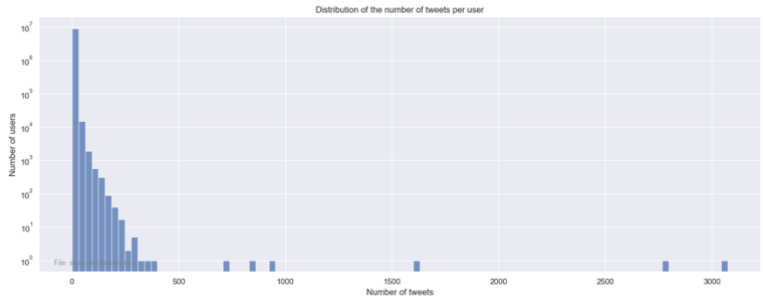

Figure 5: Twitter Streams Tweet by User

4.2.1.2 Tweet Length

4.2.1.2.1 Word Token

The mean number of tokens per tweet stands at 16.38, with a median of 17.0. The distribution of tweet lengths has a standard deviation of 10.77, illustrating a significant variance in the way users structure their tweets. The data reveals a first quartile (Q1) of 7.0 tokens and a third quartile (Q3) of 23.0 tokens. The minimum and maximum number of tokens per tweet are 1 and 191, respectively.

The distribution of tweet lengths, as shown in Figure 6, underscores the diversity of expression within the 280-character limit of Twitter. This variation in tweet length could have implications for the estimation of entropy in language. Shorter tweets may reflect simpler, more predictable structures, while longer tweets might embody more complex syntactical arrangements.

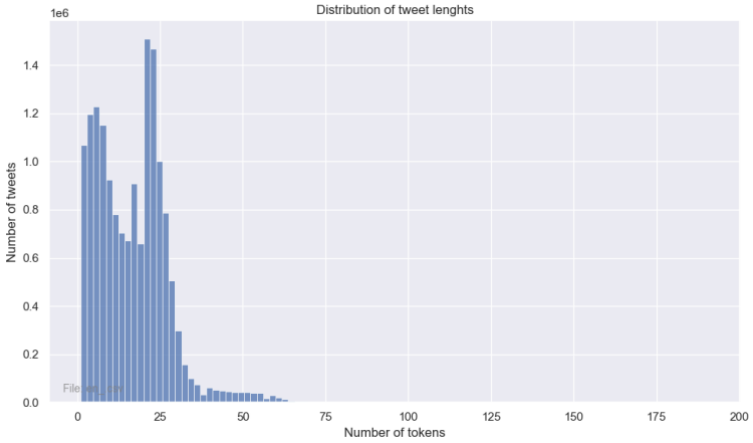

Figure 6: Twitter Streams Tweet Length (Word)

4.2.1.2.2 Character tokens

In addition to word tokens, character tokens provide further insight into the language structure of Twitter streams. The mean number of character tokens per tweet is 78.40, with a median of 84.0. The standard deviation of 50.50 character tokens per tweet points to a wide distribution in the character count across different tweets, demonstrating the extensive variability in the language patterns. The first and third quartiles (Q1 and Q3) stand at 36.0 and 107.0 character tokens, respectively, with the minimum and maximum character tokens per tweet being 1 and 970.

The distribution, as visualized in Figure 7, represents another dimension of the complexity and richness of language on Twitter. The character-level analysis might reveal subtler patterns, such as abbreviations and unconventional spellings, prevalent in informal communication channels like Twitter.

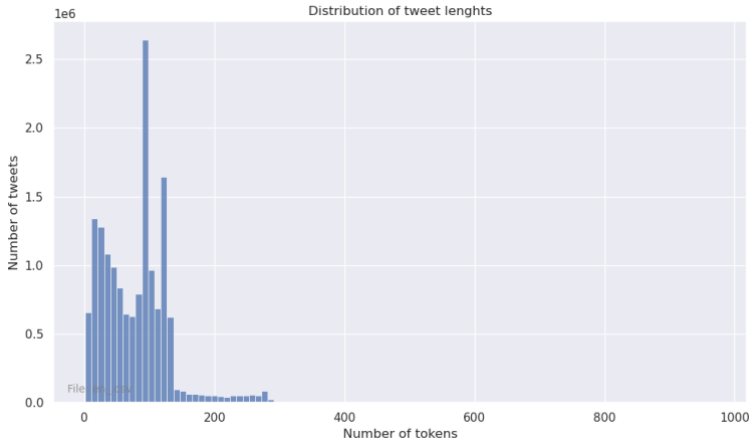

Figure 7: Twitter Stream Tweet Length (Char)

4.2.1.3 Temporal Trend

The temporal distribution of tweets (Figure 8) reveals an average of 1.2 million tweets collected daily. This consistent volume suggests a steady flow of data that might provide insights into both short-term and long-term trends in language usage and complexity.

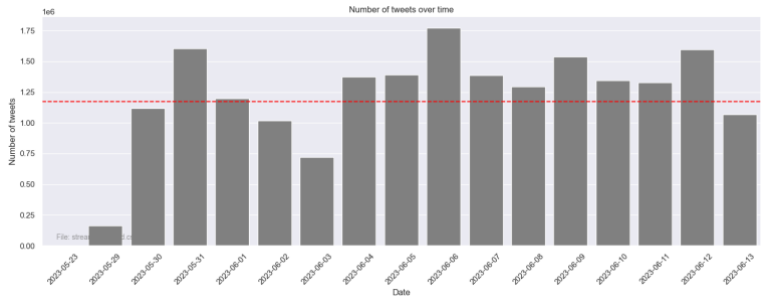

Figure 8: Twitter Streams Tweet over Time

4.2.1.4 Language Distribution

The dataset predominantly consists of English tweets (1.5e7), followed by Spanish (around 0.5e7) and French (around 0.1e7) as depicted in Figure 9. Such diversity in languages opens avenues for cross-lingual entropy comparison, particularly in assessing the inherent noise present in different languages or dialects.

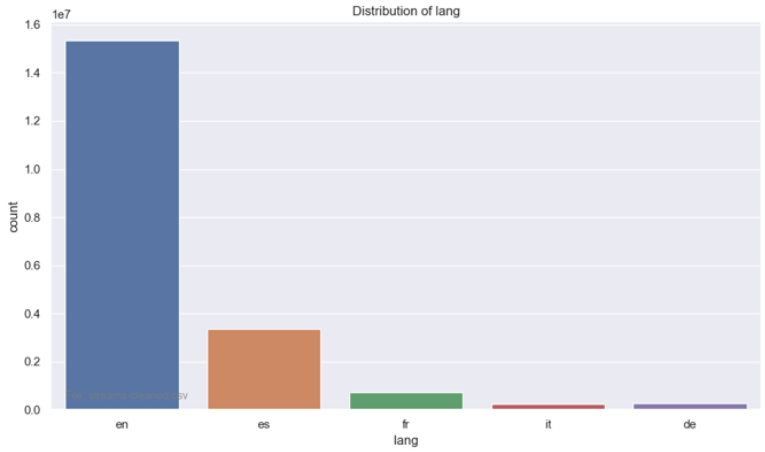

Figure 9: Twitter Streams Language Distribution

4.2.1.5 Emotion and Sentiment Distribution

Emotional expressions within the dataset are varied, with a significant presence of joy and anticipation, a medium amount of anger, sadness, disgust, optimism, and low levels of surprise, love, and pessimism (Figure 10). Such distributions might reflect common public sentiments during the data collection period. This emotional layer provides an additional dimension for understanding language

entropy, as different emotions may manifest in varying syntactic and lexical patterns.

The sentiment distribution is mostly well balanced between positive and negative, with a slight skew toward negativity and approximately 50% neutrality (Figure 11). Observing sentiment over time (Figure 12), the distribution appears mostly stable, perhaps indicative of a general mood during the collection period. These sentiment metrics offer a more nuanced understanding of language structure and could be leveraged for entropy calculations.

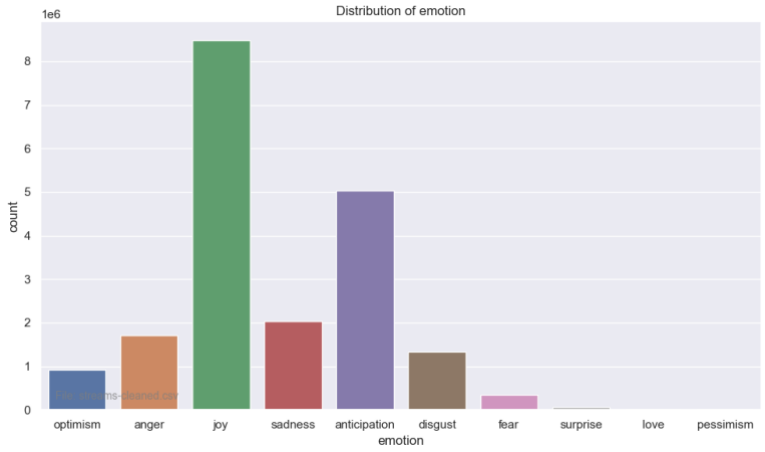

Figure 10: Twitter Streams Emotion Distribution

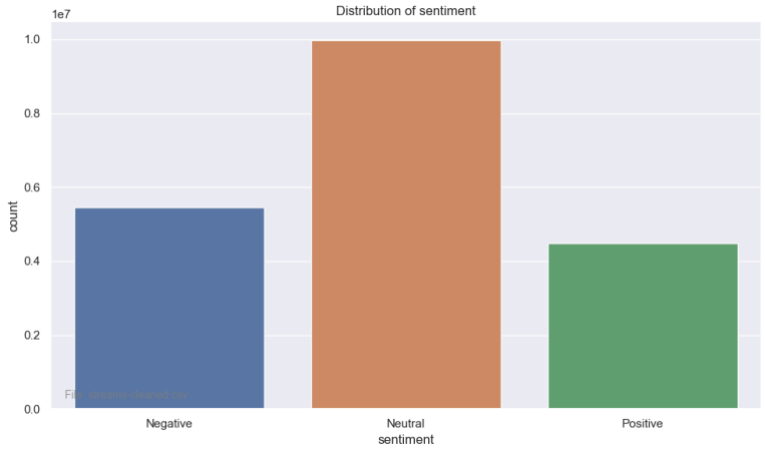

Figure 11: Twitter Streams Sentiment Distribution

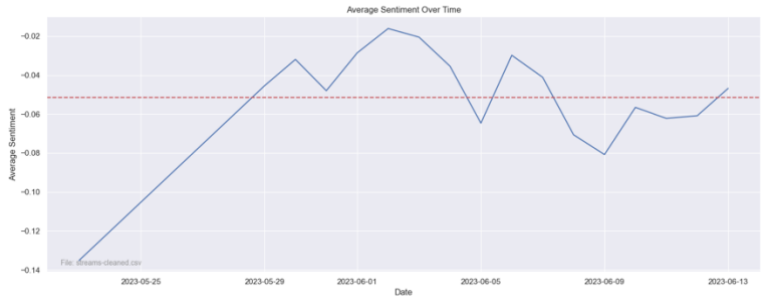

Figure 12: Twitter Streams Sentiment over Time

4.2.1.6 Topic Distribution

The topical insights (Figure 13) reveal that most tweets are centered on diaries and daily life, followed by business and entrepreneurship, news and social concerns, and then celebrities and pop culture. This distribution could influence entropy measures as different topics might inherently contain various lexical and syntactic structures.

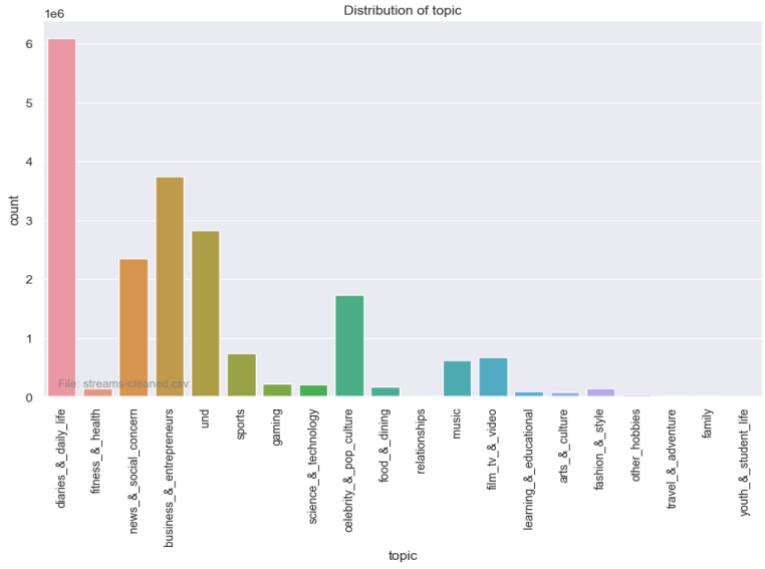

Figure 13: Twitter Streams Topic Distribution

4.2.2 Analytics per Language

Table 6 encapsulates key statistics related to the number of tweets, tokens, vocabulary size, and the Type-Token Ratio (TTR) for each language. Interestingly, the TTR percentage varies significantly across languages, being lowest for English at 0.76% and highest for French at 2.20%. The large

vocabulary size, coupled with a low TTR in English, suggests extensive reuse of the vocabulary, potentially impacting the entropy calculation.

Table 6: Twitter Streams Token Analytics (Word)

Language	Nbr. Tweet	Nbr. Token	Vocab. Size	TTR (%)
English	15 736 589	252 385 416	1 919 357	0.76
Spanish	3 506 969	54 452 795	623 123	1.14
French	751 450	11 859 961	261 279	2.20

Figure 14 further illustrates that around 55-60% of tokens are used only once in the Twitter streams, which is significant when compared to the literature book dataset. This could include a high level of noise, misspellings, and other irregularities that could influence the entropy estimation, adding an extra layer of complexity to the analysis.

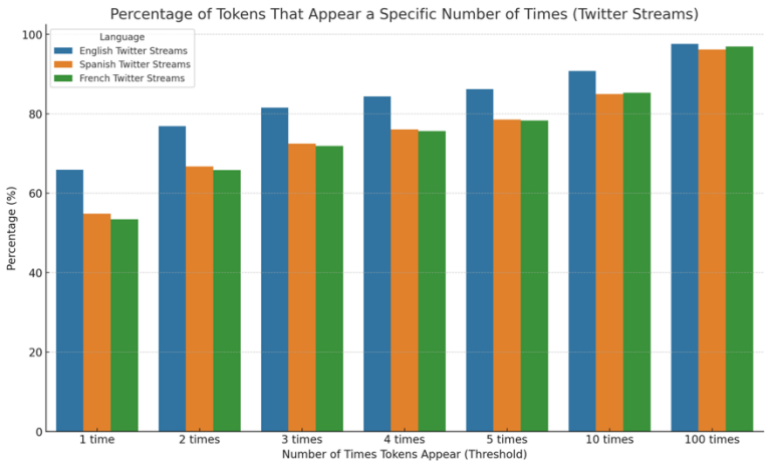

Figure 14: Twitter Streams Tokens Appearance (Word)

Figure 15: Twitter Streams English Token Distribution illustrates the skewed token distribution within the English Twitter streams. This figure reveals a pattern where most of the tokens are used only once, while a select few tokens are used with high frequency. Such a distribution aligns with Zipf's law, a well-recognized principle in linguistic studies (Zipf, 1949). This law asserts that a small number of words are used extremely often, while the vast majority of words are used rarely. The implications for entropy analysis in this context are substantial, as this

skewed distribution may introduce biases or irregularities in the entropy estimation.

The prominence of rarely-used tokens also further substantiates the earlier observation regarding the potential noise and intricacies inherent in social media language.

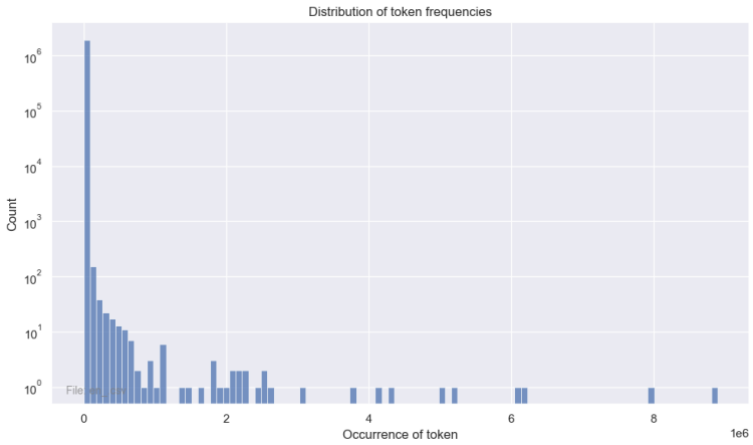

Figure 15: Twitter Streams English Token Distribution

4.2.2.1 Most Common Token (Word)

4.2.2.1.1 English

The most common tokens in the English Twitter streams are presented in Figure 16 and Table 7. The frequent occurrence of tokens such as the dollar symbol ('$') and certain words like 'airdrop,' 'loyal,' and 'claim' in Table 8 suggests a prevalence of content related to crypto giveaways.

The presence of such tokens, including specific ones like $Loyal and $psyop, could be interpreted as suspect or bot activity. This assumption can be substantiated by the abnormally high frequency of these tokens. Moreover, the exclusion of punctuation and stop-words reveals a prevalence of specific jargon related to financial or commercial activities.

These findings are crucial in the context of entropy analysis as they demonstrate how the presence of commercial or bot-related content can create a specific linguistic pattern within the dataset. Such patterns could skew the entropy

estimation, reflecting a structural characteristic of the data rather than an inherent property of the natural language itself.

Figure 16: Twitter Streams English Most Common Token (Word)

Table 7: Twitter Streams Most Common Token (Word)

Token	Count
$	8 895 213
the	7 955 775
.	6 142 332
...	6 094 321
,	5 205 729
!	5 040 109
is	4 281 371
to	4 170 831
and	3 803 227
a	3 043 008

Table 8: Twitter Streams Most Common Token without Punctuation and Stop-words (Word)

Token	Count
airdrop	2 543 314
loyal	2 109 831
live	2 012 219
claim	1 833 553

tokens	1 441 644
check	1 109 713
eligibility	968 823
psyop	769 650

4.2.2.1.2 French and Spanish Words

For the Spanish tweets, stop-words and punctuation like 'de', ',', 'que', 'la', and '.' are among the most common tokens, as shown in Table 9. However, when removing punctuation and stop-words, more informative terms emerge, as in Table 10. Words like 'solo' (only), 'hoy' (today), 'asi' (like), and 'quiero' (I want) reflect daily expressions and common conversational elements. The frequency of these words emphasizes the casual and organic nature of Twitter communication in Spanish.

In a similar vein, the French language stream illustrates a dominance of punctuation and common connective words like 'de', ',', 'la', 'le', and '.' (see Table 9). Removing these, we find frequently occurring words like 'c'est' (it is), 'fait' (done), 'j'ai' (I have), and 'tout' (all), as depicted in Table 10. Like the Spanish dataset, these words mirror everyday language use in French-speaking communities.

Table 9: Twitter Stream Most Comon Token French and Spanish (Word)

Spanish		**French**	
token	**count**	**token**	**count**
de	2 197 584	de	411 068
,	1 700 352	,	301 017
que	1 575 210	a	265 238
la	1 416 814	la	258 506
.	1 220 567	...	243 226
a	1 195 991	le	227 947
el	1 169 999	.	226 558
...	1 143 995	les	170 694
y	1 105 736	et	160 270

en	995 651	en	123 228

Table 10: Twitter Stream Most Comon Token French and Spanish without Punctuation and Stop-words (Word)

Spanish		**French**	
token	**count**	**token**	**count**
solo	97 479	c'est	98 134
hoy	93 070	fait	32 460
asi	91 436	j'ai	31 996
anos	67 924	tout	29 750
quiero	67 868	meme	25 985
ver	65 322	comme	25 323
hace	62 079	faire	25 166
vida	61 625	quand	22 293
siempre	60 944	cette	22 045
ahora	60 713	trop	18 755

The Figure 17 word clouds for both French and Spanish visually summarize these findings, highlighting the similarities between the two languages in terms of punctuation and stop-words. Such parallel structures might have implications for language entropy estimation, particularly in how noise and organic conversation are modeled.

The more "organic" nature of French and Spanish tweets, as opposed to the prevalence of bot-like activities observed in English, also becomes apparent. This

contrast may carry significance in entropy estimation, where understanding the natural flow and structure of language is crucial.

Figure 17: Twitter Stream French (left) and Spanish (right) Wordcloud

4.2.2.2 Most Common Token (Char)

Table 11 presents the number of tweets, total character tokens, and vocabulary size for each of the three languages. Interestingly, the vocabulary size at the character level is consistent across languages at 69 (excluding emoji and accents). This reflects the shared Latin script and can be considered as a factor that normalizes part of the analysis across different languages.

Table 11: Twitter Stream General Analytics (Char)

Language	Nbr. Tweet	Nbr. Token	Vocab. Size
English	15 736 589	1 228 250 902	69
Spanish	3 506 969	276 074 672	69
French	751 450	61 762 418	69

In the English dataset, the space character is the most common, followed by the letters 'e', 'o', 'a', 'i', 't', 'n', 's', 'r', and 'l', as shown in Table 12. These characters are consistent with the frequency distribution commonly found in the English language (Zipf's Law), where vowels and commonly used consonants like 't' and 'n' dominate.

The Spanish dataset reveals a somewhat different distribution. While the space character also tops the list, the characters 'e', 'a', 'o', 's', 'n', 'i', 'r', 'l', and 'd' are

the most frequent. The prominence of 'a' and 'o', particular to the grammatical structure of Spanish, reflects the language's specific characteristics.

The French dataset, while like English in some respects, has its unique distribution, with the space character leading, followed by 'e', 'a', 's', 'i', 'n', 'r', 't', 'o', and 'u'. The high frequency of vowels, particularly 'e', reflects the French language's phonetic and orthographic system.

Table 12: Twitter Streams Most Common Tokens (Char)

English		**Spanish**		**French**	
Token	**Count**	**Token**	**Count**	**Token**	**Count**
(space)	212 721 232	(space)	47 448 802	(space)	10 693 048
e	99 400 219	e	28 624 299	e	7 714 481
o	76 238 664	a	27 202 770	a	4 069 684
a	74 554 134	o	20 335 322	s	3 733 106
i	73 751 040	s	15 572 240	i	3 341 286
t	73 641 698	n	15 1536 36	n	3 309 366
n	61 295 590	i	14 076 790	r	3 236 745
s	55 562 036	r	13 692 170	t	3 155 444
r	53 622 260	l	11 283 613	o	2 825 575
l	46 772 620	d	10 074 059	u	2 822 758

4.3 Covid-19 Tweets

Table 13 illustrates the number of Covid-19 related tweets in the three languages. English dominates with a staggering 91,318,297 tweets, reflecting the global language's widespread use. Spanish follows with 28,357,319 tweets, while French accounts for 6,809,418 tweets. This distribution may be influenced by various factors, including the number of speakers, the severity of the pandemic in specific regions, and the media coverage in these languages.

Table 13: Covid-19 Tweets by Languages

Language	Number of Tweets
English	91 318 297
Spanish	28 357 319
French	6 809 418

Figure 16 depicts the number of Covid-19 related tweets over time, providing a visual representation of the public's engagement with the pandemic. Peaks in the graph may correspond to significant events or milestones, such as announcements of new variants, vaccination rollouts, or lockdown measures The data cover the following period : 23rd Mars 2020 – 31st December 2021.

Note: The data from 1st November 2020 to 31st 2020 is not available due to the shutdown of the API before the research succeed to collect everything.

The statistical analysis of the daily count of Covid-19 related tweets offers some interesting insights. On average, there are 215,110.60 tweets per day related to Covid-19 across the three languages, indicating a robust engagement with the topic. The considerable standard deviation of 88,112.48 reflects the fluctuating nature of public interest, with daily tweet counts ranging from a minimum of 61,677 to a maximum of 519,194.

Note: The statistics provided here correspond to Covid-19 related tweets collected through the streaming public GitHub repository. Therefore, they represent a sample rather than the complete universe of tweets related to Covid-19 on the platform. This distinction should be kept in mind, as the data does not fully encapsulate the overall tweeting activity on the subject but rather offers a snapshot reflecting the available subset.

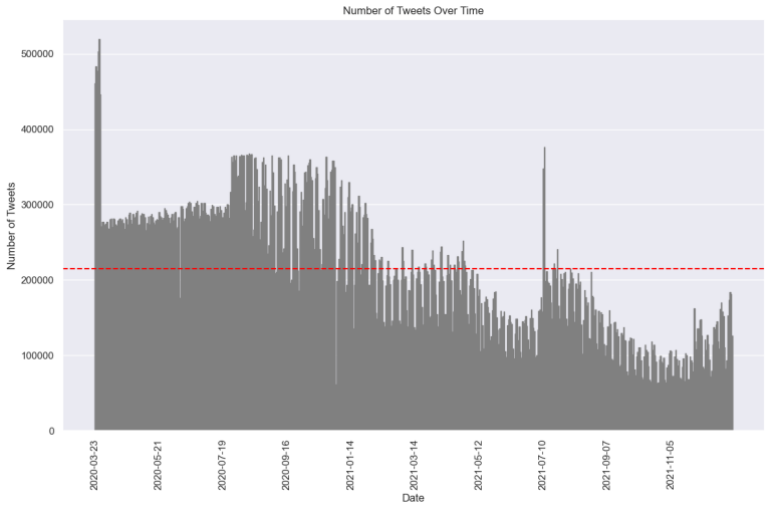

Figure 18: Covid-19 Number of Tweet over Time

4.4 Targeted User Tweets

4.4.1 Personalities

According to Table 14, the comparison between Elon Musk and Donald Trump's tweets unveils some interesting contrasts. Elon Musk has authored 17,848 tweets, resulting in 243,208 tokens and a vocabulary size of 15,720, leading to a Type-Token Ratio (TTR) of 6.46%. Donald Trump's tweets are more numerous at 53,576, with 1,206,388 tokens and a vocabulary size of 33,419, but a lower TTR of 2.77%. TTR is often used as a measure of lexical diversity; a higher TTR may indicate a more varied use of words.

As depicted in Figure 19, the distribution of token appearances between the two individuals shows subtle differences in the utilization of words. Elon Musk's tokens that appear tend to be more spread, with a broader range from 44.78% to 98.02%, compared to Donald Trump's range of 50.28% to 96.50%. This might reflect Musk's varied interests and discourses on topics such as technology, innovation, and futurism, as opposed to Trump's more political and slogan-oriented communication.

Table 14: Elon Musk, Donald Trump Token Statistics

Personality	Nbr. Tweet	Nbr. Token	Vocab. Size	TTR (%)
Elon Musk	17 848	243 208	15 720	6.46
Donald Trump	53 576	1 206 388	33 419	2.77

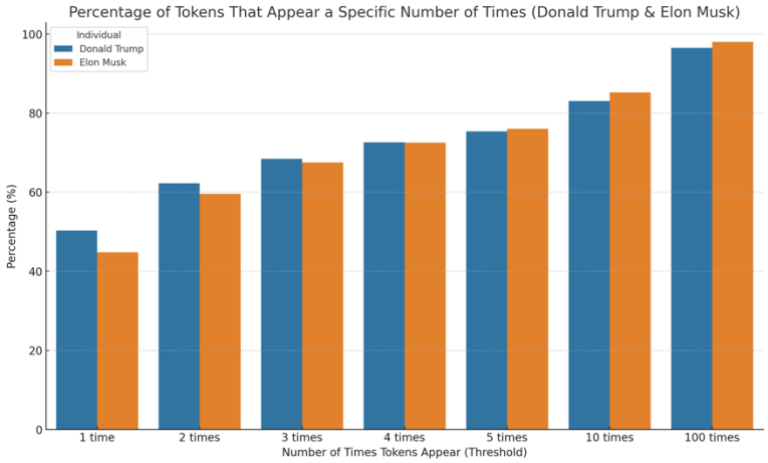

Figure 19: Elon Musk, Donald Trump Token Appearance

Examining the most common tokens without stop words and punctuation (Table 15), Musk's usage of words like "tesla," "great," "time," "model," and "car" aligns with his entrepreneurial interests, while Trump's vocabulary such as "great," "trump," "president," "country," and "america" corresponds to his political persona and leadership style. The repetition of certain words like "great" and "time" across both personalities suggests some commonality in the expression but likely divergent in context and meaning.

Table 15: Elon Musk, Donald Trump Most Common Token without Stopword and Punctuation

Elon Musk		Donald Trump	
Token	**Count**	**Token**	**Count**
tesla	1 366	great	7 455
great	566	trump	6 156
time	417	president	4 472
model	416	country	2 271

car	414	america	2 165
twitter	399	big	2 079
high	368	donald	1 893
3	364	time	1 885
make	363	make	1 829
year	333	news	1 751

4.4.2 News Outlets

Analysis on English news outlets, including prominent sources such as Reuters, NY Times, Guardian, BBC, and CNN, covering a time range from January 8, 2007, to December 31, 2022.

As observed in Table 16, the combined data for these news outlets consists of 904,358 tweets, yielding 14,568,184 tokens with a vocabulary size of 170,470. The resulting Type-Token Ratio (TTR) is 1.17%. A lower TTR percentage in this context may signify a more specialized or focused use of language, typical of formal and standardized news reporting.

Table 16: English News Outlets Token Statistics

Nbr. Tweet	Nbr. Token	Vocab. Size	TTR (%)
904 358	14 568 184	170 470	1.17

A detailed examination of Table 17 brings attention to some insights. When considering all tokens, including punctuation and stopwords, one notices an abundance of punctuation like ".", ",", and ":". The high occurrence of colons may be linked to specific tweet structures often used by news outlets, such as “NEWS:” or “BREAKING:” as a prefix to announce breaking news or declarational tweets.

When we exclude punctuation and stopwords, the remaining common tokens emphasize significant themes such as "president," "trump," "police," and "house." These keywords highlight the consistent focus on political matters and

presidency. It is reflective of the period's political climate, marked by critical political events, campaigns, elections, and administrative decisions (Figure 20).

Table 17: English News Outlets Tweet Most Common Token

w/ Punctuation & Stopwords		w/o Punctuation & Stopwords	
Token	**Count**	**Token**	**Count**
the	469 616	president	29 057
.	389 478	trump	28 471
,	377 961	police	24 553
to	359 152	live	19 231
in	304 504	uk	17 559
of	285 437	world	16 864
a	276 191	house	16 097
:	230 636	news	15 258
"	183 224	video	15 046

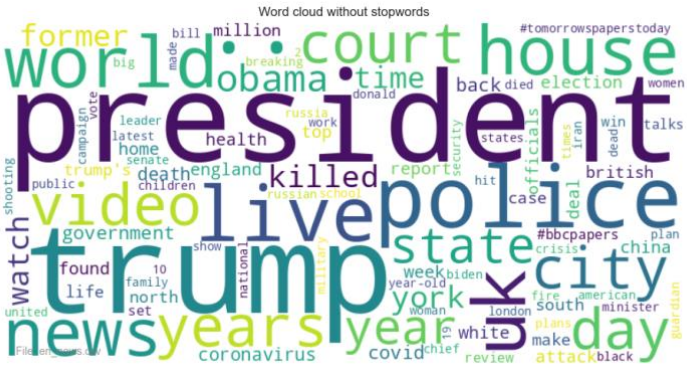

Figure 20: English News Outlets Wordcloud without Stopwords and Punctuation

4.4.3 Ukraine War Tweets

The analysis of tweets pertaining to the Ukraine war was conducted using a dataset created with keywords derived from previous research papers and scraping techniques. Specific keywords like "putin," "soviet," "Ukraine," "Russia,"

"Ukrainian," "NATO," "Moscow," "minsk," "Luhansk," "kremlin," "kiev," and "donetsk" were employed to curate this dataset.

Table 18 provides an overview of the Ukraine War Tweets tokens statistics, segmented by language (English, Spanish, French). Notably, there are 403,278 English tweets with a vocabulary size of 191,389 and a TTR of 1.38%, while Spanish and French tweets exhibit a higher TTR, 2.18% and 1.96%, respectively.

Table 18: Ukraine War Tweets Tokens Statistics

Language	Nbr. Tweet	Nbr. Token	Vocab. Size	TTR (%)
English	403 278	13 863 649	191 389	1.38
Spanish	343 322	10 203 007	222 669	2.18
French	314 065	9 578 627	187 950	1.96

An examination of the length statistics reveals insights into the complexity and composition of the tweets. The mean number of tokens per tweet is 35.51, with a median of 37, and a standard deviation of 15.59. This data indicates that tweets related to the Ukraine War tend to be more detailed and verbose, with a concentration of information within a broad range of 23 to 48 tokens (from Q1 to Q3).

A comparison with a classic English dataset further emphasizes this point. The mean length of the general English tweets stands at 16.38 tokens, significantly shorter than the Ukraine War-related tweets. Figure 21 and Figure 22 visually represent these differences, showcasing how the Ukraine War tweets exhibit a more extended length distribution.

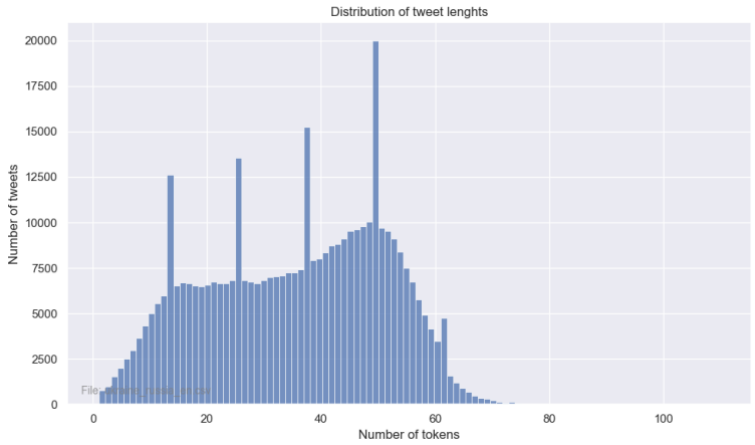

Figure 21: Ukraine War Tweet Lengths

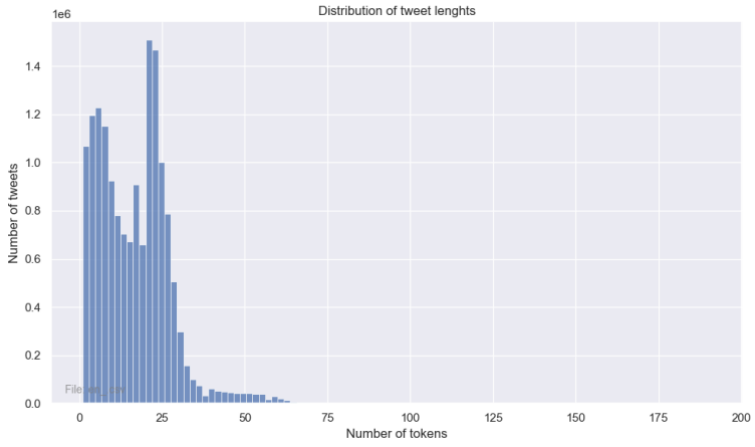

Figure 22: General English Tweet Lengths Comparison

The word clouds (Figure 23) and the lists of most common tokens without stopwords present a vivid image of the prevalent themes within the tweets. Words like "Ukraine," "Russia," "Putin," "NATO," and "Moscow" dominate the discourse across the three languages, reflecting the central figures and entities involved in the conflict.

These results underline the global concern around the Ukraine conflict and the geopolitical aspects, such as alliances (NATO), historical context (Soviet), key cities (Kiev, Donetsk), and pivotal figures (Putin). It emphasize the universal resonance of the conflict, reflecting its widespread impact and attention.

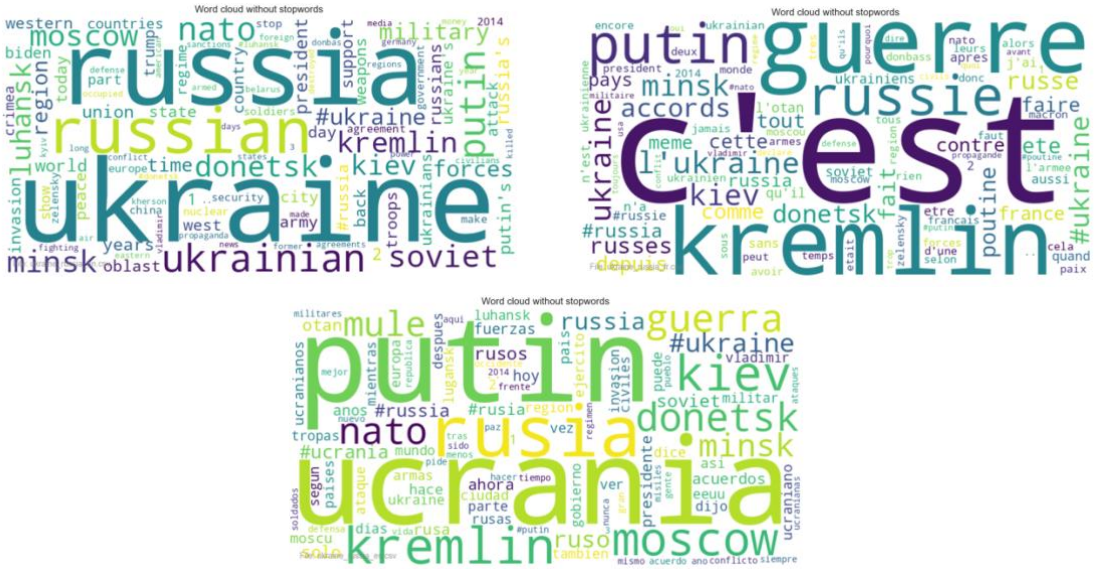

Figure 23: Ukraine War Tweets Wordcloud without Stopwords and Punctuation English (Left) French (Right) Spanish (Bottom)

4.5 Programming Languages

4.5.1 Token Statistics

Table 19 presents a broad view of token statistics for C++, Java, and Python, encompassing the number of files, tokens, vocabulary size, and TTR. The C++ language has a larger dataset, with over 4 million files and more than 1.45 billion tokens. In comparison, Java and Python have smaller datasets but exhibit distinct characteristics. For instance, Python's TTR stands at 0.21%, higher than C++ and Java at 0.08%, possibly indicative of Python's concise syntax and expressive vocabulary.

Table 19: Programming Languages Tokens Statistics

Language	Nbr. Of File	Nbr. Token	Vocab. Size	TTR (%)
C++	4 353 050	1 454 883 936	1 214 278	0.08
Java	354 983	177 870 063	144 387	0.08
Python	1 796 572	232 620 329	490 909	0.21

4.5.2 Token Appearances

The representation of token appearances in Figure 24 uncovers a substantial diversity in token distribution across different percentiles. The sudden ascent in Python's token usage, particularly at higher percentiles, manifests the language's flexibility. Unlike the more uniform appearance in C++ and Java, Python's versatility is captured in its wider token range.

This multi token distribution could be symptomatic of Python's multifunctionality, enabling applications from simple scripting to complex machine learning models. In the context of entropy, this diverse vocabulary and flexibility can contribute to higher entropy levels, reflecting more uncertainty and complexity in the language structure.

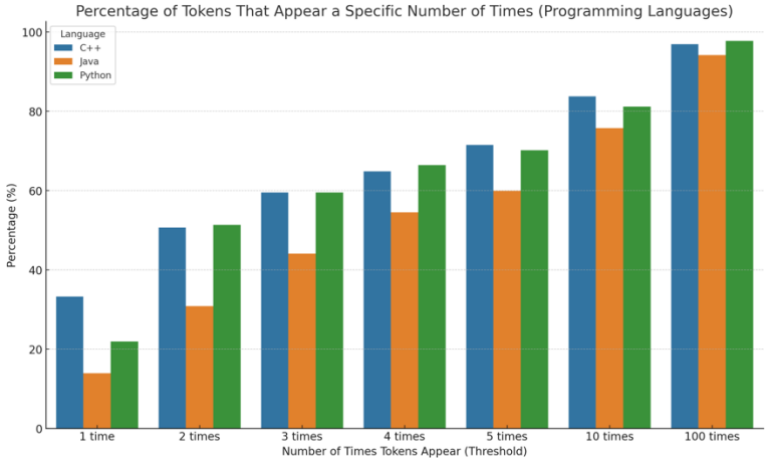

Figure 24: Programming Languages Token Appearances

4.5.3 Most Common Token

A understanding of common tokens emerges from Table 20 and Table 21. Punctuation tokens like semicolons and parentheses dominate across languages, underlining syntactic commonalities. But without punctuation, the landscape change to reveal specific language behaviors.

C++ and Java share similarities with tokens like 'int,' reflecting strong typing systems. In contrast, Python's distinct tokens, such as '#INDENT#,' paint a picture of a language leveraging readability and consistent formatting.

Table 20: Programming Languages Most Common Tokens

C++		Java		Python	
Token	**Count**	**Token**	**Count**	**Token**	**Count**
;	61 643 148	;	14 131 619	#NEWLINE#	21 950 825
)	47 744 432	(	14 041 894	(	20 352 520
(	47 744 189	)	14 041 894	)	20 352 520
,	29 480 871	.	8 328 793	,	10 312 573
=	21 186 016	=	6 974 724	=	8 945 726
i	20 953 945	}	5 971 600	:	7 698 500
[	20 866 156	{	5 971 567	[	7 307 700
]	20 866 132	[	5 535 193	]	7 307 700
}	19 969 725	]	5 535 193	#DEDENT#	6 602 325
{	19 966 174	int	4 778 662	#INDENT#	6 602 325

Table 21: Programming Languages Most Common Tokens without Punctuation

C++		Java		Python	
Token	**Count**	**Token**	**Count**	**Token**	**Count**
i	20 953 945	int	4 778 662	1	5 113 396
int	16 647 807	i	4 351 738	i	4 506 713
0	12 708 540	0	2 424 591	0	3 397 618
1	11 973 592	1	1 886 678	in	2 865 545
a	10 345 766	return	1 855 692	input	2 849 801
<<	8 550 269	new	1 802 766	int	2 764 065
n	8 106 056	public	1 734 869	for	2 708 116
return	7 739 657	a	1 704 859	print	2 682 850
if	7 087 148	n	1 604 409	if	2 607 931
ll	6 758 156	if	1 603 130	a	2 209 839

4.5.4 Unique Tokens

The insights from Table 22 and Figure 25 are multiple. C++'s substantial portion of numbers (42.15%) might mirror its use in numerical and performance-critical contexts. In contrast, Java's preponderance of variables (73.40%) could be indicative of its object-oriented paradigm, emphasizing encapsulation and structured programming.

Python's profile, balancing variables (47.98%) and numbers (43.92%), portrays a language of multiple domain application. The unique presence of comments in Python and absence in C++ and Java might be attributed to differences in commenting practices, documentation philosophies, or potential limitations in the data scraping methodology.

Note: As highlighted in the methodology the detection of variables, number, comments, and strings are not fully optimize and efficient, so the number above is a greater estimate but might not be accurate at the individual level.

Table 22: Programming Languages Number of Unique Token Identified

Language	Variables	Numbers	Strings	Comments
C++	381 977	511 807	7 4602	9
Java	105 975	16 866	13 803	0
Python	235 529	215 602	25 611	20 574

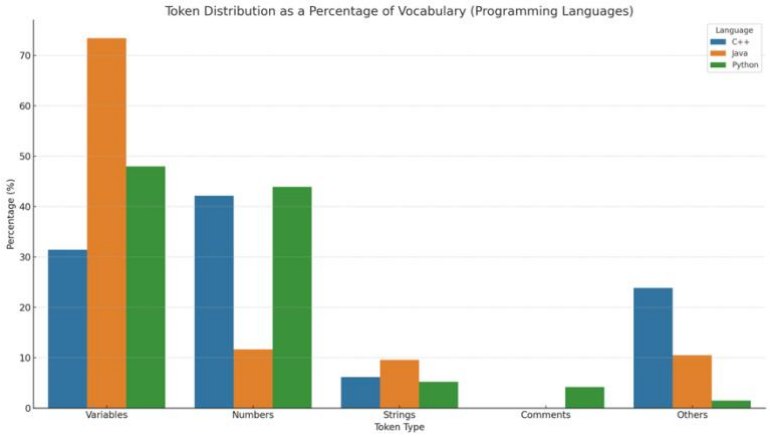

Figure 25: Programming Language Unique Token as a Percentage of Vocabulary

4.5.5 Implications

The observed characteristics offer profound insights into entropy estimation:

Complexity and Redundancy: C++'s lower TTR hints at more redundancy, potentially leading to lower entropy. Python's diversity may increase entropy, as variations and unpredictability grow.

Readability and Structure: Python's readability and human-like syntax might make it less noisy from a human perspective but introduce unique noise characteristics machine-wise. Java and C++ may present less human-like readability but more machine efficiency.

Syntactical Nuances: The unique tokens and their distribution across these languages showcase the inherent entropy differences due to syntax rules. The higher number of numbers in C++ may indicate more low-level operations, whereas Python's use of indentation tokens reveals its abstraction level.

5 RESULTS

This part will cover analysis and description of the result from the entropy estimation using the methodology detailed. A preliminary description of the result will be cover, a longer description including insights gained, criticisms and improvements will be covered in the discussion part.

5.1 Literature Books

This analysis within a controlled, non-noisy environment allows for a comprehensive examination of language complexity. By focusing on five specific literature books, the research captures insights that can be contrasted with noisier language forms like Twitter.

Note: It is imperative to note that these evaluations will not be reflective of the entirety of the English language's complexity, as the selection of only five books does not capture the breadth of the language's diversity. However, it does create a foundational basis for comparison with other datasets.

5.1.1 Word Token

The analysis begins by considering words as tokens, utilizing the Unigram model, Entropy Rate, and Prediction by Partial Matching (PPM) Entropy.

5.1.1.1 Unigram Entropy

The Unigram model, a straightforward approach, uncovers underlying complexity in the text by examining the occurrence of individual words (Figure 26):

- Alice in Wonderland: 8.31 Bits
- The Bible: 8.66 Bits
- The Great Gatsby: 8.98 Bits
- Les Misérables: 9.30 Bits
- Romeo and Juliette: 8.74 Bits

Here, the unit "Bits" refers to the binary logarithm of the probability associated with each word's occurrence. This value quantifies the unpredictability or information content in each word, with higher values indicating greater uncertainty.

The distribution represents the uncertainty associated with predicting each word in the text (at the individual level of each token).

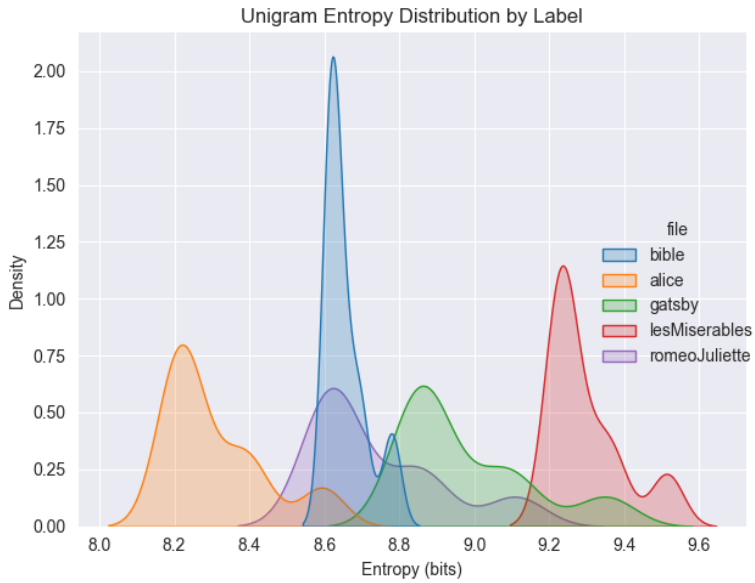

Figure 26: Literature Book Unigram Entropy Distribution

5.1.1.2 Entropy Rate

The Entropy Rate is evaluated by iterating over an increasing size of corpus, taking into account the previous knowledge and tokens.

The entropy rate values are:

- Alice in Wonderland: 5.63 Bits
- The Bible: 5.49 Bits
- The Great Gatsby: 6.31 Bits
- Les Misérables: 6.48 Bits
- Romeo and Juliette: 5.96 Bits

Figure 27 captures the rapid convergence of the entropy rate, indicating a stable measurement beyond a few thousand tokens.

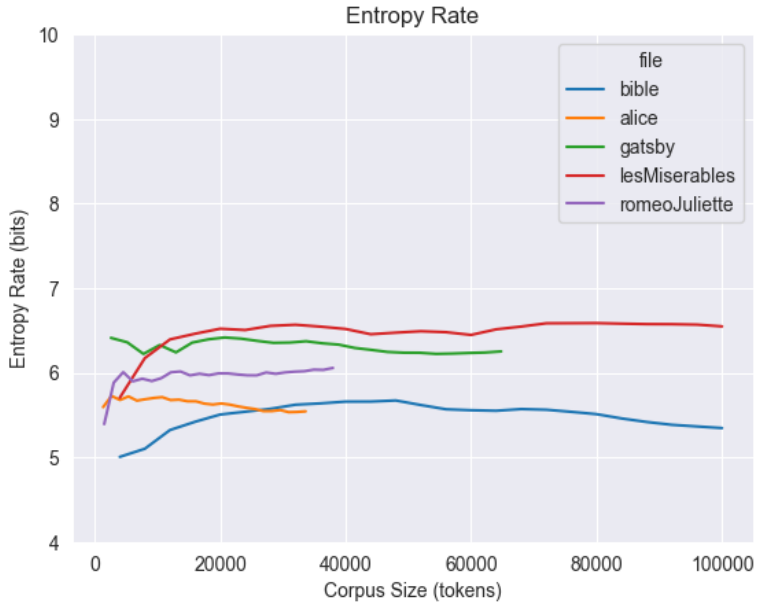

Figure 27: Literature Book Entropy Rate

The comparison between the entropy rate mean (5.97) and unigram mean (8.80) reveals an interesting shift in values. This shift between the unigram and entropy rate means indicates that considering a broader context (i.e., previous tokens) leads to a more nuanced and often lower entropy estimation. It emphasizes the role of context in shaping language's complexity and unpredictability.

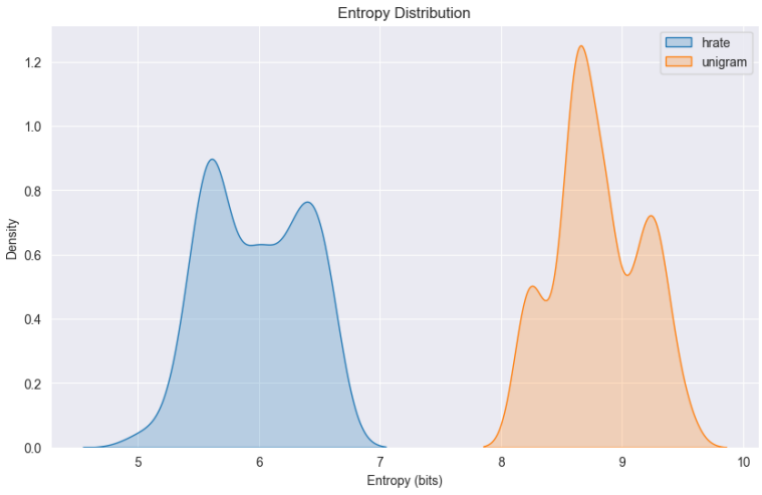

Figure 28: Literature Book Entropy Rate vs Unigram

5.1.1.3 PPM Entropy

Figure 29 offers the Entropy evaluation using a Prediction by Partial Matching by token. We observe a negative relationship, indicating a decline in entropy across tokens. This trend implies that as we gather more tokens (previous knowledge), our ability to predict future parts increases, thus lowering the entropy. It showcases the adaptive nature of language and how context is integral in understanding and predicting textual information.

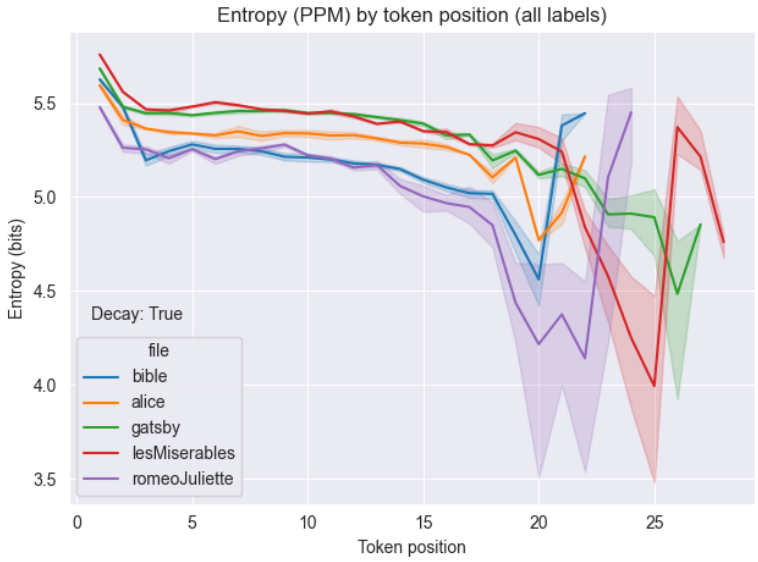

Figure 29: Literature Books PPM Entropy by Token

Figure 30 showcases a slightly skewed distribution on the right, with mean values ranging from 5.28 to 5.50 and standard deviation values between 0.27 to 0.41.

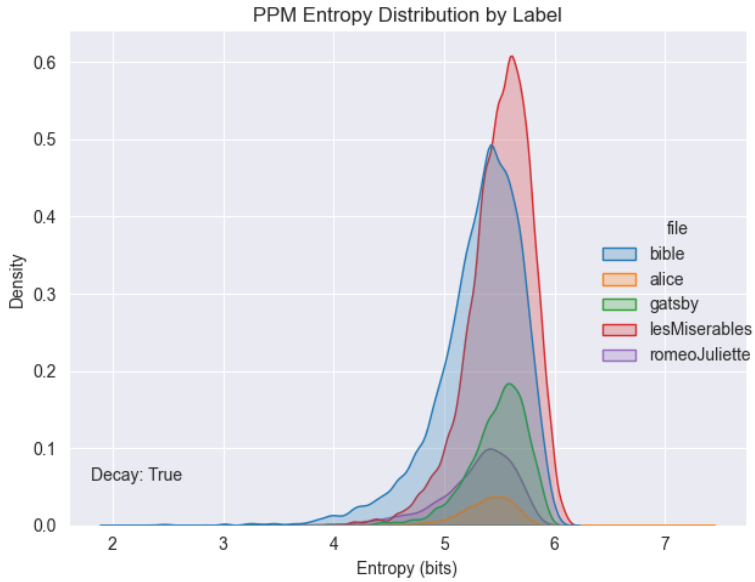

Figure 30: Literature Books PPM Entropy Distribution

and she tried her best to climb up one of the rabbit's little ! ' said the mouse , turning to alice . ` i wonder what they'll do next ! as for the fan and the pair of white kid gloves

Figure 31: Alice In Wonderland PPM Prediction of Text

and i ' d known tom buchanan and his girl and i went up together to new york — or not a little , as if she were balancing something on it which she and tom belonged .

Figure 32: The Great Gatsby PPM Prediction of Text

5.1.2 Character Token

Character token analysis explores the entropy at the granular level of individual characters, thus examining the complexity and unpredictability within a text. This

section details the results obtained from different entropy measurements and highlights key insights.

5.1.2.1 Unigram Entropy

Unigram entropy analysis focuses on the entropy of single characters within the text. The results (Figure 33) are as follows:

- Alice In Wonderland: 4.33 Bits
- The Bible: 4.34 Bits
- The Great Gatsby: 4.37 Bits
- Les Miserables: 4.31 Bits
- Romeo and Juliette: 4.38 Bits

The unigram entropy reveals subtle differences between the texts. With a mean of 4.34 and a standard deviation of 0.03. For instance, The Great Gatsby exhibits the highest unigram entropy (4.37), whereas 'Les Miserables' shows the lowest value (4.31), suggesting a more consistent pattern.

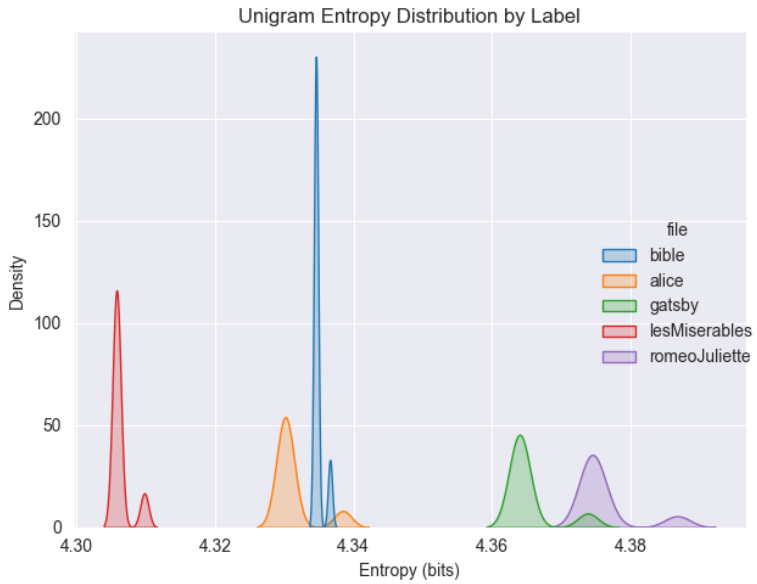

Figure 33: Literature Books Unigram Entropy Distribution (Char)

5.1.2.2 Entropy Rate

The entropy rate (Figure 34) reflects the average uncertainty per character. The results are as follows:

- Alice in Wonderland: 2.46 Bits
- The Bible: 2.13 Bits
- The Great Gatsby: 2.66 Bits
- Les Miserables: 2.54 Bits
- Romeo and Juliette: 2.60 Bits

The entropy rate has a mean value of 2.48 and a standard deviation of 0.20. Notably, The Bible possesses the lowest entropy rate, while The Great Gatsby demonstrates the highest. This contrasts with the unigram entropy values and may reflect differences in the compositional structure of these texts, such as sentence length and syntax. Figure 35 illustrates the relationship between the entropy rate and unigram entropy.

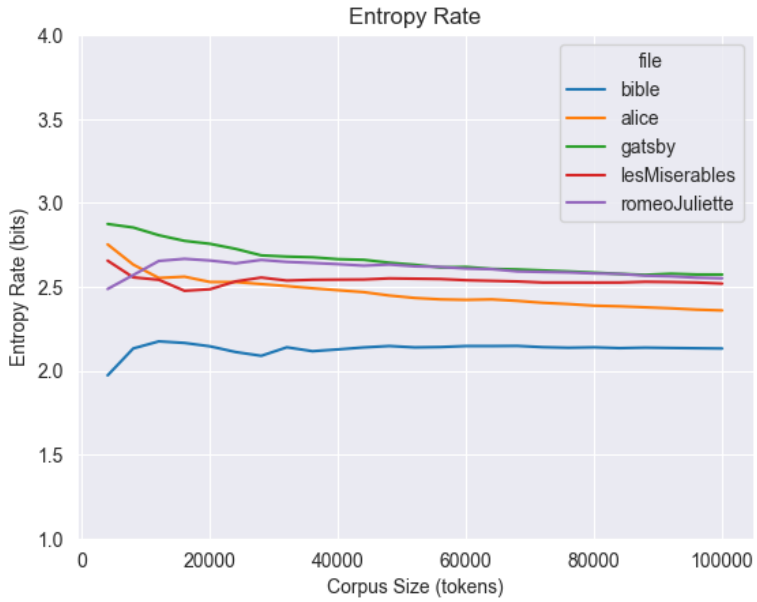

Figure 34: Literature Books Entropy Rate (Char)

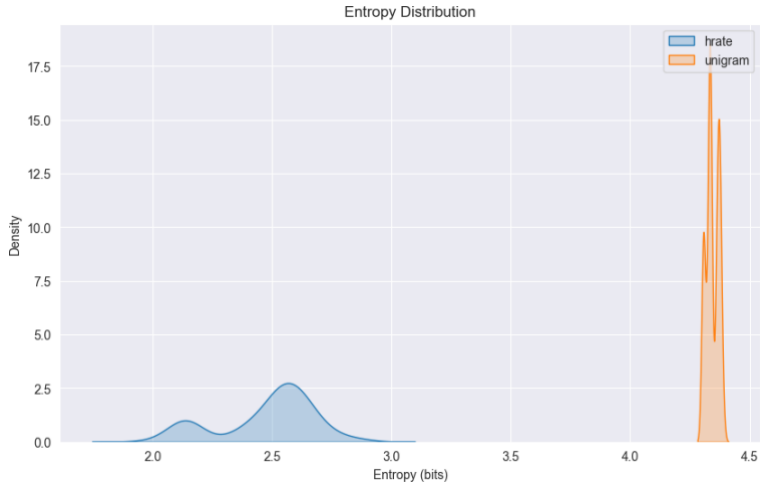

Figure 35: Literature Books Entropy Rate vs Unigram

5.1.2.3 PPM Entropy

The results of PPM entropy by token (Figure 36) and distribution (Figure 37) for each text are as follows:

- Alice in Wonderland: 2.97 Bits
- The Bible: 2.66 Bits
- The Great Gatsby: 3.04 Bits
- Les Miserables: 3.01 Bits
- Romeo and Juliette: 3.09 Bits

The trend data indicates a consistent negative trend in PPM entropy for most of the texts, except for Romeo and Juliette, which exhibits a slight positive trend. This may reflect a unique stylistic approach or lexical variety.

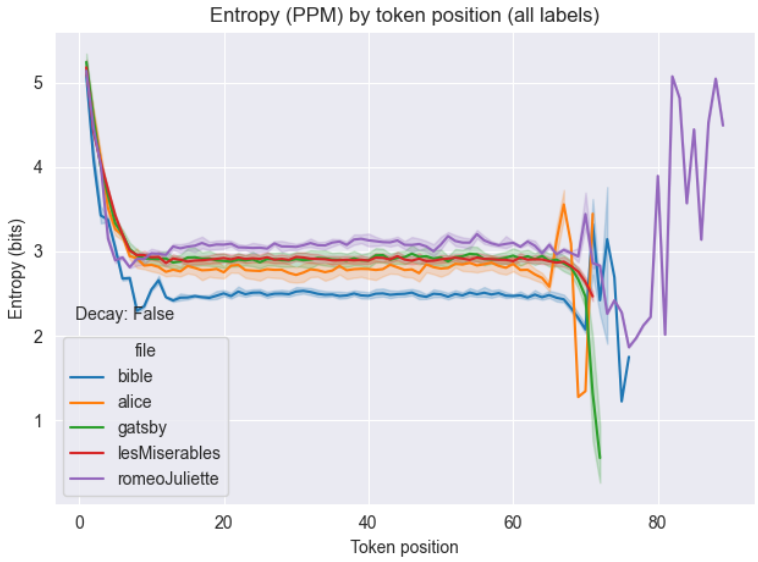

Figure 36: Literature Books PPM Entropy by Token (Char)

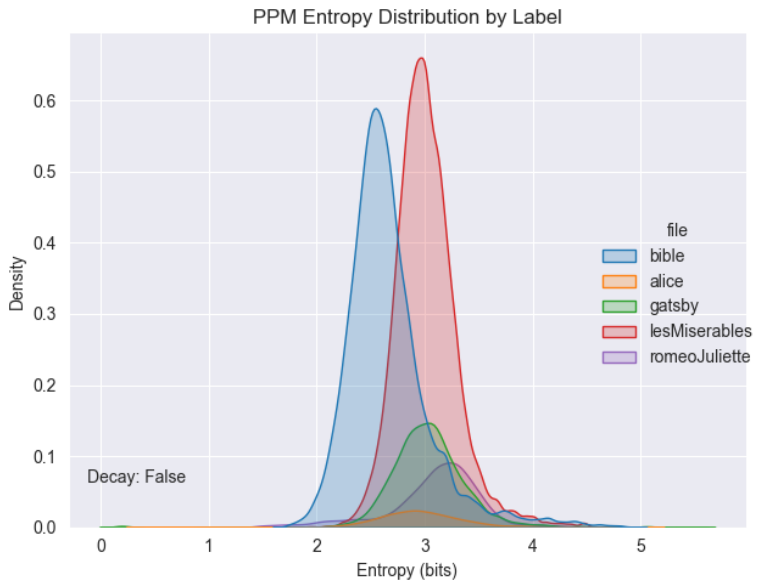

Figure 37: Literature Books PPM Entropy Distribution (Char)

5.1.2.3.1 Effect of Decay and Model Order

The analysis shows a significant effect of decay on the entropy measure. Specifically, the mean entropy with decay is 3.44 bits, compared to 2.87 bits without decay Figure 38). This increase in entropy when decay is true suggests a more complex and less predictable character structure in the literature books.

Decay in this context likely refers to a statistical property that reflects the aging or diminishing importance of older data. This might imply that considering decay allows the model to be more sensitive to recent character patterns, thereby capturing a richer and more nuanced understanding of the text.

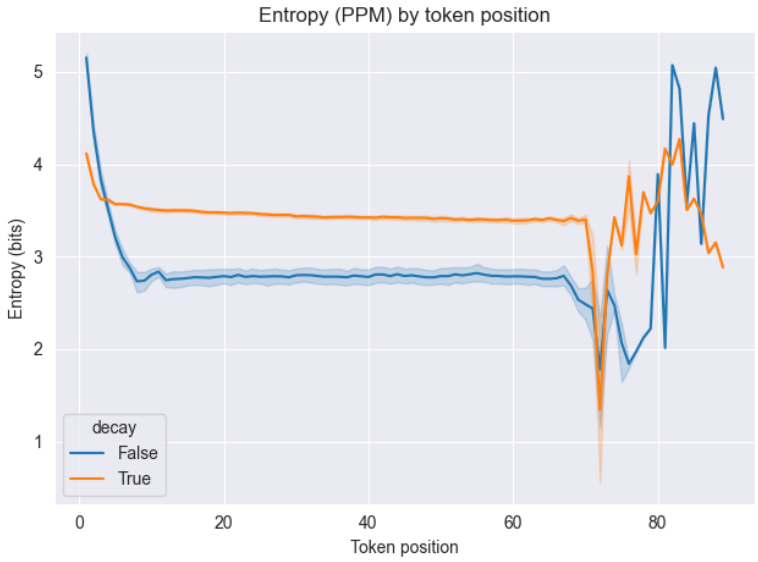

Figure 38: Literature Books PPM Entropy Decay Effect (Char)

The effect of model order on entropy is also evident, as shown in Figure 39. Without decay, the entropy rises until it reaches around 5 and then stabilizes.

However, with decay, the entropy continues to grow until it reaches 10, at which point it stabilizes.

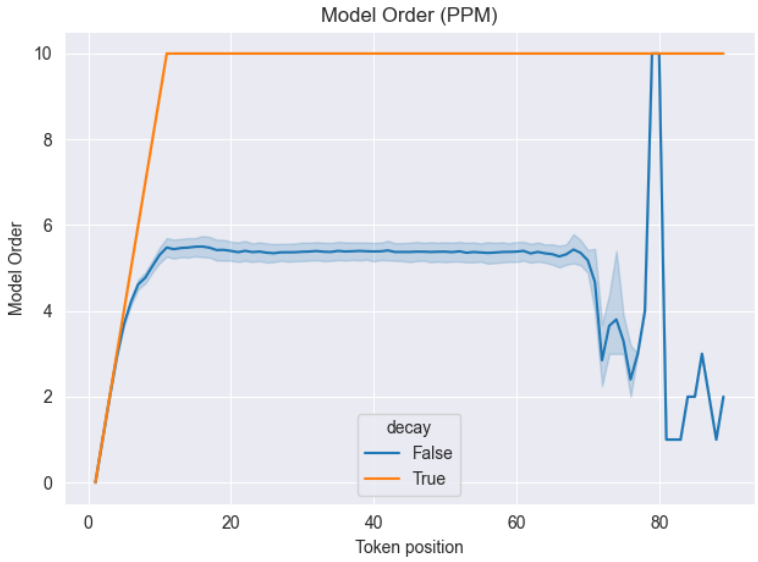

Figure 39: Literature Books PPM Entropy Model Order (Char)

5.2 Twitter Streams

The Twitter streams section offers a perspective on language entropy across three different languages: English, Spanish, and French. This allows to explore how the diversity of language is manifested in a noisy environment like social media, where the standard rules of grammar may not apply as strictly.

5.2.1 Word Token

In the word token analysis, punctuation was retained, reflecting the real-world textual structure found in Twitter streams. This decision aligns with the overall goal of the study to estimate language entropy in noisy environments.

5.2.1.1 Unigram Entropy

The Unigram Entropy for English, Spanish, and French was measured at 10.43, 10.64, and 10.89 bits, respectively. This data is captured in Figure 40, which

represents the unigram entropy's progression with an increasing number of tokens.

Interestingly, English started lower at approximately 5.5 bits for 50,000 tokens, while Spanish and French began around 7 bits. However, they all converged around the same number at approximately 300,000 tokens. This behavior suggests a common underlying structure among these languages, despite the initial differences.

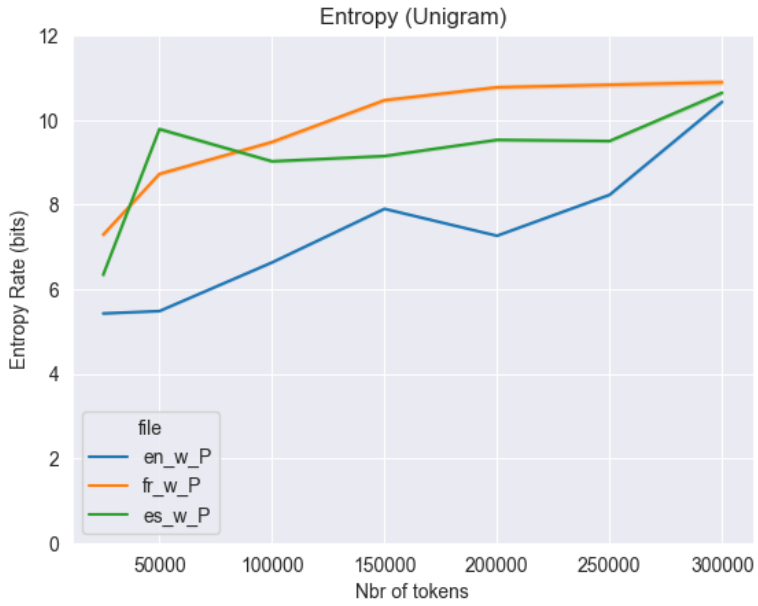

Figure 40: Twitter Streams Unigram Entropy

5.2.1.2 Entropy Rate

Figure 41 illustrates the quick and fast stabilization of the entropy rate over the increasing corpus size. The entropy rates were calculated as 6.63 for English,

7.59 for Spanish, and 7.44 for French. The low standard deviation from 40,000 tokens, as shown in Figure 42, reinforces the finding of rapid stabilization.

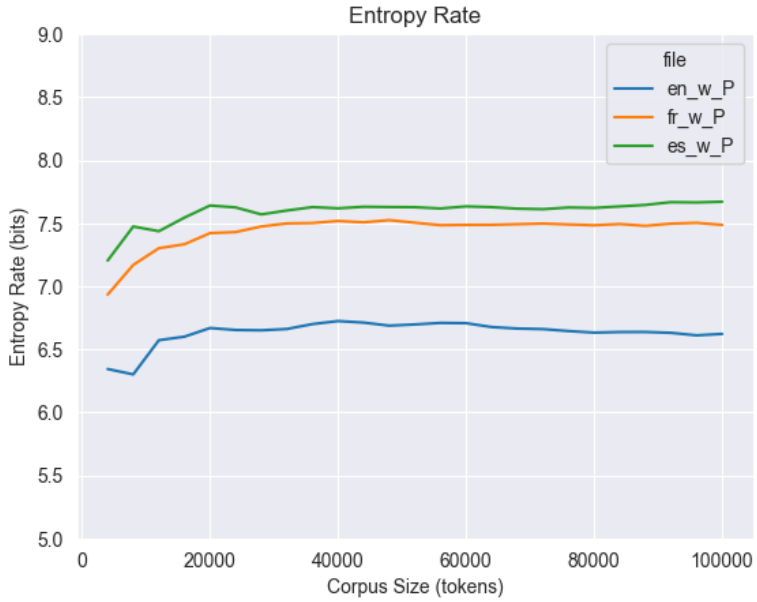

Figure 41: Twitter Streams Entropy Rate

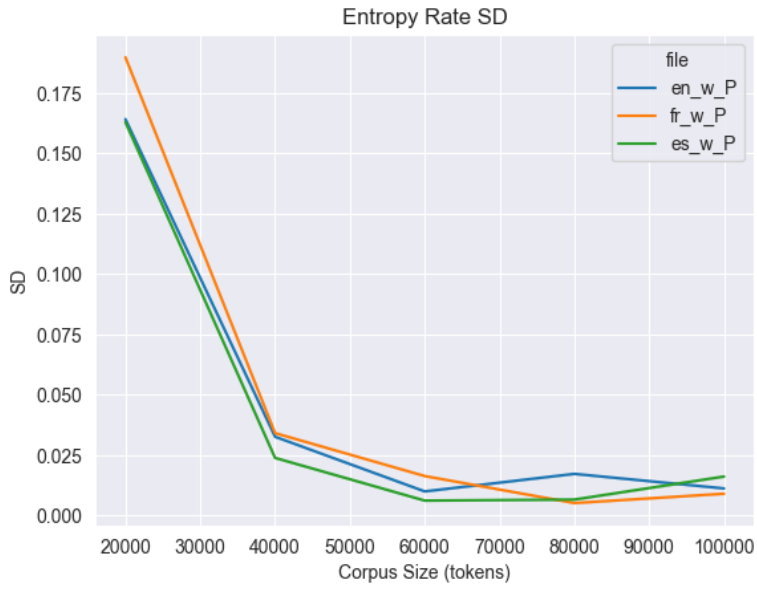

Figure 42: Twitter Streams Entropy Rate SD

The entropy rate ranges between 4 to 8 bits, whereas the unigram ranges from 6 to 11 bits (Figure 43). These numbers further elucidate the close relationships

between the languages in terms of entropy, with variations attributable to language-specific nuances and context taking into consideration.

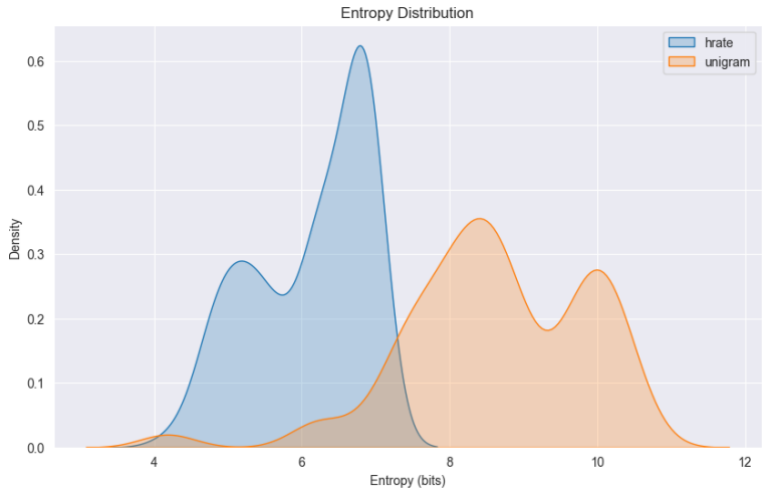

Figure 43: Twitter Streams Entropy Rate vs Unigram

5.2.1.3 PPM Entropy

The PPM Entropy results, depicted in Figure 44, provided more fine-grained insights into the data. The mean values for Spanish and French were 5.90 and 5.98, respectively, with standard deviations of 0.20 and 0.11. The trend indicated by the polyfit function for both languages showed a slight negative slope, underscoring a subtle but consistent decline in entropy as the number of tokens increased.

In contrast, the PPM Information Content, displayed in Figure 45, yielded mean values of 15.93 for Spanish and 15.94 for French, with standard deviations of 2.93 and 1.76, respectively. Here, the polyfit trend indicated a more noticeable negative slope, potentially reflecting the multifaceted nature of information content across different languages.

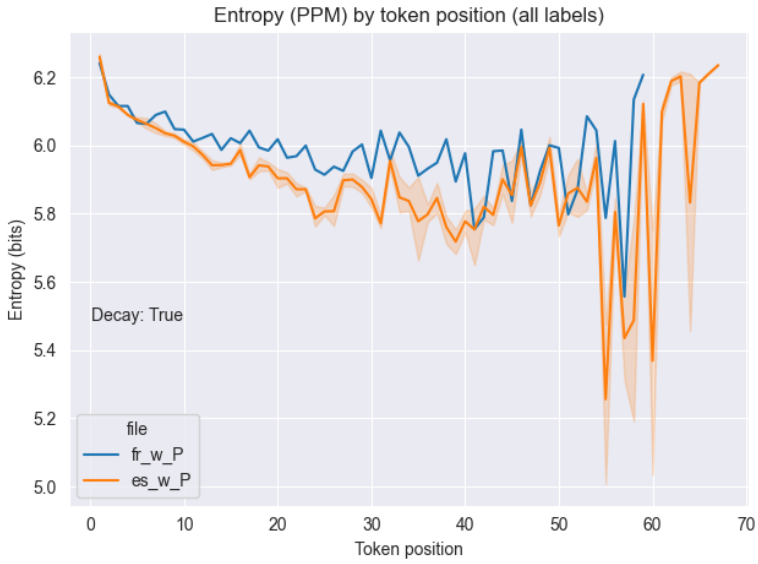

Figure 44: Twitter Streams PPM Entropy by Token

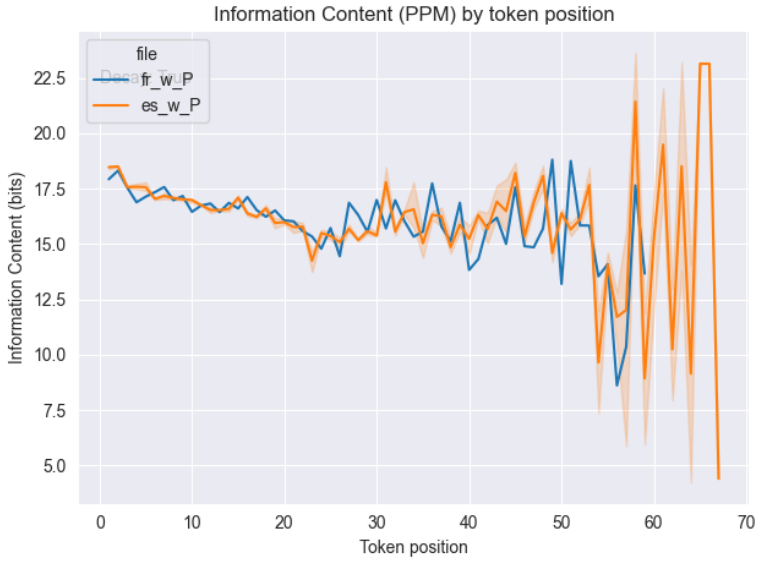

Figure 45: Twitter Streams PPM Information Content by Token

5.2.2 Character Token

The Unigram Entropy results indicate a relatively close value for both Spanish and French, with 4.26 and 4.33 bits respectively (Figure 46). These results align

with the theoretical expectations regarding the complexity and randomness of individual character occurrences within a language corpus.

- **Spanish**: 4.26 Bits
- **French**: 4.33 Bits

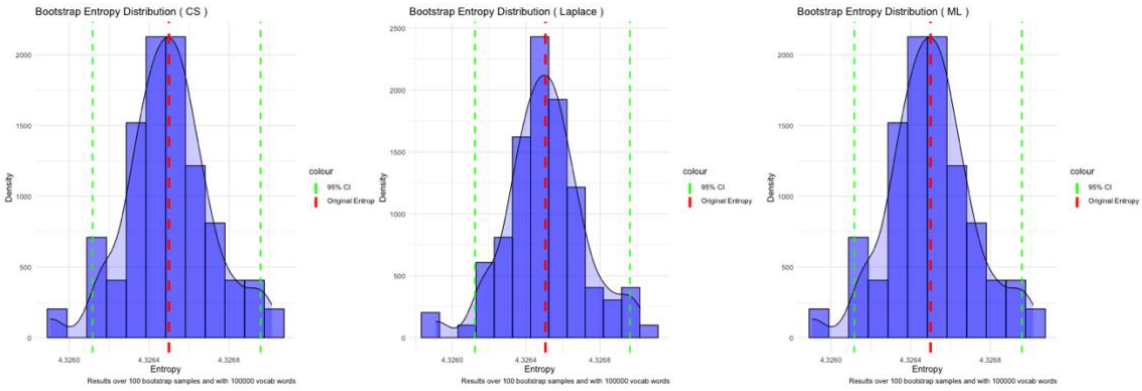

Figure 46: Twitter Streams Unigram Entropy Distribution for French (CS, Laplace, ML) by Char

The Hrate Entropy, or Entropy Rate, provides a more dynamic view of the system's entropy. Figure 47 illustrates the Hrate Entropy for Spanish and French as follows:

- **Spanish**: 2.71 Bits
- **French**: 2.73 Bits

A unique pattern is revealed in the graphical representation, where the entropy starts around 3 bits and lowers with the corpus size, stabilizing around 2.7.

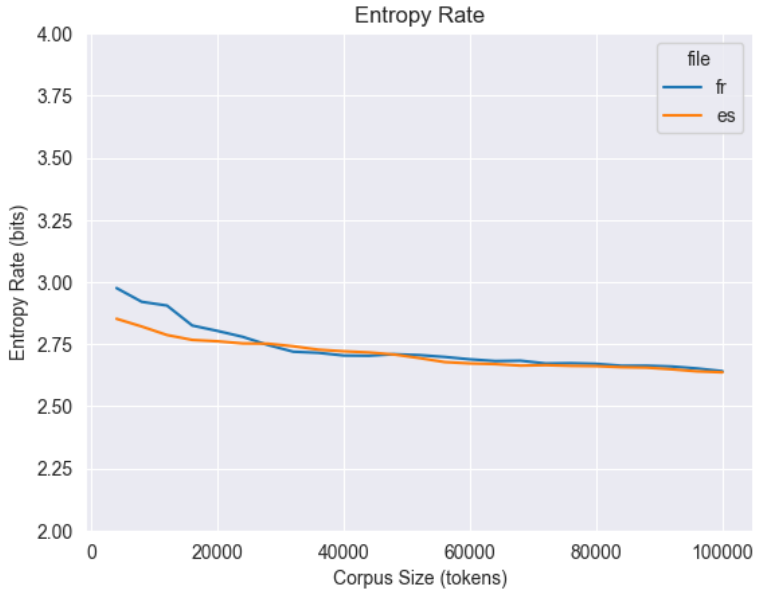

Figure 47: Twitter Streams Entropy Rate (Char)

5.2.3 Effect of Token

In this section, the focus is on understanding the role of individual tokens such as punctuation, accents, and emojis in the entropy estimation. This study highlights the impact of each type of token within a noisy environment like Twitter, considering different preprocessing techniques. Specifically, it consider four different configurations, as follows:

- Without punctuation, accent, and emoji (_)
- With punctuation (_w_P)
- With accent and emoji (_w_A_E)
- With punctuation, accent, and emoji (_w_P_A_E)

5.2.3.1 Unigram Entropy

The analysis of Unigram Entropy uncovers intriguing insights into the impact of individual tokens on language entropy. **Error! Reference source not found.** illustrates the Unigram Entropy in French for different token configurations. We observe that:

- Without punctuation, accent, and emoji, the entropy is 11.16 Bits.

- With punctuation, the entropy slightly decreases to 10.89 Bits
- With accent and emoji, the entropy is 11.40 Bits
- With punctuation, accent, and emoji, the entropy is 10.96 Bits.

Similar patterns are observed in Spanish and English, with respective entropies shown in the given data (Table 23). In all cases, the inclusion of accents and emojis tends to increase entropy, while punctuation has a decreasing effect.

Table 23: Twitter Streams Effect of Token Type on Unigram Entropy

Configuration	FR	ES	EN
w/o Accent/Emoji/Punctuation	11.16	10.96	10.87
w/ Accent/Emoji	11.40	11.21	11.05
w/ Accent/Emoji/Punctuation	10.96	10.92	10.63
w/ Punctuation	10.89	10.64	10.42

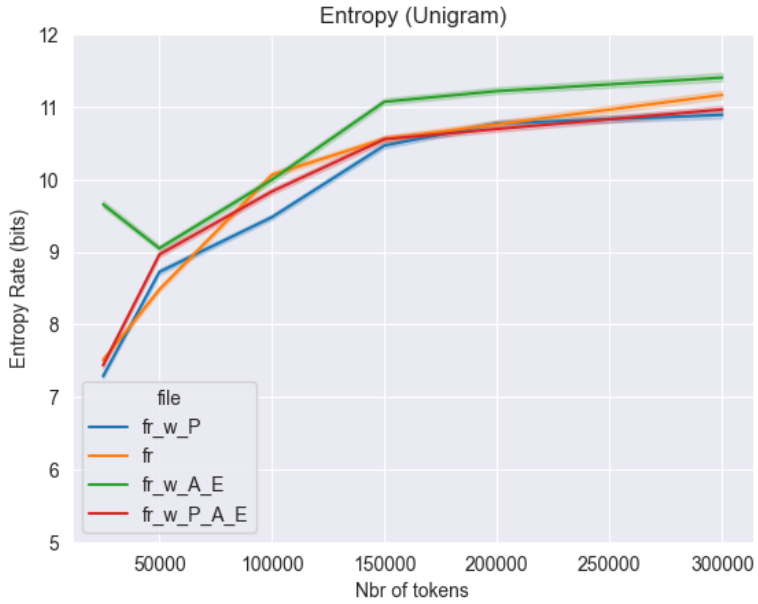

Figure 48: Twitter Stream Unigram Entropy French Effect of Token Type

5.2.3.2 Entropy Rate

The examination of Entropy Rate (Hrate Entropy) further elucidates the relationship between these tokens and the underlying entropy in a given language. Figure 49, Figure 50, Figure 51, and Figure 52 illustrate the results in French, Spanish, and English, revealing the following insights (Table 24):

- In French, the addition of accents and emojis increased the entropy rate from 7.51 to 7.67, while punctuation alone decreased it to 7.44, and the combination of all tokens resulted in an entropy rate of 7.37.
- In Spanish, similar trends are observed, with a marginal increase in entropy rate when accents and emojis are included (from 7.76 to 7.76), and a decrease with punctuation (7.59), followed by a moderate increase with all tokens (7.71).
- In English, the entropy rates follow a different pattern, with the lowest entropy observed when all tokens are included (6.53), and the highest without any tokens (6.78).

These findings emphasize the complex interplay between various tokens and entropy within a language. The results demonstrate that punctuation, accents, and emojis affect the entropy rate differently across languages, possibly reflecting cultural and grammatical nuances. It also highlights the need to consider these factors in computational models, especially in noisy environments like social media.

Table 24: Twitter Streams Effect of Token Type on Entropy Rate

Configuration	FR	ES	EN
w/o Accent/Emoji/Punctuation	7.51	7.75	6.78
w/ Accent/Emoji	7.66	7.76	6.76
w/ Accent/Emoji/Punctuation	7.37	7.70	6.53
w/ Punctuation	7.43	7.59	6.63

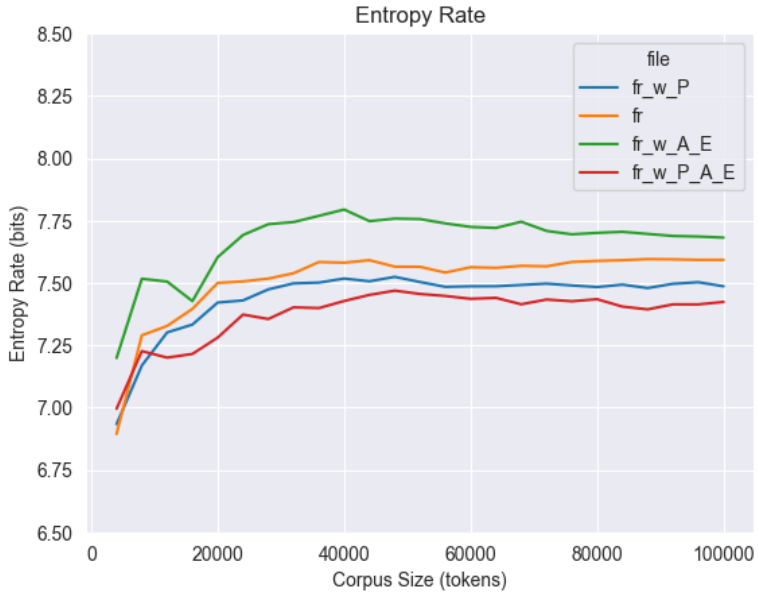

Figure 49: Twitter Stream Entropy Rate French Effect of Token Type

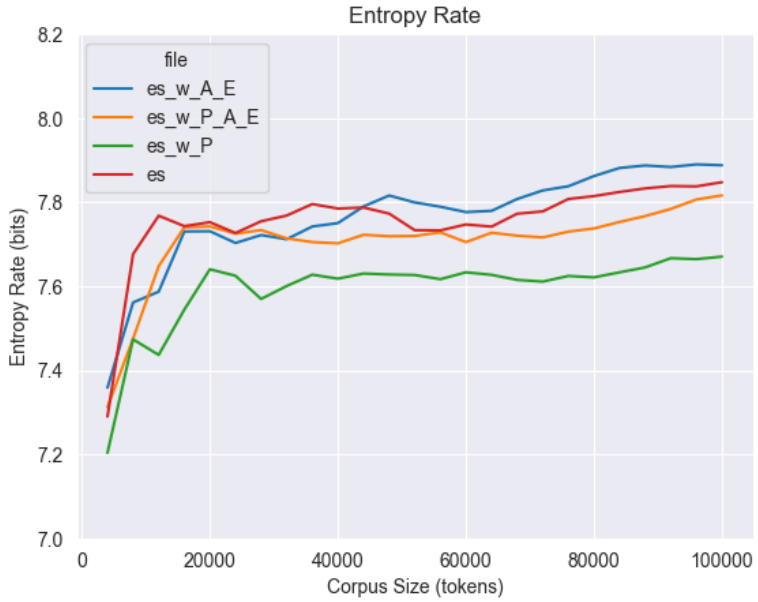

Figure 50: Twitter Stream Entropy Rate Spanish Effect of Token Type

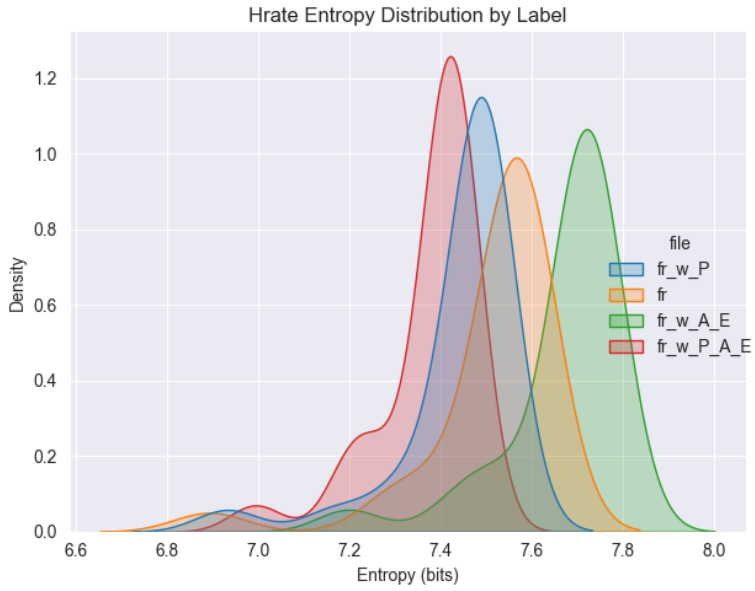

Figure 51: Twitter Stream Entropy Rate Distribution French Effect of Token Type

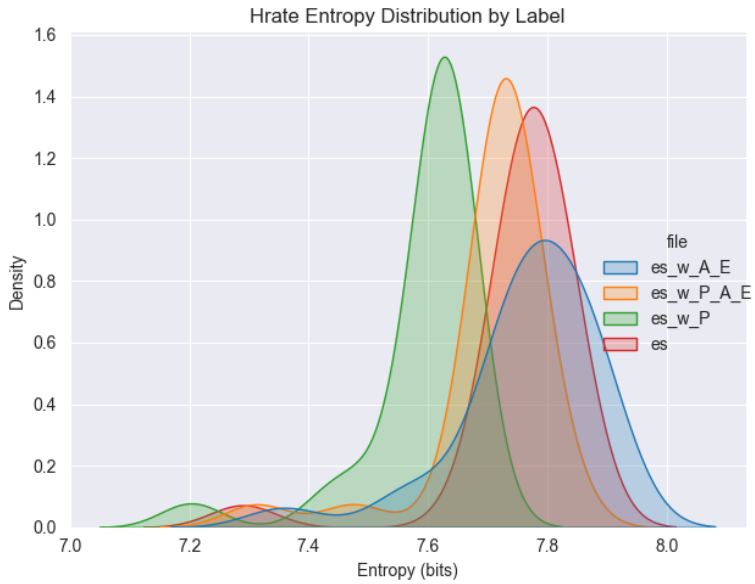

Figure 52: Twitter Stream Entropy Rate Distribution Spanish Effect of Token Type

5.2.4 Effect of Cluster

5.2.4.1 Analysis by Sentiment

The analysis of language entropy by sentiment clustering provides a distinct perspective on language complexity in a noisy environment. The following sections describe the unigram entropy and entropy rate (hrate) for French, Spanish, and English, focusing on the negative, neutral, and positive sentiment categories.

5.2.4.1.1 French Sentiments

: Figure 53illustrates the unigram entropy for the French language segmented by sentiment. Remarkably, the neutral sentiment exhibits a higher entropy value of 11.03 Bits compared to the negative (10.50 Bits) and positive (10.50 Bits) categories. This observation suggests that neutral texts tend to have a more complex and varied vocabulary structure.

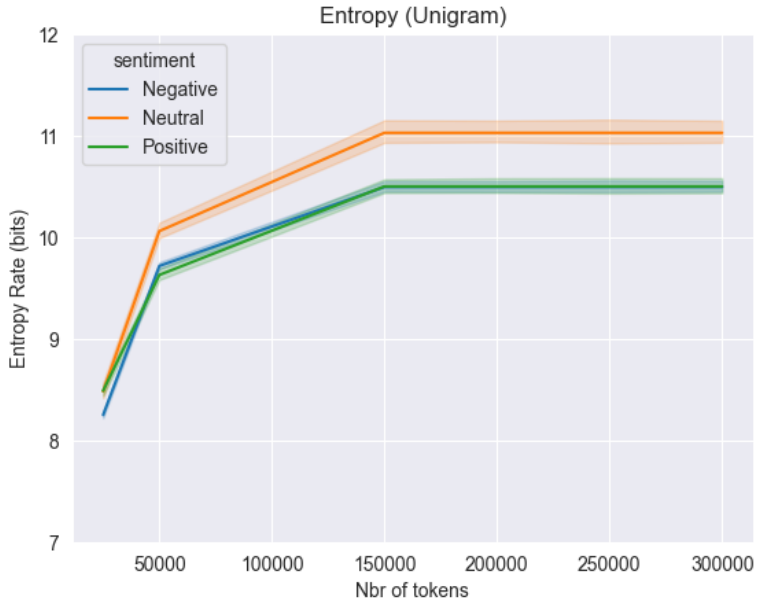

Figure 53: Twitter Stream by Sentiment French Unigram Entropy

The entropy rate (Figure 54) follows a similar pattern, with the neutral sentiment (7.08) being slightly higher than negative (6.93) and positive (6.89). This pattern emphasizes that the complexity and unpredictability within neutral sentiment texts

are consistent across different levels of analysis, and taken into account the internal context and structure of the language.

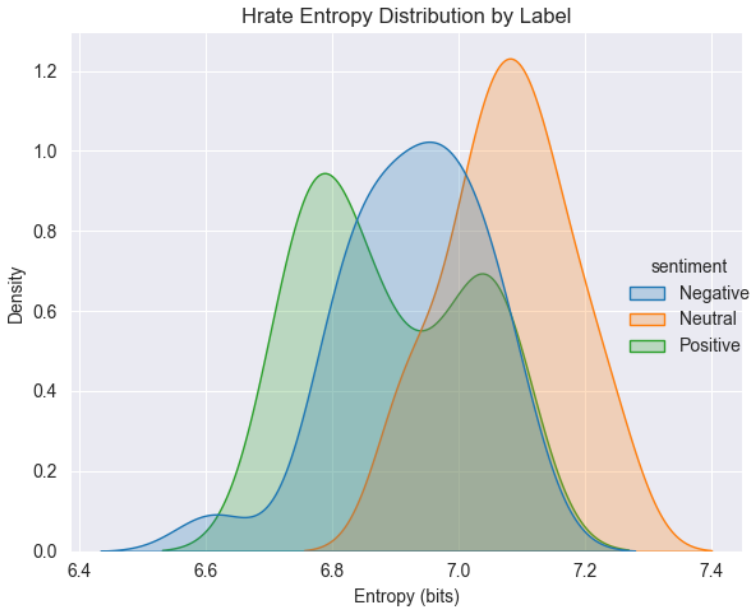

Figure 54: Twitter Stream by Sentiment French Entropy Rate Distribution

5.2.4.1.2 Spanish Sentiments

The Spanish unigram entropy (Figure 55) exhibits more uniform values across sentiments. However, an interesting trend is observed in negative sentiment, starting lower around 5.5 and converging to approximately 10.31 Bits. Neutral and positive sentiments remain relatively steady, at 10.47 and 10.25 Bits, respectively.

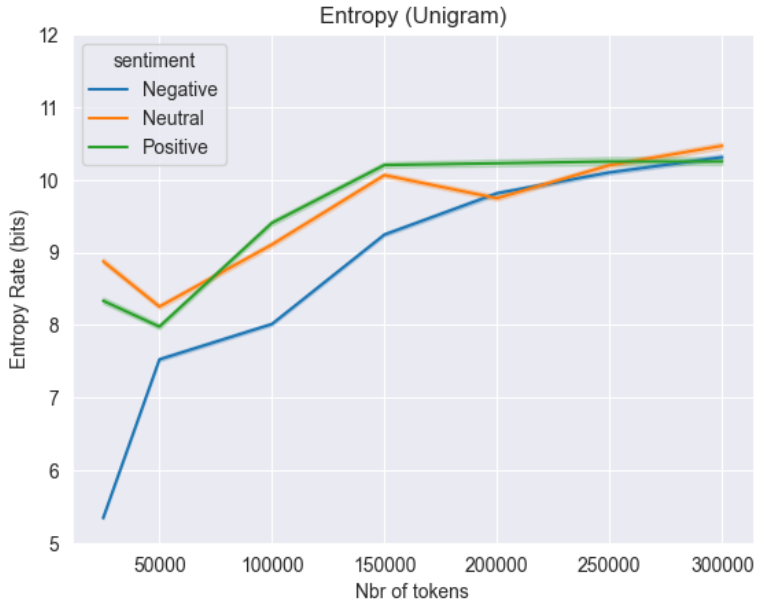

Figure 55: Twitter Stream by Sentiment Spanish Unigram Entropy

In terms of entropy rate (Figure 56), the positive sentiment seems to be distinctly lower (6.48) compared to negative (7.24) and neutral (7.22). This disparity might indicate that positive texts in Spanish follow more predictable patterns or employ a more concise vocabulary. The entropy rate with the context lower entropy suggest that internal patterns and structures repeat into positive languages.

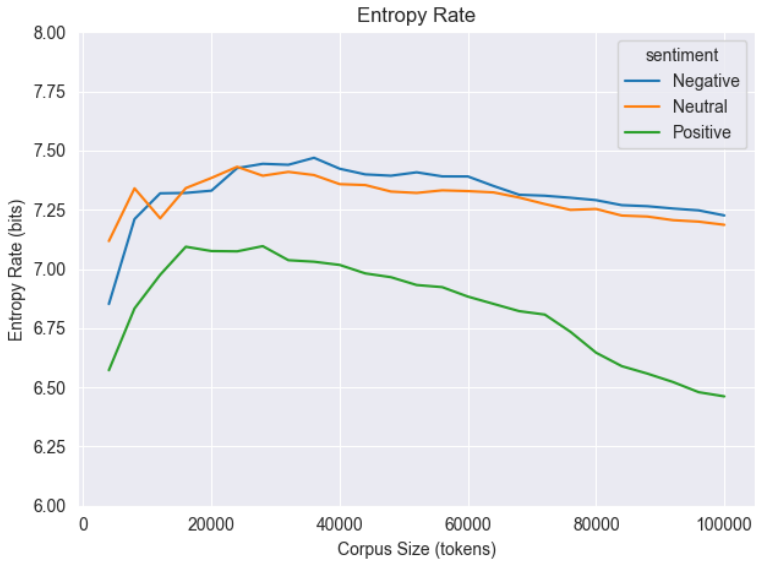

Figure 56: Twitter Stream by Sentiment Spanish Entropy Rate

5.2.4.1.3 English Sentiments

The English unigram entropy (Figure 57) portrays a more differentiated pattern, with neutral sentiment slightly taking longer to converge and resulting in a smaller entropy value of 9.61 Bits. Negative and positive sentiments stand at 10.55 and 10.21 Bits, respectively.

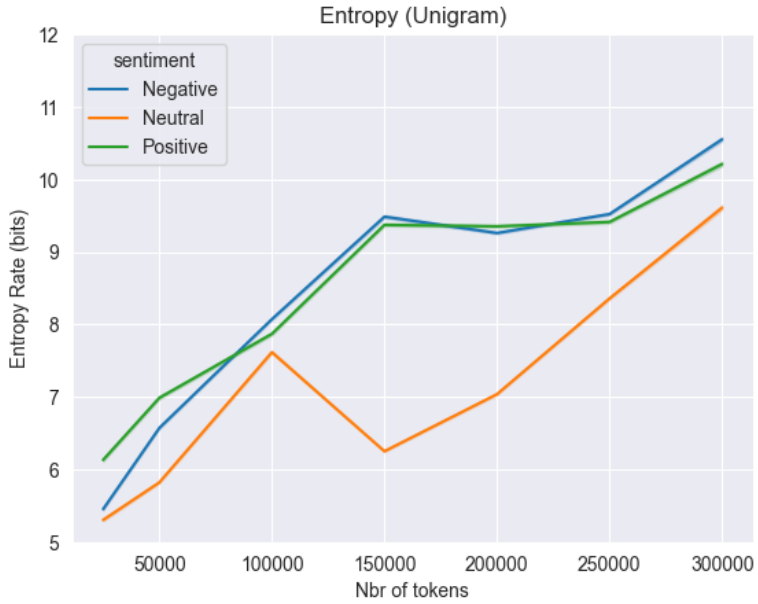

Figure 57: Twitter Stream by Sentiment English Unigram Entropy

Figure 58 presents the English entropy rate, revealing a considerable divergence between sentiments. Neutral converges to 5.5, positive around 6.5, and negative around 7.2. Such differentiation may reflect cultural language usage, structural variations, or thematic content disparities between the sentiments. The entropy

rate with higher divergence than the unigram may suggest structural differences that come with context whereas as vocabulary differences.

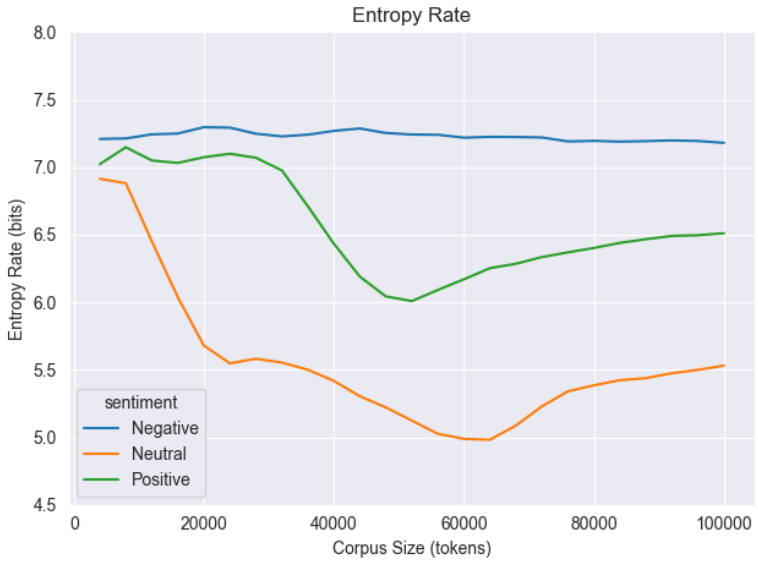

Figure 58: Twitter Stream by Sentiment English Entropy Rate

5.2.4.2 Analysis by Emotion

The following section describe the effect of emotions on English, Spannish and French.

5.2.4.2.1 English Emotions

In English, certain emotions exhibit a low standard deviation on unigram entropy, signifying a clear and less random entropy, with more structured and recurrent sentences and patterns (Figure 59). For example, 'fear' had the lowest standard deviation (0.65) compared to other emotions like 'anticipation' with a standard deviation of 1.55, indicating that expressions of fear tend to follow more consistent patterns. Surprise and Love also exhibit low standard deviation whereas optimism and anger exhibit a large one.

With unigram entropy values range from a low of 7.54 for anticipation to a high of 9.80 for fear. Notably, 'disgust' and 'sadness' both fall above the 9 Bits mark, suggesting a complex vocabulary structure within these emotions.

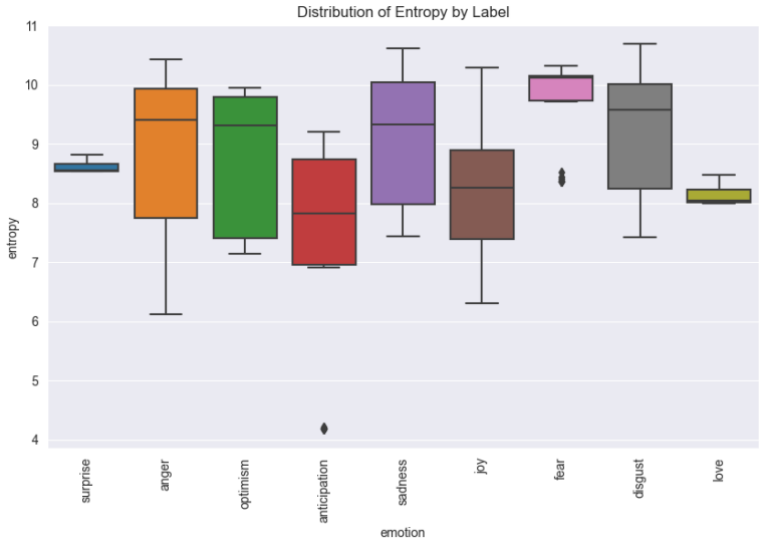

Figure 59: Twitter Streams Unigram Entropy of English Emotions

The entropy rate (Figure 60) presents another layer of understanding. The 'surprise' emotion has a significantly low entropy rate of 5.02, followed by 'anticipation' at 5.67. This could point towards the usage of more predictable patterns within these emotions. 'Anger,' on the other hand, had the highest entropy rate of 6.92, possibly reflecting a broader vocabulary and more complex internal structure.

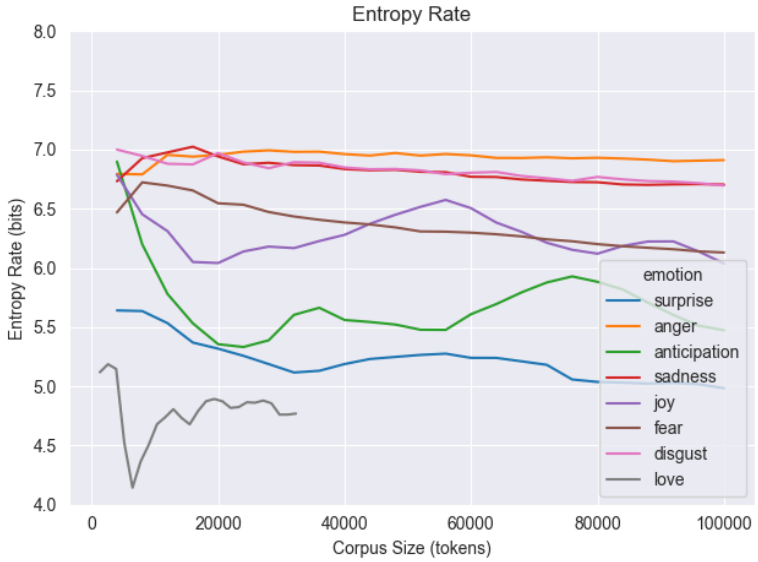

Figure 60: Twitter Streams Entropy Rate by English Emotions

5.2.4.2.2 French Emotions

In French (Figure 61), the unigram entropy values stay mostly consistent across emotions, with minimal variations. Interestingly, 'surprise' is an exception, with a mean of 8.68 and a standard deviation of 0.20, which might indicate specific characteristics of expressing surprise in French. 'Joy' has the highest standard deviation (1.05), perhaps denoting diverse ways of expressing happiness in French. As in English ‘surprise’ and ‘fear’ exhibit a low standar deviation.

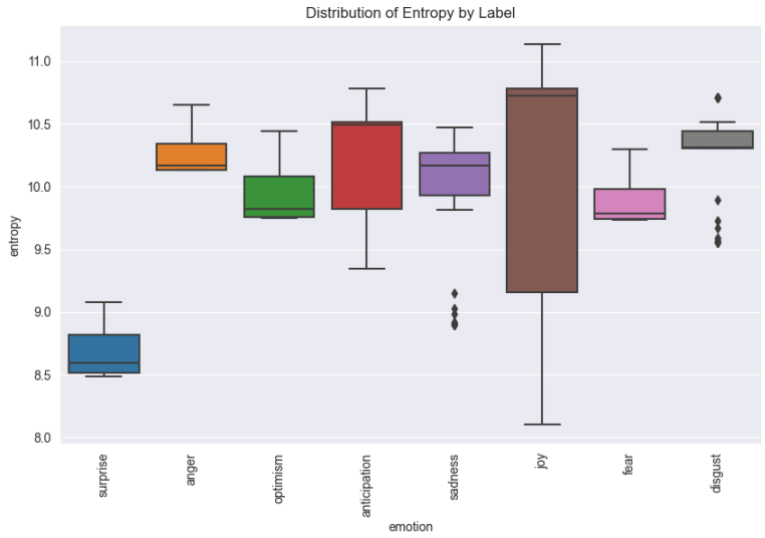

Figure 61: Twitter Streams Unigram Entropy of French Emotions

Compare to the overall French dataset, the unigram entropy remained relatively constant, whereas the entropy rate for French emotions is decreasing (Figure 62) The overall entropy rate for the French dataset is 7.44, while emotions like 'fear' and 'disgust' lie below this mark. This could highlight unique structural patterns in the French language specific to certain emotions.

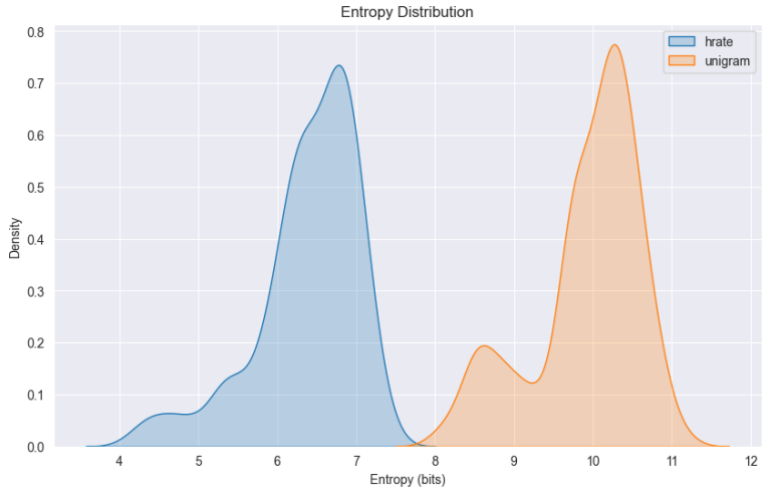

Figure 62: Twitter Streams French Emotions Entropy Rate vs Unigram

5.2.4.2.3 Spanish Emotions

The unigram entropy of Spanish emotions (Figure 63) shows a varied landscape, with 'anticipation' at 9.05 and 'anger' at 9.97. 'Surprise' stands out with a mean of 8.33 and a similar standard deviation (0.21), reflecting the peculiarity of this emotion in Spanish. As in French and English ‘surprise’ and ‘fear’ have low standard deviation. As French but on the contrary to English anger also exhibits a low standard deviation of unigram entropy.

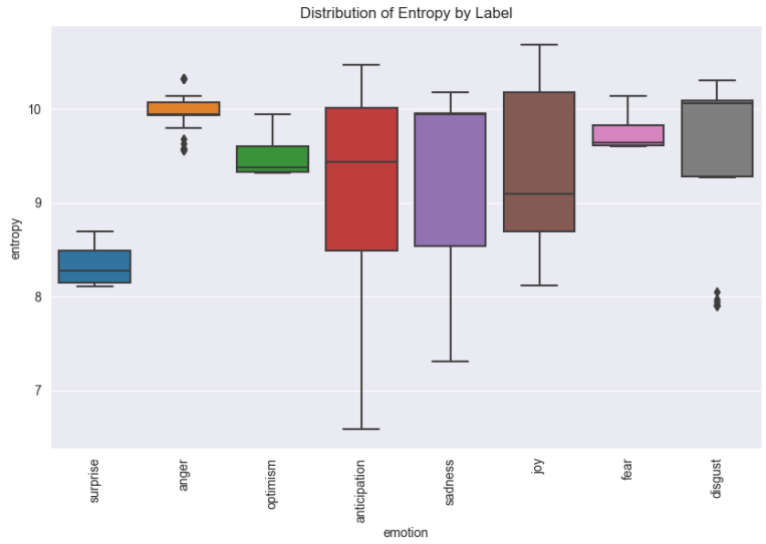

Figure 63: Twitter Streams Unigram Entropy of Spanish Emotions

The Spanish emotions' entropy rate (Figure 64) reveals intriguing insights. Emotions like 'joy' (7.37) and 'anticipation' (7.13) have higher entropy rates, potentially indicating a complex structure. In contrast, emotions like 'anger' (5.87) and 'fear' (5.64) have lower entropy rates, suggesting more predictable patterns.

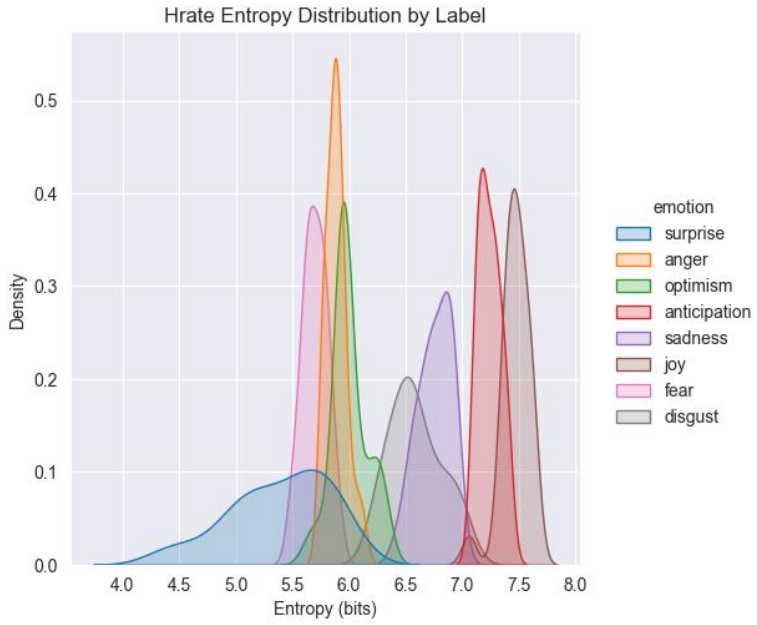

Figure 64: Twitter Streams Entropy Rate Distribution of Spanish Emotions

5.3 Covid-19 Timeline

In this section, we analyze the language entropy within tweets related to Covid-19, focusing on three main languages: English, Spanish, and French. The goal is to explore the evolution of entropy in a specific subject that has resonated with people across the globe. It begins with an overall analysis of the dataset before diving into a time-series examination.

5.3.1 Overall Analysis

5.3.1.1 Unigram Entropy

The Twitter Streams dataset represents a valuable comparison point for the noisy entropy of overall discussions on Twitter.

Table 25 presents the unigram entropy for Covid-19 related tweets across three languages. The unigram entropy in bits for English is 10.50, Spanish is 10.05, and French is 10.13. When compared to the broader Twitter Streams dataset, English shows a slight increase in entropy (+0.67%), while Spanish and French display a reduction of -5.54% and -6.97%, respectively (Figure 65).

This increase in English entropy suggests a potentially richer and more complex vocabulary structure in Covid-19 related discussions. The reduction in entropy for Spanish and French might indicate more consistent patterns or repeated structures.

Table 25: Covid-19 Tweets Overall Unigram Entropy by Language, with comparison of the Twitter Streams dataset

Language	English	Spanish	French
Unigram (Bits)	10.50	10.05	10.13
Diff. in % with Twitter Streams	+0.67%	-5.54%	-6.97%

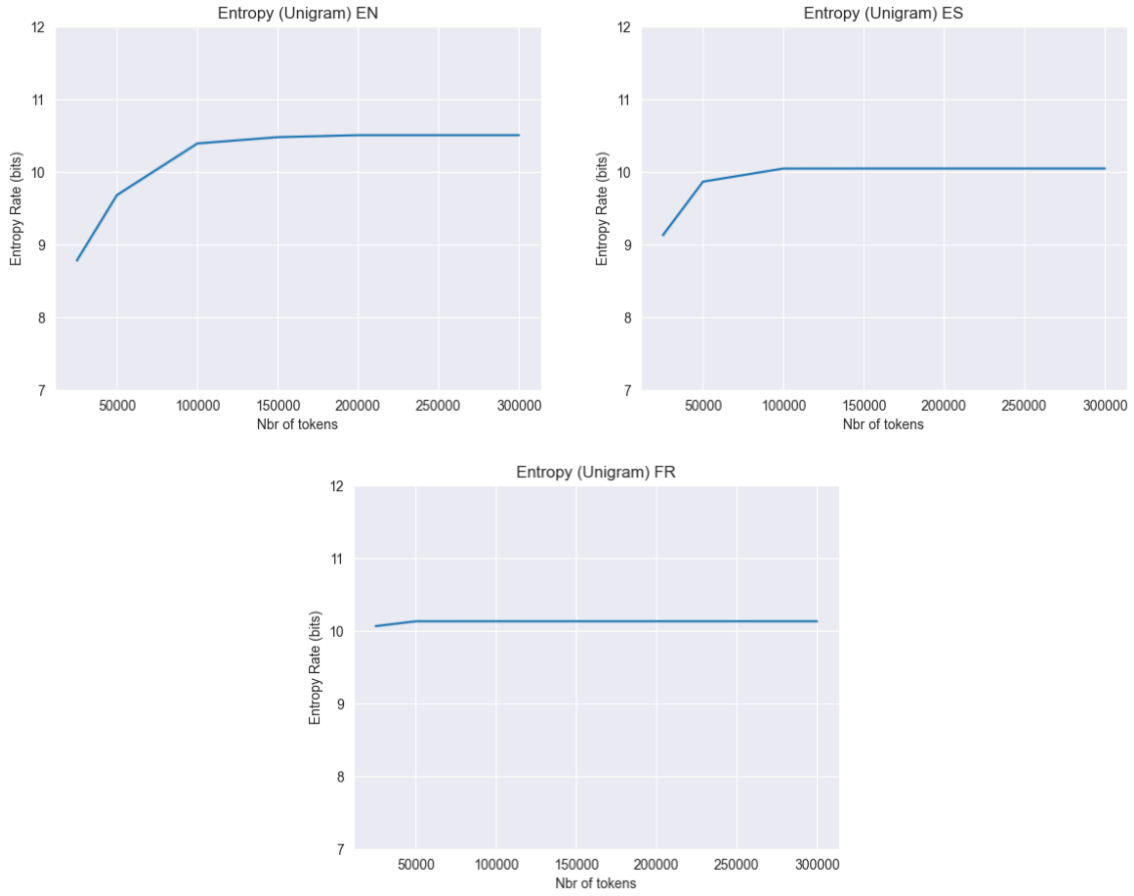

Figure 65: Covid-19 Tweets Unigram Entropy

5.3.1.2 Entropy Rate

Table 26 outlines the overall entropy rate by language for Covid-19 tweets. English has an entropy rate of 6.72 bits, Spanish 6.69 bits, and French 6.61 bits. These values represent a difference in comparison with the Twitter Streams dataset of +1.35% for English, and decreases of -12.08% and -11.86% for Spanish and French, respectively (Figure 66).

The increased entropy rate in English may align with the greater unigram entropy, possibly reflecting a more nuanced and varied discussion. The decrease in Spanish and French could be indicative of more predictable patterns within the language structure in these tweets.

Table 26: Covid-19 Tweets Overall Entropy Rate by Language, with comparison of the Twitter Streams dataset

Language	English	Spanish	French
Hrate (Bits)	6.72	6.69	6.61
Diff. in % with Twitter Streams	+1,35%	-12.08%	-11,86%

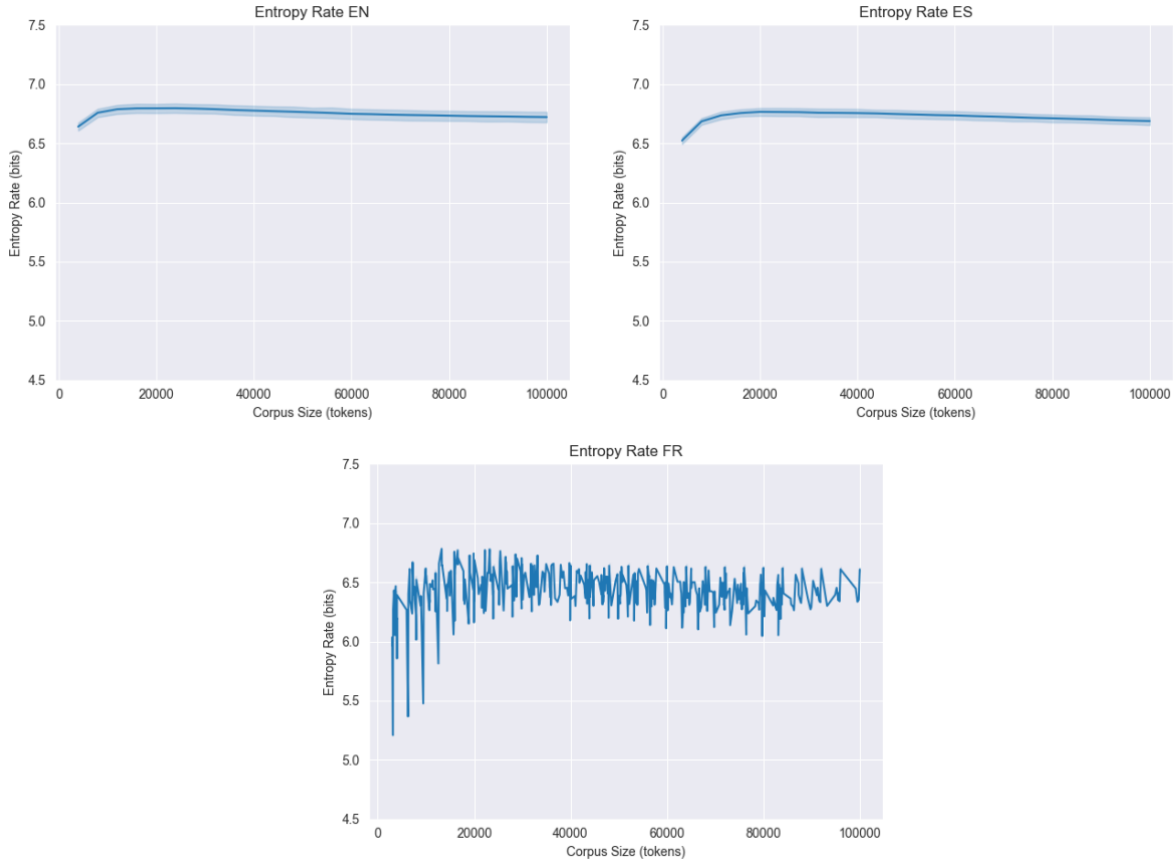

Figure 66: Covid-19 Tweets Entropy Rate

5.3.2 Time Serie Effect

5.3.2.1 Unigram Entropy

A time series analysis of the unigram entropy reveals interesting trends over time. The trend for French (FR) is -0.00043, Spanish (ES) is -0.00024, and English (EN) is +0.00034 (Figure 67).

These trends suggest that while English exhibits a slight increase in complexity over time, both French and Spanish show a decrease. This may signify evolving patterns of expression as the pandemic unfolds, possibly reflecting changes in public sentiment or discourse.

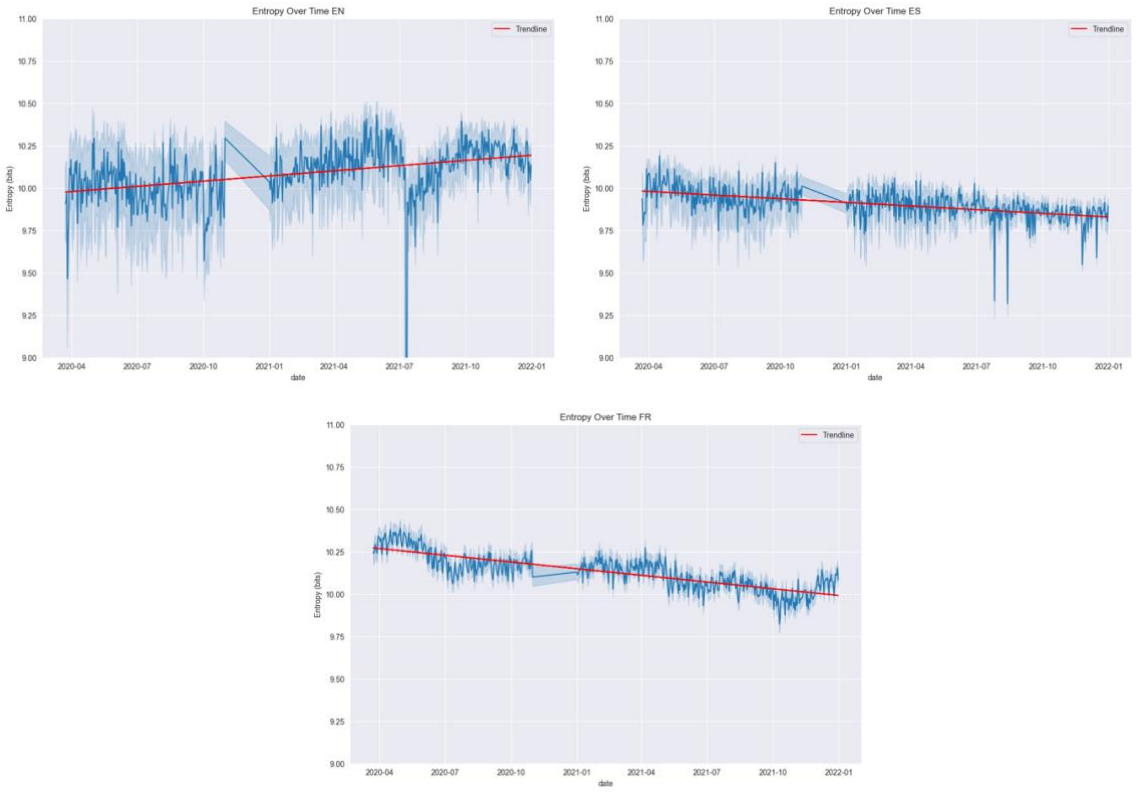

Figure 67: Covid-19 Tweet Unigram Entropy Over Time

5.3.2.2 Entropy Rate

The entropy rate trends over time for the three languages are more pronounced. The French trend is -0.00125, Spanish is -0.00165, and English is -0.00170 (Figure 68).

These negative trends in entropy rate may indicate a general trend towards more predictable language patterns over time across all three languages. This could be a result of the saturation of information or the solidification of public opinion.

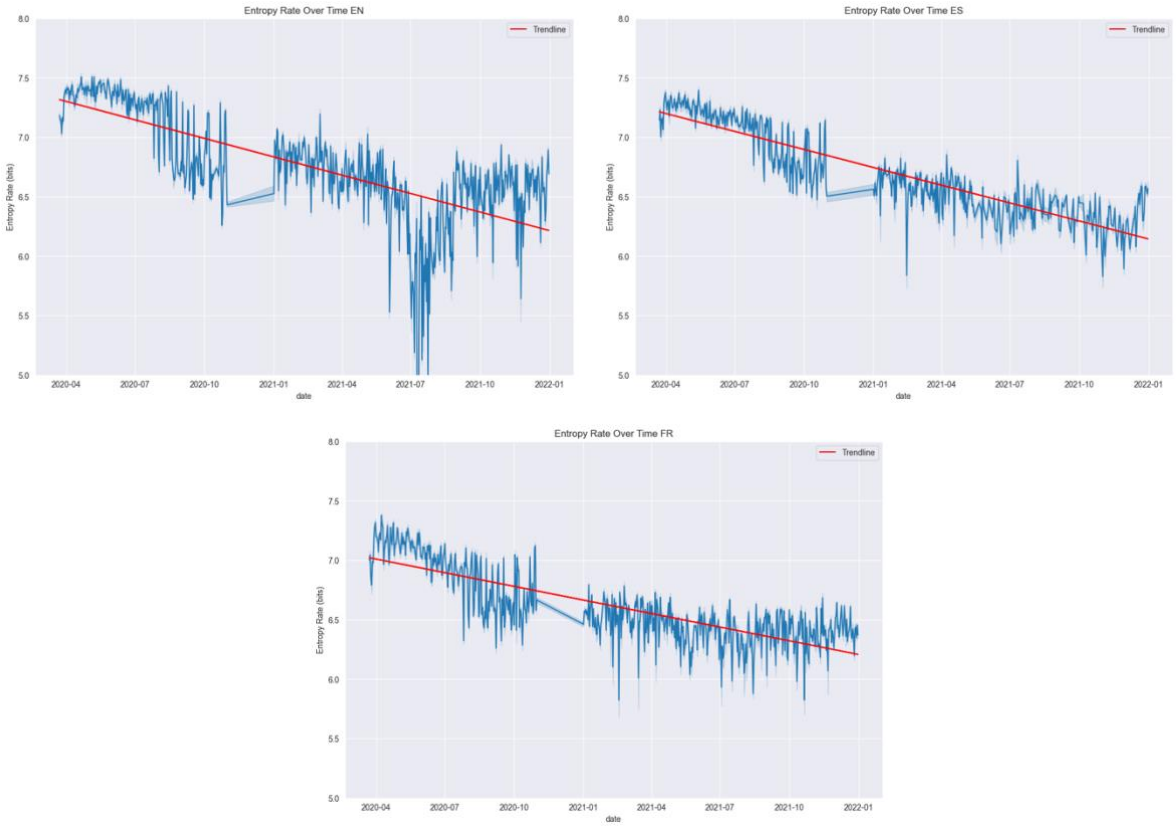

Figure 68: Covid-19 Tweet Entropy Rate Over Time

5.4 Target Users Tweets

In this section, we delve into the analysis of specific target users and events, focusing on the entropy measurements. We explore unigram entropy and entropy rate (Hrate) for news outlets, prominent individuals, and an international event.

5.4.1 News Outlets

News outlets analyzed include Guardian, BBC, NY Times, CNN, and Reuters.

The unigram entropy presents an interesting distribution across these outlets (Figure 69). Notably:

- **Guardian:** Highest unigram entropy of 11.42, indicating the richest vocabulary structure.
- **BBC:** An entropy of 10.98, closely followed by Reuters at 11.12.
- **NY Times and CNN:** Entropies of 10.69 and 10.63 respectively, implying a slightly less diverse vocabulary usage.

The standard deviation of the unigram entropy further suggests differences in the consistency of language use, with the NY Times showing the greatest fluctuation (std=0.534) and Reuters the least (std=0.263).

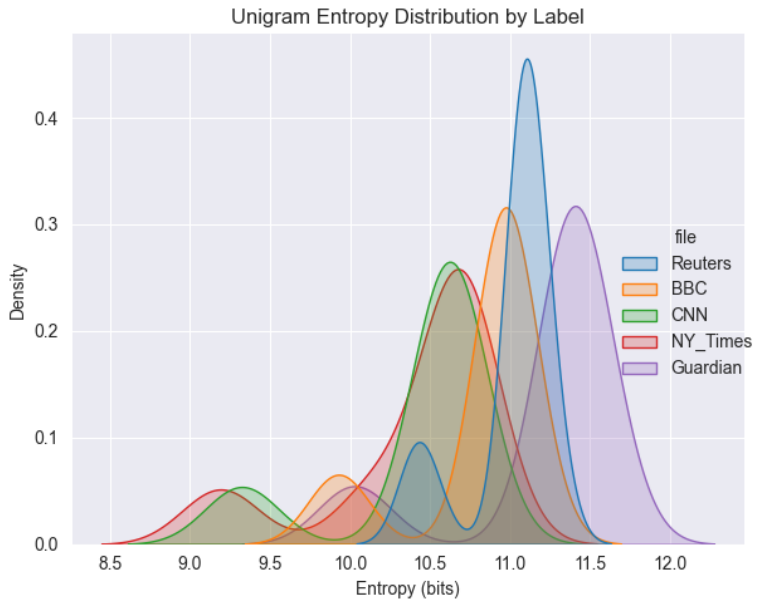

Figure 69: News Outlets Unigram Entropy Distribution

The entropy rate (Figure 70) reveals another dimension:

- **CNN:** Highest entropy rate at 7.67, reflecting more nuanced and varied discussion.
- **Guardian:** Lower entropy rate at 7.06 despite having the highest unigram entropy, suggesting more predictable patterns within the complex vocabulary.
- **Reuters:** Lowest at 6.48, possibly indicative of a more standardized reporting style.
- **BBC, NY Times:** Almost same entropy rate at 7.49 bits, between the CNN complexity and The Guardian.

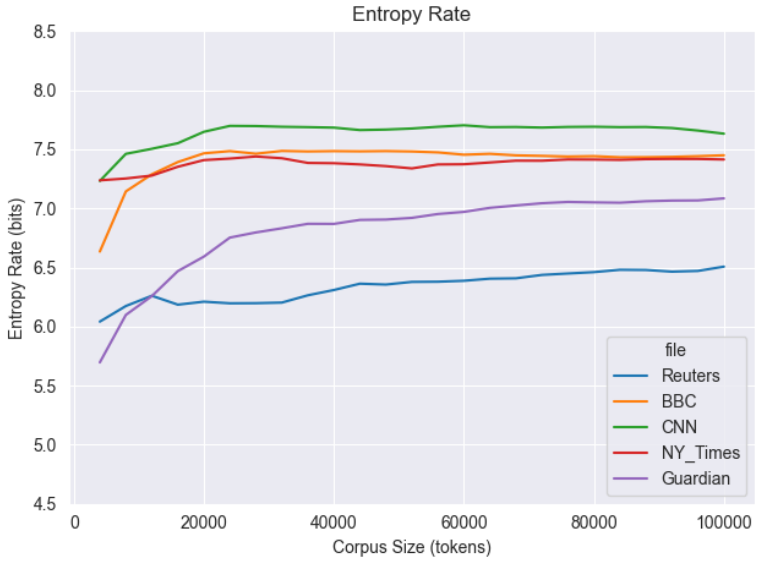

Figure 70: News Outlets Entropy Rate

5.4.2 Elon Musk vs Donald Trump

The comparison between Elon Musk and Donald Trump provides an interesting perspective on individual communication styles.

- **Elon Musk:** Higher unigram entropy of 10.16 bits (Figure 71), signifying a more varied vocabulary.
- **Donald Trump:** Lower unigram entropy of 9.69bits , indicating a more consistent or repetitive word choice.

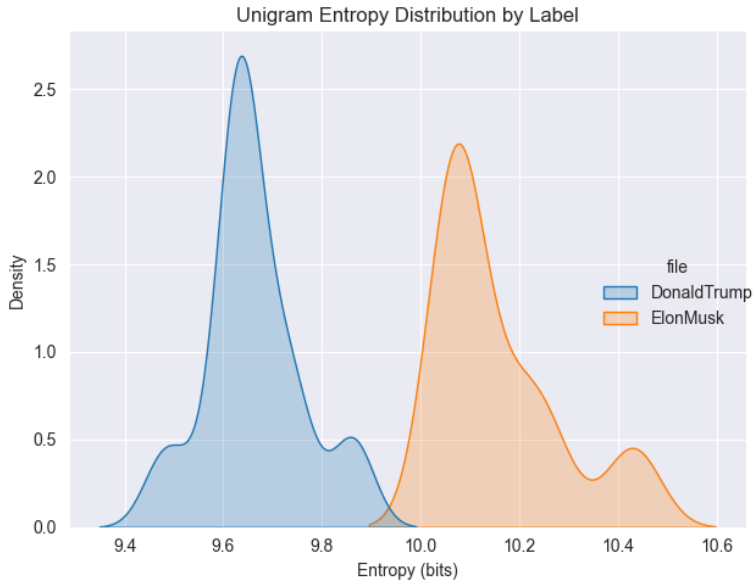

Figure 71: Elon Musk vs Donald Trump Unigram Entropy Distribution

- **Elon Musk:** Hrate entropy of 7.09, reflecting complexity in word sequences (Figure 72).
- **Donald Trump:** Lower Hrate entropy of 6.47, indicating more predictability in language structure.

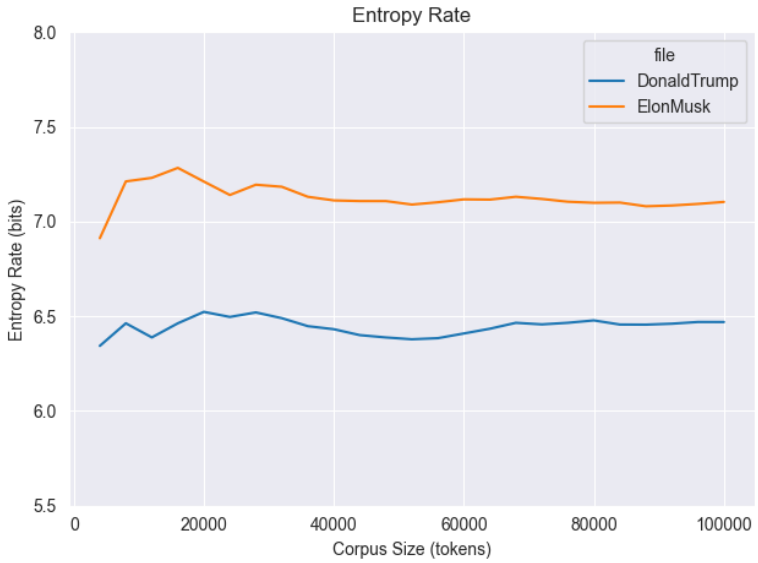

Figure 72: Elon Musk vs Donald Trump Entropy Rate

5.4.2.1 Effect of Unigram Estimators

An exploration of the effect of various unigram estimators (e.g., Laplace, Jeffreys Bayesians estimators, Chao Shen) reveals that different methods may lead to slight overestimation on smaller samples like individual accounts (Figure 73). The standard deviation across methods ranges from 0.240 (ML) to 0.313 (Laplace), demonstrating the need for careful selection and interpretation of estimators in entropy analysis.

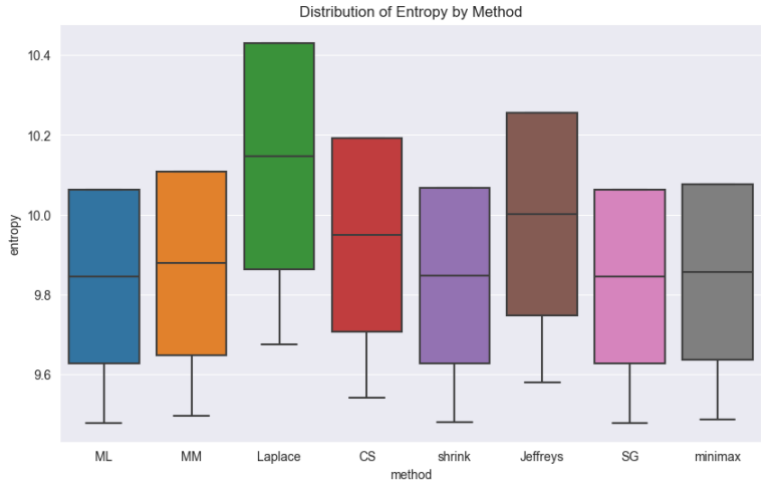

Figure 73: Effect of Unigram Estimators

5.4.3 Ukraine War

The analysis of tweets related to the Ukraine War presents some unexpected findings:

The unigram entropies for English (EN), Spanish (ES), and French (FR) converge around 10.30 - 10.42 bits (Figure 74). This surprising convergence may suggest a global unity in discourse, reflecting shared concerns and themes across language barriers.

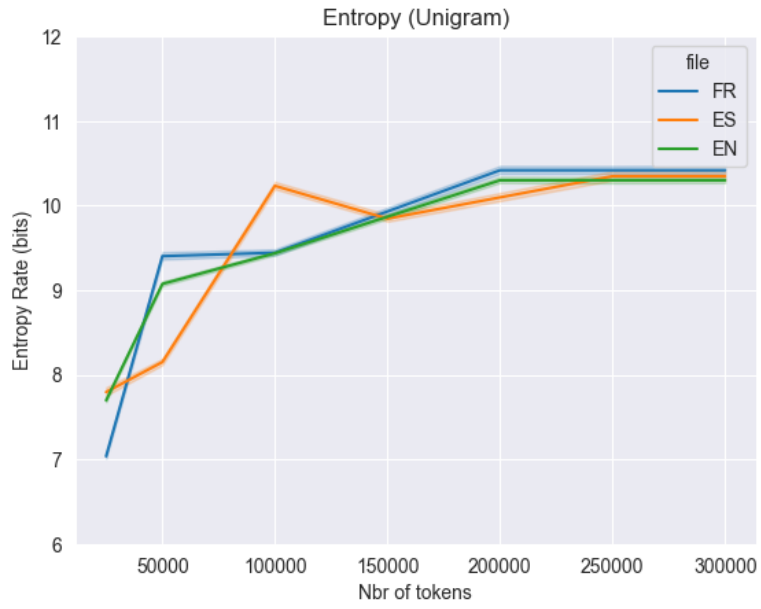

Figure 74: Ukraine War Unigram Entropy

The Hrate entropy (Figure 75) shows more divergence:

- **FR:** Highest at 7.45, suggesting more nuanced discussion in French.
- **EN and ES:** Very close values at 6.93 and 6.92, indicating similar levels of predictability in English and Spanish.

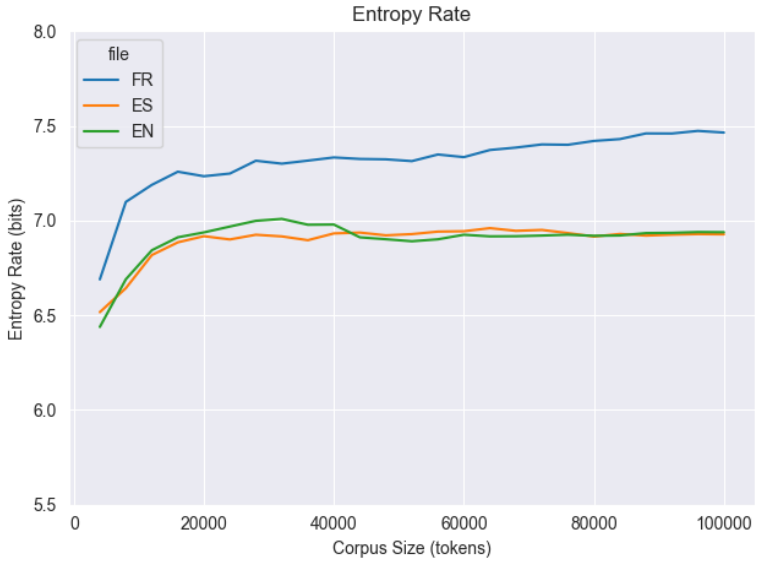

Figure 75: Ukraine War Entropy Rate

5.5 Programming Language

5.5.1 Language Analysis

In this section, the analysis is aimed at understanding the intrinsic characteristics of different programming languages: Python, C++, and Java. By examining these languages with variables, strings, numbers, and comments removed and replaced by specific tokens (e.g., #NUM#, #VAR#, etc.), we can gain insight into the language's structure without the interference of user input noise.

5.5.1.1 Unigram and Entropy Rate

The unigram entropy for the analyzed languages was found to be closely clustered, with C++ at 4.65, Java at 4.74, and Python at 4.60 (Table 27). This closeness in values may indicate a similarity in the richness and diversity of vocabulary structures across these languages.

Table 27: Programming Languages, Entropy Analysis per Language

Entropy	C++	Python	Java
Unigram (Bits)	4.645	4.603	4.736
Entropy Rate (Bits)	1.012	0.801	0.791

The hrate entropy revealed more nuanced differences among the languages (Figure 76). Specifically, C++ exhibited the highest entropy rate at 1.01 bits, followed by Python at 0.79 bits, and Java at 0.80 bits. The mean and standard deviation for hrate were 1.01 and 0.32, respectively.

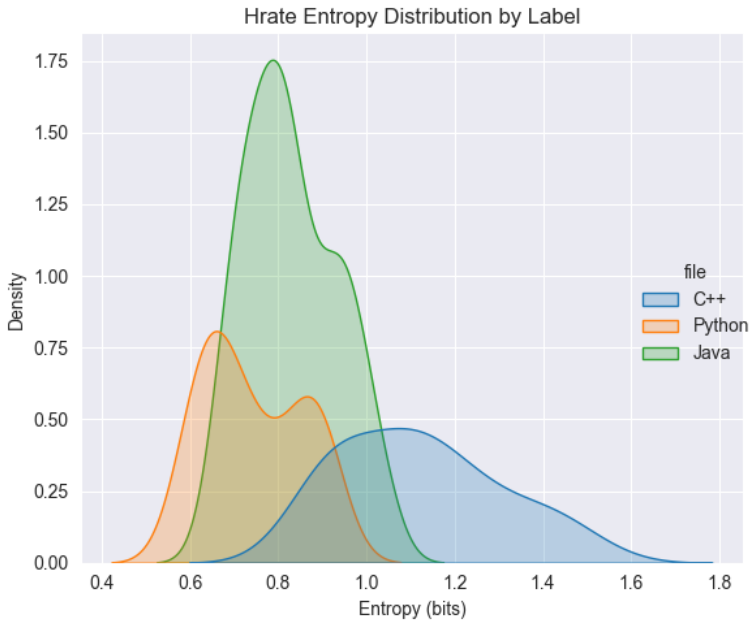

Figure 76: Programming Language Entropy Rate by Language

5.5.1.2 Entropy Rate vs Unigram

In examining the hrate and unigram entropy across the programming languages of Python, C++, and Java, a substantial shift in entropy values is observed when considering the context and not merely evaluating entropy at the individual token level.

The mean unigram entropy was found to be 4.63 with a standard deviation of 0.91. In contrast, the mean hrate entropy was significantly lower at 1.01, with a standard deviation of 0.32 (Figure 77). This stark difference between the two measures illuminates the sensitivity of entropy calculations to the inclusion of contextual information.

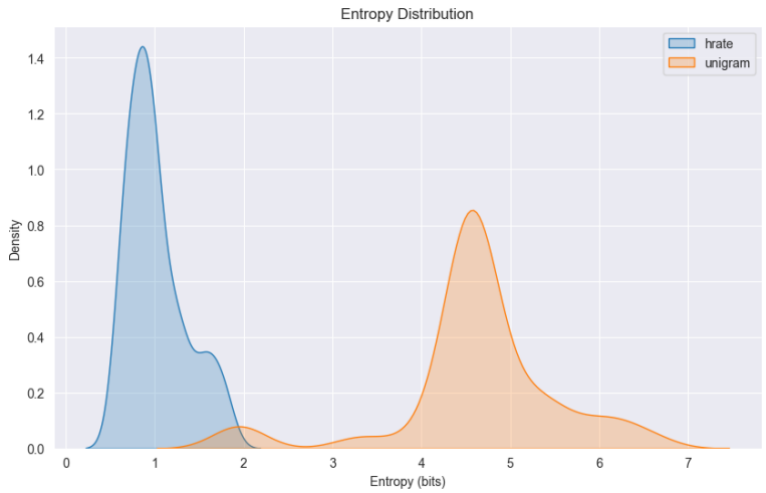

Figure 77: Programming Language Entropy Rate vs Unigram

5.5.1.3 PPM Entropy

The Prediction by Partial Matching (PPM) entropy analysis further complements our understanding of the languages. Both Java (mean 2.10, std 0.42) and Python (mean 2.38, std 0.28) demonstrated an interesting behaviour where the entropy starts high around 6-7 and quickly converges to around 2 as the number of tokens increase (Figure 78). This convergence may suggest a stabilization in the

complexity as the language structure grows, reflecting the intrinsic properties of these languages.

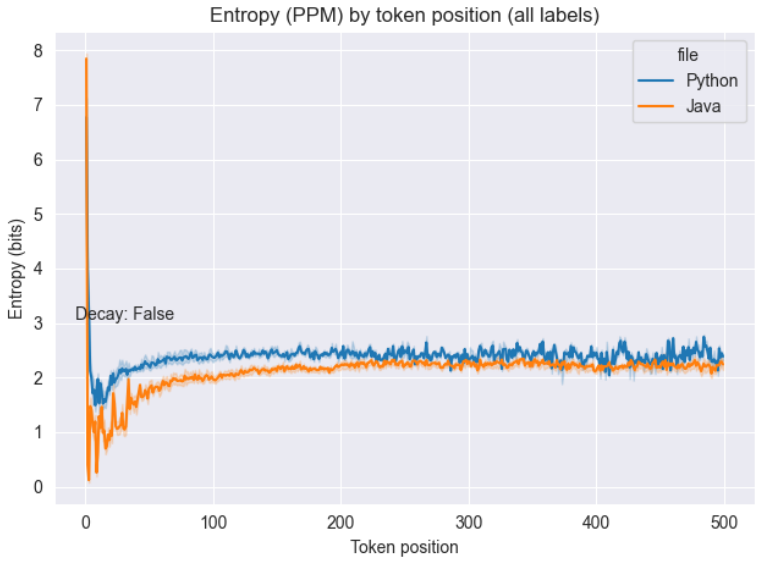

Figure 78: Programming Language PPM Entropy by Token

5.5.2 Token Type Analysis

The token type analysis aims to study the effect of different language elements on the entropy, focusing on variables, comments, strings, and numbers.

For the reader's comprehension, it is vital to define the specific nomenclature used in the analysis, which aids in understanding the graphs and figures presented in this section:

- **w_VAR**: Only Variables are included along with the native language structure; all other elements are replaced with specific tokens like #NUM#, #STR#, etc.
- **w_NUM**: Only Numbers are included, with other elements represented as specific tokens.
- **w_STR**: Only Strings are included, while other components are replaced with tokens.

- **w_ALL**: Variables, Strings, and Numbers are all included in the analysis.
- **C++ or Python or Java**: The analysis considers only the native language structure, with other elements replaced by specific tokens.

5.5.2.1 Unigram Entropy

The inclusion of different token types led to a substantial increase in unigram entropy for all languages (Table 28). Notably, when all tokens (Variables, Strings, Numbers) were included, the entropy increased to 7.02 for C++, 6.55 for Python, and 6.81 for Java. When isolating variables, the entropy remained high for all languages, while the inclusion of only strings or numbers brought the entropy closer to the native language structure.

Table 28: Programming Language Unigram Entropy by Token Type

Options	C++	Python	Java
Native	4.645	4.603	4.736
w/ All (Variables, Strings, Numbers)	7.017	6.547	6.806
w/ Variables	6.770	6.124	6.668
w/ Strings	4.655	4.587	4.752
w/ Numbers	4.615	4.517	4.651

5.5.2.2 Entropy Rate

A closer examination of the entropy rate by token type reveals distinct variations across the languages (Figure 79, Figure 80, Figure 81). In C++, the inclusion of all tokens increased the entropy rate to 1.45 bits from 0.94 bits. Similarly, in Python, the rate varied from 0.89 bits (native) to 1.78 bits (w_ALL), highlighting the significant effect of user-defined tokens like variables, strings, and numbers. Java exhibited similar patterns, with the highest entropy rate observed when all

tokens were included (1.10 bits). In all languages the addition of only String or only Number had small or almost no impact on the entropy rate.

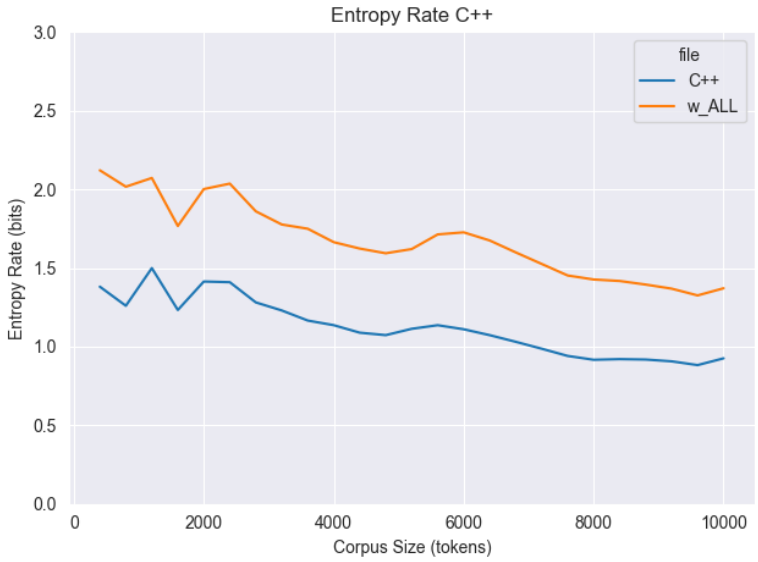

Figure 79: Programming Languages, C++ Entropy Rate by Token Type

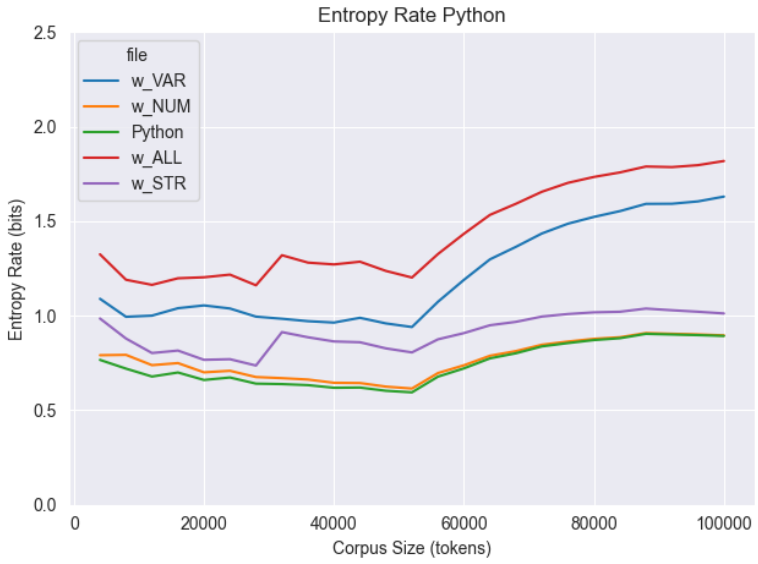

Figure 80: Programming Languages, Python Entropy Rate by Token Type

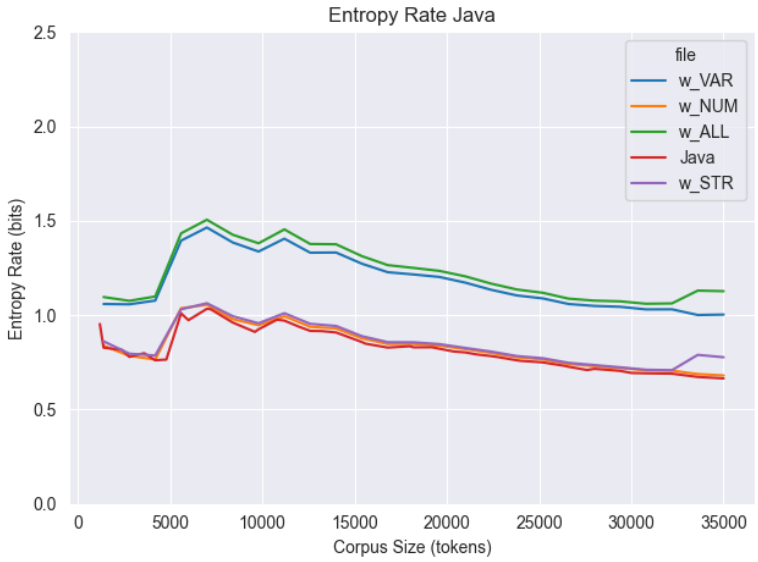

Figure 81: Programming Languages, Java Entropy Rate by Token Type

6 DISCUSSION

6.1 Literature Books

6.1.1 Word Token

6.1.1.1 Key Insights

The study provides a nuanced insight into the entropy of different literature texts, varying from 5-9 Bits (depending on the method and context). This range underscores the diversity of the English language and the complexity within various writing styles. The utilization of different entropy models, including Unigram, Entropy Rate, and PPM Entropy, offers multiple perspectives.

The values for Unigram entropy (Figure 26) reflect the uncertainty associated with predicting individual word occurrences. The diversity in Bits (e.g., Alice in Wonderland at 8.31, Les Misérables at 9.30) corresponds well with the lexical richness or Type-Token Ratio (TTR) (see Table 1), demonstrating how word variety is a key determinant of text complexity.

The transition from token-level to considering a broader context leads to a shift in entropy values, as seen in Figure 28. This decrease in entropy rate (mean of 5.97 Bits) when context is considered aligns with established linguistic theories emphasizing the role of context in shaping language complexity and confirms the 3 Bits shift observed by (Bentz & Alikaniotis, 2016) on natural languages.

The observation in Les Miserables (Table 2, Figure 4) where punctuation and common words are among the most frequent tokens brings forth a critical insight into the noisy nature of texts. This emphasizes the importance of preprocessing decisions and also opens up a path to consider entropy estimations that include punctuation, numbers, and special characters.

The application of entropy analysis in the literature can extend to educational strategies. Educators can select texts with appropriate complexity levels based on entropy and TTR measurements, facilitating learner-centric pedagogy. The complexity measures can be used to design reading materials that align with students' reading levels, assisting in individualized learning. The understanding

of text complexity can further improve literacy research, particularly in assessing readability and creating adaptive learning materials for different age groups.

In conclusion, our literature book analysis establishes a foundational range for classic literature from 5 to 9 Bits. This supports previous studies on entropy estimation and enhances our understanding of text complexity in non-noisy environments.

6.1.2 Character Token

6.1.2.1 Comparison and Implications

The unigram entropy results ranged from 4.31 to 4.38 bits per character for the analyzed texts. This finding contrasts with classical estimations by renowned researchers in the field. For instance, (Shannon, 1951) posited a range between 1.3 and 4.2 bits per character, whereas Cover and King estimated between 1.25 and 1.35 bits per character. Other works by (P. E. Brown et al., 1992), and (Schürmann & Grassberger, 1996) reported values within the 1.46 to 1.75 bits per character range.

The divergence between the current analysis and classical estimations may stem from a difference in the methodology employed, considering the broader vocabulary set in the current analysis. Whereas traditional literature often restricted the analysis to 27 characters, the present study included a larger character set, with Les Miserables displaying a vocabulary size of 83 (Table 3). This likely captures additional noise in the system and results in higher entropy values, echoing the study's primary goal of entropy estimation in noisy environments.

6.1.2.2 Structural Insights

The entropy rate figures reveal further insights, with The Great Gatsby demonstrating the highest value and The Bible the lowest. Interestingly, the lower entropy rate in The Bible is consistent with the dominance of spaces and common lowercase letters in its character frequency (Table 4). This aligns with the expected character distribution in the English language.

A comparison of unigram entropy and entropy rate provides an interesting observation. The two measures sometimes contrast, possibly indicating different underlying patterns, such as variations in sentence length, word choice, and syntactic structure. For instance, Les Miserables show lower unigram entropy but a relatively high entropy rate, potentially reflecting its richer syntactic structure, diverse character set, and complex specialized terms.

6.1.2.3 PPM Model Perspectives

The PPM entropy values, particularly when compared with word token analysis, offer additional insights. It's intriguing to note that the PPM model gives lower entropy without decay activated for character tokens, whereas it was activated for word tokens. This contrast may result from the model's sensitivity to granular lexical structures and the way characters interact within words.

6.1.2.3.1 Effect of Decay

The finding that mean entropy is higher with decay activated (3.44 bits) than without (2.87 bits) has significant implications for the study. This pattern indicates that a decay mechanism, which gives more importance to recent observations, can unravel the complexities within the text more efficiently.

The observation that means entropy with decay is higher suggests a more nuanced understanding of textual patterns, with potential applications in natural language processing (NLP) and literary analysis. Future work could explore the utilization of decay mechanisms in context-sensitive applications like sentiment analysis or semantic interpretation.

The enhancement in entropy with decay reflects an ability to detect subtler variations in character sequences and possibly identify stylistic elements specific to different authors or literary genres. This could be harnessed in various applications such as authorship attribution, genre classification, or literary criticism.

The relationship between model order and entropy provides perspectives on character complexity within the text. The continued increase in entropy with decay, up to a model order of 10, reveals that the literature books' character

structures have multiple layers of complexity. The stability after reaching this point might suggest a saturation in information gain at that depth of context.

Without decay, the stabilization occurs earlier, at a model order of around 5. This might be an indication that without considering the recentness of data, the model quickly reaches a point where additional context does not yield new insights.

6.1.3 Criticisms and Improvements

While the selection of five books offers valuable insights at the individual level of a book, a broader selection encompassing various genres, authors, and time periods could offer a more comprehensive understanding of English entropy.

Including literature from different eras could enrich the analysis by providing a diachronic view of English structure and entropy, revealing how language complexity evolves in response to historical and cultural dynamics. This time series approach would offer a dynamic perspective on how language adapts and changes, reflecting societal transformations, technological advancements, and cultural shifts.

Grouping and analyzing entropy and language characteristics by genre could lead to significant findings. The quantification of entropy could facilitate the automated categorization of books, providing libraries with a faster, less computation-intensive method for classification. Moreover, understanding the entropy patterns within specific genres could contribute to literary studies, offering insights into conventions, motifs, and the underlying philosophy of different literary movements.

6.1.4 Major Takeaways

The unigram and entropy rate (hrate) at the word token level exhibit the same entropy values as those found in the literature and previous work with a ranging value of 8-9 Bits. Contrarily, the character-level examination reveals differences in both unigram and hrate entropy. This discrepancy arises primarily from the consideration of a larger vocabulary that includes more noise, particularly in the form of unique symbols, special characters, and formatting. The incorporation of

additional elements such as unique symbols, special characters, and formatting introduces greater variability and complexity within the character-level analysis. This nuanced approach accounts for the observed deviation from traditional literature books and opens new avenues for exploring language complexity at the intersection of structural and semantic analysis.

6.2 Twitter Streams

The investigation into language entropy, particularly in the noisy environment of Twitter, revealed insights into the complexity and nuances of natural languages. This analysis highlights several mechanisms that govern the statistical behaviour of languages in social media contexts, extending the foundational works of (Montemurro & Zanette, 2011), (Bentz & Alikaniotis, 2016) who reported entropy estimates within the range of 4-8 bits per word in traditional literature.

6.2.1 Word Token

In the noisy context of Twitter, word tokens are influenced by many factors such as the informal nature of tweets, character limitations, and creative expressions like hashtags and slang.

Understanding the variety and frequency of word tokens in the data set (as described in the Analysis part) provides insights into the complexity and the vocabulary richness of the language used. In a social media environment, these tokens can be more fluid and unpredictable, reflecting cultural trends and real-time events.

The noisy nature of Twitter data can lead to a high degree of variability in word tokens. Misspellings, abbreviations, emojis, and creative expressions often result in a vast and unpredictable set of word tokens.

6.2.1.1 Unigram Entropy and Noise

The analysis reveals that the noisy environment of Twitter, characterized by high individual tokens and misspellings, contributes to elevated unigram entropy, a value higher than the literature review and higher than the literacy books. The

extensive vocabulary size and the high frequency of tokens used only once highlight the influence of noise at the individual token level. Tokens used only once, termed as "hapax legomena," emphasize the chaotic nature of individual word usage.

The findings concerning unigram entropy in social media are significant, considering the disruptive and chaotic manner in which language is often used online. Such an increase in entropy levels compared to traditional literature may reflect a broader shift in language dynamics, influenced by the globalization and democratization of information and communication. The elevated unigram entropy underscores the importance of recognizing the diverse and dynamic nature of language on platforms like Twitter.

This phenomenon manifests as a divergence from the entropy estimates found in traditional literature, with the unigram entropy reaching up to 10.89 bits. In contrast, prior studies by (Montemurro & Zanette, 2011)., and others have noted values ranging from 6 to 9 bits per word in more controlled contexts.

The high unigram entropy illustrates the scattered and chaotic nature of individual tokens in social media language. It also confirms the thesis's focus on noisy entropy, where misspellings, abbreviations, and colloquial expressions contribute to the perceived complexity.

The observed lower entropy in English within the Twitter streams might be attributed to the effect of bot activity, as highlighted in the analysis section. It is important to consider the impact of automated content generation on the nature and structure of language online, as this can lead to more predictable and homogenous patterns of expression, as seen in the repeated use of specific tokens like the dollar symbol and crypto-related words. The recurrent appearance of specific tokens, such as the dollar symbol ('$') and words related to crypto giveaways like 'airdrop,' 'loyal,' and 'claim,' implies a pattern of automated or scripted interactions. Such repeated patterns inevitably lower the language's entropy, reflecting a constrained and predictable structure rather than a complex and diverse one inherent in natural language. The prevalence of this bot-related content could therefore have resulted in the English language's unique entropy

profile within the dataset, and its understanding is essential for an accurate interpretation of the entropy measures.

6.2.1.2 Contextual Entropy

The analysis using Prediction by Partial Matching (PPM) entropy and entropy rate resulted in lower values compared to unigram entropy. These measurements consider sequences of tokens and are more resilient to noise, reflecting language's underlying structure.

This is consistent with (Bentz & Alikaniotis, 2016) observation of a 3-bit entropy shift between individual and context levels. The context-aware measurements like PPM account for language structure and grammar, revealing that despite the noise, the underlying linguistic patterns remain intact.

6.2.1.3 Implications

The observed discrepancy between unigram entropy and context-aware entropy measurements, such as PPM and entropy rate, presents practical implications for AI and Natural Language Processing (NLP) applications. Understanding the concentration of noise at the individual token level suggests the need for custom preprocessing techniques.

Specific pre-processing methods could be designed to filter out the noise inherent in social media language without losing the underlying linguistic structure. These methods can include employing statistical models to identify noise patterns, utilizing deep learning architectures like recurrent neural networks (RNNs) to learn the underlying structure, or implementing probabilistic filters based on Bayesian inference.

Such approaches could involve noise reduction algorithms, spelling correction, or removal of rare tokens, all aimed at preparing data more efficiently for machine learning models. Such methods could involve:

- Noise Reduction Algorithms: Specialized filters to remove irrelevant or nonsensical tokens.

- Spelling Correction: Using tools like Hunspell to correct common misspellings.
- Removal of Rare Tokens: Filtering out tokens that appear only once or infrequently.
- Lemmatization: Using algorithms like WordNet Lemmatizer to reduce words to their base or root form.
- Stemming: Applying algorithms like the Porter Stemmer to cut back words to their root form.
- Embedding: Utilizing pre-trained models like Word2Vec to convert words into numerical vectors, capturing semantic meanings.

The findings also extend to sociolinguistics, offering a nuanced perspective on how language evolves and adapts within digital platforms. The observed patterns in entropy can help scholars understand the dynamism of language in online environments, its adaptability, and its resilience to noise.

6.2.2 Character Token

While the analysis of word tokens has revealed the impact of noise at the lexical level, the investigation of character tokens presents a more fundamental view of language structure. This leads to an examination of character-level entropy in social media, revealing consistencies with traditional literature.

The results give us a new way to look at language, even in noisy places like Twitter. What is surprising is that the individual letters in Twitter messages, known as character-level entropy, behave the same way as they do in the literature books. This tells us that even in Twitter, where language can look very messy, the basic building blocks of language don't change.

One of the big takeaways from this research is that the noise seen in Twitter does not show up at the level of individual letters but instead at the level of whole words. The fact that the character-level entropy in Twitter streams is very similar to that in literature books (around 4.33 bits for both) means that the randomness

and complexity of individual letters are the same in both places. This finding is saying that the noise of Twitter does not come within the letters themselves but rather with how they are put together to make words.

This consistency between Twitter and books is not just interesting; it helps us understand how solid and unchanging the basic rules of language can be. Even in a place like Twitter, where people use a lot of slang and shortcuts, the fundamental ways that letters are used do not seem to change. This discovery can help us better understand how language works and adapts, even in new and different environments.

The consistency in character-level entropy between Twitter and literature books may be indicative of the fundamental rules of syntax and orthography that govern the usage of individual letters. Despite the variations in word formation and expression, the basic structure and constraints of character usage remain relatively constant, preserving the integrity of the underlying linguistic system.

The pattern found in the Hrate Entropy, starting at 3 bits and then settling down at 2.7, gives us even more to think about. It could be a sign of how language finds a balance even in the busy and noisy world of social media. This balance may tell us more about how language keeps its shape and meaning, even when it's used in unconventional ways.

These insights could be useful in areas such as computer language understanding. Knowing that the basic rules of language hold steady even in noisy texts could lead to better and more adaptable computer programs. It shows that the basic rules of language are strong and reliable, even in today's fast-changing world.

6.2.3 Effect of Token

The results generated from the exploration of the effect of various tokens (i.e., punctuation, accents, and emojis) on the entropy of language, especially within a noisy environment like Twitter, present a nuanced and multiple understanding of the interaction between language complexity and individual textual elements.

6.2.3.1 Effect of Noise

One of the most interesting revelations of this analysis is the result of the quantitative effect of noise on language entropy. The inclusion of accents and emojis increased the entropy, signifying an increase in the complexity and unpredictability of the language. However, punctuation acted as a structuring element, reducing entropy and making the text more readable. The disparity in entropy values, especially between the configurations without any tokens and those with all tokens, signifies the substantial role these tokens play in affecting language complexity.

Increase in Complexity

Accents and emojis contribute to an increase in entropy, indicating greater complexity and unpredictability in the language structure. This was consistently observed across French, Spanish, and English languages, though variations in degree were apparent.

The addition of emojis and accents expands the vocabulary, giving rise to more unique expressions and subtleties within the language. This enhancement in vocabulary could signify an enriched expressive capability, reflecting emotions, intentions, and cultural contexts.

Accents and emojis might also carry cultural significance, reflecting language diversity. This could partly explain the differences observed in entropy rates across the languages (Figures 45-48). Emojis are not just decorative symbols; they serve as a form of self-expression, cultural reference, or tone indicator. This trend highlights a broader shift in communication, where these icons become essential elements conveying emotions, affiliations, and nuanced meanings.

Decrease in Complexity

Contrary to common perception, punctuation decreased entropy. This contradicts the often-held view of punctuation as noise and underscores its role in improving readability by organizing sentences. Punctuation marks, such as full stops, commas, and question marks, offer breaks and segmentation in language,

potentially reducing the possible combinations of subsequent characters or words and thereby decreasing unpredictability.

While this pattern was consistent across French and Spanish, English showed a distinct behaviour with the lowest entropy when all tokens were included. This may suggest a language-specific relationship between punctuation and entropy, requiring further investigation. Or it could also be a reflection of the bot usage and non-human generated content that was highlighted in the analysis section. Such content tends to follow structured templates which can significantly reduce overall entropy.

6.2.3.2 Implications

The varying effects of tokens on entropy have direct applications in developing more refined pre-processing techniques, especially for noisy environments. By understanding these dynamics, it's possible to create methods that selectively filter or retain elements to suit specific analysis goals.

These insights contribute to a broader understanding of linguistic complexity and have practical applications in natural language processing and machine learning. They can inform models that account for nuances such as accents and emojis, leading to more accurate language analysis.

The observed effect of punctuation on reducing complexity provides valuable insights for enhancing communication clarity. Organizations and educators can use these findings to create more accessible content, improving engagement and comprehension.

Understanding how different tokens affect language complexity has broad societal implications. For example, considering the role of punctuation in improving readability, policymakers and educators could emphasize proper punctuation usage in educational curricula, thus enhancing literacy rates and communication efficiency.

The language-specific differences observed in entropy effects highlight the importance of considering cultural and linguistic nuances in building language

models. Such consideration would result in more culturally sensitive and relevant applications, particularly in global communication contexts. Considering the role of emojis in reflecting emotions and cultural nuances, sentiment analysis models can be optimized to treat emojis not just as tokens, but as significant carriers of sentiment value.

6.2.4 Effect of Clusters

6.2.4.1 Sentiment Analysis

The analysis of language entropy in a noisy environment, segmented by sentiment, offers a detailed understanding of language complexity across three diverse languages.

6.2.4.1.1 Key Insights

The distinct higher entropy in French neutral texts may indicate a less constrained use of language. While positive and negative sentiments often employ specific vocabulary, neutral language may encompass a broader range of topics and expressions. This could reflect a tendency to use a more formal and intricate vocabulary in neutral contexts, which might be tied to the cultural or literary tradition of the French language. Historically, French literature and communication have demonstrated admiration for detailed discourse, which might carry over to modern digital communication.

The significant divergence between sentiments in English, particularly in entropy rate, could hint at underlying structural variations or thematic content disparities. The English language's global use might contribute to this, as different cultures employ English in varied ways. The difference in entropy rate specifically may reflect a more complex interplay of syntax and semantics within different sentiments. For instance, the use of English in Asian countries, influenced by native language structures and cultural expressions, might exhibit unique patterns of complexity when expressing sentiments.

The consistently lower entropy rate in positive Spanish texts might reflect cultural tendencies to express positivity in more standardized or conventional ways. This could have implications for understanding how positive emotions are articulated

across cultures, drawing parallels with sociolinguistic studies on emotion expression.

The unique pattern observed in Spanish, where the negative sentiment's entropy starts lower and converges with the increase in corpus size, might signify a particular property of the Spanish language structure. This pattern could reveal that negative sentiment expressions in Spanish initially appear to be simpler but become more complex as more data is considered. It contrasts with other sentiments, where the entropy values remain more consistent. This phenomenon may be related to the distribution and frequency of specific words used in negative contexts in Spanish and might indicate an intricate balance between common and rare words.

6.2.4.2 Emotions Analysis

6.2.4.2.1 Key Insights

One of the most striking observations is the consistent patterns in emotions like 'surprise' and 'fear' across all three languages. These emotions tend to exhibit low standard deviation in unigram entropy, suggesting universal traits in the expression of these emotions (Figures 55, 57, and 59). This consistency might be indicative of commonality in human emotional expression transcending linguistic barriers. One could say that these emotions represent clear and genuine emotions intrinsic to human beings resulting in a more natural and less noisy structure across all languages.

The results also reveal unique aspects of individual languages. For instance, 'anger' in Spanish and French displays a low standard deviation of unigram entropy, unlike English (Figures 55, 57, and 59). This may imply cultural or linguistic influences on the expression of certain emotions.

An interesting pattern emerges from the French data, where unigram entropy stays constant, but the entropy rate decreases compared to the overall French dataset studied previously. This indicates that individual words maintain similar levels of unpredictability, but the sequence of these words, when tied to specific emotions, tends to be more predictable. This could underline a tendency in

French for certain phrases or structures to be repeated more often in specific emotional contexts, resulting in a lower entropy rate. A study of specific linguistic constructs and emotional idioms in French could provide a more granular understanding of this phenomenon. This insight resonates with the broader understanding of syntactic and semantic relationships in language modelling (Chomsky, 1957).

6.2.4.2.2 Implications

These insights are invaluable for linguistics, psychology, and computational social sciences. The patterns and complexities unveiled can enrich theories of emotion, and sociolinguistics.

The findings have immediate applications in Natural Language Processing (NLP) technologies such as emotion recognition systems, chatbots, and digital assistants. For example, understanding the emotional structure of languages can help in tailoring responses in customer service bots, thereby enhancing user experience.

The particular pattern observed in the French data offers valuable insights for computational linguistics, including sentiment analysis and machine translation. Recognizing the interplay between individual word uncertainty and sequence predictability in emotion-driven texts could inform more nuanced algorithms, leading to improvements in natural language understanding technologies.

In educational settings and therapeutic practices, insights into the structured and patterned expression of emotions can inform strategies for emotional literacy training and therapy.

6.3 Covid-19 Tweets

6.3.1 Language Complexity

The slight increase in English unigram entropy over time might be reflective of a broader and more diversified global conversation. English, being a global language, might encompass diverse viewpoints and cultural narratives, contributing to increased complexity.

The decrease in Spanish and French unigram entropy could be indicative of a more localized and culturally cohesive response to the pandemic. This might also reflect a more centralized communication strategy from authorities in these language regions.

The observed patterns may also be influenced by the way media in different linguistic regions cover the pandemic. The increase in English entropy might be correlated with more fragmented and varied media landscapes, whereas the consistency in Spanish and French could be tied to more homogeneous media reporting.

The growing predictability in the sequence of words might be a reflection of the effects of official communications and guidelines. Repeated messages from authorities, scientific experts, and media can lead to a consolidation of specific terms and phrases, making the discourse more standardized. This could be a manifestation of the power of consistent messaging in shaping public discourse and opinion,

6.3.2 Temporal Dynamism

The trends in time-series unigram entropy (Figure 67) and entropy rate (Figure 68) may underline human adaptability and resilience. The general decline in entropy over time might symbolize a societal adaptation to the new normal, with language becoming more predictable as people acclimate to the ongoing crisis. Conversely, the slight increase in English unigram entropy over time might be reflective of the continuing evolution of the discourse and the incorporation of new ideas and perspectives, indicating a broader and more diversified global conversation.

The initial stages of the pandemic might have been characterized by higher entropy, reflecting confusion, fear, and uncertainty. This might have gradually given way to more structured discourse as information became more consistent and reliable.

The observed trends could also suggest a convergence of public opinion and sentiment over time. The uniform decline in entropy rate across all three

languages might symbolize a coalescing of global sentiment and understanding as the pandemic evolved.

6.3.3 Implications

Understanding the nuanced differences in entropy across languages could enable more effective and targeted communication. For instance, the richer and more complex vocabulary in English might necessitate a different approach compared to the more predictable and structured discourse in Spanish and French.

These insights may also extend to other crises or global events, offering a framework for understanding how public discourse evolves. The trends and patterns observed here could be applied to future communication strategies, leveraging the lessons learned during the Covid-19 pandemic.

The decline in entropy can also serve as a valuable metric for monitoring public response to ongoing crises. It might signal not only the effectiveness of communication but also the public's acceptance or resistance to the implemented measures. These observations provide a window into the dynamic nature of human behaviour during crises, illustrating how social narratives evolve, adapt, and converge. By understanding these patterns, policymakers and communicators can craft more resonant and effective messages during future crises, capitalizing on the observed linguistic trends.

6.4 Target Users

6.4.1 Key Insights

6.4.1.1 News Outlets

The differing unigram entropy among various news outlets sheds light on the diversity of journalistic practices and editorial policies. Each outlet's unique approach has created specific language patterns.

With the highest unigram entropy (Figure 69), The Guardian appears to have the most diverse vocabulary. This could be indicative of more in-depth analysis and a willingness to explore complex topics, aligning with their reputation for critical and independent reporting.

Reuters, with the lowest standard deviation and entropy rate (Figures 65, 66), perhaps follows a more standardized reporting style. This may enhance readability but potentially limit the range of expression, reflecting the agency's focus on financial and business news, where precision and clarity are needed.

The higher standard deviation in NY Times could signify a more flexible approach to language use, reflecting a broader spectrum of topics and writing styles. This aligns with the paper's focus on both hard news and feature stories. The significant standard deviation in NY Times, as compared to Reuters, further supports the notion of a more dynamic and adaptable approach to language use in the former, adding texture to their reporting across various subjects.

CNN's highest entropy rate (Figure 70) suggests more unpredictable and nuanced discussions. This might be a strategic choice to engage a broader audience, reflecting the network's role as a leading international news provider.

The varying entropy rate highlights that a rich vocabulary does not necessarily translate to more complex sequences of words. The lower rate in Guardian despite its higher unigram entropy may indicate more standardized sentence structures, even with diverse vocabulary.

6.4.1.2 Individual Communication

The contrasting entropies between Elon Musk and Donald Trump provide insights into personal communication tactics.

Musk's higher unigram entropy (Figure 71) might be emblematic of his innovative thinking and willingness to explore diverse topics. It resonates with his role as an entrepreneur in cutting-edge industries like space exploration and electric vehicles.

Trump's more consistent or repetitive word choice (Figure 72) aligns with his straightforward, populist style, designed to resonate with a broad base of supporters. This may have played a role in his political success.

6.4.1.3 Entropy Estimators

Certain methods, like Laplace, may lead to the overestimation of smaller samples (Figure 73). This highlights the need to consider sample size and specific characteristics when selecting an estimator, especially in analysing individual accounts.

The range of standard deviations across methods emphasizes the requirement for methodological rigor, especially in analysing a noisy environment, where the choice of estimator can significantly impact the results.

6.4.1.4 Ukraine War

The convergence in unigram entropy (Figure 74) could reflect a shared human response to a crisis, transcending cultural and linguistic barriers. This might signify the power of global media in shaping a unified narrative. This also may underscore the universality of certain themes and emotions in times of conflict, transcending linguistic differences.

The higher Hrate entropy in French (Figure 75) may point to more nuanced discussions within French media, possibly reflecting France's historical and political interest in the region.

6.4.2 Implications

The entropy measurements provide insights into journalistic practices, revealing potential biases, standardizations, and varying degrees of complexity across different news outlets. These findings can be crucial for media literacy, enabling readers to understand how different editorial policies and reporting styles may shape information presentation. They might also inform automated bias detection tools by understanding the linguistic nuances tied to specific news sources.

The analysis of public figures like Musk and Trump offers a new way to understand political and business leadership communication strategies. It may

have real-world applications in political campaign analysis and public relations strategy development.

The shared unigram entropy across different languages in the Ukraine War tweets may provide a foundational basis for future cross-cultural communication research. It opens avenues for exploring how language and cultural barriers can be transcended during international crises, potentially informing global media strategies and diplomatic communication.

6.5 Programming Languages

6.5.1 Key Insights

6.5.1.1 Language Characteristics

The observed unigram entropies for C++, Python, and Java (Table 27) highlight a clustering effect, reflecting the intrinsic similarities in the vocabulary structures of these languages. This observation can be associated with the concept of universality in programming languages where core syntactical and grammatical structures are shared across them, contributing to similar entropy measures. For example, the same structure of "FOR LOOPS" or "IF-ELSE" statements is common in many programming languages, serving as a testament to this universality.

The hrate entropy, revealing more nuanced differences (Figure 76), particularly underscores the sensitivity of entropy calculations to contextual information. A high entropy rate in C++ might signify a richer expression in language syntax and structure, lending it a higher complexity compared to Python and Java.

6.5.1.2 The Effect of Token Types and Noise

The marked increase in unigram entropy when all tokens (Variables, Strings, Numbers) are included in the analysis for C++, Python, and Java (Table 28) is indicative of the multifaceted complexity introduced by these elements. Variables, being user-defined, introduce a layer of uncertainty and unpredictability into the language structure. This observation may be linked to the open-ended nature of

programming languages, where variables allow for customization and, therefore, higher entropy. It sheds light on the creative aspect of programming and may inform compiler designers about potential optimization paths, particularly concerning variable handling.

The minimal change in entropy rate with the inclusion of only Strings or Numbers (Figures 75-77) suggests an underlying pattern within the languages. This observation can be explained by the constrained set of rules governing the use of numbers and strings within programming languages. Unlike variables, the use of numbers and strings tends to follow a more defined and predictable pattern, which is reflected in their limited impact on the overall entropy.

6.5.1.3 Divergence of Entropy Rate and Unigram

Programming languages are highly structured entities, with defined patterns, block structures, brackets, and specific syntactical rules that guide their composition. This codified structure presents unique challenges and opportunities in understanding entropy within these languages, especially when considering both hrate and unigram analyses.

The unigram analysis, focusing on individual tokens, captures the surface-level complexity of a language. At this level, programming languages might appear chaotic or noisy due to the variety of tokens and syntax rules involved. However, this perspective fails to capture the underlying structure that governs how these tokens are organized and interact.

The hrate analysis, on the other hand, takes into consideration the broader context, revealing how tokens are patterned and arranged within the language. This perspective uncovers the inherent predictability in the programming languages, driven by block structures, conditional statements, loops, and other syntactical constructs. The 'noisiness' observed at the unigram level translates into structured patterns when seen in a broader context.

6.5.2 Implications

This insight has practical implications for areas such as code completion and error detection, where understanding the predictable nature of strings and numbers can lead to more efficient algorithms and tools.

The significant increase in entropy with the inclusion of all tokens (w_ALL) reveals the essential role of user-defined tokens in defining the language's structure and complexity. This can be particularly useful for educational purposes, highlighting the importance of user-defined elements in understanding and learning programming languages.

Moreover, this insight into the complexity introduced by variables emphasizes the importance of adhering to consistent naming conventions for variables, functions, and other user-defined elements. Whether working in team environments or contributing to open-source projects, maintaining clear and standardized naming practices can reduce complexity and enhance code readability. This, in turn, can emphasise collaboration and streamline the development process, reinforcing that the aesthetic and organizational aspects of coding are as vital as functional correctness and efficiency.

These findings may also be leveraged to influence compiler design. By recognizing the entropy effects of different token types, compiler optimization techniques could be developed to prioritize or handle specific tokens differently, leading to more efficient code translation and execution.

7 FUTURE WORKS

The research conducted in this thesis has led the way for further exploration in the field of language entropy, especially in noisy environments. While the findings offer substantial insights into various domains such as journalism, political communication, the effect of punctuation and special character in languages, and social media discourse, there remain several avenues for further investigation and improvement.

One area of interest is to investigate how different preprocessing methods affect the performance and accuracy of language models. Implementing various preprocessing techniques, such as tokenization, stemming, lemmatization, noise reduction, and feature extraction, and comparing their effects on model accuracy, efficiency, and interpretability could lead to the development of optimized preprocessing pipelines tailored to specific tasks and domains, enhancing model performance and robustness.

Another promising avenue is to examine how the entropy of the input text impacts the accuracy and performance of different language models. Developing a systematic methodology to quantify the relationship between input entropy and model accuracy and investigating how variations in vocabulary richness, syntactic complexity, and noise levels affect model performance could enable more precise model tuning. Understanding this relationship may allow for adaptive adjustments based on the characteristics of the input text.

The exploration of the writing quality and entropy of various large-scale language models such as GPT-3, BERT, and Transformer-based models is also an exciting prospect. Implementing comprehensive metrics to assess writing quality, including readability, coherence, grammatical accuracy, and stylistic appropriateness, along with analyzing the entropy of generated texts to understand the models' creativity and variability, could lead to the identification of best practices for utilizing large language models in diverse applications, from creative writing to professional documentation.

Extending the entropy analysis to spoken language presents another valuable opportunity. This includes analyzing random conversations, podcasts, structured videos, news, and TV programs, among others. Utilizing tools like Whisper AI to convert spoken content into text or leveraging auto-generated subtitles from platforms like YouTube and applying entropy analysis methods to assess the complexity and diversity of spoken language across different contexts may uncover novel insights into the dynamics of spoken communication. This exploration could reveal patterns related to spontaneity, informality, and real-time interaction.

An intriguing direction for future work also involves quantifying the relationship between prompts and output entropy in large language models such as ChatGPT. As these models respond to varied prompts, the output's clarity, information content, structure, and complexity may differ substantially. Investigating how specific prompt characteristics, such as ambiguity, specificity, length, and context, influence the resulting output entropy could provide critical insights into optimizing human-model interactions. Understanding the dynamics of prompt-output relationships may lead to the development of guidelines for crafting more effective prompts, tailoring model responses for different audiences, and implementing adaptive strategies to enhance communication efficacy. This inquiry could further our understanding of how language models perceive and process information, paving the way for more nuanced and intelligent dialogue systems that can cater to diverse informational and communicative needs. This line of research could bridge the gap between theoretical modelling and real-world applicability, facilitating more personalized and context-aware human-machine communication.

8 CONCLUSION

Much like physical laws that govern the natural world, entropy in language serves as a fundamental principle that dictates the unpredictability and complexity inherent in various forms of communication. Whether through the everyday activity of social media platforms like Twitter, the expressive capacities of literature, or the structured syntax of programming languages, entropy in language is as significant as it is subtle. Through rigorous study and mathematical precision, this research has mapped the contours of entropy with a focus on noisy environments, thereby transforming abstract ideas into tangible insights.

The methodology employed in this study was rooted in a multifaceted approach, tailored to examine noisy entropy across an array of domains. It began with the selection of diverse data sources, encompassing natural languages, social media, news outlets, individual communication, contemporary events, and programming languages. The extraction of this wide-ranging data was performed using targeted algorithms and predefined criteria to ensure relevance and consistency. Various entropy estimators were utilized, including character-level, word-level, unigram, and context-focus estimator as PPM and Hrate, among others, to provide a comprehensive and nuanced analysis. The decision to analyze distinct aspects like punctuation, emojis, accents, and specific contexts like the COVID-19 pandemic or the Ukraine War was a deliberate move to explore the multi-dimensional characteristics of noisy entropy.

In terms of the experimental procedure, rigorous measures were taken to ensure the validity and reliability of the results. The study employed robust statistical analyses using several uncertainty analyses and implemented time-series analysis, emotional cluster analysis, and comparisons across multiple datasets. The choice of specific entropy measures and the inclusion of a wide variety of text samples were carefully calibrated to illuminate the hidden patterns and complexities within the data.

8.1 Key Insights

8.1.1 Entropy in Literature

The in-depth analysis began with the investigation of language entropy in the literature, primarily focused on the noisy characteristics of languages by considering a broader set of characters, including punctuation, special characters, and numbers. By examining five literature books, the results propose a novel estimate at the character level of 4.31-4.38 bits per character, highlighting a more realistic and chaotic environment when considering a larger alphabet. This contrasted starkly with previous estimates and signified a crucial advancement in our understanding of English entropy. At the word level, it confirmed previous work, validating the entropy range of 5-9 bits per token, depending on the estimators used, with an observed 3 bits entropy shift when considering the context.

8.1.2 Entropy on Twitter

The chaos of individual word usage, accentuated by hapax legomena, highlighted the seemingly disordered nature of language, particularly on platforms like Twitter. Yet, a parallel study of character-level entropy in both Twitter streams and literature books uncovered a profound consistency, revealing that the randomness and complexity of individual letters are the same in both domains. This discovery signalled that the perceived noise in Twitter lies not in the individual letters but in their amalgamation into words. At the level of the words, Twitter yielded a higher unigram entropy, 2 bits higher than previously studied, accentuating the chaotic nature of individual word usage. Importantly the 3 bits shift in entropy when considering context is still present on Twitter highlighting the inside logical structure of language even when the noise remains at the individual level of token.

Furthermore, the inclusion of accents and emojis, which often contributed to increased entropy, and the contrasting effect of punctuation as a structuring element, challenges conventional perspectives. Punctuation's role in reducing

entropy and enhancing readability underscores its importance in the organization of sentences, contradicting the widely-held view of it as mere noise.

The cluster analysis displayed consistent patterns in certain emotions across languages, indicating universal traits in human emotional expression. This consistency underlines a profound human connection, transcends linguistic boundaries, and gives insight into the nature of human emotional discourse. Emotions inherent of human nature like love, fear or surprise tend to exhibit a lower entropy, representing clear and genuine emotions intrinsic of human being resulting in a more natural and less noisy structure across all languages.

An analysis of various news outlets' styles revealed distinct entropy patterns reflecting their unique approaches. The Guardian's diverse vocabulary, Reuters' standardized style, CNN's nuanced discussions. These findings could be instrumental in understanding how media outlets tailor their language to suit their readership's needs and preferences. The entropy measure on language affecting public interest could lead to the shape of a biased detector, uncertainty analysis or fact-check technology leading to more truthful and less biased information.

The contrasting entropies between Elon Musk and Donald Trump provided significant insights into personal communication tactics. While Musk's higher entropy resonated with his innovative approach, Trump's more consistent word choice reflected his populist style, shedding light on different communication strategies. With a person more focus on a specific goal when using Twitter lead to a lower and uniform entropy measure.

8.1.3 Entropy over Time

The time-series analysis of entropy during the COVID-19 pandemic was especially enlightening. A gradual decline in entropy might signify societal adaptation, with language evolving to become more predictable. This also might indicate a convergence of public opinion, a symbolic unity across languages as the pandemic progressed. Similarly, the analysis of language during the Ukraine War exposed a convergence in unigram entropy, possibly representing a shared human response, and emphasized the global media's role in shaping a unified

narrative. The convergence in entropy during significant global events such as the COVID-19 pandemic and the Ukraine war also underlines human adaptability and the power of global media in forming a unified narrative.

8.1.4 Entropy in Programming Languages

The exploration of programming languages like C++, Python, and Java opened a new dimension to entropy. The clustering effect observed in unigram entropies reflects the universality in programming languages, with its core syntactical and grammatical structures shared across them. Moreover, the marked increase in unigram entropy with the inclusion of variables, strings, and numbers exposes the multifaceted complexity and creativity in programming with most of the noise introduced in the realm of naming customization allowed for the user.

The study of programming languages also revealed the distinction between surface-level complexity, evident in unigram analysis, and the underlying predictable structure, as uncovered through hrate analysis. This duality emphasizes the importance of context in understanding the ordered chaos inherent in programming languages. The very low Entropy Rate of programming languages also highlighted their highly structural patterns and codify conventions, leading to a high predictability when considering previous tokens.

8.2 Final Words

Language is a daily experience so ingrained in human nature that its complex behaviours often feel instinctive and obvious. We intuitively navigate its nuances, much like we inherently understand the pull of gravity. However, what seems obvious is not always simple to quantify or prove. The research on language entropy has achieved exactly that, transforming abstract understandings of language into quantifiable measurements. By applying mathematical rigor to language's intricate patterns, it has illuminated its underlying structure and behavior. This isn't merely an exercise in putting numbers to what we already know; it's a vital step towards the future of science, paving the way for innovations and insights that can only come from translating the abstract into the tangible.

Like the quantification of physical laws, this research offers a foundation upon which new theories, applications, and understandings can be built.

This research has unravelled the complex interplay of structure and chaos in language, providing a nuanced perspective on entropy in various domains. The insights gained through this comprehensive study not only contribute to the understanding of language but also lay the groundwork for future exploration and applications, both theoretical and practical, in the fascinating realm of noisy entropy. The multifaceted analysis of diverse language sources, from literature and social media to news outlets and programming languages, has illuminated the inherent complexities and beautiful order embedded within the chaos of language.

9 REFERENCES

Alshaabi, T., Rushing Dewhurst, D., Minot, J. R., Arnold, M. V, Adams, J. L., Danforth, C. M., & Dodds, P. S. (2021). *The growing amplification of social media: measuring temporal and social contagion dynamics for over 150 languages on Twitter for 2009-2020*. https://doi.org/10.1140/epjds/s13688-021-00271-0

Ao, C. H. (2003). Nonparametric estimation of Shannon's index of diversity when there are unseen species in sample. *Environmental and Ecological Statistics*, *10*.

Baldwin, T., Cook, P., Lui, M., MacKinlay, A., & Wang, L. (2013). *How Noisy Social Media Text, How Diffrnt Social Media Sources?* (pp. 356–364). https://aclanthology.org/I13-1041

Banda, J. M., Tekumalla, R., Wang, G., Yu, J., Liu, T., Ding, Y., Artemova, K., Tutubalina, E., & Chowell, G. (2020). A large-scale COVID-19 Twitter chatter dataset for open scientific research -- an international collaboration. *Epidemiologia*, *2*(3), 315–324. https://doi.org/10.3390/epidemiologia2030024

Beaufort, R., Roekhaut, S., Cougnon, L.-A., & Fairon, C. (2010). *A Hybrid Rule/Model-Based Finite-State Framework for Normalizing SMS Messages* (pp. 770–779). Association for Computational Linguistics. https://aclanthology.org/P10-1079

Bell, T., Witten, I. H., & Cleary, J. G. (1989). Modeling for text compression. *ACM Computing Surveys (CSUR)*, *21*(4), 557–591. https://doi.org/10.1145/76894.76896

Bentz, C., & Alikaniotis, D. (2016). *The Word Entropy of Natural Languages*.

Bentz, C., Alikaniotis, D., Cysouw, M., & Ferrer-i-Cancho, R. (2017). The Entropy of Words—Learnability and Expressivity across More than 1000 Languages. *Entropy 2017, Vol. 19, Page 275*, *19*(6), 275. https://doi.org/10.3390/E19060275

Bentz, C., Ruzsics, T., Koplenig, A., & Samardži´c, T. S. (2016). *A Comparison Between Morphological Complexity Measures: Typological Data vs. Language Corpora* (pp. 142–153). https://aclanthology.org/W16-4117

Benveniste, A., Juditsky, A., Delyon, B., Zhang, Q., Glorennec, P., Kontoyiannis, I., Member, S., Algoet, P. H., Suhov, Y. M., & Wyner, A. J. (1998). The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems. *IEEE TRANSACTIONS ON INFORMATION THEORY*, *44*(3), 1391–1407.

Bird, Ewan, K., Steven, & Edward, L. (2009). *Natural Language Processing with Python*. O'Reilly Media Inc.

Brown, P. E., Della Pietra, V. J., Mercer, R. L., Della Pietra, S. A., & Lai, J. C. (1992). *An Estimate of an Upper Bound for the Entropy of English*.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, *2020-December*. https://arxiv.org/abs/2005.14165v4

Camacho-Collados, J., Rezaee, K., Riahi, T., Ushio, A., Loureiro, D., Antypas, D., Boisson, J., Espinosa-Anke, L., Liu, F., Martínez-Cámara, E., Medina, G., Buhrmann, T., Neves, L., & Barbieri, F. (2022). *TweetNLP: Cutting-Edge Natural Language Processing for Social Media*. https://spacy.io

Chen, E., & Ferrara, E. (2022). Tweets in Time of Conflict: A Public Dataset Tracking the Twitter Discourse on the War Between Ukraine and Russia. *Proceedings of the International AAAI Conference on Web and Social Media*, *17*, 1006–1013. https://doi.org/10.1609/icwsm.v17i1.22208

Chomsky, N. (1957). Syntactic Structures. *Syntactic Structures*. https://doi.org/10.1515/9783112316009/HTML

Cover, T. M., & King, R. C. (1978). A Convergent Gambling Estimate of the Entropy of English. *IEEE TRANSACTIONS ON INFORMATION THEORY*, *4*, 413.

Cover, T. M., Thomas, J. A., Schilling, D. L., Bellamy, J., & Freeman, R. L. (2006). *Elements of Information Theory Elements of Information Theory WILEY SERIES IN TELECOMMUNICATIONS Telecommunication Transmission Handbook, 3rd Edition Introduction to Communications Engineering, 2nd Edition*.

Crystal, D. (2003). *English as a global language, Second edition*.

Daniel Jurafsky, & James H. Martin. (2023). *Speech and Language Processing* (Third Edition). https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf

Dretske, F. I. (1983). Précis of Knowledge and the Flow of Information. *THE BEHAVIORAL AND BRAIN SCIENCES*, *6*, 55–90.

Eisenstein, J. (2013). *What to do about bad language on the internet* (pp. 359–369). Association for Computational Linguistics. https://aclanthology.org/N13-1037

Feder, M., Merhav, N., & Gutman, M. (1992). Universal Prediction of Individual Sequences. *IEEE Transactions on Information Theory*, *38*(4), 1258–1270. https://doi.org/10.1109/18.144706

Feldman, R., & Sanger, J. (2007). *An Application of Porters Stemming Algorithm for Text Mining in Healthcare Ashwini*. 410. https://www.researchgate.net/publication/200504395_The_text_mining_handbook_Advanced_approaches_in_analyzing_unstructured_data

Gao, Y., Kontoyiannis, I., & Bienenstock, E. (2008). Estimating the entropy of binary time series: Methodology, some theory and a simulation study. *Entropy*, *10*(2), 71–99. https://doi.org/10.3390/entropy-e10020071

Goodman, J. T. (2001). A bit of progress in language modeling. *Computer Speech & Language*, *15*(4), 403–434. https://doi.org/10.1006/CSLA.2001.0174

Han, B., & Baldwin, T. (2011). *Lexical Normalisation of Short Text Messages: Makn Sens a #twitter* (pp. 368–378). https://aclanthology.org/P11-1038

Harrison, P. (2021). *pmcharrison/ppm: ppm v0.3.0*. https://doi.org/10.5281/ZENODO.4884795

Hashimoto, K., Kontonatsios, G., Miwa, M., & Ananiadou, S. (2016). *Topic detection using paragraph vectors to support active learning in systematic reviews*. https://doi.org/10.1016/j.jbi.2016.06.001

Hausser, J., Maintainer, K. S., & Strimmer, K. (2022). *Package "entropy" Title Estimation of Entropy, Mutual Information and Related Quantities*.

James, W., & Stein, C. (1961). Estimation with Quadratic Loss. *Https://Doi.Org/*, *4.1*, 361–380. https://projecteuclid.org/ebooks/berkeley-symposium-on-mathematical-statistics-and-probability/Proceedings-of-the-Fourth-Berkeley-Symposium-on-Mathematical-Statistics-and/chapter/Estimation-with-Quadratic-Loss/bsmsp/1200512173

Jiang, Z., Yang, M. Y. R., Tsirlin, M., Tang, R., Dai, Y., & Lin, J. (2023). *"Low-Resource" Text Classification: A Parameter-Free Classification Method with Compressors* (pp. 6810–6828). https://aclanthology.org/2023.findings-acl.426

John R. Pierce. (1980). *An Introduction to Information Theory: Symbols, Signals and Noise - John R. Pierce - Google Books*. https://books.google.fr/books?hl=en&lr=&id=eKvhiI2ogwEC&oi=fnd&pg=PR2&dq=Pierce,+J.+R.+(1980).+An+Introduction+to+Information+Theory:+Symbols,+Signals+and+Noise&ots=bMmALnxrFm&sig=T9phOqpwILxSdQ-RY8j2MW46-lQ&redir_esc=y#v=onepage&q=Pierce%2C%20J.%20R.%20(1980).%20An

%20Introduction%20to%20Information%20Theory%3A%20Symbols%2C%20Signals%20and%20Noise&f=false

Kullback, S., & Leibler, R. A. (1951). On Information and Sufficiency. *Https://Doi.Org/10.1214/Aoms/1177729694*, *22*(1), 79–86. https://doi.org/10.1214/AOMS/1177729694

Mackay, D. J. C. (1995). *Information Theory, Inference, and Learning Algorithms*. http://www.inference.phy.cam.ac.uk/mackay/itila/

Manning, C. D., & Schiitze, H. (1999). *Foundations of Statistical Natural Language Processing E0123734*.

Mayer-Schönberger, V., & Cukier, K. (2013). *Big data: A revolution that will transform how we live, work, and think.* https://psycnet.apa.org/record/2013-17650-000

Miller, G., & Miller, G. (1955). *Note on the bias of information estimates*.

Montemurro, M. A., & Zanette, D. H. (2011). *Universal Entropy of Word Ordering Across Linguistic Families*. https://doi.org/10.1371/journal.pone.0019875

Nemenman, I., Shafee, F., & Bialek, W. (2002). *Entropy and Inference, Revisited*.

Nielsen, M. A., & Chuang, I. L. (2010). *Quantum Computation and Quantum Information*. www.cambridge.org

Paninski, L. (2003). *Estimation of Entropy and Mutual Information*.

Piantadosi, S. T. (2014). *Zipf's word frequency law in natural language: A critical review and future directions*. https://doi.org/10.3758/s13423-014-0585-6

Piantadosi, S. T., Tily, H., Gibson, E., & Kay, P. (2011). *Word lengths are optimized for efficient communication*. https://doi.org/10.1073/pnas.1012551108

project-codenet, ufinkler, Janssen, G., VlZolotov, Reiss, F., Chen, J., Choudhury, M., Martinelli, S., Thost, V., giacomo-domeniconi, lindseydecker,

lucaburatti7, & Puri, R. (2021). *IBM/Project_CodeNet: Initial release 1.0 (May 5, 2021)*. https://doi.org/10.5281/ZENODO.4814770

Rosenfeld, R. (2000). Two decdes of statistical language modeling where do we go form here? Where do we go from here? *Proceedings of the IEEE*, *88*(8), 1270–1275. https://doi.org/10.1109/5.880083

Rosenkrantz, R. D. (1989). Information Theory and Statistical Mechanics I (1957). *E. T. Jaynes: Papers on Probability, Statistics and Statistical Physics*, 4–16. https://doi.org/10.1007/978-94-009-6581-2_2

Salomon, D., & Motta, G. (2010). Handbook of data compression. *Handbook of Data Compression*, 1–1359. https://doi.org/10.1007/978-1-84882-903-9/COVER

Schürmann, T., & Grassberger, P. (1996). Entropy estimation of symbol sequences. *Chaos*, *6*(3), 414–427. https://doi.org/10.1063/1.166191

Shannon, C. E. (1948). A Mathematical Theory of Communication. *The Bell System Technical Journal*, *27*, 623–656.

Shannon, C. E. (1951). *Prediction and Entropy of Printed English*.

Silva, J. F. (2018). Shannon Entropy Estimation in ∞-Alphabets from Convergence Results: Studying Plug-In Estimators. *Entropy 2018, Vol. 20, Page 397*, *20*(6), 397. https://doi.org/10.3390/E20060397

Smaya, H. (2022). The Influence of Big Data Analytics in the Industry. *OALib*, *09*(02), 1–12. https://doi.org/10.4236/OALIB.1108383

Teahan, W. J., & Cleary, J. G. (1996). *THE ENTROPY OF ENGLISH USING PPM-BASED MODELS*.

TIOBE. (2023). *TIOBE Index for January 2023*. Https://Www.Tiobe.Com/Tiobe-Index/.

Tufekci, Z. (2017). *Twitter and Tear Gas. The Power and Fragility of Networked Protest*. https://doi.org/10.25969/MEDIAREP/14848

Turing, A. M. (1936). *ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO THE ENTSCHEIDUNGSPROBLEM*.

Urgen Pfeffer, J. ¨, Mooseder, A., Lasser, J., Hammer, L., Stritzel, O., & Garcia, D. (2023). *This Sample seems to be good enough! Assessing Coverage and Temporal Reliability of Twitter's Academic API*. https://developer.twitter.com/en/products/twitter-api/

van Erven Peter Harremoës, T. (2007). *Rényi Divergence and Kullback-Leibler Divergence*. *6*(1).

Wiener, N. (1948). *CYBERNETICS*. https://www.jstor.org/stable/24945913

Witten, I. H., Neal, R. M., & Cleary, J. G. (1987). Arithmetic coding for data compression. *Communications of the ACM*, *30*(6), 520–540. https://doi.org/10.1145/214762.214771

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., … Wen, J.-R. (2023). *A Survey of Large Language Models*. https://www.bing.com/new

Zipf, G. (1936). *The psycho-biology of language: An introduction to dynamic philology*. https://api.taylorfrancis.com/content/books/mono/download?identifierName=doi&identifierValue=10.4324/9781315009421&type=googlepdf

Zipf G. (1949). *Human Behavior and the Principle of Least Effort: An Introduction to Human ... - George Kingsley Zipf - Google Livres*. https://books.google.fr/books?hl=fr&lr=&id=m-XDCwAAQBAJ&oi=fnd&pg=PT4&ots=Dn5ZqpbV_z&sig=CH2-cwPdbzCmjbZw3yGVBrWLuIQ&redir_esc=y#v=onepage&q&f=false

Ziv, J., & Lempel, A. (1977). A Universal Algorithm for Sequential Data Compression. *IEEE Transactions on Information Theory*, *23*(3), 337–343. https://doi.org/10.1109/TIT.1977.1055714

Ziv, J., & Lempel, A. (1978). Compression of Individual Sequences via Variable-Rate Coding. *IEEE Transactions on Information Theory*, *24*(5), 530–536. https://doi.org/10.1109/TIT.1978.1055934

10 APPENDIX

10.1 Meeting Minutes

Supervision Meeting Notes

Taught ☒ Research ☐

Student Name	Simeon FEREZ							
Student Number	S392371							
Course	Computational & Software Techniques in Eng. (CIDA option)							
Supervisor	Dr. Jun Li							
Date of Meeting	11th May 2023							
Meeting by	In person	☐	Telephone	☐	Skype	☐	Other	☒

Decisions / Actions agreed and by whom
Supervision Meeting #1 Review project proposal Define the objectives of the project Define the overall Methodology to follow

Date of next meeting
\

Supervision Meeting Notes

Taught ☒ Research ☐

Student Name	Simeon FEREZ							
Student Number	S392371							
Course	Computational & Software Techniques in Eng. (CIDA option)							
Supervisor	Dr. Jun Li							
Date of Meeting	22nd May 2023							
Meeting by	In person	☐	Telephone	☐	Skype	☐	Other	☒

Decisions / Actions agreed and by whom
Supervision Meeting #2 Review of Literature review proposal: - Focus on one aspect, either Entropy estimation or Language Modelling - A large literature has been done, now the focus is on more specific aspects and recent projects Review methodology and objectives for the next weeks Discussed several entropy estimation methods and previous works on the domain

Date of next meeting
\

Supervision Meeting Notes

Taught ☒ Research ☐

Student Name	Simeon FEREZ							
Student Number	S392371							
Course	Computational & Software Techniques in Eng. (CIDA option)							
Supervisor	Dr. Jun Li							
Date of Meeting	19th June 2023							
Meeting by	In person	☐	Telephone	☐	Skype	☐	Other	☒

Decisions / Actions agreed and by whom
Review Methodology Review current decisions on Data Collection over Twitter Review the different entropy Estimators proposed for the thesis (Plug-in, Entropy Rate, PPM) Discussion about application of Entorpy in the AI Landscape Possible outcome and applications of Entropy quantification on AI

Date of next meeting
\

Supervision Meeting Notes

Taught ☒ Research ☐

Student Name	Simeon FEREZ							
Student Number	S392371							
Course	Computational & Software Techniques in Eng. (CIDA option)							
Supervisor	Dr. Jun Li							
Date of Meeting	24th July 2023							
Meeting by	In person	☐	Telephone	☐	Skype	☐	Other	☒

Decisions / Actions agreed and by whom
Review the Technical Presentation did a few weeks ago. Gather feedback and advice on the next step of the project. Discussion about the writing methodology, internal organisation of the report. Discussion about the next steps of the thesis project. Discussion about the future after the thesis at Cranfield University.

Date of next meeting
No Next Meeting

10.2 CURES Application

15 May 2023

Dear Mr Ferez ,

Reference: CURES/18440/2023
Project ID: 21419
Title: Noisy entropy estimation and language modelling for various languages using social media data and beyond

Thank you for your application to the Cranfield University Research Ethics System (CURES).

We are pleased to inform you your CURES application, reference CURES/18440/2023 **has been reviewed. You may now proceed with the research activities you have sought approval for.**

If you have any queries, please contact CURES Support.

We wish you every success with your project.

Regards,

CURES Team