How to Compute the Probability of a Word

Tiago Pimentel, Clara Meister
ETH
{tiago.pimentel, clara.meister}@inf.ethz.ch

Abstract

Language models (LMs) estimate the probability distribution over sequences of natural language; these distributions are crucial for computing perplexity and surprisal in linguistics research. While we are usually concerned with measuring these values for words, most LMs operate over subwords. Despite seemingly straightforward, accurately computing probabilities over one unit given probabilities over the other requires care. Indeed, we show here that many recent linguistic studies have been incorrectly computing these values. This paper derives the correct methods for computing word probabilities, highlighting issues when relying on language models that use beginning-of-word (bow)-marking tokenisers, e.g., the GPT family. Empirically, we show that correcting the widespread bug in probability computations affects measured outcomes in sentence comprehension and lexical optimisation analyses.

1 Introduction

Language models (LMs) define probability distributions. After being trained on language data, these models can be used to compute estimates of the probability of a sequence of characters $\mathbf{c} \in \mathcal{C}^*$, or of a word $w_t \in \mathcal{W}$ in context $\mathbf{w}_{<t} \in \mathcal{W}^*$. While deriving such estimates is now rarely the explicit goal of training such models,[1] this use case is still critical in several fields. Estimating the probability of a sequence of characters, for instance, is necessary to compute a model's perplexity; a core evaluation metric in LM training. Estimating the probability of a word in context is necessary to compute a **word's surprisal**: $-\log p(w_t \mid \mathbf{w}_{<t})$, an important value in both psycho- and computational linguistics (Hale, 2001; Levy and Jaeger, 2007; Piantadosi et al., 2011; Pimentel et al., 2023a).

[1] Rather, LMs have become known for their high performance on downstream natural language processing (NLP) tasks (Radford et al., 2019; Touvron et al., 2023).

TL;DR: How to correctly compute word probabilities

Given a word w in context $\mathbf{w}_{<t}$, let $\mathbf{s}^{w}$ and $\mathbf{s}^{\mathbf{w}_{<t}}$ be their respective subword sequences output by a tokeniser. Further, let:

$$p(\mathbf{s}^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}}) = \prod_{i=1}^{|\mathbf{s}^{w}|} p(s_i^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}_{<i}^{w})$$

- LM with end-of-word marking tokeniser
$$p(w \mid \mathbf{w}_{<t}) = p(\mathbf{s}^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}})$$
- LM with beginning-of-word marking tokeniser
$$p(w \mid \mathbf{w}_{<t}) = p(\mathbf{s}^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}}) \underbrace{\frac{\sum_{\{s \in \overline{\mathcal{S}}_{\text{bow}}\}} p(s \mid \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w})}{\sum_{\{s \in \overline{\mathcal{S}}_{\text{bow}}\}} p(s \mid \mathbf{s}^{\mathbf{w}_{<t}})}}_{\text{“bug” fix}}$$

Figure 1: Equations for computing a word's contextual probability $p(w \mid \mathbf{w}_{<t})$ using a subword-based LM $p(s_t \mid \mathbf{s}_{<t})$. $\overline{\mathcal{S}}_{\text{bow}}$ is a subset of the tokeniser's vocabulary which marks beginnings of words.

Notably, most recent LMs operate over **subwords** (Sennrich et al., 2016; Kudo and Richardson, 2018): sequences of characters that frequently occur together. This is done for both optimisation and efficiency reasons (Gallé, 2019; Mielke et al., 2021; Zouhar et al., 2023). Subwords, however, do not necessarily constitute actual words, as defined by a language's lexicon.[2] At least superficially, converting from a probability distribution over subwords $p(\mathbf{s})$ into one over characters $p(\mathbf{c})$ or words $p(\mathbf{w})$ appears straightforward. However, some technical details are easy to overlook. For example, several sequences of subwords $\mathbf{s}$ can map to a single sequence of characters $\mathbf{c}$, implying an accurate computation of $p(\mathbf{c})$ should marginalise over these options (Cao and Rimell, 2021).

In this work, we discuss how to correctly compute a word's contextual probability: $p(w_t \mid \mathbf{w}_{<t})$.

[2] Despite the name, which we use out of convention, a subword need not strictly be subunits of words. For example, subwords can contain the marker that one chooses to use as the delineation between words (e.g., white space).

This value's computation depends on the choice of tokeniser used to define an LM's vocabulary. When using an end-of-word (eow)-marking tokeniser, computing $p(w_t | \mathbf{w}_{<t})$ is simple. However, when using a beginning-of-word (bow)-marking tokeniser, correctly computing this value is not as straightforward. We derive methods for these tokenisation schemes, which we present in Fig. 1. Since many widely-used LMs employ bow-marking tokenisers (e.g., the GPT models, Pythia, Mistral), this highlights a wide-spread "bug" in how most recent psycholinguistics and computational linguistics works compute word probabilities (present in, e.g., Oh and Schuler, 2023b; Wilcox et al., 2023; Pimentel et al., 2023a; Shain et al., 2024).

Empirically, we evaluate how correcting this computation affects the results of two prior empirical analyses: one on sentence comprehension and another on the lexicons' communicative efficiency. While these studies' conclusions do not change, we do observe statistically significant differences between the measured quantities when using the correct vs. buggy methods for computing word probabilities. We conclude this methodological choice has the ability to impact empirical analyses, and that future work should adopt our corrections.

2 What is a Word?

Despite decades of discussion and debate, there is no single, widely accepted definition of what constitutes a word (Haspelmath, 2023). Typically, definitions are made with respect to some system(s) within the language, such as its orthography, phonology, or grammar. As a concrete example, one can delineate words using the sound system of a language, assuming they delineate the domain over which certain phonological processes, such as vowel harmony, operate (Hall and Kleinhenz, 1999). Alternatively, one could define words as grammatical elements (e.g., a root plus affixes) that are cohesive, occur in a fixed order, and have a coherent meaning (Dixon and Aikhenvald, 2003). Notably grammatical and phonological words aren't in a one-to-one relationship. For example, English hyphenated elements like *editor-in-chief* or *mother-in-law* are typically analysed as a single grammatical word that contains multiple phonological words (Dixon and Aikhenvald, 2003).

We abstain from this broader discussion here. While we use the definition common to natural language processing applications—where words are defined orthographically[3]—our methods only assume the existence of a deterministic set of rules for segmenting a string into words.

3 Words and Distributions Over Them

Let $\mathcal{W}$ be a lexicon—the (potentially infinite) set of all words in a language—and $w \in \mathcal{W}$ a word in this lexicon. Further, let $\mathbf{w} \in \mathcal{W}^*$ be a sequence of words; $\mathcal{W}^*$ denotes the set of all finite-length word sequences. Now, assume distribution p describes the probability with which users of this language produce sequences $\mathbf{w}$. We can decompose these probabilities autoregressively as:

$$p(\mathbf{w}) = p(\texttt{eos} \mid \mathbf{w}) \prod_{t=1}^{|\mathbf{w}|} p(w_t \mid \mathbf{w}_{<t}) \tag{1}$$

where eos is a special end-of-sequence symbol that makes this probability distribution over $\mathcal{W}^*$ valid.[4]

This paper is concerned with the proper method for computing the probability of a word in context, i.e., $p(w_t \mid \mathbf{w}_{<t})$, using a pretrained language model. To this end, we first discuss its equivalence to other quantities, which will ultimately reveal a flaw in prior approaches to its computation. We start by defining a probability function $\mathbb{P}_{\mathcal{W}}$, which operates over sets of strings $\Psi_{\mathcal{W}} \subseteq \mathcal{W}^*$.

Definition 1. *Given distribution $p(\mathbf{w})$, we define the probability function $\mathbb{P}_{\mathcal{W}} : \mathcal{P}(\mathcal{W}^*) \to [0, 1]$ that returns the probability of any event $w \in \Psi_{\mathcal{W}} \subseteq \mathcal{W}^*$ occurring. As these events are disjoint, $\mathbb{P}_{\mathcal{W}}(\Psi_{\mathcal{W}})$ can be defined as:*

$$\mathbb{P}_{\mathcal{W}}(\Psi_{\mathcal{W}}) \stackrel{\text{def}}{=} \sum_{\mathbf{w} \in \Psi_{\mathcal{W}}} p(\mathbf{w}) \tag{2}$$

Now, let $\circ$ denote concatenation (between either strings or sets of strings), and $\mathbf{w} \circ \mathcal{W}^*$ represent the set of all strings with prefix $\mathbf{w}$: $\{\mathbf{w} \circ \mathbf{w}' \mid \mathbf{w}' \in \mathcal{W}^*\}$. We can compute our desired conditional distribution as the quotient of two evaluations of $\mathbb{P}_{\mathcal{W}}$:

$$p(w \mid \mathbf{w}_{<t}) = \frac{\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ w \circ \mathcal{W}^*)}{\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ \mathcal{W}^*)} \tag{3}$$

Note that this is a trivial invocation of the joint rule of probability: the conditional $p(w \mid \mathbf{w}_{<t})$ is equal to the probability of observing prefix $\mathbf{w}_{<t} \circ w$—represented by $\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ w \circ \mathcal{W}^*)$—divided by the

[3] Orthographic words are defined as sequences of characters surrounded by whitespace or other special delimiters, such as ' in the English clitic 's.

[4] See Du et al. (2023) for a longer discussion on when probability distributions over $\mathcal{W}^*$ are valid.

probability of observing prefix $\mathbf{w}_{<t}$—represented by $\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ \mathcal{W}^*)$. We call probabilities of the form $\mathbb{P}_{\mathcal{W}}(\mathbf{w} \circ \mathcal{W}^*)$ the **prefix probability** of $\mathbf{w}$.

Orthography. We assume here this language can be written, and that it has a standardised orthographic convention. Formally, given a language's alphabet $\mathcal{C}$, each string $\mathbf{w}$ can be mapped to a sequence of characters $\mathbf{c} \in \mathcal{C}^*$ via function ${}_{\mathcal{W}^* \to \mathcal{C}^*}\mathbb{S} : \mathcal{W}^* \to \mathcal{C}^*$. Further, we assume this language allows for straightforward segmentation from orthography. Given a sequence of characters $\mathbf{c}$, we can thus extract a sequence of words as ${}_{\mathcal{C}^* \to \mathcal{W}^*}\mathbb{S}(\mathbf{c}) = \mathbf{w}$.

4 Subwords and Language Models

Most modern language models are not defined directly as distributions over words $\mathbf{w}$, but rather as distributions over *sub*words. These subwords are themselves defined by a choice of **tokeniser**.[5] In this section, we first introduce tokenisers, and how they map words to subwords (and back). We then use these building blocks to show how we can compute word probabilities from subword probabilities.

4.1 From Words to Subwords and Back

A tokeniser comes equipped with a vocabulary $\mathcal{S}$, whose elements are subwords $s \in \mathcal{S}$; each of these subwords represents a sequence of characters $\mathbf{c} \in \mathcal{C}^*$.[6] A **detokenisation function** ${}_{\mathcal{S}^* \to \mathcal{C}^*}\mathbb{S} : \mathcal{S}^* \to \mathcal{C}^*$ can then be defined by simply mapping a sequence of subwords to the characters they represent and concatenating them together. Tokenisers also provide a **tokenisation function** ${}_{\mathcal{C}^* \to \mathcal{S}^*}\mathbb{S} : \mathcal{C}^* \to \mathcal{S}^*$, which takes as input a character sequence and maps it to a subword sequence. Notably, multiple subword sequences may map to the same character sequence; most tokenisers, however, choose one of these subword sequences as a canonical choice and use a deterministic tokenisation function.

Collectively, the mapping functions we have defined give us the ability to convert between words and subwords, which will be necessary when using subword distributions to compute word probabilities. We write word-to-subword mappings as:

$$ {}_{\mathcal{W}^* \to \mathcal{S}^*}\mathbb{S} \stackrel{\text{def}}{=} {}_{\mathcal{W}^* \to \mathcal{C}^*}\mathbb{S} \bullet {}_{\mathcal{C}^* \to \mathcal{S}^*}\mathbb{S}, \quad {}_{\mathcal{S}^* \to \mathcal{W}^*}\mathbb{S} \stackrel{\text{def}}{=} {}_{\mathcal{S}^* \to \mathcal{C}^*}\mathbb{S} \bullet {}_{\mathcal{C}^* \to \mathcal{W}^*}\mathbb{S} \tag{4} $$

[5]We are not concerned with most aspects of individual tokenisers, and will focus on general considerations here. See Mielke et al. (2021) for a more comprehensive discussion.

[6]While subwords can be mapped back to a set of characters, they need not consist of only characters from the alphabet $\mathcal{C}$. Additional markers—such as bow—can be used.

Importantly, these functions reverse each other when applied as ${}_{\mathcal{S}^* \to \mathcal{W}^*}\mathbb{S}({}_{\mathcal{W}^* \to \mathcal{S}^*}\mathbb{S}(\mathbf{w})) = \mathbf{w}$, but not necessarily when applied in the opposite order. The implication of this is that each $\mathbf{w}$ maps to a unique $\mathbf{s}$, and every $\mathbf{w}$ can be represented by some $\mathbf{s}$; but there are subword sequences that will *not* be mapped to by our tokenisation function. For example, if a tokeniser maps word *probability* to subwords [*_prob*, *ability*], then the subword sequence [*_p*, *r*, *o*, *b*, ...] will never be mapped to. We denote **unmapped subword sequences** as:

$$ \mathcal{S}_{\mathsf{x}} \stackrel{\text{def}}{=} \mathcal{S}^* \setminus \left\{ {}_{\mathcal{W}^* \to \mathcal{S}^*}\mathbb{S}(\mathbf{w}) \mid \mathbf{w} \in \mathcal{W}^* \right\} \tag{5} $$

4.2 From Word to Subword Probabilities

Now let $p_{\boldsymbol{\theta}}$ be a language model with parameters $\boldsymbol{\theta}$ and a vocabulary $\mathcal{S}$. This model defines a probability distribution over the set of all finite subword sequences $\mathbf{s} \in \mathcal{S}^*$ and its parameters are optimized to provide good estimates of the true distribution over subwords, given by:

$$ p(\mathbf{s}) = \sum_{\mathbf{w} \in \mathcal{W}^*} p(\mathbf{w}) \, \mathbb{1}\left\{ \mathbf{s} = {}_{\mathcal{W}^* \to \mathcal{S}^*}\mathbb{S}(\mathbf{w}) \right\} \tag{6} $$

As not all subword sequences are mapped to, and because each mapping in ${}_{\mathcal{W}^* \to \mathcal{S}^*}\mathbb{S}$ is unique, we can re-write this distribution as:

$$ p(\mathbf{s}) = \begin{cases} p(\mathbf{w}) & \texttt{if } \mathbf{s} = {}_{\mathcal{W}^* \to \mathcal{S}^*}\mathbb{S}(\mathbf{w}) \\ 0 & \texttt{if } \mathbf{s} \in \mathcal{S}_{\mathsf{x}} \end{cases} \tag{7} $$

Throughout this paper, we focus on **exact language models**, which we define as a $p_{\boldsymbol{\theta}}$ with the same support as p; formally, $p_{\boldsymbol{\theta}}(\mathbf{s}) = 0$ when $p(\mathbf{s}) = 0$. However, we briefly discuss how to generalise our findings to non-exact models in the next section.

4.3 From Subword to Word Probabilities

Eq. (7) suggests a way to extract probabilities over words from a language model; we can simply use the equivalence:

$$ p(\mathbf{w}) = p(\mathbf{s}), \quad \texttt{for } \mathbf{s} = {}_{\mathcal{W}^* \to \mathcal{S}^*}\mathbb{S}(\mathbf{w}) \tag{8} $$

Notably, to apply this equivalence in practice with language models $p_{\boldsymbol{\theta}}(\mathbf{s})$, the model needs to be exact.[7] While we focus on exact distributions here,

[7]Note that most neural language models *cannot* assign zero probability to any subword sequence due to their use of a softmax projection in the final step of computing probabilities; they will thus not be exact in this sense. Unmapped subword sequences can thus typically still be generated by language models. Prior work discusses whether marginalising over these sequences is important for inference and evaluation (Cao and Rimell, 2021; Chirkova et al., 2023).

we note that extending our results to inexact distributions simply requires marginalising out potential ambiguities: i.e., computing $p(\mathbf{w})$ for a given word requires summing over the (finite) set of subword sequences which map to it (Cao and Rimell, 2021).

The implication of eq. (8) is that if we can create a subword set $\Psi_{\mathcal{S}}$ that is "equivalent" to a chosen word set $\Psi_{\mathcal{W}}$, we would be able to compute $\Psi_{\mathcal{W}}$'s probability by summing over the subwords in $\Psi_{\mathcal{S}}$. Formally, we define the equivalence between two sets as:

$$\Psi_{\mathcal{W}} \triangleq \Psi_{\mathcal{S}} \implies \left(w \in \Psi_{\mathcal{W}} \iff \underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) \in \Psi_{\mathcal{S}} \right) \quad (9)$$

Now let $\mathbb{P}_{\mathcal{S}}$ be a probability function defined analogously to $\mathbb{P}_{\mathcal{W}}$ (in Defn. 1). It then follows that:

$$\mathbb{P}_{\mathcal{W}}(\Psi_{\mathcal{W}}) = \mathbb{P}_{\mathcal{S}}(\Psi_{\mathcal{S}}), \quad \texttt{for } \Psi_{\mathcal{W}} \triangleq \Psi_{\mathcal{S}} \quad (10)$$

We are now in a position to define our quantity of interest $p(w \mid \mathbf{w}_{<t})$ in terms of subword probabilities: it is simply the quotient of $\mathbb{P}_{\mathcal{S}}(\cdot)$ for two different sets $\Psi_{\mathcal{S}}$.

Lemma 1. *The contextual probability of a word can be computed using probability distributions over subwords as:*

$$p(w \mid \mathbf{w}_{<t}) = \frac{\mathbb{P}_{\mathcal{S}}(\Psi'_{\mathcal{S}})}{\mathbb{P}_{\mathcal{S}}(\Psi''_{\mathcal{S}})} \quad (11)$$

where $\Psi'_{\mathcal{S}} \triangleq \mathbf{w}_{<t} \circ w \circ \mathcal{W}^*$ *and* $\Psi''_{\mathcal{S}} \triangleq \mathbf{w}_{<t} \circ \mathcal{W}^*$.

Proof. This result follows from a simple application of the equivalence in eq. (10) to the definition of $p(w \mid \mathbf{w}_{<t})$ in eq. (3). □

Luckily, it is straightforward to find the sets $\Psi'_{\mathcal{S}}$ and $\Psi''_{\mathcal{S}}$ required by Lemma 1. This is because, for a given word set $\Psi_{\mathcal{W}}$, the subword set

$$\Psi_{\mathcal{S}} = \{ \underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) \mid \mathbf{w} \in \Psi_{\mathcal{W}} \} \quad (12)$$

meets the equivalence $\Psi_{\mathcal{W}} \triangleq \Psi_{\mathcal{S}}$. By construction, we have that $\mathbf{w} \in \Psi_{\mathcal{W}} \implies \underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) \in \Psi_{\mathcal{S}}$. Further, due to the injectivity of $\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}$, it must be that $\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) \in \Psi_{\mathcal{S}} \implies \mathbf{w} \in \Psi_{\mathcal{W}}$, proving both sides of the equivalence in eq. (9).

Before making use of eq. (11) for computing contextual probabilities, however, there is still one hurdle to overcome: the two sets $\Psi'_{\mathcal{W}} = (\mathbf{w}_{<t} \circ w \circ \mathcal{W}^*)$ and $\Psi''_{\mathcal{W}} = (\mathbf{w}_{<t} \circ \mathcal{W}^*)$ are both infinite. We must thus find a more efficient strategy to compute probabilities than summing over the (also infinite) sets $\Psi'_{\mathcal{S}}$ and $\Psi''_{\mathcal{S}}$.

4.4 Leveraging LMs' Autoregressiveness

We now discuss how we can leverage the fact that most LMs compute probabilities autoregressively to efficiently compute the probabilities in Lemma 1. In short, most LMs provide estimates of conditional probabilities: $p(s \mid \mathbf{s}_{<t})$. Given eq. (3) and the fact that $\mathbb{P}_{\mathcal{S}}(\mathcal{S}^*) = 1$, we can use these conditionals to compute prefix probabilities efficiently.

Lemma 2. *We can use conditional probabilities to compute prefix probabilities as:*

$$\mathbb{P}_{\mathcal{S}}(\mathbf{s} \circ \mathcal{S}^*) = \prod_{t=1}^{|\mathbf{s}|} \frac{\mathbb{P}_{\mathcal{S}}(\mathbf{s}_{<t} \circ s_t \circ \mathcal{S}^*)}{\mathbb{P}_{\mathcal{S}}(\mathbf{s}_{<t} \circ \mathcal{S}^*)} = \prod_{t=1}^{|\mathbf{s}|} p(s_t \mid \mathbf{s}_{<t}) \quad (13)$$

It follows that, if we can find a set of subword sequences $\Psi_{\mathcal{S}} = \{\mathbf{s}^{(k)}\}_{k=1}^{K}$ for which we have the equivalence $\mathbf{w} \circ \mathcal{W}^* \triangleq \bigcup_{\mathbf{s} \in \Psi_{\mathcal{S}}} \mathbf{s} \circ \mathcal{S}^*$, then we can compute prefix probabilities as:[8]

$$\mathbb{P}_{\mathcal{W}}(\mathbf{w} \circ \mathcal{W}^*) = \mathbb{P}_{\mathcal{S}}\left(\bigcup_{\mathbf{s} \in \Psi_{\mathcal{S}}} \mathbf{s} \circ \mathcal{S}^* \right) \quad (14a)$$

$$= \sum_{\mathbf{s} \in \Psi_{\mathcal{S}}} \mathbb{P}_{\mathcal{S}}(\mathbf{s} \circ \mathcal{S}^*) \quad (14b)$$

In turn, these let us compute $p(w \mid \mathbf{w}_{<t})$ efficiently.

5 The Nuances of Mapping: Tokeniser-dependent Strategies

We are left with the task of finding a set of subword prefixes which will allow us to compute the probabilities of $\Psi'_{\mathcal{S}} \triangleq \Psi'_{\mathcal{W}}$ and $\Psi''_{\mathcal{S}} \triangleq \Psi''_{\mathcal{W}}$. In this section, we discuss how our tokeniser's specification—specifically whether it uses end- or beginning-of-word markings in its vocabulary—affects this task.

5.1 Segmentation-aware Tokenisers

In the following sections, we consider $\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}$ that operate independently over words in a sequence $\mathbf{w}$. This is necessary for our methods below, and is a common practice in NLP (typically called pre-tokenisation) where a text is segmented according to some criterion (e.g., white space) before being converted into subwords by a tokeniser. Here, we consider pre-tokenisation to be one of the steps implemented by $\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}$. We formalise this in the following definition.

[8] Note that the prefix sets $\mathbf{s} \circ \mathcal{S}^*$ are disjoint for $\mathbf{s} \in \Psi_{\mathcal{S}}$.

$$\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{How}\circ\texttt{do}\circ\texttt{you}\circ\texttt{compute}\circ\texttt{a}\circ\texttt{word}\circ\texttt{'s}\circ\texttt{probability}\circ\texttt{?})\qquad \text{eow-marked, split punctuation}$$

$$= \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{How})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{do})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{you})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{compute})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{a})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{word})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{'s})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{probability})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{?})$$

$$= \textit{How\_}\circ\textit{do\_}\circ\textit{you\_}\circ\textit{comp}\circ\textit{ute\_}\circ\textit{a\_}\circ\textit{word}\circ\textit{'s\_}\circ\textit{prob}\circ\textit{ability}\circ\textit{?}$$

$$\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{How}\circ\texttt{do}\circ\texttt{you}\circ\texttt{compute}\circ\texttt{a}\circ\texttt{word}\circ\texttt{'s}\circ\texttt{probability}\circ\texttt{?})\qquad \text{bow-marked, split punctuation}$$

$$= \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{How})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{do})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{you})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{compute})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{a})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{word})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{'s})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{probability})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\texttt{?})$$

$$= \textit{How}\circ\textit{\_do}\circ\textit{\_you}\circ\textit{\_comp}\circ\textit{ute}\circ\textit{\_a}\circ\textit{\_word}\circ\textit{'s}\circ\textit{\_prob}\circ\textit{ability}\circ\textit{?}$$

Figure 2: The output of tokenisers with different methods of handling word delineations.

Definition 2. *We define a **segmentation-aware tokeniser** as one whose operations can be decomposed across words in a sequence, i.e.:*

$$\begin{aligned}\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) &= \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}_{<t})\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(w_t)\circ\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}_{>t}) \\ &= \underset{\mathcal{W}\to\mathcal{S}^*}{\mathbb{S}}(w_1)\circ\underset{\mathcal{W}\to\mathcal{S}^*}{\mathbb{S}}(w_2)\circ\cdots\circ\underset{\mathcal{W}\to\mathcal{S}^*}{\mathbb{S}}(w_{|\mathbf{w}|})\end{aligned} \tag{15}$$

While it is possible to create tokenisers with vocabularies in which subwords can cross word boundaries, the majority of them meet this criterion. E.g., the `sentencepiece` library (Kudo and Richardson, 2018) has an option to allow for multi-word subwords when learning a tokeniser's vocabulary, but by default it does not allow them.

The decomposition in Defn. 2 has an important implication. As discussed in §4.1, the (sequence-level) tokenisation function $\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}$ must be injective, meaning that each word sequence must map to a unique subword sequence. The word-level tokenisation function $\underset{\mathcal{W}\to\mathcal{S}^*}{\mathbb{S}}$, thus, must have the property that concatenating its outputs always leads to a unique string. This property is known in the compression literature as unique decodability (Cover and Thomas, 2006, page 105). While there are several ways to guarantee unique decodability, most tokenisers rely on relatively simple strategies: they either mark the ends or beginnings of words (eow or bow) using a subset of the subwords in $\mathcal{S}$. We discuss these strategies next.

5.2 End of Word Markings

We now consider eow-marking tokenisers. These tokenisers use a subset of their vocabulary $\mathcal{S}_{\texttt{eow}}\subseteq\mathcal{S}$ to indicate the end of words,[9] with the rest of the vocabulary $\mathcal{S}_{\texttt{mid}}\overset{\text{def}}{=}\mathcal{S}\setminus\mathcal{S}_{\texttt{eow}}$ mapping back to the beginning or middle of words.

Definition 3. *An **eow-marking tokeniser** is a segmentation-aware tokeniser which marks ends of words. Its word-level tokenisation function can be written as* $\underset{\mathcal{W}\to\mathcal{S}^*}{\mathbb{S}^{\texttt{eow}}}:\mathcal{W}\to\mathcal{S}^*_{\texttt{mid}}\circ\mathcal{S}_{\texttt{eow}}$.[10]

Importantly, given the definition above, when a subword $s_t\in\mathcal{S}_{\texttt{eow}}$ is observed, it means that the current subsequence $\mathbf{s}_{t':t}$ (where $t'\le t$) can be mapped back to a word, and that a subsequence representing a new word will begin at s_{t+1}. (The current subsequence $\mathbf{s}_{t':t}$ is thus determined by the smallest t' for which $\mathbf{s}_{t':t-1}\in\mathcal{S}^*_{\texttt{mid}}$; note that this means either $t'=1$ or $s_{t'-1}\in\mathcal{S}_{\texttt{eow}}$.) The implication of this property is that eow-marking tokenisers provide **instantaneous decodability** (Cover and Thomas, 2006, page 106): prefix $\mathbf{s}_{\le t}$ with $s_t\in\mathcal{S}_{\texttt{eow}}$ is instantaneously decodable, as it always maps to the same words, regardless of its continuation $\mathbf{s}_{>t}$. Instantaneous decodability allows us to compute the contextual probability of a word as follows.

Theorem 1. *Let $\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}$ be a eow-marking tokeniser. Further, let $\mathbf{s}^{\mathbf{w}}\overset{\text{def}}{=}\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w})$. We can show the following equivalence:*

$$\begin{aligned}\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t}\circ\mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}}\circ\mathcal{S}^*) \\ \mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t}\circ w\circ\mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}}\circ\mathbf{s}^{w}\circ\mathcal{S}^*)\end{aligned} \tag{16}$$

Further, we can compute a word's probability as:

$$p(w\mid\mathbf{w}_{<t}) = \underbrace{\prod_{t'=1}^{|\mathbf{s}^w|} p\left(s^w_{t'}\mid\mathbf{s}^{\mathbf{w}_{<t}}\circ\mathbf{s}^w_{<t'}\right)}_{p(\mathbf{s}^w\mid\mathbf{s}^{\mathbf{w}_{<t}})} \tag{17}$$

Proof. See App. C.1 for formal proof. □

Eq. (16) follows from instantaneous decodability, as every sequence $\mathbf{s}\in\mathbf{s}^{\mathbf{w}}\circ\mathcal{S}^*$ maps back to $\mathbf{w}\circ\mathcal{W}^*$.

[9] The case of $\mathcal{S}_{\texttt{eow}}=\mathcal{S}$ or $\mathcal{S}_{\texttt{bow}}=\mathcal{S}$ happens when $\mathcal{S}=\mathcal{W}$; while possible in theory, it will not happen in practice since a language model cannot have an infinite vocabulary.

[10] Note that only subword sequences of the form $\{\texttt{eos}\}\cup(\mathcal{S}^*\circ\mathcal{S}_{\texttt{eow}})$ are valid under this tokeniser. This is because: $\{\texttt{eos}\}\cup(\mathcal{S}^*\circ\mathcal{S}_{\texttt{eow}})=\bigcup_{i=0}^{\infty}(\mathcal{S}^*_{\texttt{mid}}\circ\mathcal{S}_{\texttt{eow}})^i$. Invalid sequences do not affect results, as they are in $\mathcal{S}_{\text{x}}$ and thus have $p(\mathbf{s})=0$.

Eq. (17) then follows from a simple application of Lemmas 1 and 2:

$$p\left(\mathbf{s}^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}}\right) = \frac{\prod_{t'=1}^{|\mathbf{s}^{\mathbf{w}_{<t}\circ w}|} p\left(\mathbf{s}_{t'}^{\mathbf{w}_{<t}\circ w} \mid \mathbf{s}_{<t'}^{\mathbf{w}_{<t}\circ w}\right)}{\prod_{t'=1}^{|\mathbf{s}^{\mathbf{w}_{<t}}|} p\left(\mathbf{s}_{t'}^{\mathbf{w}_{<t}} \mid \mathbf{s}_{<t'}^{\mathbf{w}_{<t}}\right)} \tag{18}$$

Notably, eq. (17) is fairly straightforward and is the way in which most NLP practioners would compute a word's probability. In the next section, however, we see that it would not compute the correct probabilities if using bow-marking tokenisers.

5.3 Beginning of Word Markings

We now consider bow-marking tokenisers. Analogously to the eow case, a subset of a bow-marking tokeniser's vocabulary $\mathcal{S}_{\text{bow}} \subseteq \mathcal{S}$ is used exclusively to indicate word beginnings. The rest of the vocabulary $\mathcal{S}_{\text{mid}} \overset{\text{def}}{=} \mathcal{S} \setminus \mathcal{S}_{\text{bow}}$ then represents either the middle or end of words. We provide a formal definition of this tokeniser below.

Definition 4. *A* **bow-*marking tokeniser*** *is a segmentation-aware tokeniser which marks beginnings of words. Its word-level tokenisation function can be written as* $\mathbb{S}^{\text{bow}}_{\mathcal{W}\to\mathcal{S}^*} : \mathcal{W} \to \mathcal{S}_{\text{bow}} \circ \mathcal{S}^*_{\text{mid}}$.[11]

Given the definition above, when a subword $s_t \in \mathcal{S}_{\text{bow}}$ is observed, it thus means that a *previous* subsequence $\mathbf{s}_{t':t-1}$ can be mapped back to a word, and that a subsequence representing a new word begins at s_t. (The previous subsequence $\mathbf{s}_{t':t-1}$ is determined by $s_{t'} \in \mathcal{S}_{\text{bow}}$ and $\mathbf{s}_{t'+1:t-1} \in \mathcal{S}^*_{\text{mid}}$.) Such tokenisers are thus *not* instantaneously decodable. They only provide what we term **near-instantaneous decodability**: a prefix $\mathbf{s}_{\le t}$ does *not* always map to the same words, as its mapping depends on whether the following subword s_{t+1} is in $\mathcal{S}_{\text{bow}} \cup \{\text{eos}\}$.[12] Computing probabilities with near-instantaneous codes thus requires discounting the probability of continuations $s_{t+1} \notin \mathcal{S}_{\text{bow}} \cup \{\text{eos}\}$; we label this discount factor as **Bug Fix** 1.

Theorem 2. *Let* $\mathbb{S}_{\mathcal{W}^*\to\mathcal{S}^*}$ *be a* bow*-marking tokeniser. Further, let* $\overline{\cdot}$ *represent the union of a set with* eos*, e.g.,* $\overline{\mathcal{S}}_{\text{bow}} = \mathcal{S}_{\text{bow}} \cup \{\text{eos}\}$*. We can show the following equivalence:*

$$\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ \mathcal{W}^*) = \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}) \tag{19}$$

$$\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ w \circ \mathcal{W}^*) = \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*})$$

Further, we can compute a word's probability as:

$$p(w \mid \mathbf{w}_{<t}) = \tag{20}$$

$$\underbrace{\prod_{t'=1}^{|\mathbf{s}^{w}|} p\left(\mathbf{s}_{t'}^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}_{<t'}^{w}\right)}_{p(\mathbf{s}^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}})} \underbrace{\frac{\sum_{\{s\in\overline{\mathcal{S}}_{\text{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w}\right)}{\sum_{\{s\in\overline{\mathcal{S}}_{\text{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t}}\right)}}_{\textbf{Bug Fix } 1}$$

Proof. See App. C.2 for formal proof. □

Eq. (19) follows from near-instantaneous decodability, as every sequence $\mathbf{s}^{\mathbf{w}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}$ maps back to $\mathbf{w} \circ \mathcal{W}^*$, but sequences in $\mathbf{s}^{\mathbf{w}} \circ \mathcal{S}_{\text{mid}} \circ \mathcal{S}^*$ do not.

5.4 Practical Concerns and Corner Cases

In this section, we discuss corner cases that deserve special consideration. Many of these cases arise because of practical demands, e.g., ensuring the presence or absence of white space where appropriate. Notably, the need for these corner cases is often language-dependent, as they arise due to orthographic conventions. We discuss the implications of two tokeniser conventions that handle special cases: the treatment of the beginnings and ends of sequences.

Non-eow-marked Final Words. Several eow-marking tokenisers do not decompose exactly as in eq. (15), but treat the final word in a sequence differently. Specifically, they override the behaviour of $\mathbb{S}_{\mathcal{W}\to\mathcal{S}^*}$ on the these words and do *not* use subwords from $\mathcal{S}_{\text{eow}}$ to mark its ends. This is also often the treatment applied to words followed immediately by punctuation. This mechanism allows tokenisers to avoid implying the existence of a white space that does not exist, e.g., after the end of a string. Notably, this breaks instantaneous decodability—making this code only near-instantaneous. Let $\mathbf{s}^{w}_{\text{mid}} \overset{\text{def}}{=} \mathbb{S}^{\text{mid}}_{\mathcal{W}\to\mathcal{S}^*}(w)$, where $\mathbb{S}^{\text{mid}}_{\mathcal{W}\to\mathcal{S}^*} : \mathcal{W} \to \mathcal{S}^*_{\text{mid}}$. Upon observing subsequence $\mathbf{s}^{w}_{\text{mid}}$, we cannot instantaneously map it back to w, and must wait for the next symbol: if $\mathbf{s}^{w}_{\text{mid}}$ is followed by either eos or punctuation, then it is mapped back to w; if not, it is mapped to another word. Handling this thus requires the following fix (termed **Bug Fix** 2 here):

$$p(w \mid \mathbf{w}_{<t}) = \tag{21}$$

$$\left(p\left(\mathbf{s}^{w}_{\text{mid}} \mid \mathbf{s}^{\mathbf{w}_{<t}}\right) \underbrace{\sum_{s\in\overline{\mathcal{S}}_{!?}} p(s \mid \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w}_{\text{mid}})\right) + p(\mathbf{s}^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}})}_{\textbf{Bug Fix } 2}$$

[11]Similarly to the eow case, not all subword sequences are valid under bow tokenisers, only sequences of the form $\{\text{eos}\} \cup (\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*)$.

[12]Here, we define the concatenation of any sequence with eos to be itself, e.g., $\mathbf{s} \circ \text{eos} = \mathbf{s}$.

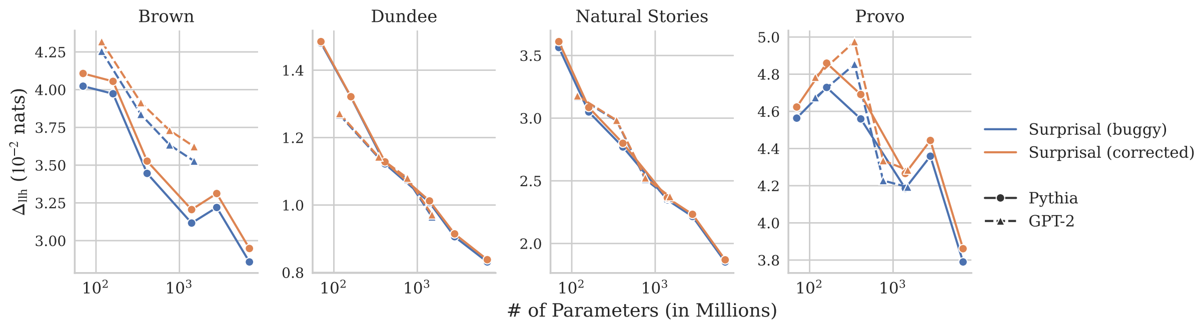

Figure 3: Δ_{llh} between regressors with and without surprisal as a predictor. We include Δ_{llh} when using surprisal estimates computed from language models across several sizes and families. Results are presented both when using the buggy and correct methods for surprisal estimation.

Non-bow-marked First Words. Just as eow-marking tokenisers often treat final words differently, bow-marking tokenisers treat the first word in a sequence differently to handle white space appropriately. These tokenisers typically do *not* mark first words with bow, and instead apply $\mathbb{S}^{\texttt{mid}}_{\mathcal{W}\to\mathcal{S}^*}$ to w_1. This affects the probability computation of the first word in a sequence. In such cases, the prefix $\mathbf{w}_{<t}$ of the first word is empty (denoted here as ""). While computing a word's contextual probability according to eq. (19) requires computing $\mathbb{P}_{\mathcal{S}}(\overline{\mathcal{S}_{\texttt{bow}} \circ \mathcal{S}^*})$, the first subword in a sequence will not be in $\mathcal{S}_{\texttt{bow}}$, but in $\mathcal{S}_{\texttt{mid}}$ instead. The probability computations of such words thus requires the following correction (**Bug Fix** 3):

$$p(w \mid \text{``''}) = p\left(\mathbf{s}^{w}_{\texttt{mid}} \mid \text{``''}\right) \underbrace{\frac{\sum_{\{s \in \bar{\mathcal{S}}_{\texttt{bow}}\}} p\left(s \mid \mathbf{s}^{w}\right)}{\sum_{\{s \in \bar{\mathcal{S}}_{\texttt{mid}}\}} p\left(s \mid \text{``''}\right)}}_{\textbf{Bug Fix } 3} \tag{22}$$

6 Experiments

We now investigate how correcting the computation of word probabilities affects the results of prior studies. We explore two settings: psycholinguistics experiments surrounding sentence comprehension (Hale, 2001; Levy, 2008) and computational linguistics experiments assessing the lexicon's communicative efficiency (Piantadosi et al., 2011; Gibson et al., 2019). We follow these works' experimental methodologies, observing how the use of corrected surprisal estimates impacts the conclusions originally drawn using their standard (buggy) surprisal estimates.

Models. In our first experiment, we estimate contextual probabilities using GPT-2 (Radford et al., 2019) and Pythia (Biderman et al., 2023); in the second, we focus only on Pythia. Both these suites contain language models of various sizes. We use these models' open-source versions from the `transformers` library (Wolf et al., 2020). GPT-2 and Pythia use bow-marking tokenisers, meaning we employ the methods discussed in §5.3 to compute words' contextual probabilities.

6.1 Sentence Comprehension

Surprisal theory (Hale, 2001; Levy, 2008) hypothesises that readers keep a belief distribution over meanings while reading; after observing each word in a sentence, they must thus update this distribution. Under some assumptions about how these belief updates are performed, surprisal theory then predicts that their cost is related to a word's **surprisal**, defined as the negative log-probability:

$$h(w_t) \stackrel{\text{def}}{=} -\log p(w_t \mid \mathbf{w}_{<t}) \tag{23}$$

Surprisal theory is widely-accepted as a model of comprehension effort, with numerous works empirically supporting it (Smith and Levy, 2008, 2013; Goodkind and Bicknell, 2018; Shain, 2019; Wilcox et al., 2020; Oh et al., 2022; Wilcox et al., 2023; Shain et al., 2024, *inter alia*). Notably, the true contextual probabilities $p(w_t \mid \mathbf{w}_{<t})$ required to compute surprisal are unknown, and must be approximated. All of the works above use language models to do so, with the most recent using LMs which operate on top of subwords produced by bow-marking tokenisers (e.g., Oh and Schuler, 2023b,a; Shain et al., 2024; Pimentel et al., 2023b). Notably, these works compute surprisal estimates using the "buggy" versions of $p(w_t \mid \mathbf{w}_{<t})$. In this section, we reproduce some of these prior works' results, observing how this correction affects results.

Setup Summary. We run our analyses on 4 reading times datasets—Brown, Dundee, Natural Sto-

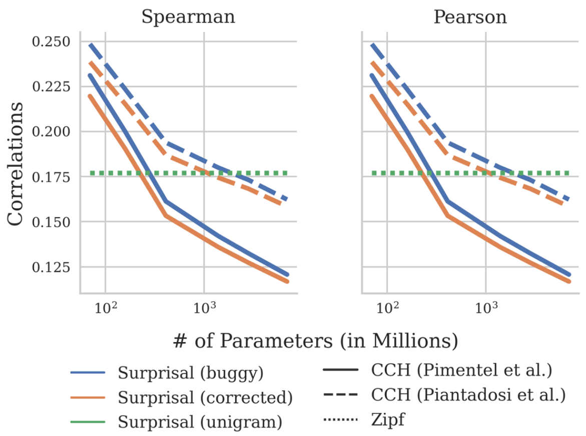

Figure 4: Correlation between English word lengths and the values predicted by either a Zipfian notion of optimality, or the channel capacity hypothesis. CCH (Pimentel et al.) and CCH (Piantadosi et al.) refer to eqs. (24) and (25).

ries, and Provo. Further, following prior work (Wilcox et al., 2023; Oh and Schuler, 2023b), we evaluate surprisal's predictive power over reading times by measuring the change in data log-likelihood Δ_{llh} when using linear regressors with and without surprisal as a predictor. More details about our experimental setup are in App. A.1.

Results. Fig. 3 shows the change in data log-likelihood under regressors with and without surprisal as a predictor; values are detailed in Tab. 1 (in the appendix). We first note that the predictive power of surprisal decreases as language model size increases, as observed in prior work (Oh and Schuler, 2023b; Shain et al., 2024). Here however, we are more interested in the effect of our corrections on these results—labelled as buggy vs. corrected surprisal. Interestingly, we observe only small changes in predictive power due to our correction, and these changes are not significant individually for each model. However, when analysed in aggregate for all models, we see this positive improvement is consistent and significant for Brown, Natural Stories and Provo ($\alpha < 0.01$ in our permutation tests). These results can be seen in Tab. 1.

6.2 Communicative Efficiency

Languages' lexicons have been studied for decades in the effort to gain better insights about the forces that shape natural languages (Zipf, 1935; Howes, 1968; Bentz and Ferrer-i-Cancho, 2016; Levshina, 2022). One characteristic of particular interest has been word lengths and how a tendency for communicative efficiency has influenced them. There are several hypotheses about the exact way in which this tendency takes effect. Zipf (1935) argues that speakers have a tendency towards minimising utterance lengths, and therefore that word lengths should correlate with frequencies. Piantadosi et al. (2011) argues that speakers maximise information transfer, and thus word lengths should correlate with a word's expected surprisal instead:

$$\mathbb{E}[h(w_t)] \stackrel{\text{def}}{=} \mathop{\mathbb{E}}_{\mathbf{w}_{<t}} \left[-\log p(w_t \mid \mathbf{w}_{<t}) \mid w_t\right] \quad (24)$$

We follow Pimentel et al. (2023a) in calling this the **channel capacity hypothesis** (CCH). Finally, Pimentel et al. (2023a) point out an issue with Piantadosi et al.'s solution, and argues that to maximise information transfer, lengths should correlate the following value instead:[13]

$$\frac{\mathbb{E}[h^2(w_t)]}{\mathbb{E}[h(w_t)]} \stackrel{\text{def}}{=} \frac{\mathbb{E}_{\mathbf{w}_{<t}}\left[(-\log p(w_t \mid \mathbf{w}_{<t}))^2 \mid w_t\right]}{\mathbb{E}_{\mathbf{w}_{<t}}\left[-\log p(w_t \mid \mathbf{w}_{<t}) \mid w_t\right]} \quad (25)$$

Setup Summary. We run our analysis using a subset of the English portion of Wiki-40B (Guo et al., 2020). We compare the three values above (unigram frequency, and eqs. (24) and (25)); evaluating them based on their correlation with words' lengths. Two of these values depend on a word's contextual probability, and we thus also compare their fixed vs. buggy versions.

Results. The results in Fig. 4 agree with the findings of Pimentel et al. (2023a): once larger (and better) language models are used to estimate words' surprisals, the metrics under the CCH hypothesis (both Piantadosi et al.'s and Pimentel et al.'s versions) become weaker predictors of word lengths. Interestingly, correcting the computation of surprisals also leads to a drop in the correlations between CCH predictors and word lengths. Improving CCH's predictors thus consistently hurts its predictive power over word lengths—either when using better models, Pimentel et al.'s fix to CCH's optimal solution, or our fix to probability computations. We conclude, as Pimentel et al., that word lengths are best predicted by Zipf's hypothesis.

7 Conclusion

This work expounds on the intricacies of accurately computing contextual word probabilities using language models. We focus on the challenges posed by the use of subword vocabularies. We show that

[13] See their paper for a derivation for this fix.

subword vocabularies defined using beginning-of-word (bow) tokenisers—common in many modern LMs—introduce complexities that are often overlooked. We point out that this has led to potential inaccuracies in the probability estimation of various prior empirical analyses. Our methodological corrections lead to significant differences in results, although the overarching conclusions of the previous studies that we explore remain the same. This finding underscores the importance of precise computational methods in linguistic research. Future work should ensure these corrections are adopted to enhance the reliability of their analyses.

Limitations

The authors see limitations with both the theoretical and empirical aspects of this work. Perhaps the main theoretical limitation is the lack of consideration of all potential corner cases which tokenisers might implement (similar to, e.g., those discussed in §5.4). The use of white space differs from language to language, and many corner cases of tokeniser behavior are designed specifically to handle this. There are likely other fixes to probability computations that would need to be derived in order to handle paradigms not discussed in §5.4. In Spanish, for instance, words following "¿" are usually not bow-marked, and might thus require the use of an approach similar to **Bug Fix** 3. Our theoretical results are also limited to autoregressive models. While the majority of today's language models meet this criterion, it is feasible that future language models would be designed differently and consequently, our methods would no longer be necessarily applicable. On the empirical side, a large limitation of our work is the exploration of the impact of our methods in only two studies. Further, our experiments are limited to English. Additional studies are thus needed to understand the full extent to which our corrections impact empirical results in other languages and in other areas of computational linguistics (and of NLP, more broadly).

Acknowledgments

We thank Ethan Wilcox for many discussions about this paper, and for helping to write parts of it. We also thank Sotiris Anagnostidis for feedback on an earlier version of this manuscript.

References

Christian Bentz and Ramon Ferrer-i-Cancho. 2016. Zipf's law of abbreviation as a language universal. In *Proceedings of the Leiden Workshop on Capturing Phylogenetic Algorithms for Linguistics*. Universität Tübingen.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23.

Kris Cao and Laura Rimell. 2021. You should evaluate your language model on marginal likelihood over tokenisations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2104–2114, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nadezhda Chirkova, Germán Kruszewski, Jos Rozen, and Marc Dymetman. 2023. Should you marginalize over possible tokenizations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–12, Toronto, Canada. Association for Computational Linguistics.

Thomas M. Cover and Joy A. Thomas. 2006. *Elements of Information Theory*, second edition. Wiley-Interscience.

Robert M. W. Dixon and Alexandra Y. Aikhenvald. 2003. Word: a typological framework. In *Word: A Cross-linguistic Typology*, page 1–41. Cambridge University Press.

Li Du, Lucas Torroba Hennigen, Tiago Pimentel, Clara Meister, Jason Eisner, and Ryan Cotterell. 2023. A measure-theoretic characterization of tight language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9744–9770, Toronto, Canada. Association for Computational Linguistics.

Richard Futrell, Edward Gibson, Harry J. Tily, Idan Blank, Anastasia Vishnevetsky, Steven Piantadosi, and Evelina Fedorenko. 2018. The natural stories corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*. European Language Resources Association.

Matthias Gallé. 2019. Investigating the effectiveness of BPE: The power of shorter sequences. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1375–1381, Hong Kong, China. Association for Computational Linguistics.

Edward Gibson, Richard Futrell, Steven T. Piantadosi, Isabelle Dautriche, Kyle Mahowald, Leon Bergen, and Roger Levy. 2019. How efficiency shapes human language. *Trends in Cognitive Sciences*, 23(5):389–407.

Adam Goodkind and Klinton Bicknell. 2018. Predictive power of word surprisal for reading times is a linear function of language model quality. In *Proceedings of the 8th Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2018)*, pages 10–18, Salt Lake City, Utah. Association for Computational Linguistics.

Mandy Guo, Zihang Dai, Denny Vrandečić, and Rami Al-Rfou. 2020. Wiki-40B: Multilingual language model dataset. In *Proceedings of the 12th Language Resources and Evaluation Conference*, page 2440–2452, Marseille, France. European Language Resources Association.

John Hale. 2001. A probabilistic Earley parser as a psycholinguistic model. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 1–8.

Tracy Alan Hall and Ursula Kleinhenz. 1999. *Studies on the phonological word*. John Benjamins.

Martin Haspelmath. 2023. Defining the word. *WORD*, 69(3):283–297.

Davis Howes. 1968. Zipf's law and Miller's random-monkey model. *The American Journal of Psychology*, 81(2):269–272.

Alan Kennedy, Robin Hill, and Joel Pynte. 2003. The Dundee corpus. In *Proceedings of the 12th European Conference on Eye Movements*.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Natalia Levshina. 2022. Frequency, informativity and word length: Insights from typologically diverse corpora. *Entropy*, 24(2).

Roger Levy. 2008. Expectation-based syntactic comprehension. *Cognition*, 106(3):1126–1177.

Roger Levy and T. Florian Jaeger. 2007. Speakers optimize information density through syntactic reduction. In *Advances in Neural Information Processing Systems*, volume 19. MIT Press.

Steven G. Luke and Kiel Christianson. 2018. The Provo corpus: A large eye-tracking corpus with predictability norms. *Behavior Research Methods*, 50(2):826–833.

Sabrina J. Mielke, Zaid Alyafeai, Elizabeth Salesky, Colin Raffel, Manan Dey, Matthias Gallé, Arun Raja, Chenglei Si, Wilson Y. Lee, Benoît Sagot, and Samson Tan. 2021. Between words and characters: A brief history of open-vocabulary modeling and tokenization in nlp. *CoRR*, abs/2112.10508.

Byung-Doh Oh, Christian Clark, and William Schuler. 2022. Comparison of structural parsers and neural language models as surprisal estimators. *Frontiers in Artificial Intelligence*, 5.

Byung-Doh Oh and William Schuler. 2023a. Transformer-based language model surprisal predicts human reading times best with about two billion training tokens. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1915–1921, Singapore. Association for Computational Linguistics.

Byung-Doh Oh and William Schuler. 2023b. Why does surprisal from larger transformer-based language models provide a poorer fit to human reading times? *Transactions of the Association for Computational Linguistics*, 11:336–350.

Steven T. Piantadosi, Harry Tily, and Edward Gibson. 2011. Word lengths are optimized for efficient communication. *Proceedings of the National Academy of Sciences*, 108(9):3526–3529.

Tiago Pimentel, Clara Meister, Ethan Wilcox, Kyle Mahowald, and Ryan Cotterell. 2023a. Revisiting the optimality of word lengths. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2240–2255, Singapore. Association for Computational Linguistics.

Tiago Pimentel, Clara Meister, Ethan G. Wilcox, Roger P. Levy, and Ryan Cotterell. 2023b. On the effect of anticipation on reading times. *Transactions of the Association for Computational Linguistics*, 11:1624–1642.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Cory Shain. 2019. A large-scale study of the effects of word frequency and predictability in naturalistic reading. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4086–4094, Minneapolis, Minnesota. Association for Computational Linguistics.

Cory Shain, Clara Meister, Tiago Pimentel, Ryan Cotterell, and Roger Levy. 2024. Large-scale evidence for logarithmic effects of word predictability on reading time. *Proceedings of the National Academy of Sciences*, 121(10):e2307876121.

Nathaniel J. Smith and Roger Levy. 2008. Optimal processing times in reading: a formal model and empirical investigation. In *Proceedings of the Cognitive Science Society*, volume 30, pages 595–600.

Nathaniel J. Smith and Roger Levy. 2013. The effect of word predictability on reading time is logarithmic. *Cognition*, 128(3):302–319.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Ethan Wilcox, Clara Meister, Ryan Cotterell, and Tiago Pimentel. 2023. Language model quality correlates with psychometric predictive power in multiple languages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7503–7511, Singapore. Association for Computational Linguistics.

Ethan Gotlieb Wilcox, Jon Gauthier, Jennifer Hu, Peng Qian, and Roger Levy. 2020. On the predictive power of neural language models for human real-time comprehension behavior. In *Proceedings of the Cognitive Science Society*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

George K. Zipf. 1935. *The Psychobiology of Language*. London: Routledge.

Vilém Zouhar, Clara Meister, Juan Gastaldi, Li Du, Mrinmaya Sachan, and Ryan Cotterell. 2023. Tokenization and the noiseless channel. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5184–5207, Toronto, Canada. Association for Computational Linguistics.

A Experimental Setup

A.1 Sentence Comprehension

Data. We use four well-established reading time datasets, in which participants were given text passages to read and their reading time was recorded. For two of these datasets—Natural Stories (Futrell et al., 2018) and Brown (Smith and Levy, 2013)—measurements were collected using the self-paced paradigm. For the other two datasets—Provo (Luke and Christianson, 2018) and Dundee (Kennedy et al., 2003)—eye-tracking movements were recorded. Each of these datasets provides the reading time each participant spent on a word. For the purpose of our analysis, we aggregate reading times per word (i.e., across participants). We thus analyse the average reading time participants spent on a word.

Evaluation. Studies of sentence comprehension are often concerned with a variable's **predictive power**: its ability to predict sentence comprehension data. Formally, let $\mathcal{D} = \{\mathbf{x}_n, y_n\}_{n=1}^{N}$ be a reading times dataset, where $y_n \in \mathbb{R}_+$ represents the average time participants spent reading a word w_n, and $\mathbf{x}_n \in \mathbb{R}^d$ is a vector containing a number of measurements taken on that word. Among these quantities is a word's length (in characters) and unigram frequency. Further, let f_ψ be a regressor that takes $\mathbf{x}_n$ as input and predicts y_n. We use ψ to denote this regressor's parameters. A variable's predictive power is then the change in $\mathcal{D}$'s log-likelihood (denoted as Δ_{llh}) under two regressors: one where $\mathbf{x}$ includes this variable (f_{ψ_1}), and one where it does not (f_{ψ_2}):

$$\Delta_{\mathrm{llh}} \overset{\mathrm{def}}{=} \mathrm{llh}(f_{\psi_1}, \mathcal{D}) - \mathrm{llh}(f_{\psi_2}, \mathcal{D}) \tag{26}$$

Here, we use this equation to measure surprisal's predictive power. Further, we estimate this change in data log-likelihood (denoted as Δ_{llh}) using 10-fold cross-validation, and we leverage these results to run paired permutation tests. Finally, we account for spillover effects by including features of word w_n as well as its three preceding words in $\mathbf{x}$.

A.2 Communicative Efficiency

We largely follow the setup of Pimentel et al. (2023a). We highlight the points where our setups differ below.

Data. We use the publicly available Wiki40b dataset (Guo et al., 2020), a large text corpus derived from Wikipedia articles. We use only the English portion of this dataset because the language models that we consider were trained solely on English data. We randomly sample a subset of the data, of size $\approx$ 20M tokens. We do not perform any pre-processing of the text, beyond that carried out by the native HuggingFace tokenisers for the respective language models. Unigram frequencies—which are used to estimate the unigram surprisals required by the Zipfian hypothesis—are computed on a separate subset of this same dataset.

Evaluation. We look at correlations between word lengths and the quantities put forward by various hypotheses about the influencing factors in a lexicon's word lengths. We expect to see that the hypotheses offering more accurate accounts of such factors have higher correlations with word lengths. In line with prior work, we specifically look at Spearman correlations.

B Detailed Surprisal Theory Results

		Surprisal		
		Improvement	Fixed	Buggy
gpt-small	Brown	0.06	4.32***	4.25***
	Natural Stories	0.01	3.18***	3.17***
	Provo	0.11	4.78***	4.67***
	Dundee	0.00	1.27***	1.27***
gpt-medium	Brown	0.08	3.91***	3.84***
	Natural Stories	0.00	2.98***	2.98***
	Provo	0.12	4.98***	4.85***
	Dundee	0.00	1.14***	1.14***
gpt-large	Brown	0.10	3.73***	3.63***
	Natural Stories	0.01	2.52***	2.51***
	Provo	0.11	4.33***	4.23***
	Dundee	0.00	1.08***	1.08***
gpt-xl	Brown	0.10	3.62***	3.53***
	Natural Stories	0.01	2.37***	2.36***
	Provo	0.09	4.28***	4.19***
	Dundee	0.01	0.97***	0.96***
pythia-70m	Brown	0.08	4.11***	4.02***
	Natural Stories	0.05	3.61***	3.56***
	Provo	0.06	4.62***	4.56***
	Dundee	0.00	1.48***	1.48***
pythia-160m	Brown	0.08	4.05***	3.97***
	Natural Stories	0.04	3.09***	3.05***
	Provo	0.13	4.86***	4.73***
	Dundee	0.00	1.32***	1.32***
pythia-410m	Brown	0.08	3.53***	3.45***
	Natural Stories	0.03	2.80***	2.77***
	Provo	0.13	4.69***	4.56***
pythia-1.4b	Brown	0.09	3.21***	3.12***
	Natural Stories	0.02	2.36***	2.35***
	Provo	0.08	4.27***	4.19***
pythia-2.8b	Brown	0.09	3.31***	3.22***
	Natural Stories	0.02	2.23***	2.22***
	Provo	0.08	4.44***	4.36***
pythia-6.9b	Brown	0.09	2.95***	2.86***
	Natural Stories	0.02	1.87***	1.85***
	Provo	0.07	3.86***	3.79***

Table 1: Δ_{llh} between regressors with and without surprisal as a predictor.

C Proofs of Lemmas and Theorems

C.1 Proof of End-of-Word Tokeniser's Theorem 1

Lemma 3. *Let* $\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}$ *be a* `eow`*-marking tokeniser. We can show the following equivalence:*

$$\begin{aligned} \mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ \mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*) \\ \mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ w \circ \mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w} \circ \mathcal{S}^*) \end{aligned} \tag{27}$$

Proof. This lemma assumes a tokeniser which runs on top of pre-tokenised text. Therefore, we can rely on Defn. 2, whose equation we rewrite here for convenience:

$$\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) = \underset{\mathcal{W}\to\mathcal{S}^*}{\mathbb{S}}(w_1) \circ \underset{\mathcal{W}\to\mathcal{S}^*}{\mathbb{S}}(w_2) \circ \cdots \circ \underset{\mathcal{W}\to\mathcal{S}^*}{\mathbb{S}}(w_{|\mathbf{w}|}) \tag{28}$$

Further, as this tokeniser is eow-marking, we have that: $\underset{\mathcal{W}\to\mathcal{S}^*}{\mathbb{S}} : \mathcal{W} \to \mathcal{S}^*_{\texttt{mid}} \circ \mathcal{S}_{\texttt{eow}}$. We now prove the equivalences above. First, we show that $\mathbf{w}' \in (\mathbf{w}_{<t} \circ \mathcal{W}^*) \implies \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \in (\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*)$; this shows that all strings $\mathbf{w}' \in (\mathbf{w}_{<t} \circ \mathcal{W}^*)$ are considered by the set $(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*)$.

$$\begin{aligned}
\mathbf{w}_{<t} \circ \mathcal{W}^* &= \left\{\mathbf{w}_{<t} \circ \mathbf{w}' \mid \mathbf{w}' \in \mathcal{W}^*\right\} && \text{definition of } \circ && (29a)\\
&\triangleq \left\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}_{<t} \circ \mathbf{w}') \mid \mathbf{w}' \in \mathcal{W}^*\right\} && \text{definition of } \overset{\mathcal{W}^*\to\mathcal{S}^*}{\Longrightarrow} && (29b)\\
&= \left\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}_{<t}) \circ \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \mid \mathbf{w}' \in \mathcal{W}^*\right\} && \text{decomposition of } \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}} && (29c)\\
&= \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}_{<t}) \circ \left\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \mid \mathbf{w}' \in \mathcal{W}^*\right\} && \text{definition of } \circ \text{ over sets} && (29d)\\
&= \mathbf{s}^{\mathbf{w}_{<t}} \circ \left\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \mid \mathbf{w}' \in \mathcal{W}^*\right\} && \text{definition of } \mathbf{s}^{\mathbf{w}_{<t}} && (29e)\\
&\subseteq \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^* && && (29f)
\end{aligned}$$

We now define the set $\Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*} \stackrel{\text{def}}{=} \{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \mid \mathbf{w}' \in (\mathbf{w}_{<t} \circ \mathcal{W}^*)\}$, and note that $\mathbf{w}_{<t} \circ \mathcal{W}^* \triangleq \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}$. We can thus split the probability we are computing into two parts:

$$\mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*) = \mathbb{P}_{\mathcal{S}}(\Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) + \mathbb{P}_{\mathcal{S}}((\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*) \setminus \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) \tag{30}$$

If we prove that $\mathbb{P}_{\mathcal{S}}((\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*) \setminus \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) = 0$, then we have that $\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ \mathcal{W}^*) = \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*)$. To prove that, we first note that:

$$p(\mathbf{s}) = \sum_{\mathbf{w}\in\mathcal{W}^*} p(\mathbf{w})\, \mathbb{1}\left\{\mathbf{s} = \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w})\right\} \quad\implies\quad p(\mathbf{s}) = \begin{cases} p(\mathbf{w}) & \texttt{if } \mathbf{s} = \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) \\ 0 & \texttt{if } \mathbf{s} \in \mathcal{S}_{\texttt{x}} \end{cases} \tag{31}$$

We now show that $\mathbf{s}' \in (\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*) \implies \underset{\mathcal{S}^*\to\mathcal{W}^*}{\mathbb{S}}(\mathbf{s}') \in (\mathbf{w}_{<t} \circ \mathcal{W}^*)$. This result implies that no other strings $\mathbf{w}' \notin (\mathbf{w}_{<t} \circ \mathcal{W}^*)$ are considered by the set $(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*)$, which itself implies that $((\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*) \setminus \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) \cap (\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) \mid \mathbf{w} \in \mathcal{W}^*\}) = \emptyset$.

$$\begin{aligned}
\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^* &= \left\{\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}' \mid \mathbf{s}' \in \mathcal{S}^*\right\} && \text{definition of } \circ && (32a)\\
&\overset{\mathcal{S}^*\to\mathcal{W}^*}{\Longrightarrow} \left\{\underset{\mathcal{S}^*\to\mathcal{W}^*}{\mathbb{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}') \mid \mathbf{s}' \in \mathcal{S}^*\right\} && \text{definition of } \overset{\mathcal{S}^*\to\mathcal{W}^*}{\Longrightarrow} && (32b)\\
&= \left\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{s}^{\mathbf{w}_{<t}}) \circ \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{s}') \mid \mathbf{s}' \in \mathcal{S}^*\right\} && \mathbf{s}^{\mathbf{w}_{<t}} \text{ ends in } \mathcal{S}_{\texttt{eow}}, \text{ decomposition of } \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}} && (32c)\\
&= \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{s}^{\mathbf{w}_{<t}}) \circ \left\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{s}') \mid \mathbf{s}' \in \mathcal{S}^*\right\} && \text{definition of } \circ \text{ over sets} && (32d)\\
&= \mathbf{w}_{<t} \circ \left\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{s}') \mid \mathbf{s}' \in \mathcal{S}^*\right\} && \text{definition of } \mathbf{s}^{\mathbf{w}_{<t}} && (32e)\\
&= \mathbf{w}_{<t} \circ \mathcal{W}^* && \text{co-domain of } \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}} && (32f)
\end{aligned}$$

Since $\mathbf{s} \in ((\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*) \setminus \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) \implies \mathbf{s} \notin (\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) \mid \mathbf{w} \in \mathcal{W}^*\})$, we have that $\mathbb{P}_{\mathcal{S}}((\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*) \setminus \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) = 0$, which completes the proof. □

Theorem 1. *Let $\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}$ be a* eow*-marking tokeniser. Further, let $\mathbf{s}^{\mathbf{w}} \stackrel{\text{def}}{=} \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w})$. We can show the following equivalence:*

$$\begin{aligned}
\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ \mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*) && (16)\\
\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ w \circ \mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w} \circ \mathcal{S}^*)
\end{aligned}$$

Further, we can compute a word's probability as:

$$p(w \mid \mathbf{w}_{<t}) = \underbrace{\prod_{t'=1}^{|\mathbf{s}^{w}|} p\left(\mathbf{s}^{w}_{t'} \mid \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w}_{<t'}\right)}_{p(\mathbf{s}^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}})} \tag{17}$$

Proof. The first part of this theorem basically re-writes Lemma 3. We now derive the probabilities in eq. (17) as:

$$\begin{aligned}
p(w \mid \mathbf{w}_{<t}) &= \frac{\mathbb{P}(\mathbf{w}_{<t} \circ w \circ \mathcal{W}^*)}{\mathbb{P}(\mathbf{w}_{<t} \circ \mathcal{W}^*)} && \text{(33a)}\\
&= \frac{\mathbb{P}\left(\mathbf{s}^{\mathbf{w}_{<t} \circ w} \circ \mathcal{S}^*\right)}{\mathbb{P}\left(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*\right)} && \text{(33b)}\\
&= \frac{\prod_{t'=1}^{|\mathbf{s}^{\mathbf{w}_{<t} \circ w}|} p\left(\mathbf{s}^{\mathbf{w}_{<t} \circ w}_{t'} \mid \mathbf{s}^{\mathbf{w}_{<t} \circ w}_{<t'}\right)}{\prod_{t'=1}^{|\mathbf{s}^{\mathbf{w}_{<t}}|} p\left(\mathbf{s}^{\mathbf{w}_{<t}}_{t'} \mid \mathbf{s}^{\mathbf{w}_{<t}}_{<t'}\right)} && \text{(33c)}\\
&= \prod_{t'=|\mathbf{s}^{\mathbf{w}_{<t}}|+1}^{|\mathbf{s}^{\mathbf{w}_{<t} \circ w}|} p\left(\mathbf{s}^{\mathbf{w}_{<t} \circ w}_{t'} \mid \mathbf{s}^{\mathbf{w}_{<t} \circ w}_{<t'}\right) && \text{(33d)}\\
&= \prod_{t'=1}^{|\mathbf{s}^{w}|} p\left(\mathbf{s}^{w}_{t'} \mid \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w}_{<t'}\right) && \text{(33e)}
\end{aligned}$$

This completes the proof. $\square$

C.2 Proof of Beginning-of-Word Tokeniser's Theorem 2

Lemma 4. *Let* $\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}$ *be a* bow*-marking tokeniser. We can show the following equivalence:*

$$\begin{aligned}
\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ \mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\texttt{bow}} \circ \mathcal{S}^*}) \\
\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ w \circ \mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w} \circ \overline{\mathcal{S}_{\texttt{bow}} \circ \mathcal{S}^*})
\end{aligned} \tag{34}$$

Proof. This lemma assumes a tokeniser which runs on top of pre-tokenised text. Therefore, we can rely on Defn. 2, whose equation we rewrite here for convenience:

$$\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) = \underset{\mathcal{W} \to \mathcal{S}^*}{\mathbb{S}}(w_1) \circ \underset{\mathcal{W} \to \mathcal{S}^*}{\mathbb{S}}(w_2) \circ \cdots \circ \underset{\mathcal{W} \to \mathcal{S}^*}{\mathbb{S}}(w_{|\mathbf{w}|}) \tag{35}$$

Further, as this tokeniser is bow-marking, we have that: $\underset{\mathcal{W} \to \mathcal{S}^*}{\mathbb{S}} : \mathcal{W} \to \mathcal{S}_{\texttt{bow}} \circ \mathcal{S}^*_{\texttt{mid}}$. We now prove the equivalences above. First, we show that $\mathbf{w}' \in (\mathbf{w}_{<t} \circ \mathcal{W}^*) \implies \underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \in (\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\texttt{bow}} \circ \mathcal{S}^*})$; this shows that all strings $\mathbf{w}' \in (\mathbf{w}_{<t} \circ \mathcal{W}^*)$ are considered by the set $(\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\texttt{bow}} \circ \mathcal{S}^*})$.

$$\begin{aligned}
\mathbf{w}_{<t} \circ \mathcal{W}^* &= \left\{\mathbf{w}_{<t} \circ \mathbf{w}' \mid \mathbf{w}' \in \mathcal{W}^*\right\} && \text{definition of } \circ \quad \text{(36a)}\\
&\triangleq \left\{\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}_{<t} \circ \mathbf{w}') \mid \mathbf{w}' \in \mathcal{W}^*\right\} && \text{definition of } \overset{\mathcal{W}^* \to \mathcal{S}^*}{\Longrightarrow} \quad \text{(36b)}\\
&= \left\{\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}_{<t}) \circ \underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \mid \mathbf{w}' \in \mathcal{W}^*\right\} && \text{decomposition of } \underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}} \quad \text{(36c)}\\
&= \underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}_{<t}) \circ \left\{\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \mid \mathbf{w}' \in \mathcal{W}^*\right\} && \text{definition of } \circ \text{ over sets} \quad \text{(36d)}\\
&= \mathbf{s}^{\mathbf{w}_{<t}} \circ \left\{\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \mid \mathbf{w}' \in \mathcal{W}^*\right\} && \text{definition of } \mathbf{s}^{\mathbf{w}_{<t}} \quad \text{(36e)}\\
&= \mathbf{s}^{\mathbf{w}_{<t}} \circ \left(\{\texttt{eos}\} \cup \left(\mathcal{S}_{\texttt{bow}} \circ \mathcal{S}^*_{\texttt{mid}} \circ \left\{\underset{\mathcal{W}^* \to \mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \mid \mathbf{w}' \in \mathcal{W}^*\right\}\right)\right) && \text{(36f)}\\
&\subseteq \mathbf{s}^{\mathbf{w}_{<t}} \circ (\{\texttt{eos}\} \cup (\mathcal{S}_{\texttt{bow}} \circ \mathcal{S}^*_{\texttt{mid}} \circ \mathcal{S}^*)) && \text{(36g)}\\
&\subseteq \mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\texttt{bow}} \circ \mathcal{S}^*} && \text{(36h)}
\end{aligned}$$

We now define the set $\Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*} \stackrel{\text{def}}{=} \{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}') \mid \mathbf{w}' \in (\mathbf{w}_{<t}\circ\mathcal{W}^*)\}$, and note that $\mathbf{w}_{<t}\circ\mathcal{W}^* \triangleq \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}$. We can thus split the probability we are computing into two parts:

$$\mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}) = \mathbb{P}_{\mathcal{S}}(\Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) + \mathbb{P}_{\mathcal{S}}((\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}) \setminus \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) \tag{37}$$

If we prove that $\mathbb{P}_{\mathcal{S}}((\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}) \setminus \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) = 0$, then we have that $\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t}\circ\mathcal{W}^*) = \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*})$. To prove that, we first note that:

$$p(\mathbf{s}) = \begin{cases} p(\mathbf{w}) & \texttt{if } \mathbf{s} = \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{w}) \\ 0 & \texttt{if } \mathbf{s} \in \mathcal{S}_{\text{x}} \end{cases} \tag{38}$$

We now show that $\mathbf{s}' \in (\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}) \implies \underset{\mathcal{S}^*\to\mathcal{W}^*}{\mathbb{S}}(\mathbf{s}') \in (\mathbf{w}_{<t}\circ\mathcal{W}^*)$. This result implies that no other strings $\mathbf{w}' \notin (\mathbf{w}_{<t}\circ\mathcal{W}^*)$ are considered by the set $(\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*})$, which itself implies that $(\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}) \setminus \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*} \subseteq \mathcal{S}_{\text{x}}$.

$$\begin{aligned}
\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*} &= \{\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}' \mid \mathbf{s}' \in \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}\} && \text{definition of } \circ \quad \text{(39a)} \\
&\overset{\mathcal{S}^*\to\mathcal{W}^*}{\Longrightarrow} \left\{\underset{\mathcal{S}^*\to\mathcal{W}^*}{\mathbb{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}') \mid \mathbf{s}' \in \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}\right\} && \text{definition of } \overset{\mathcal{S}^*\to\mathcal{W}^*}{\Longrightarrow} \quad \text{(39b)} \\
&= \left\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{s}^{\mathbf{w}_{<t}}) \circ \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{s}') \mid \mathbf{s}' \in \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}\right\} && \quad \text{(39c)} \\
& && \mathbf{s}' \text{ is either empty, or starts in } \mathcal{S}_{\text{bow}}, \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}} \text{ thus decomposes} \quad \text{(39d)} \\
&= \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{s}^{\mathbf{w}_{<t}}) \circ \left\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{s}') \mid \mathbf{s}' \in \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}\right\} && \text{definition of } \circ \text{ over sets} \quad \text{(39e)} \\
&= \mathbf{w}_{<t} \circ \left\{\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}(\mathbf{s}') \mid \mathbf{s}' \in \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}\right\} && \text{definition of } \mathbf{s}^{\mathbf{w}_{<t}} \quad \text{(39f)} \\
&= \mathbf{w}_{<t} \circ \mathcal{W}^* && \text{co-domain of } \underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}} \quad \text{(39g)}
\end{aligned}$$

Since $\mathbf{s} \in ((\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}) \setminus \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) \implies \mathbf{s} \in \mathcal{S}_{\text{x}}$, we have that $\mathbb{P}_{\mathcal{S}}((\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}) \setminus \Psi_{\mathcal{S}}^{\mathbf{w}_{<t}\circ\mathcal{W}^*}) = 0$, which completes the proof. □

Theorem 2. *Let* $\underset{\mathcal{W}^*\to\mathcal{S}^*}{\mathbb{S}}$ *be a* bow*-marking tokeniser. Further, let* $\overline{\cdot}$ *represent the union of a set with* eos*, e.g.,* $\overline{\mathcal{S}_{\text{bow}}} = \mathcal{S}_{\text{bow}} \cup \{\texttt{eos}\}$. *We can show the following equivalence:*

$$\begin{gathered}
\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t}\circ\mathcal{W}^*) = \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*}) \\
\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t}\circ w\circ\mathcal{W}^*) = \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w} \circ \overline{\mathcal{S}_{\text{bow}} \circ \mathcal{S}^*})
\end{gathered} \tag{19}$$

Further, we can compute a word's probability as:

$$p(w \mid \mathbf{w}_{<t}) = \underbrace{\prod_{t'=1}^{|\mathbf{s}^{w}|} p\left(\mathbf{s}^{w}_{t'} \mid \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w}_{<t'}\right)}_{p(\mathbf{s}^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}})} \underbrace{\frac{\sum_{\{s\in\overline{\mathcal{S}}_{\text{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w}\right)}{\sum_{\{s\in\overline{\mathcal{S}}_{\text{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t}}\right)}}_{\textit{Bug Fix } 1} \tag{20}$$

Proof. The first part of this theorem basically re-writes Lemma 4. We now derive the probabilities in eq. (20) as:

$$
\begin{aligned}
p(w \mid \mathbf{w}_{<t}) &= \frac{\mathbb{P}(\mathbf{w}_{<t} \circ w \circ \mathcal{W}^*)}{\mathbb{P}(\mathbf{w}_{<t} \circ \mathcal{W}^*)} && \text{(40a)}\\
&= \frac{\sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} \mathbb{P}\left(\mathbf{s}^{\mathbf{w}_{<t} \circ w} \circ s \circ \mathcal{S}^*\right)}{\sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} \mathbb{P}\left(\mathbf{s}^{\mathbf{w}_{<t}} \circ s \circ \mathcal{S}^*\right)} && \text{(40b)}\\
&= \frac{\sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t} \circ w}\right) \prod_{t'=1}^{|\mathbf{s}^{\mathbf{w}_{<t} \circ w}|} p\left(\mathbf{s}_{t'}^{\mathbf{w}_{<t}} \mid \mathbf{s}_{<t'}^{\mathbf{w}_{<t}}\right)}{\sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t}}\right) \prod_{t'=1}^{|\mathbf{s}^{\mathbf{w}_{<t}}|} p\left(\mathbf{s}_{t'}^{\mathbf{w}_{<t}} \mid \mathbf{s}_{<t'}^{\mathbf{w}_{<t}}\right)} && \text{(40c)}\\
&= \frac{\prod_{t'=1}^{|\mathbf{s}^{\mathbf{w}_{<t} \circ w}|} p\left(\mathbf{s}_{t'}^{\mathbf{w}_{<t} \circ w} \mid \mathbf{s}_{<t'}^{\mathbf{w}_{<t} \circ w}\right) \sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t} \circ w}\right)}{\prod_{t'=1}^{|\mathbf{s}^{\mathbf{w}_{<t}}|} p\left(\mathbf{s}_{t'}^{\mathbf{w}_{<t}} \mid \mathbf{s}_{<t'}^{\mathbf{w}_{<t}}\right) \sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t}}\right)} && \text{(40d)}\\
&= \frac{\prod_{t'=|\mathbf{s}^{\mathbf{w}_{<t}}+1|}^{|\mathbf{s}^{\mathbf{w}_{<t} \circ w}|} p\left(\mathbf{s}_{t'}^{\mathbf{w}_{<t} \circ w} \mid \mathbf{s}_{<t'}^{\mathbf{w}_{<t} \circ w}\right) \sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t} \circ w}\right)}{\sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t}}\right)} && \text{(40e)}\\
&= \prod_{t'=1}^{|\mathbf{s}^{w}|} p\left(\mathbf{s}_{t'}^{w} \mid \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}_{<t'}^{w}\right) \frac{\sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t} \circ w}\right)}{\sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} p\left(s \mid \mathbf{s}^{\mathbf{w}_{<t}}\right)} && \text{(40f)}
\end{aligned}
$$

This completes the proof. □

C.3 Theorem of Non-eow-marking Final-word Tokeniser's

Theorem 3. *Let* ${}_{\mathcal{W}^* \to \mathcal{S}^*}\mathbb{S}$ *be a* eow*-marking tokeniser with unmarked final word. We can show the following equivalence:*

$$
\begin{aligned}
\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ \mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathcal{S}^*) && \text{(41)}\\
\mathbb{P}_{\mathcal{W}}(\mathbf{w}_{<t} \circ w \circ \mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}((\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}^{w} \circ \mathcal{S}^*) \cup \{\mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}_{\texttt{mid}}^{w}\})
\end{aligned}
$$

Further, we can compute a word's probability as:

$$
p(w \mid \mathbf{w}_{<t}) = \Bigg(p\left(\mathbf{s}_{\texttt{mid}}^{w} | \, \mathbf{s}^{\mathbf{w}_{<t}}\right) \underbrace{\sum_{s \in \mathcal{S}_{!?}} p(s | \, \mathbf{s}^{\mathbf{w}_{<t}} \circ \mathbf{s}_{\texttt{mid}}^{w})\Bigg) + p(\mathbf{s}^{w} | \, \mathbf{s}^{\mathbf{w}_{<t}})}_{\textit{Bug Fix } ②} \tag{42}
$$

C.4 Theorem of Non-bow-marking First-word Tokeniser's

Theorem 4. *Let* ${}_{\mathcal{W}^* \to \mathcal{S}^*}\mathbb{S}$ *be a* bow*-marking tokeniser with unmarked first words. We can show the following equivalence:*

$$
\begin{aligned}
\mathbb{P}_{\mathcal{W}}(\mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\overline{\mathcal{S}_{\texttt{mid}} \circ \mathcal{S}^*}) && \text{(43)}\\
\mathbb{P}_{\mathcal{W}}(w \circ \mathcal{W}^*) &= \mathbb{P}_{\mathcal{S}}(\mathbf{s}_{\texttt{mid}}^{w} \circ \overline{\mathcal{S}_{\texttt{bow}} \circ \mathcal{S}^*})
\end{aligned}
$$

Further, we can compute a word's probability as:

$$
p(w \mid \mathbf{w}_{<t}) = p\left(\mathbf{s}_{\texttt{mid}}^{w} \mid \text{``''}\right) \underbrace{\frac{\sum_{\{s \in \overline{\mathcal{S}}_{\texttt{bow}}\}} p\left(s \mid \mathbf{s}^{w}\right)}{\sum_{\{s \in \overline{\mathcal{S}}_{\texttt{mid}}\}} p\left(s \mid \text{``''}\right)}}_{\textit{Bug Fix } ③} \tag{44}
$$