Consistent Bidirectional Language Modelling: Expressive Power and Representational Conciseness

Georgi Shopov[1] **Stefan Gerdjikov**[1,2]
[1]IICT, Bulgarian Academy of Sciences [2]FMI, Sofia University
gshopov@lml.bas.bg stefangerdzhikov@fmi.uni-sofia.bg

Abstract

The inability to utilise future contexts and the pre-determined left-to-right generation order are major limitations of unidirectional language models. Bidirectionality has been introduced to address those deficiencies. However, a crucial shortcoming of bidirectional language models is the potential inconsistency of their conditional distributions. This fundamental flaw greatly diminishes their applicability and hinders their capability of tractable sampling and likelihood computation. In this work, we introduce a class of bidirectional language models, called *latent language models*, that are consistent by definition and can be efficiently used both for generation and scoring of sequences. We define latent language models based on the well-understood formalism of bisequential decompositions from automata theory. This formal correspondence allows us to precisely charaterise the abilities and limitations of a subclass of latent language models, called *rational language models*. As a result, we obtain that latent language models are exponentially more concise and significantly more expressive than unidirectional language models.

1 Introduction

Recently, language models (Bengio et al., 2003; Mikolov et al., 2010; Brown et al., 2020) have established themselves as the primary approach for solving natural language processing tasks (Devlin et al., 2019; Raffel et al., 2020; Brown et al., 2020). They have also exhibited noteworthy capabilities in computer programming (Chen et al., 2021; Fried et al., 2023) and commonsense and mathematical reasoning (Wei et al., 2022; Zhou et al., 2023).

Language models are traditionally differentiated based on the contextual conditioning that they use for token prediction. *Unidirectional* (or autoregressive) language models, such as those based on recurrent neural networks (RNNs) (Mikolov et al., 2010) as well as the Transformer-based GPT models (Radford et al., 2019; Brown et al., 2020), condition the prediction of a given token only on its left context. On the other hand, *bidirectional* (or masked) language models, such as those based on bidirectional RNNs (Arisoy et al., 2015; Mousa and Schuller, 2017) as well as the Transformer-based BERT (Devlin et al., 2019; Liu et al., 2019) and T5 (Raffel et al., 2020; Xue et al., 2021), predict tokens based on both their left and right contexts.

Naturally, because of the access to richer contextual information, bidirectional language models have proven to produce stronger learned representations (Devlin et al., 2019; Raffel et al., 2020). In this regard, Brown et al. (2020) speculate that the inability of GPT-3 to benefit from future contexts could explain its inferior performance on certain tasks where bidirectionality is important. Additionally, several studies have brought attention to the importance of the order in which sequences are processed (Vinyals et al., 2015; Ford et al., 2018). To this end, both empirical (Emelianenko et al., 2019; Li et al., 2021) and theoretical (Lin et al., 2021) results have implied that, as opposed to bidirectional language models, the pre-determined left-to-right order used by unidirectional language models is often suboptimal for tasks that require exploration, planning or strategic lookahead (Yao et al., 2023).

Despite of the aforementioned advantages of bidirectional language models, it is currently unclear how to tractably ensure the *consistency* of their conditional distributions.[1] In other words, it is non-trivial to guarantee the existence of a joint distribution whose conditionals coincide with those of a given bidirectional language model (Goyal et al., 2022; Torroba Hennigen and Kim, 2023; Young et al., 2024). Furthermore, even if such a joint distribution exists, it is often computationally ex-

[1]On the other hand, for unidirectional language models, Du et al. (2023) have given sufficient conditions for consistency that can be easily enforced.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 5724–5768
November 12-16, 2024

pensive to access it explicitly. Those fundamental flaws of bidirectional language models greatly hinder their applicability for sampling (Ghazvininejad et al., 2019) and likelihood computation (Salazar et al., 2020), and often lead to self-contradictory behavior during inference (Young et al., 2024).

In this work, we introduce a class of bidirectional language models that are consistent by definition and can be efficiently used both for generation and scoring of sequences. To achieve this, we consider language modelling from the point of view of automata theory (Eilenberg, 1974; Sakarovitch, 2009; Mihov and Schulz, 2019). Svete and Cotterell (2023) have already explored the relations between unidirectional language models and *sequential transducers*. We extend their work by considering the bidirectional formalism of *bisequential decompositions* (Elgot and Mezei, 1965). By examining how bisequential decompositions represent probability distributions, we derive a class of bidirectional language models that we call *latent language models*. This formal correspondence allows us to precisely charaterise the abilities and limitations of a subclass of latent language models, called *rational language models*. As a result, we obtain that latent language models are exponentially more concise and significantly more expressive than unidirectional language models. We argue that such knowledge about the abilities and limitations of language models is essential whenever we require formal guarantees of the correctness and consistency of their outputs.

2 Factorisations of Language Models

Given a finite set Σ, we shall use Σ^* to denote the set of finite sequences of elements of Σ and ϵ to denote the empty sequence. In this case, Σ is called an *alphabet*, the elements of Σ are called *letters* and the elements of Σ^* are called *words*. Additionally, given a word α, we shall write $|\alpha|$ for the length of α, α_i for the i-th letter of α, $\alpha_{\leq i}$ (or $\alpha_{<i+1}$) for the prefix $\alpha_1\alpha_2\cdots\alpha_i$ and $\alpha_{\geq i}$ (or $\alpha_{>i-1}$) for the suffix $\alpha_i\alpha_{i+1}\cdots\alpha_{|\alpha|}$. Lastly, we shall naturally extend concatenation of words to sets of words.

Definition 2.1. Let Σ be an alphabet. A *language model over* Σ is a discrete probability distribution over Σ^*; i.e., a function $\mathbb{P}\colon \Sigma^* \to [0,1]$ such that[2]

$$\sum_{\alpha\in\Sigma^*} \mathbb{P}(\alpha) = 1. \tag{1}$$

[2]As usual, we assume that $\mathbb{P}$ is extended to the power set $\mathfrak{P}(\Sigma^*)$ of Σ^* by countable additivity.

The support of a real-valued function f is the set

$$\operatorname{Supp}(f) \coloneqq \big\{\alpha \in \operatorname{Dom}(f) \mid f(\alpha) \neq 0\big\}.$$

We call a probability distribution $\mathbb{P}$ over X *positive* if $\operatorname{Supp}(\mathbb{P}) = X$. Naturally, a family of positive probability distributions is also called *positive*.

In practice, it is not difficult to model a function from Σ^* to $[0,1]$ (e.g., with a logistic curve). However, satisfying the normalisation constraint (1) is non-trivial due to the infinitarity of Σ^*. Consequently, in this section, we consider how language models can be factorised into families of *finite* conditional probability distributions that can then be modelled efficiently (e.g., with a $\mathrm{softmax}$-based linear prediction head (Bengio et al., 2003)).

2.1 Prefix Factorisations

Unidirectional language models are typically defined in terms of finite local probability distributions conditioned solely on the left context.

Definition 2.2. Let Σ be an alphabet. A *prefix factorisation over* Σ is a family $(\phi_\alpha)_{\alpha\in\Sigma^*}$ of discrete probability distributions over $\Sigma_\$ \coloneqq \Sigma \cup \{\$\}$.[3]

The value $\phi_\alpha(a)$ is intended to be interpreted as the probability of the current letter being a given that α is the word formed by the previous letters.[4]

Definition 2.3. Let $\Phi \coloneqq (\phi_\alpha)_{\alpha\in\Sigma^*}$ be a prefix factorisation over Σ. Φ is called *consistent* if there exists a language model $\mathbb{P}$ over Σ that is *compatible with* Φ; i.e., for $\alpha \in \Sigma^*$ and $a \in \Sigma$,[5]

$$\mathbb{P}(\alpha\Sigma^*) \neq 0 \implies \begin{cases} \mathbb{P}(\alpha a\Sigma^* \mid \alpha\Sigma^*) = \phi_\alpha(a) \\ \mathbb{P}(\alpha \mid \alpha\Sigma^*) = \phi_\alpha(\$) \end{cases}.$$

Now, it is apparent that we can use the distributions of a consistent prefix factorisation to define, via the chain rule of probability, its unique compatible language model (see Appendix B.1 for a proof). Furthermore, the chain rule provides an efficient method for sampling and scoring of words.

Definition 2.4. Let $\Phi \coloneqq (\phi_\alpha)_{\alpha\in\Sigma^*}$ be a prefix factorisation over Σ. *The prefix model generated by* Φ is the function $M\colon \Sigma^* \to [0,1]$ defined as

$$M(\alpha) \coloneqq \Big(\prod_{i=1}^{|\alpha|} \phi_{\alpha_{<i}}(\alpha_i)\Big)\phi_\alpha(\$).$$

[3]We shall use $\$$ to denote a special *end-marker* letter that is considered not to be a member of any declared alphabet.

[4]Similarly, the value $\phi_\alpha(\$)$ is intended to be interpreted as the probability of the word ending after α.

[5]We note that $\alpha\Sigma^*$ is the set of all words over Σ that begin with α. Thus, $\mathbb{P}(\alpha a\Sigma^* \mid \alpha\Sigma^*)$ is the probability of a word to begin with αa given that it begins with α, and $\mathbb{P}(\alpha \mid \alpha\Sigma^*)$ is the probability of a word to be α given that it begins with α.

Theorem 2.1. *A prefix factorisation Φ over Σ is consistent if and only if the prefix model M generated by Φ is a language model over Σ. In this case, M is the only language model compatible with Φ.*

In the literature, the term 'unidirectional language model' is commonly used to mean 'a prefix model', which raises some formal issues since the class of language models and the class of prefix models do not coincide. It is easily observed that every language model is compatible with a consistent prefix factorisation and thus is a prefix model. However, the converse is true only under certain conditions as described by Du et al. (2023). Fortunately, those conditions are satisfied in most practical settings (e.g., when the prefix model is represented by a Transformer (Vaswani et al., 2017) or a bounded RNN (Elman, 1990; Hochreiter and Schmidhuber, 1997; Cho et al., 2014) with a $\mathrm{softmax}$-based linear prediction head).

2.2 Confix Factorisations

Bidirectional language models, such as BERT, T5 and their RNN-based alternatives, represent a family of finite local probability distributions conditioned both on the left and the right contexts.

Definition 2.5. Let Σ be an alphabet. A *confix*[6] *factorisation over* Σ is a family $(\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$ of discrete probability distributions over Σ.[7,8]

Intuitively, the value $\phi_{\alpha,\beta}(a)$ could be interpreted as the probability of the current letter being a given that α and β are the words formed by the previous and the following letters, respectively.

Definition 2.6. Let $\Phi \coloneqq (\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$ be a confix factorisation over Σ. Φ is called *consistent* if there exists a language model $\mathbb{P}$ over Σ that is *compatible with* Φ; i.e., for $\alpha, \beta \in \Sigma^*$ and $a \in \Sigma$,

$$\mathbb{P}(\alpha\Sigma\beta) \neq 0 \implies \mathbb{P}(\alpha a\beta \mid \alpha\Sigma\beta) = \phi_{\alpha,\beta}(a).$$

Next, we consider whether the distributions of a consistent confix factorisation Φ can be used to define a language model that is compatible with Φ. A classical result by Besag (1974) shows that the distributions of a consistent positive confix factorisation can be used to express the quotients of the probabilities of words of equal length (see Appendix B.2 for a proof).

[6] A confix (or a *circumfix*) is a pair of a prefix and a suffix.

[7] To be more precise, BERT and T5 actually represent joint distributions over multiple *masked* positions in a word by conditioning on the remaining letters. Here, for the sake of simplicity, we assume that there is a single masked position.

[8] See also Remark B.3 in Appendix B.2.

Proposition 2.1. *Let $\mathbb{P}$ be a language model over Σ that is compatible with a positive confix factorisation $(\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$. Then, for $\alpha \in \Sigma^*$ and $\beta \in \Sigma^{|\alpha|}$,*

$$\mathbb{P}(\alpha) = \mathbb{P}(\beta) \prod_{i=1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)}.$$

Nevertheless, the distributions of a consistent confix factorisation contain no information about the distribution of the word lengths. Hence, in order to define a language model $\mathbb{P}$ over Σ that is compatible with a given confix factorisation, it is necessary to additionally specify the probabilities of the events Σ^n for $n \in \mathbb{N}$.

Definition 2.7. A *complete confix factorisation over* Σ is a tuple $(\Phi, \mathbb{P}_L)$, where Φ is a positive confix factorisation over Σ and $\mathbb{P}_L$ is a probability distribution over $\mathbb{N}$. $(\Phi, \mathbb{P}_L)$ is *consistent* if there exists a language model $\mathbb{P}$ over Σ that is *compatible with* $(\Phi, \mathbb{P}_L)$; i.e., $\mathbb{P}$ is compatible with Φ and

$$(\forall n \in \mathbb{N})\big(\mathbb{P}(\Sigma^n) = \mathbb{P}_L(n)\big).$$

Now, Proposition 2.1 implies that we can use the distributions of a consistent complete confix factorisation to express its unique compatible language model (see Appendix B.2 for a proof).

Definition 2.8. Let $(\Phi, \mathbb{P}_L)$ be a complete confix factorisation over Σ such that $\Phi \coloneqq (\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$. *The confix model generated by* $(\Phi, \mathbb{P}_L)$ is the function $M\colon \Sigma^* \to [0, \infty)$ defined as

$$M(\alpha) \coloneqq \frac{\mathbb{P}_L\big(|\alpha|\big)}{\sum_{\beta\in\Sigma^{|\alpha|}} \prod_{i=1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}}. \quad (2)$$

Theorem 2.2. *Let $(\Phi, \mathbb{P}_L)$ be a consistent complete confix factorisation over Σ. Then, the confix model generated by $(\Phi, \mathbb{P}_L)$ is the only language model over Σ that is compatible with $(\Phi, \mathbb{P}_L)$.*

Unfortunately, the intractable sum in the denominator makes the expression in (2) inapplicable for generation and scoring. More importantly, complete confix factorisations lack an easy to enforce condition that implies their consistency. Thus, it is unclear how one could guarantee the existence of an underlying compatible language model.

3 Sequential Language Models

In this section, we consider language modelling with sequential transducers. We argue that they provide a meaningful abstraction of unidirectional

language models. Thus, attaining deeper knowledge of the properties of sequential transducers could lead to better understanding of the abilities and limitations of unidirectional language models.

3.1 Transducers

Definition 3.1. A *monoid* is a tuple $(M, \circ, e)$, where M is a set, called *the carrier*, $\circ$ is an associative binary operation on M and $e \in M$ is a *neutral element for* $\circ$ (that is, $a \circ e = e \circ a = a$).

In this work, we shall consider *the free monoid* $\Sigma^* := (\Sigma^*, \cdot, \epsilon)$, where $\cdot$ denotes concatenation, and *the probability monoid* $\mathcal{R}_{[0,1]} := \big([0,1], \cdot, 1\big)$, where $\cdot$ denotes multiplication.

Definition 3.2. A $(\Sigma, \mathcal{M})$-*transducer* is a tuple $(\Sigma, \mathcal{M}, Q, \mathbb{I}, \mathbb{F}, \Delta)$, where Σ is an alphabet; $\mathcal{M}$ is a monoid with carrier M; Q is a finite set (of *states*); $\mathbb{I}, \mathbb{F} \colon Q \to M$ are *the initial and final output functions*; $I := \mathrm{Dom}(\mathbb{I})$ and $F := \mathrm{Dom}(\mathbb{F})$ are *the sets of initial and final states*; and $\Delta \subseteq Q \times \Sigma \times M \times Q$ is a finite *transition relation*.[9]

Definition 3.3. Let $\mathcal{T} := \big(\Sigma, (M, \circ, e), Q, \mathbb{I}, \mathbb{F}, \Delta\big)$ be a transducer. *The generalised transition relation* $\Delta^* \subseteq Q \times \Sigma^* \times M \times Q$ is the set of all tuples

$$(q_0, a_1 a_2 \cdots a_n, m_1 \circ m_2 \circ \cdots \circ m_n, q_n)$$

such that $\big((q_{i-1}, a_i, m_i, q_i)\big)_{i=1}^{n}$ is a finite sequence of transitions. *The behaviour of* $\mathcal{T}$ is the relation $[\![\mathcal{T}]\!]$ from Σ^* to M that maps $\alpha \in \Sigma^*$ to

$$\bigcup_{(i,f) \in I \times F} \big\{\mathbb{I}(i) \circ m \circ \mathbb{F}(f) \mid (i, \alpha, m, f) \in \Delta^*\big\}.$$

We shall say that $\mathcal{T}$ *represents* (or *realises*) $[\![\mathcal{T}]\!]$.

Definition 3.4. Two transducers are called *equivalent* if their behaviours coincide. A function from Σ^* to $\mathcal{M}$ is called *rational* if it can be realised by a $(\Sigma, \mathcal{M})$-transducer.[10]

3.2 Sequential Transducers

Since we are interested in the efficient representation of functions and, in particular, language models, we shall focus primarily on deterministic transducers (Schützenberger, 1977) because they enable the computation of $[\![\mathcal{T}]\!](\alpha)$ in $O\big(|\alpha|\big)$ time.

Definition 3.5. Let $\mathcal{T} := (\Sigma, \mathcal{M}, Q, \mathbb{I}, \mathbb{F}, \Delta)$ be a transducer. $\mathcal{T}$ is called *sequential* if $\mathbb{I} = (i, \iota)$ and, for every $(p, a) \in Q \times \Sigma$, there is at most one $(m, q) \in M \times Q$ such that $(p, a, m, q) \in \Delta$. Thus, if $\mathcal{T}$ is sequential, we define *the transition function* $\delta \colon Q \times \Sigma \to Q$ and *the transition output function* $\lambda \colon Q \times \Sigma \to M$ such that $\delta(p, a) := q$ and $\lambda(p, a) := m$ if and only if $(p, a, m, q) \in \Delta$. Furthermore, we define *the generalised transition function* $\delta^* \colon Q \times \Sigma^* \to Q$ and *the generalised transition output function* $\lambda^* \colon Q \times \Sigma^* \to M$ such that $\delta^*(p, \alpha) := q$ and $\lambda^*(p, \alpha) := m$ if and only if $(p, \alpha, m, q) \in \Delta^*$. Lastly, if $\mathcal{T}$ is sequential, we shall also denote it as $\big(\Sigma, \mathcal{M}, Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$.

Consequently, the behaviour of a sequential transducer $\mathcal{T}$ is a function that can be expressed as

$$[\![\mathcal{T}]\!](\alpha) = \iota \circ \lambda^*(i, \alpha) \circ \mathbb{F}\big(\delta^*(i, \alpha)\big).$$

Definition 3.6. A function from Σ^* to $\mathcal{M}$ is called *sequential* if it can be realised by a sequential $(\Sigma, \mathcal{M})$-transducer.

Note that unidirectional language models based on saturated RNNs (Merrill, 2019) or RNNs using the Heaviside activation (Svete and Cotterell, 2023) are in fact sequential because they transition between a finite number of states in a deterministic manner.[11] Similarly, unidirectional Transformer-based language models are sequential functions because of the boundedness of the lengths of their contexts (Vaswani et al., 2017) (if N is the maximum context length, then the number of states is bounded by $|\Sigma|^N$). This connection with language models that are used in practice motivates the consideration of sequential $(\Sigma, \mathcal{R}_{[0,1]})$-transducers.[12]

3.3 Stochastic Sequential Transducers

Definition 3.7. A sequential $(\Sigma, \mathcal{R}_{[0,1]})$-transducer $\mathcal{T}$ is *probabilistic* if $[\![\mathcal{T}]\!]$ is a probability distribution over Σ^*; and *stochastic* if $\iota = 1$ and

$$(\forall q \in Q)\Big(\mathbb{F}(q) + \sum_{a \in \Sigma} \lambda(q, a) = 1\Big).$$

Obviously, probabilistic sequential transducers represent exactly the class of sequential language models. Likewise, stochastic sequential transducers realise a subclass of sequential prefix models. Indeed, every stochastic sequential transducer $\mathcal{T}$ defines a prefix factorisation

$$\phi_\alpha(a) := \begin{cases} \lambda\big(\delta^*(i, \alpha), a\big) & \text{if } a \in \Sigma \\ \mathbb{F}\big(\delta^*(i, \alpha)\big) & \text{if } a = \$ \end{cases}$$

[9, 10] See Appendix C.1 for a justification of the definition.

[11] A similar argument could be made for any RNN whose activation function maps onto a finite set. In that sense, all deployed RNN language models are sequential transducers, albeit with a very large state space.

[12] When working with sequential $(\Sigma, \mathcal{R}_{[0,1]})$-transducers, we shall implicitly assume that $\mathbb{F}$, δ and λ are total functions. In this case, it follows that δ^*, λ^* and $[\![\mathcal{T}]\!]$ are also total.

that generates $[\![\mathcal{T}]\!]$ (see Appendix C.2). Compared to probabilistic sequential transducers, the main advantage of stochastic sequential transducers is the fact that they are extremely efficient at sampling since each of their states defines a probability distribution over $\Sigma_{\$}$.[13]

Despite of the fact that stochastic sequential transducers cannot realise all sequential prefix models, they can represent the class of sequential language models. In fact, when applied to a probabilistic sequential transducer, the classical *canonisation* construction of Mohri et al. (2008) (see Appendix C.3) produces an equivalent sequential transducer that is stochastic (see Appendix C.4).

Theorem 3.1. *Every probabilistic sequential transducer is equivalent to a stochastic one.*

Hence, in order to represent sequential language models, it is sufficient to consider only stochastic sequential transducers and thus work with tractable and easy to sample from finite distributions. Next, we show that we can also easily avoid the stochastic sequential transducers that are not probabilistic.

Theorem 3.2. *A stochastic sequential transducer $\mathcal{T}$ is probabilistic if and only if every accessible state of $\mathcal{T}$ is co-accessible.*[14]

The condition from Theorem 3.2 is a corollary of a classical result about absorbing Markov chains (see Appendix C.5). The condition is trivially satisfied if the transition and final outputs of every state define a positive distribution over $\Sigma_{\$}$. In practice, this is true whenever the $\mathrm{softmax}$ activation is used (see Appendix C.6 for further discussion).

3.4 Limitations of Sequential Transducers

Unlike regular languages, which can be recognised by deterministic automata, not all rational language models can be represented by sequential transducers. Thus, in order to obtain better understanding of the representational limitations of sequential transducers, we characterise the class of language models that they can realise (see Appendix C.7).

Definition 3.8. Let $\alpha, \beta \in \Sigma^*$. *The prefix distance between α and β* is defined as

$$d_p(\alpha, \beta) \coloneqq |\alpha| + |\beta| - 2|\alpha \wedge \beta|,$$

where $\wedge$ is the longest common prefix operation.

[13] In this case, the probability of \$ is represented by the final output of the corresponding state.

[14] In this setting, a state q is called *accessible* if there exists a word α such that $\delta^*(i, \alpha) = q$ and $\iota\lambda^*(i, \alpha) \neq 0$; and *co-accessible* if there exists a word α such that $\delta^*(q, \alpha) \in F$ and $\lambda^*(q, \alpha)\mathbb{F}(\delta^*(q, \alpha)) \neq 0$.

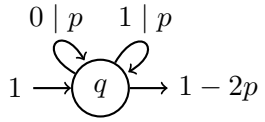

Figure 1: Sequential transducer realising $\mathbb{P}$ and $\mathbb{P}(\alpha^\top)$.

Theorem 3.3. *A rational language model $\mathbb{P}$ over Σ is sequential if and only if*

$$\left\{ \frac{\mathbb{P}(\alpha)}{\mathbb{P}(\beta)} \;\middle|\; \alpha, \beta \in \mathrm{Supp}(\mathbb{P}) \wedge d_p(\alpha, \beta) \leq n \right\}$$

is finite for all $n \in \mathbb{N}$.

Intuitively, since (Σ^*, d_p) is a metric space, Theorem 3.3 states that sequential language models map bounded sets of words into finite sets of probabilities (see Section 4.1 for a rational language model that violates this constraint). To alleviate the above-mentioned limitations of sequential transducers, we could also consider the class of *co-sequential* functions, which can be represented by a sequential transducer that scans the input from right to left and then reverses the output.

Definition 3.9. For a word α, we define *the reverse of α* as $\alpha^\top \coloneqq \alpha_{|\alpha|} \cdots \alpha_1$, and, for a real number x, we define *the reverse of x* as $x^\top \coloneqq x$. We call a function f from Σ^* to Γ^* or $\mathcal{R}_{[0,1]}$ *co-sequential* if $f(\alpha) = g(\alpha^\top)^\top$ for some sequential function g.

Co-sequential functions are rational. Moreover, they are complementary to sequential functions. Indeed, in a fixed integer base, multiplication by a given integer can be implemented by a sequential transducer if and only if it reads from right to left, while it is the converse that is true for division. Nevertheless, sequential and co-sequential language models do not exhaust the whole class of rational language models (see Section 4.1).

4 Rational Language Models

In this section, we consider bidirectional language modelling with a pair of right-to-left and left-to-right sequential transducers. We show that, compared to single independent sequential transducers, such bidirectional representations are exponentially more concise and significantly more expressive because they can realise any rational language model.

4.1 Motivating Example

Let $\Sigma \coloneqq \{0, 1\}$ and $p \in (0, 0.5)$. Consider the language model $\mathbb{P}$ over Σ such that

$$\mathbb{P}(\alpha) \coloneqq (1 - 2p)p^{|\alpha|}.$$

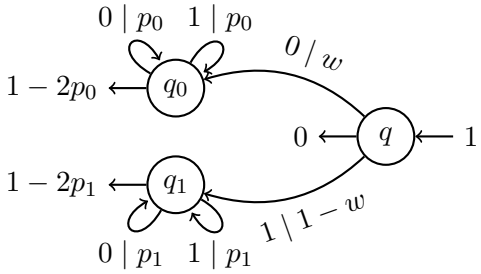

Figure 2: Sequential transducer representing $\overline{\mathbb{P}}(\alpha^\top)$.

It is easy to see that this language model is both sequential and co-sequential (see Figure 1).

Now, let us consider language models that are more discriminative with respect to the last letter. In particular, for $\alpha \in \Sigma^*$ and $i \in \Sigma$, let

$$\mathbb{P}_i(\alpha i) \coloneqq (1 - 2p_i)p_i^{|\alpha|},$$

where $p_i \in (0, 0.5)$. The language models $\mathbb{P}_i$ have supports $\Sigma^* i$ and are just as easy to represent as $\mathbb{P}$. However, this is not the case for their mixture

$$\overline{\mathbb{P}}(\alpha) \coloneqq w\mathbb{P}_0(\alpha) + (1 - w)\mathbb{P}_1(\alpha),$$

where $w \in (0, 1)$. Indeed, on input αi a sequential transducer $\mathcal{T}$ that represents $\overline{\mathbb{P}}$ should anticipate the last letter i in order to distinguish $\mathbb{P}_0$ from $\mathbb{P}_1$. However, if $p_0 \neq p_1$, $\mathcal{T}$ would be unable to tell whether the probability of αi is

$$w(1 - 2p_0)p_0^{|\alpha|} \quad \text{or} \quad (1 - w)(1 - 2p_1)p_1^{|\alpha|}$$

until it scans the last letter. It should be intuitively clear that $\mathcal{T}$ cannot resolve this uncertainty with finite memory.[15] On the other hand, a sequential transducer that scans the input in reverse would begin with the last letter i and could immediately determine the correct distribution $\mathbb{P}_i$ (see Figure 2). Thus, $\overline{\mathbb{P}}$ is co-sequential but not sequential. Of course, one can symmetrically construct a language model that is sequential but not co-sequential.

Next, we show that there are rational language models that are neither sequential nor co-sequential. To achieve this, we make both the first and the last letter of an input word crucial for determining its probability. Formally, for $\alpha \in \Sigma^*$ and $i, j \in \Sigma$, let

$$\mathbb{P}_{ij}(i\alpha j) \coloneqq (1 - 2p_{ij})p_{ij}^{|\alpha|},$$

where $p_{ij} \in (0, 0.5)$. Now, consider the mixture

$$\widetilde{\mathbb{P}}(\alpha) \coloneqq \sum_{i,j \in \Sigma} w_{ij}\mathbb{P}_{ij}(\alpha), \tag{3}$$

where $w_{ij} \in (0, 1)$ sum to 1. It should be clear that, if p_{ij} are pairwise distinct, then $\widetilde{\mathbb{P}}$ is neither sequential nor co-sequential (see Appendix D.1). However, by using two sequential transducers – a right-to-left and a left-to-right one, we can easily represent this language model (see Figure 3). Indeed, suppose that the right-to-left sequential transducer runs first and scans the input $\beta \coloneqq i\alpha j$ in reverse. The first letter it reads is j and subsequently it augments each of the letters of β with the additional *feature* j; that is, it transforms β into $\gamma \coloneqq (\beta_1, j)(\beta_2, j) \cdots (\beta_{|\beta|}, j)$. This helps the left-to-right transducer because it runs on γ and once it reads $(\beta_1, j) = (i, j)$ it can uniquely identify the language model $\mathbb{P}_{ij}$ that should be simulated.

4.2 Bisequential Decompositions

The example above motivates the study of the well-established notion of a bisequential decomposition.

Definition 4.1. A *bisequential decomposition of* $f\colon \Sigma^* \to M$ is a tuple (Γ, η, g), where

(i) Γ is an alphabet;

(ii) $\eta\colon \Sigma^* \to \Gamma^*$ is a co-sequential function;

(iii) $g\colon \Gamma^* \to M$ is a sequential function;

(iv) $f = \eta \circ g$.[16]

We say that a tuple $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ is a *representation* of the bisequential decomposition (Γ, η, g) if $\mathcal{T}_\eta$ and $\mathcal{T}_g$ are sequential transducers such that

$$[\![\mathcal{T}_\eta]\!](\alpha^\top)^\top = \eta(\alpha) \quad \text{and} \quad [\![\mathcal{T}_g]\!] = g.$$

In the previous section, we described a bisequential decomposition of the language model in (3). In fact, the decomposition had a very particular form; namely, $\Gamma = \Sigma \times \Sigma$ and η preserved the input and augmented it with additional features.

Definition 4.2. A bisequential decomposition (Γ, η, g) of a function from Σ^* to M is called *standard* if $\Gamma = \Sigma \times \Gamma'$ and $\eta \circ \pi_{\Sigma^*} = \mathrm{id}_{\Sigma^*}$.[17]

It is well-known that the class of functions that admit a bisequential decomposition is exactly the class of rational functions (Elgot and Mezei, 1965). Thus, for language models we obtain the following.

Theorem 4.1. *A language model is rational if and only if it admits a bisequential decomposition. In this case, it also admits a standard decomposition.*

[15] While the presented argument is quite informal, a rigorous proof, by means of Theorem 3.3, is given in Appendix D.1.

[16] We define $(\eta \circ g)(\alpha) \coloneqq g\big(\eta(\alpha)\big)$.

[17] π_{Σ^*} is the projection function from $(\Sigma \times \Gamma')^*$ to Σ^* and id_{Σ^*} is the identity function on Σ^*.

One way to think of a bisequential decomposition (Γ, η, g) of a language model $\mathbb{P}$ is the following: Γ is a set of *regular features*, the co-sequential function η *encodes* an input word α into a sequence of features and the function g computes, based on those features, the probability of α. On the other hand, given a standard bisequential decomposition (Γ, η, g) of a language model $\mathbb{P}$, the function g can be viewed as a *generator*. Indeed, since $\mathbb{P} = \eta \circ g$ and η is injective, it follows that $g\mathbf{1}_{\mathrm{Im}(\eta)}$ is a language model over Γ.[18] Furthermore, due to the special form of η, sampling from $\mathbb{P}$ is equivalent to sampling from $g\mathbf{1}_{\mathrm{Im}(\eta)}$ and projecting onto Σ^*.

Theorem 4.2. *Let (Γ, η, g) be a standard bisequential decomposition of $\mathbb{P}\colon \Sigma^* \to [0, 1]$. Then, $\mathbb{P}$ is a language model over Σ if and only if $g\mathbf{1}_{\mathrm{Im}(\eta)}$ is a sequential language model over Γ.*

For a formal verification of Theorem 4.2, see Appendix E.1. Now, Theorem 3.1 implies that, if we want to represent rational language models, it is sufficient to consider only representations $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ of standard bisequential decompositions where $\mathcal{T}_g$ is stochastic. Moreover, recalling Theorem 3.2, we could easily guarantee that the represented function is a language model by making sure that every accessible state of the stochastic $\mathcal{T}_g$ is co-accessible.

4.3 Conciseness of Decompositions

Next, we note that even in the cases where a language model $\mathbb{P}$ is (co-)sequential, $\mathbb{P}$ might admit an exponentially more concise representation as a bisequential decomposition. For example, this occurs when $\mathrm{Supp}(\mathbb{P})$ is difficult to be recognised.

Consider, the class $\mathcal{P}_{\Sigma,n}$ of language models $\mathbb{P}$ over Σ such that

$$\mathrm{Supp}(\mathbb{P}) = \bigcup_{a,b\in\Sigma} a\Sigma^n a\Sigma^* b\Sigma^n b.$$

Theorem 4.3. *Every sequential transducer that represents (either sequentially of co-sequentially) a language model from $\mathcal{P}_{\Sigma,n}$ has $\Omega\big(|\Sigma|^n\big)$ states.*

Regardless of the direction, such a sequential transducer should check that the letters at positions 1 and $n+2$ in α and $\alpha^\top$ match. Intuitively, in order to do that, it has to maintain the last $n+1$ scanned letters, which requires $\Omega\big(|\Sigma|^n\big)$ states.

Theorem 4.4. *There exist (co-)sequential language models in $\mathcal{P}_{\Sigma,n}$ that admit a bisequential decomposition with a representation $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ such that $\mathcal{T}_\eta$ and $\mathcal{T}_g$ have $O\big(n|\Sigma|\big)$ states.*

Intuitively, a representation $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ of a bisequential decomposition of a language model from $\mathcal{P}_{\Sigma,n}$ could function as follows. The encoder $\mathcal{T}_\eta$ could co-sequentially verify that the letters at positions 1 and $n+2$ of $\alpha^\top$ coincide. To do so, it needs to remember the first letter of $\alpha^\top$ and count to $n+1$. This can be achieved with $O\big(n|\Sigma|\big)$ states. Similarly, the generator $\mathcal{T}_g$ could sequentially check that the letters at positions 1 and $n+2$ of α match with $O\big(n|\Sigma|\big)$ states. For a formal treatment of the arguments presented above, see Appendix D.2.

4.4 Expressiveness of Decompositions

The example from Section 4.1 might misleadingly suggest that rational language models that are not sequential (or co-sequential) can capture only local features from the ending (or the beginning) of a word. This is not true and the following theorem, whose proof can be found in Appendix D.3, shows that rational language models are closed under arbitrary mixing and regular conditioning.

Theorem 4.5. *Let $w \in (0, 1)$, $L \subseteq \Sigma^*$ be a regular language and $\mathbb{P}_1, \mathbb{P}_2$ be language models over Σ.*

(i) If $\mathbb{P}_1$ is (co-)sequential, then $\mathbb{P}_1$ is rational.

(ii) If $\mathbb{P}_1$ is rational and $\mathbb{P}_1(L) \neq 0$, then the conditional language model $\mathbb{P}_1(\bullet \mid L)$ is rational.

(iii) If $\mathbb{P}_1$ and $\mathbb{P}_2$ are rational with disjoint supports, then so is the mixture $w\mathbb{P}_1 + (1-w)\mathbb{P}_2$.

Note that regular languages are closed under union and complement; that is, they form an algebra[19]. This is why conditioning on such sets is both probabilistically sound and algorithmically tractable. Other natural classes of formal languages, such as the context-free languages, lack those properties. Furthermore, for the context-sensitive languages, the problem whether $L \neq \emptyset$ is not decidable, which means that it is algorithmically intractable to verify if the conditional language model $\mathbb{P}_1(\bullet \mid L)$ is well-defined.

Next, we shed some light on the structure of a standard bisequential decomposition (Γ, η, g) of a language model $\mathbb{P}$ over Σ, where $\Gamma \coloneqq \Sigma \times \Gamma'$. In particular, the following theorem describes a set of useful additional features Γ' that can be maintained by the encoder η and then used by the generator g to output, via the chain rule, the correct probabilities.

[18] $\mathbf{1}_{\mathrm{Im}(\eta)}$ is the indicator function of the image $\mathrm{Im}(\eta)$ of η. Thus, $g\mathbf{1}_{\mathrm{Im}(\eta)}(\alpha)$ equals $g(\alpha)$ if $\alpha \in \mathrm{Im}(\eta)$, and 0 otherwise.

[19] However, they do not form a σ-algebra

Theorem 4.6. *Let $\mathbb{P}$ be a language model over Σ. Then, $\mathbb{P}$ is rational if and only if there exists a finite partition $\{L_i\}_{i=1}^n$ of Σ^* such that, for $1 \leq i \leq n$, L_i is regular and $\left\{\mathbb{P}(\,\cdot\,\alpha \mid L_i\alpha)\right\}_{\alpha\in\Sigma^*}$ is finite.*

As the proof of Theorem 4.6 in Appendix D.4 reveals, the additional features can be defined as

$$\Gamma' \coloneqq \left\{\left(\mathbb{P}(\,\cdot\,\alpha \mid L_i\alpha)\right)_{i=1}^n \;\middle|\; \alpha \in \Sigma^*\right\}.$$

Then, a representation $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ of (Γ, η, g) can be constructed such that

(i) the encoder $\mathcal{T}_\eta$ has states Γ', which enables it to preserve the input α and maintain in a co-sequential manner the appropriate additional feature $f_j \coloneqq \left(\mathbb{P}(\,\cdot\,\alpha_{>j} \mid L_i\alpha_{>j})\right)_{i=1}^n$;

(ii) given the encoding $\eta(\alpha) = \left((\alpha_j, f_j)\right)_{j=1}^{|\alpha|}$, the generator $\mathcal{T}_g$ maintains in a sequential manner the set L_{i_j} that contains the prefix $\alpha_{<j}$;

(iii) using the information about L_{i_j} as well as the feature f_j, the generator $\mathcal{T}_g$ correctly outputs $\mathbb{P}(L_{i_{j+1}}\alpha_{\geq j} \mid L_{i_j}\alpha_{>j})$ on input (α_j, f_j).

Thus, via the chain rule, the generator can sequentially compute $\mathbb{P}(\alpha)$.

4.5 Minimal Co-sequential Lookahead

A natural question that arises from the discussion above is about the minimal co-sequential lookahead or the minimal information from the future that is required in order to represent a rational language model $\mathbb{P}$. Quantitatively, it should correspond to the number of states of the minimal sequential transducer $\mathcal{T}_\eta$ such that $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ is a representation of a bisequential decomposition of $\mathbb{P}$. The following theorem gives the answer to this question (see Appendix D.5 for a proof).

Theorem 4.7. *Let $\mathbb{P}$ be a positive language model over Σ. Then, $\equiv_\mathbb{P} \subseteq \Sigma^* \times \Sigma^*$, defined as*

$$\alpha \equiv_\mathbb{P} \beta \iff \left\{\frac{\mathbb{P}(\gamma\alpha)}{\mathbb{P}(\gamma\beta)} \;\middle|\; \gamma \in \Sigma^*\right\} \text{ is finite},$$

is a left congruence of finite index. Furthermore, if $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ is a representation of a bisequential decomposition of $\mathbb{P}$, then $\mathcal{T}_\eta$ has at least $|\Sigma^*/{\equiv_\mathbb{P}}|$ states and this bound is tight.[20]

Obviously, sequential language models require no co-sequential lookahead. Thus, the left congruence $\equiv_\mathbb{P}$ of a sequential language model $\mathbb{P}$ should have a single equivalence class. This observation clearly illustrates the difference in expressivity between the sequential and the rational language models (see Appendix D.5 for a formal verification).

[20] With $\Sigma^*/{\equiv_\mathbb{P}}$ we denote the equivalence classes of $\equiv_\mathbb{P}$.

5 Latent Language Models

The study of bisequential decompositions in Section 4 suggests that for language modelling it might be beneficial to map (via an encoding function η) the input probability space Σ^* onto a latent probability space Γ^* where the corresponding probability measure $\mathbb{P}$ could be easier to represent. To make sure that the composition $\eta \circ \mathbb{P}$ is a probability measure on Σ^*, we have to ensure that no probability mass in the latent space 'leaks' onto inaccessible elements (that is, $\operatorname{Supp}(\mathbb{P}) \subseteq \operatorname{Im}(\eta)$) and that the encoding function η does not map different elements of Σ^* onto a single latent element with non-zero probability (that is, η should be injective on $\eta^{-1}\big(\operatorname{Supp}(\mathbb{P})\big)$). Indeed, those properties are sufficient for $\eta \circ \mathbb{P}$ to be a language model over Σ. In this case, we also have that η is a bijection from $\eta^{-1}\big(\operatorname{Supp}(\mathbb{P})\big)$ to $\operatorname{Supp}(\mathbb{P})$; thus, sampling from $\eta \circ \mathbb{P}$ is equivalent to sampling $\gamma \in \operatorname{Supp}(\mathbb{P})$ from $\mathbb{P}$ and then computing $\eta^{-1}(\gamma)$.

5.1 Latent Decompositions

The discussion above motivates the following definition of a latent decomposition.

Definition 5.1. A *latent decomposition of* $f\colon \Sigma^* \to [0, 1]$ is a tuple (Γ, η, g), where

(i) Γ is an alphabet;

(ii) $\eta\colon \Sigma^* \to \Gamma^*$ is a function that is injective on $\eta^{-1}\big(\operatorname{Supp}(g)\big)$ and $\operatorname{Supp}(g) \subseteq \operatorname{Im}(\eta)$;

(iii) $g\colon \Gamma^* \to [0, 1]$ is a sequential function;

(iv) $f = \eta \circ g$.

A *latent language model* is a language model that admits a latent decomposition.

Note that latent decompositions generalise standard bisequential decompositions by relaxing the constraints on the encoder. Indeed, if (Γ, η, g) is a standard bisequential decomposition of a language model $\mathbb{P}$, then $(\Gamma, \eta, g\mathbf{1}_{\operatorname{Im}(\eta)})$ is a latent decomposition of $\mathbb{P}$. Thus, we can conclude the following.

Theorem 5.1. *Every rational language model is a latent language model.*

Naturally, we can also generalise Theorem 4.2 to the class of latent decompositions.

Theorem 5.2. *Let (Γ, η, g) be a latent decomposition of $\mathbb{P}\colon \Sigma^* \to [0,1]$. Then, $\mathbb{P}$ is a language model over Σ if and only if g is a sequential language model over Γ.*

For detailed proofs, see Appendix E.1.

5.2 Expressiveness of Latent Decompositions

Next, we argue that latent language models provide a meaningful generalisation of rational language models. To this end, we note that standard bisequential decompositions are as expressive as non-standard ones (see Theorem 4.1). However, this is not the case for standard latent decompositions.

Definition 5.2. A latent decomposition (Γ, η, g) of a function from Σ^* to $[0,1]$ is called *standard* if $\Gamma = \Sigma \times \Gamma'$ and $\eta \circ \pi_{\Sigma^*} = \mathrm{id}_{\Sigma^*}$.

In fact, suppose that $(\Sigma \times \Gamma', \eta, g)$ is a standard latent decomposition. Then, $\mathrm{Im}(\eta)$ is the graph of the function $\eta \circ \pi_{\Gamma'^*}$. Without loss of generality, assume that $\mathrm{Im}(\eta) = \mathrm{Supp}(g)$. Since g is sequential, it follows that $\mathrm{Im}(\eta)$ is regular. Therefore, $\eta \circ \pi_{\Gamma'^*}$ is a rational function. This implies that η is rational and thus $\eta \circ g$ is also rational.

Theorem 5.3. *Every latent language model that admits a standard latent decomposition is rational.*

Nevertheless, non-standard latent decompositions are strictly more expressive. Indeed, consider the language model $\mathbb{P}$ that assigns probability $1/2^{n+1}$ to $a^n b^n$. Obviously, $\mathbb{P}$ is not rational because $\mathrm{Supp}(\mathbb{P}) = \{a^n b^n\}_{n \in \mathbb{N}}$ is not a regular language. However, $g\big((ab)^n\big) \coloneqq 1/2^{n+1}$ is a sequential language model. Thus, $\big(\{a,b\}, \eta, g\big)$, where the encoder $\eta(a^n b^n) \coloneqq (ab)^n$ drastically simplifies the support $\mathbb{P}$, is a latent decomposition of $\mathbb{P}$.

Theorem 5.4. *Latent language models are strictly more expressive than rational language models.*

For a more formal and detailed treatment of the arguments presented above, see Appendix E.2.

5.3 Comparison with Other Latent Models

Finally, we compare latent language models with a couple of other well-established latent models.

First, we consider vq-wav2vec (Baevski et al., 2020). Similarly to a latent language model, vq-wav2vec consists of an encoder η that maps from an input space Σ^* to a latent space Γ^*, and a language model $\mathbb{P}$ over Γ. However, in vq-wav2vec, η is not restricted to be injective, which leads to several issues. First, classical maximum likelihood estimation cannot be used to train the composition $\eta \circ \mathbb{P}$ because of the existence of a degenerate optimum in which the encoder collapses; that is, η maps all elements of Σ^* to a single latent element $\gamma \in \Gamma^*$ and $\mathbb{P}$ places all of the probability mass on γ. This necessitates the use of more indirect and inefficient contrastive estimation methods (Gutmann and Hyvärinen, 2012; van den Oord et al., 2019). Furthermore, $\eta \circ \mathbb{P}$ is not guaranteed to be a language model and, even if it is, $\mathbb{P}$ cannot be used to efficiently sample from $\eta \circ \mathbb{P}$.

Second, we consider Discrete Flows (Tran et al., 2019), which are comprised of a bijective encoder $\eta\colon \Sigma^* \to \Gamma^*$ and a sequential language model over Γ; i.e., Discrete Flows are latent language models. Tran et al. (2019) propose two types of encoders. The *bipartite* encoder $\eta \coloneqq \eta^{(1)} \circ \eta^{(2)} \circ \cdots \circ \eta^{(l)}$ is such that, for $\alpha \in \Sigma^*$, $1 \leq i \leq |\alpha|$ and $1 \leq j \leq l$,

$$\eta^{(j)}(\alpha)_i \coloneqq \begin{cases} \alpha_i & \text{if } i \equiv j \pmod 2 \\ f^{(j)}(\alpha^{(j)}, \alpha_i) & \text{otherwise} \end{cases}.$$

In the expression above, $\alpha^{(j)}$ denotes the concatenation of the letters α_i such that $i \equiv j \pmod 2$. Therefore, each layer $\eta^{(j)}$ of a bipartite encoder η preserves the letters $\alpha^{(j)}$ of the input α and modifies the remaining letters based on $\alpha^{(j)}$. Furthermore, in order for $\eta^{(j)}$ to be bijective, $f^{(j)}$ is chosen such that $\alpha^{(j)}$ and $f^{(j)}(\alpha^{(j)}, \alpha_i)$ uniquely identify α_i. Lastly, note that Discrete Flows correspond to a subclass of non-standard latent decompositions that can represent non-rational language models. In fact, a sufficiently deep multi-layer bipartite encoder can implement the mapping $a^n b^n \mapsto (ab)^n$. For further empirical evidence of the advantages of Discrete Flows, we refer to Tran et al. (2019).

In addition, we compare latent language models with discrete diffusion language models such as D3PM (Austin et al., 2021) in Appendix E.3.

6 Conclusion

We introduced a class of consistent bidirectional language models, called latent language models, that allow for efficient sampling and scoring of sequences. We defined latent language models based on the well-understood formalism of bisequential decompositions. This formal correspondence allowed us to precisely charaterise the abilities and limitations of a subclass of latent language models, called rational language models. As a result, we showed that latent language models are exponentially more concise and significantly more expressive than unidirectional language models.

Limitations

The primary focus of this paper is the characterisation of the representational capacity and conciseness of rational language models, which constitute a strict subclass of latent language models. Consequently, the question about the expressive power of the full class of latent language models remains unanswered. In other words, it is not clear how latent language models relate to other classes of formal languages; for example, whether they also subsume the class of (deterministic) context-free language models. Furthermore, we do not explore the limitations of latent language models. While a straightforward constraint that latent language models impose is that their images should be rational sets, further work is required to obtain deeper understanding of the limits of their capabilities.

Another notable limitation is that we do not assess the learnability of latent language models; that is, we do not consider the problem of searching for such models via gradient-based or other optimisation methods. A latent language model consists of a latent encoder η and a unidirectional language model g. It is well known that unidirectional language models, such as g, are effectively learnable. However, it is not obvious if η and g could also be jointly optimised and if so what is an appropriate parametric family for η.

Lastly, we do not explore the applicability of latent language models to natural language processing tasks. That is, it remains to be empirically verified whether the increased expressive power of latent language models could lead to better downstream task performance when compared to classical unidirectional language models.

Acknowledgements

Georgi Shopov acknowledges that this work is partially supported by CLaDA-BG, *the Bulgarian National Interdisciplinary Research e-Infrastructure for Resources and Technologies in favor of the Bulgarian Language and Cultural Heritage, part of the EU infrastructures CLARIN and DARIAH*, funded by the Ministry of Education and Science of Bulgaria (support for the Bulgarian National Roadmap for Research Infrastructure).

Stefan Gerdjikov acknowledges that this study is financed by the European Union – NextGenerationEU, through the National Recovery and Resilience Plan of the Republic of Bulgaria, project No. BG-RRP-2.004-0008.

References

Ebru Arisoy, Abhinav Sethy, Bhuvana Ramabhadran, and Stanley Chen. 2015. Bidirectional recurrent neural network language models for automatic speech recognition. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5421–5425.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. 2021. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, volume 34, pages 17981–17993. Curran Associates, Inc.

Alexei Baevski, Steffen Schneider, and Michael Auli. 2020. vq-wav2vec: Self-supervised learning of discrete speech representations. In *International Conference on Learning Representations*.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3(44):1137–1155.

Julian Besag. 1974. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):192–236.

Kiante Brantley, Kyunghyun Cho, Hal Daumé, and Sean Welleck. 2019. Non-monotonic sequential text generation. In *Proceedings of the 2019 Workshop on Widening NLP*, pages 57–59, Florence, Italy. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N.

Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating large language models trained on code. *Preprint*, arXiv:2107.03374.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Christian Choffrut. 1977. Une caractérisation des fonctions séquentielles et des fonctions sous-séquentielles en tant que relations rationnelles. *Theoretical Computer Science*, 5(3):325–337.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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.

Samuel Eilenberg. 1974. *Automata, Languages, and Machines*, volume A. Academic Press.

Calvin C. Elgot and Jorge E. Mezei. 1965. On relations defined by generalized finite automata. *IBM Journal of Research and Development*, 9(1):47–68.

Jeffrey L. Elman. 1990. Finding structure in time. *Cognitive Science*, 14(2):179–211.

Dmitrii Emelianenko, Elena Voita, and Pavel Serdyukov. 2019. Sequence modeling with unconstrained generation order. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Nicolas Ford, Daniel Duckworth, Mohammad Norouzi, and George Dahl. 2018. The importance of generation order in language modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2942–2946, Brussels, Belgium. Association for Computational Linguistics.

Daniel Fried, Armen Aghajanyan, Jessy Lin, Sida Wang, Eric Wallace, Freda Shi, Ruiqi Zhong, Scott Yih, Luke Zettlemoyer, and Mike Lewis. 2023. InCoder: A generative model for code infilling and synthesis. In *International Conference on Learning Representations*.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019. Mask-predict: Parallel decoding of conditional masked language models. 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 6112–6121, Hong Kong, China. Association for Computational Linguistics.

Kartik Goyal, Chris Dyer, and Taylor Berg-Kirkpatrick. 2022. Exposing the implicit energy networks behind masked language models via Metropolis–Hastings. In *International Conference on Learning Representations*.

Charles M. Grinstead and J. Laurie Snell. 1997. *Introduction to probability*. American Mathematical Society.

Jiatao Gu, Qi Liu, and Kyunghyun Cho. 2019. Insertion-based decoding with automatically inferred generation order. *Transactions of the Association for Computational Linguistics*, 7:661–676.

Michael U. Gutmann and Aapo Hyvärinen. 2012. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, 13(11):307–361.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In *International Conference on Learning Representations*.

Xuanlin Li, Brandon Trabucco, Dong Huk Park, Michael Luo, Sheng Shen, Trevor Darrell, and Yang Gao. 2021. Discovering non-monotonic autoregressive orderings with variational inference. In *International Conference on Learning Representations*.

Chu-Cheng Lin, Aaron Jaech, Xin Li, Matthew R. Gormley, and Jason Eisner. 2021. Limitations of autoregressive models and their alternatives. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5147–5173, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *Preprint*, arXiv:1907.11692.

William Merrill. 2019. Sequential neural networks as automata. In *Proceedings of the Workshop on Deep Learning and Formal Languages: Building Bridges*, pages 1–13, Florence. Association for Computational Linguistics.

Stoyan Mihov and Klaus U. Schulz. 2019. *Finite-State Techniques: Automata, Transducers and Bimachines*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Proc. Interspeech 2010*, pages 1045–1048.

Mehryar Mohri. 1997. Finite-state transducers in language and speech processing. *Computational Linguistics*, 23(2):269–311.

Mehryar Mohri, Fernando Pereira, and Michael Riley. 2008. Speech recognition with weighted finite-state transducers. In Jacob Benesty, M. Mohan Sondhi, and Yiteng Arden Huang, editors, *Springer Handbook of Speech Processing*, pages 559–584. Springer Berlin Heidelberg, Berlin, Heidelberg.

Amr Mousa and Björn Schuller. 2017. Contextual bidirectional long short-term memory recurrent neural network language models: A generative approach to sentiment analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1023–1032, Valencia, Spain. Association for Computational Linguistics.

J. R. Norris. 1997. *Markov Chains*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Christophe Reutenauer and Marcel-Paul Schützenberger. 1991. Minimization of rational word functions. *SIAM Journal on Computing*, 20(4):669–685.

Jacques Sakarovitch. 2009. *Elements of Automata Theory*. Cambridge University Press.

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, Online. Association for Computational Linguistics.

M.P. Schützenberger. 1961. A remark on finite transducers. *Information and Control*, 4(2):185–196.

M.P. Schützenberger. 1977. Sur une variante des fonctions séquentielles. *Theoretical Computer Science*, 4(1):47–57.

Mitchell Stern, William Chan, Jamie Kiros, and Jakob Uszkoreit. 2019. Insertion transformer: Flexible sequence generation via insertion operations. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5976–5985. PMLR.

Anej Svete and Ryan Cotterell. 2023. Recurrent neural language models as probabilistic finite-state automata. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8069–8086, Singapore. Association for Computational Linguistics.

Lucas Torroba Hennigen and Yoon Kim. 2023. Deriving language models from masked language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1149–1159, Toronto, Canada. Association for Computational Linguistics.

Dustin Tran, Keyon Vafa, Kumar Agrawal, Laurent Dinh, and Ben Poole. 2019. Discrete flows: Invertible generative models of discrete data. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding. *Preprint*, arXiv:1807.03748.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. 2015. Order matters: Sequence to sequence for sets. In *International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving

with large language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 11809–11822. Curran Associates, Inc.

Tom Young, Yunan Chen, and Yang You. 2024. Inconsistencies in masked language models. *Preprint*, arXiv:2301.00068.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V Le, and Ed H. Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. In *International Conference on Learning Representations*.

A Related Work

The problem of characterising the consistent prefix factorisations is tackled by Du et al. (2023). They give an easy to enforce condition that is sufficient for consistency. Moreover, the authors show that many of the models used in practice to represent prefix factorisations satisfy this property. In particular, the authors demonstrate that all prefix factorisations realised by Transformers or RNNs (such as GRU and LSTM) with a $\mathrm{softmax}$ prediction head are in fact consistent.

Nevertheless, we are not aware of the existence of such results for confix factorisations. Goyal et al. (2022); Torroba Hennigen and Kim (2023) acknowledge the problem of ensuring the consistency of confix factorisations as well as the intractability of sampling from them. To this end, Goyal et al. (2022) attempt to sidestep the issue by interpreting confix factorisations as energy-based models and deriving a different incompatible distribution from them. Similarly, Torroba Hennigen and Kim (2023) explore methods for deriving incompatible joint distributions from confix factorisation. However, they focus only on distributions over two-letter alphabets.

Several studies (Vinyals et al., 2015; Ford et al., 2018) have suggested that the order in which sequences are processed is important for language modelling. Furthermore, there have been many different proposals for achieving bidirectionality by learning the decoding order (Brantley et al., 2019; Stern et al., 2019; Gu et al., 2019; Li et al., 2021). Nonetheless, each of those solutions requires either expensive beam search or variational inference for decoding. Moreover, many of the methods do not allow for efficient scoring of sequences. The only bidirectional models we are aware of that achieve both efficient generation and scoring are the Discrete Flows by Tran et al. (2019), which are in fact a special case of the latent language models that we introduce.

We note that Svete and Cotterell (2023) have already explored the connection between unidirectional language models and sequential transducers. The authors show that unidirectional language models based on Heaviside RNNs are as expressive as sequential transducers. However, we are not aware of similar developments for bidirectional models. More importantly, we do not know of other works that have attempted to define bidirectional language models in a principled way and have explored their expressive power and representational conciseness in terms of formal abstractions from automata and formal language theory.

B Factorisations of Language Models

B.1 Language Modelling with Prefix Factorisations

In this appendix, we describe in more details the correspondence between the language models that are compatible with a given prefix factorisation Φ and the prefix model generated by Φ. We begin by considering several basic properties of prefix factorisations.[21]

Definition B.1. Let Σ be an alphabet and $A \subseteq \Sigma^*$. For $n \in \mathbb{N}$, we define the sets A^n, $A^{<n}$ and $A^{\leq n}$ as

$$A^n \coloneqq \begin{cases} \epsilon & \text{if } n = 0 \\ A^{n-1}A & \text{otherwise} \end{cases}, \quad A^{<n} \coloneqq \bigcup_{i=0}^{n-1} A^i \quad \text{and} \quad A^{\leq n} \coloneqq A^{<n} \cup A^n.$$

Definition B.2. Let $(\phi_\alpha)_{\alpha \in \Sigma^*}$ be a prefix factorisation over Σ. For every $\alpha \in \Sigma^*$, we use ϕ^*_α to denote the extension of ϕ_α to $\Sigma^*\$^{\leq 1}$ defined as

$$\phi^*_\alpha(\beta) \coloneqq \prod_{i=1}^{|\beta|} \phi_{\alpha\beta_{<i}}(\beta_i).$$

Remark B.1. Now, given a prefix factorisation $(\phi_\alpha)_{\alpha \in \Sigma^*}$ over Σ, the prefix model M that is generated by it can be expressed as

$$M(\alpha) = \phi^*_\epsilon(\alpha\$).$$

[21] In accord with general use, we shall identify a singleton set with the element that it contains and omit the braces.

Proposition B.1. *Let $(\phi_\alpha)_{\alpha\in\Sigma^*}$ be a prefix factorisation over Σ. Then, for $\alpha,\beta\in\Sigma^*$ and $\gamma\in\Sigma^*\$^{\leq 1}$,*

$$\phi^*_\alpha(\beta\gamma) = \phi^*_\alpha(\beta)\phi^*_{\alpha\beta}(\gamma).$$

Proof. Follows directly from Definition B.2. □

Remark B.2. Given a prefix factorisation $(\phi_\alpha)_{\alpha\in\Sigma^*}$, we additively extend every $f\in\bigcup_{\alpha\in\Sigma^*}\{\phi_\alpha,\phi^*_\alpha\}$ to $\mathfrak{P}\big(\mathrm{Dom}(f)\big)$ as

$$f(A) \coloneqq \sum_{\alpha\in A} f(\alpha).$$

Proposition B.2. *Let $(\phi_\alpha)_{\alpha\in\Sigma^*}$ be a prefix factorisation over Σ. Then,*

$$(\forall\alpha\in\Sigma^*)\big(\phi^*_\alpha(\Sigma^*\$)\leq 1\big).$$

Proof. Let $\alpha\in\Sigma^*$. First, we prove by induction on n that

$$(\forall n\in\mathbb{N})\big(\phi^*_\alpha(\Sigma^{\leq n}\$) + \phi^*_\alpha(\Sigma^{n+1}) = 1\big). \tag{4}$$

For $n=0$, we have that

$$\phi^*_\alpha(\Sigma^{\leq 0}\$) + \phi^*_\alpha(\Sigma) = \phi_\alpha(\$) + \phi_\alpha(\Sigma) = \phi_\alpha(\Sigma_\$) = 1.$$

Suppose the statement holds for $n\in\mathbb{N}$. Then,

$$\begin{aligned}
\phi^*_\alpha(\Sigma^{\leq n+1}\$) + \phi^*_\alpha(\Sigma^{n+2}) &= \phi^*_\alpha(\Sigma^{\leq n}\$) + \phi^*_\alpha(\Sigma^{n+1}\$) + \phi^*_\alpha(\Sigma^{n+1}\Sigma)\\
&= \phi^*_\alpha(\Sigma^{\leq n}\$) + \phi^*_\alpha(\Sigma^{n+1}\Sigma_\$)\\
&= \phi^*_\alpha(\Sigma^{\leq n}\$) + \sum_{\beta\in\Sigma^{n+1}} \phi^*_\alpha(\beta)\cancelto{1}{\phi_{\alpha\beta}(\Sigma_\$)}\\
&= \phi^*_\alpha(\Sigma^{\leq n}\$) + \phi^*_\alpha(\Sigma^{n+1})\\
&= 1.
\end{aligned}$$

Now, since $\phi^*_\alpha(\Sigma^*\$)$ is the limit of the partial sums $\big(\phi^*_\alpha(\Sigma^{\leq n}\$)\big)_{n\in\mathbb{N}}$, from (4), we obtain that

$$\phi^*_\alpha(\Sigma^*\$) = \lim_{n\to\infty}\phi^*_\alpha(\Sigma^{\leq n}\$) \leq \lim_{n\to\infty}\big(\phi^*_\alpha(\Sigma^{\leq n}\$) + \phi^*_\alpha(\Sigma^{n+1})\big) = 1.$$ □

Proposition B.3. *Let $(\phi_\alpha)_{\alpha\in\Sigma^*}$ be a prefix factorisation over Σ and $\alpha\in\Sigma^*$ be such that $\phi^*_\epsilon(\alpha)\neq 0$ and $\phi^*_\alpha(\Sigma^*\$)<1$. Then,*

$$\big(\forall 0\leq n\leq|\alpha|\big)\big(\phi^*_\epsilon(\alpha_{\leq n})\neq 0 \wedge \phi^*_{\alpha_{\leq n}}(\Sigma^*\$)<1\big).$$

Proof. We proceed by downward induction on n. For $n=|\alpha|$, the statement is true by assumption. Suppose that the statement holds for $n>0$. Then, it is obvious that $\phi^*_\epsilon(\alpha_{<n})\neq 0$. Furthermore,

$$\phi^*_{\alpha_{<n}}(\Sigma^*\$) = \phi_{\alpha_{<n}}(\$) + \phi_{\alpha_{<n}}(\alpha_n)\phi^*_{\alpha_{\leq n}}(\Sigma^*\$) + \sum_{a\in\Sigma\setminus\alpha_n}\phi_{\alpha_{<n}}(a)\phi^*_{\alpha_{<n}a}(\Sigma^*\$).$$

However, $\phi^*_{\alpha_{\leq n}}(\Sigma^*\$)<1$ by the inductive hypothesis and $\phi^*_{\alpha_{<n}a}(\Sigma^*\$)\leq 1$ by Proposition B.2. Therefore, we conclude that

$$\phi^*_{\alpha_{<n}}(\Sigma^*\$) < \phi_{\alpha_{<n}}(\$) + \phi_{\alpha_{<n}}(\alpha_n) + \sum_{a\in\Sigma\setminus\alpha_n}\phi_{\alpha_{<n}}(a) = \phi_{\alpha_{<n}}(\Sigma_\$) = 1.$$ □

Proposition B.4. *Let $\Phi\coloneqq(\phi_\alpha)_{\alpha\in\Sigma^*}$ be a prefix factorisation over Σ and M be the prefix model generated by Φ. Then, M is a language model if and only if*

$$(\forall\alpha\in\Sigma^*)\big(\phi^*_\epsilon(\alpha)\neq 0 \implies \phi^*_\alpha(\Sigma^*\$)=1\big).$$

Proof. Assume that M is a language model and let $\alpha \in \Sigma^*$ be such that $\phi^*_\epsilon(\alpha) \neq 0$. Now, suppose that $\phi^*_\alpha(\Sigma^*\$) \neq 1$. Then, $\phi^*_\alpha(\Sigma^*\$) < 1$ by Proposition B.2. Therefore, Proposition B.3 implies that

$$\sum_{\beta \in \Sigma^*} M(\beta) = \sum_{\beta \in \Sigma^*} \phi^*_\epsilon(\beta\$) = \phi^*_\epsilon(\Sigma^*\$) < 1,$$

which contradicts the fact that M is a language model. Thus, $\phi^*_\alpha(\Sigma^*\$) = 1$.

The backward direction is trivial since $\phi^*_\epsilon(\epsilon) = 1 \neq 0$ by definition and $\phi^*_\epsilon(\Sigma^*\$) = 1$ is equivalent to M being a language model. □

Theorem B.1. *A prefix factorisation Φ over Σ is consistent if and only if the prefix model M generated by Φ is a language model over Σ. In this case, M is the only language model compatible with Φ.*

Proof. Let $\Phi \coloneqq (\phi_\alpha)_{\alpha \in \Sigma^*}$ be a prefix factorisation and M be the prefix model generated by Φ.

First, assume that Φ is consistent and let $\mathbb{P}$ be a language model that is compatible with Φ. Then, for $\alpha \in \Sigma^*$, the chain rule implies that[22]

$$M(\alpha) = \Big(\prod_{i=1}^{|\alpha|} \phi_{\alpha_{<i}}(\alpha_i)\Big)\phi_\alpha(\$) = \Big(\prod_{i=1}^{|\alpha|} \mathbb{P}(\alpha_{\leq i}\Sigma^* \mid \alpha_{<i}\Sigma^*)\Big)\mathbb{P}(\alpha \mid \alpha\Sigma^*) = \mathbb{P}(\alpha).$$

Therefore, M is a language model. Furthermore, since $\mathbb{P}$ is arbitrary, every language model that is compatible with Φ coincides with M; that is, M is the only language model compatible with Φ.

Now, assume that M is a language model. Let $\alpha \in \Sigma^*$ be such that $M(\alpha\Sigma^*) \neq 0$. Then, since

$$M(\alpha\Sigma^*) = \sum_{\beta \in \Sigma^*} M(\alpha\beta) = \sum_{\beta \in \Sigma^*} \phi^*_\epsilon(\alpha)\phi^*_\alpha(\beta\$) = \phi^*_\epsilon(\alpha) \sum_{\beta \in \Sigma^*} \phi^*_\alpha(\beta\$) = \phi^*_\epsilon(\alpha)\phi^*_\alpha(\Sigma^*\$),$$

it follows that $\phi^*_\epsilon(\alpha) \neq 0$. Therefore, Proposition B.4 implies that, for $a \in \Sigma$,

$$M(\alpha a\Sigma^* \mid \alpha\Sigma^*) = \frac{M(\alpha a\Sigma^*)}{M(\alpha\Sigma^*)} = \frac{\cancel{\phi^*_\epsilon(\alpha)}\,\phi_\alpha(a)\phi^*_{\alpha a}(\Sigma^*\$)}{\cancel{\phi^*_\epsilon(\alpha)}\,\cancel{\phi^*_\alpha(\Sigma^*\$)}^{\,1}} = \phi_\alpha(a)\phi^*_{\alpha a}(\Sigma^*\$) = \phi_\alpha(a),$$

where the last equality holds because $\phi^*_{\alpha a}(\Sigma^*\$) = 1$ whenever $\phi_\alpha(a) \neq 0$. Similarly, we derive that

$$M(\alpha \mid \alpha\Sigma^*) = \frac{M(\alpha)}{M(\alpha\Sigma^*)} = \frac{\cancel{\phi^*_\epsilon(\alpha)}\,\phi_\alpha(\$)}{\cancel{\phi^*_\epsilon(\alpha)}\,\cancel{\phi^*_\alpha(\Sigma^*\$)}^{\,1}} = \phi_\alpha(\$).$$

Therefore, Φ is consistent and M is compatible with Φ. □

B.2 Language Modelling with Confix Factorisations

In this appendix, we describe in more details the correspondence between the language models that are compatible with a given complete confix factorisation $(\Phi, \mathbb{P}_L)$ and the confix model generated by $(\Phi, \mathbb{P}_L)$. Additionally, we demonstrate that, unlike prefix factorisations, there exist inconsistent complete confix factorisations whose confix models are language models.

Remark B.3. Note that, according to Definition 2.5, there are no confix factorisations over $\emptyset$ because there are no probability distributions over $\emptyset$. Nevertheless, we shall extend Definition 2.5 by considering the empty family as *the confix factorisation over* $\emptyset$.

Remark B.4. The extension of Definition 2.5 leads to the following additional amendments.

[22] Note that, for $\big(\mathbb{P}(\alpha_{\leq i}\Sigma^* \mid \alpha_{<i}\Sigma^*)\big)_{i=1}^{|\alpha|}$ and $\mathbb{P}(\alpha \mid \alpha\Sigma^*)$ to be defined and the derivation of $M(\alpha) = \mathbb{P}(\alpha)$ to be valid, it is necessary that $\mathbb{P}(\alpha\Sigma^*) \neq 0$. However, even if $\mathbb{P}(\alpha\Sigma^*) = 0$, $M(\alpha) = \mathbb{P}(\alpha)$ still holds since $\mathbb{P}(\alpha) = 0$ because $\mathbb{P}(\alpha) \leq \mathbb{P}(\alpha\Sigma^*)$, and $M(\alpha) = 0$ because there exists $1 \leq i \leq |\alpha|$ such that $\mathbb{P}(\alpha_{<i}\Sigma^*) \neq 0$ and $\phi_{\alpha_{<i}}(\alpha_i) = \mathbb{P}(\alpha_{\leq i}\Sigma^* \mid \alpha_{<i}\Sigma^*) = 0$.

(i) A *complete confix factorisation over* Σ is a tuple $(\Phi, \mathbb{P}_L)$ such that Φ is a positive confix factorisation over Σ and $\mathbb{P}_L$ is a probability distribution over $\mathbb{N}_\Sigma$, where

$$\mathbb{N}_\Sigma \coloneqq \{|\alpha| \mid \alpha \in \Sigma^*\} = \begin{cases} 0 & \text{if } \Sigma = \emptyset \\ \mathbb{N} & \text{otherwise} \end{cases}.$$

(ii) Let $\mathbb{P}$ be a language mode over Σ and $\mathbb{P}_L$ is a probability distribution over $\mathbb{N}_\Sigma$. We say that $\mathbb{P}$ is *compatible with* $\mathbb{P}_L$ if

$$(\forall n \in \mathbb{N}_\Sigma)\big(\mathbb{P}(\Sigma^n) = \mathbb{P}_L(n)\big).$$

(iii) Let $(\Phi, \mathbb{P}_L)$ is a complete confix factorisation over Σ. $(\Phi, \mathbb{P}_L)$ is called *consistent* if there exists a language model $\mathbb{P}$ over Σ that is *compatible with* $(\Phi, \mathbb{P}_L)$; i.e., $\mathbb{P}$ is compatible with both Φ and $\mathbb{P}_L$.

By considering complete confix factorisations, we restrict our attention only to positive confix factorisation. However, not every language model is compatible with a positive confix factorisation.

Proposition B.5. *Let $\mathbb{P}$ be a language model over Σ. Then, $\mathbb{P}$ is compatible with a positive confix factorisation over Σ if and only if, for every $n \in \mathbb{N}_\Sigma$,*

$$(\exists \alpha \in \Sigma^n)\big(\mathbb{P}(\alpha) = 0\big) \iff \mathbb{P}(\Sigma^n) = 0. \tag{5}$$

Proof. Assume that $\mathbb{P}$ is compatible with the positive confix factorisation $\Phi \coloneqq (\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$. The backward direction of (5) holds trivially; thus, we focus on the forward direction. We note that, since $\mathbb{P}(\Sigma^0) = \mathbb{P}(\epsilon)$, the forward direction of (5) holds trivially for $n = 0$. In what follows, we consider the case where $n > 0$.

Let $\alpha \in \Sigma^* \setminus \epsilon$ be such that $\mathbb{P}(\alpha) = 0$. We prove by induction on i that

$$\big(\forall 0 \le i \le |\alpha|\big)\big(\mathbb{P}(\Sigma^i \alpha_{>i}) = 0\big).$$

Note that, when $i = |\alpha|$, we obtain that $\mathbb{P}(\Sigma^{|\alpha|}) = 0$ (that is, the forward direction of (5) holds for $n > 0$).

For $i = 0$, it is obvious that $\mathbb{P}(\Sigma^i \alpha_{>i}) = \mathbb{P}(\alpha) = 0$. Now, assume that $\mathbb{P}(\Sigma^i \alpha_{>i}) = 0$ for $0 \le i < |\alpha|$, and suppose that $\mathbb{P}(\Sigma^{i+1}\alpha_{>i+1}) \neq 0$. Then, there exists $\beta \in \Sigma^i$ such that $\mathbb{P}(\beta\Sigma\alpha_{>i+1}) \neq 0$ and

$$\mathbb{P}(\beta\alpha_{>i}) = \mathbb{P}(\beta\Sigma\alpha_{>i+1})\mathbb{P}(\beta\alpha_{>i} \mid \beta\Sigma\alpha_{>i+1}) = \mathbb{P}(\beta\Sigma\alpha_{>i+1})\phi_{\beta,\alpha_{>i+1}}(\alpha_{i+1}) \neq 0.$$

However, this contradicts with $\mathbb{P}(\beta\alpha_{>i}) \le \mathbb{P}(\Sigma^i\alpha_{>i}) = 0$. Therefore, $\mathbb{P}(\Sigma^{i+1}\alpha_{>i+1}) = 0$.

Next, assume that (5) holds. Consider the confix factorisation $\Phi \coloneqq (\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$ defined, for $\alpha, \beta \in \Sigma^*$, as follows:

(i) if $\mathbb{P}(\alpha\Sigma\beta) \neq 0$, then $\phi_{\alpha,\beta} \coloneqq \mathbb{P}(\alpha \bullet \beta \mid \alpha\Sigma\beta)$;

(ii) if $\mathbb{P}(\alpha\Sigma\beta) = 0$, then let $\phi_{\alpha,\beta}$ be an arbitrary positive probability distribution over Σ.

Now, for $\alpha, \beta \in \Sigma^*$, if $\mathbb{P}(\alpha\Sigma\beta) \neq 0$, then, for $a \in \Sigma$,

$$\mathbb{P}(\alpha a\beta) \neq 0 \quad \text{and} \quad \phi_{\alpha,\beta}(a) = \mathbb{P}(\alpha a\beta \mid \alpha\Sigma\beta) = \frac{\mathbb{P}(\alpha a\beta)}{\mathbb{P}(\alpha\Sigma\beta)} \neq 0;$$

otherwise, if $\mathbb{P}(\alpha\Sigma\beta) = 0$, then $\phi_{\alpha,\beta}$ is positive by definition. □

Proposition B.6. *Let $\mathbb{P}$ be a language model over Σ that is compatible with a positive confix factorisation $(\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$ over Σ. Then, for $\alpha \in \Sigma^*$ and $\beta \in \Sigma^{|\alpha|}$,*

$$\mathbb{P}(\alpha) = \mathbb{P}(\beta) \prod_{i=1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)}. \tag{6}$$

Proof. Let $\alpha \in \Sigma^*$ and $\beta \in \Sigma^{|\alpha|}$. First, assume that $\mathbb{P}(\alpha) \neq 0$. Then, from Proposition B.5, it follows that $\mathbb{P}(\Sigma^{|\alpha|}) \neq 0$. Now, we prove by downward induction on $0 \leq j \leq |\alpha|$ that

$$\mathbb{P}(\alpha) = \mathbb{P}(\alpha_{\leq j}\beta_{>j}) \prod_{i=j+1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)}. \tag{7}$$

Note that, when $j = 0$, (7) is equivalent to (6).

For $j = |\alpha|$, (7) holds trivially. Now, assume that (7) is true for $0 < j \leq |\alpha|$. Then,

$$\begin{aligned}
\mathbb{P}(\alpha) &= \mathbb{P}(\alpha_{\leq j}\beta_{>j}) \prod_{i=j+1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)} \\
&= \mathbb{P}(\alpha_{<j}\Sigma\beta_{>j})\mathbb{P}(\alpha_{\leq j}\beta_{>j} \mid \alpha_{<j}\Sigma\beta_{>j}) \prod_{i=j+1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)} \\
&= \mathbb{P}(\alpha_{<j}\beta_{\geq j}) \frac{\mathbb{P}(\alpha_{\leq j}\beta_{>j} \mid \alpha_{<j}\Sigma\beta_{>j})}{\mathbb{P}(\alpha_{<j}\beta_{\geq j} \mid \alpha_{<j}\Sigma\beta_{>j})} \prod_{i=j+1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)} \\
&= \mathbb{P}(\alpha_{\leq j-1}\beta_{>j-1}) \prod_{i=j}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)}.
\end{aligned}$$

□

Next, assume that $\mathbb{P}(\alpha) = 0$. Then, $\mathbb{P}(\Sigma^{|\alpha|}) = 0$ by Proposition B.5, which implies that $\mathbb{P}(\beta) = 0$ and thus (6) holds.

Definition B.3. Let $\Phi \coloneqq (\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$ be a positive confix factorisation over Σ. For $\alpha \in \Sigma^*$ and $n \in \mathbb{N}_\Sigma$, we define

$$\Phi_\alpha \coloneqq \frac{1}{\sum_{\beta\in\Sigma^{|\alpha|}} \prod_{i=1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}} \quad \text{and} \quad \Phi_n \coloneqq \sum_{\beta\in\Sigma^n} \Phi_\beta.$$

Proposition B.7. *Let $\mathbb{P}$ be a language model over Σ that is compatible with a complete confix factorisation $(\Phi, \mathbb{P}_L)$ over Σ. Then, for $\alpha \in \Sigma^*$,*

$$\mathbb{P}(\alpha) = \mathbb{P}_L\big(|\alpha|\big)\Phi_\alpha.$$

Furthermore, for $n \in \mathbb{N}_\Sigma$ such that $\mathbb{P}_L(n) \neq 0$,

$$(\forall \alpha \in \Sigma^n)\big(\Phi_\alpha = \mathbb{P}(\alpha \mid \Sigma^n)\big) \quad \textit{and} \quad \Phi_n = 1.$$

Proof. Let $\Phi =: (\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$ and $\alpha \in \Sigma^*$. Then, from Proposition B.6, it follows that, for $\beta \in \Sigma^{|\alpha|}$,

$$\mathbb{P}(\beta) = \mathbb{P}(\alpha) \prod_{i=1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}.$$

Summing over $\beta \in \Sigma^{|\alpha|}$, we obtain that

$$\mathbb{P}(\Sigma^{|\alpha|}) = \mathbb{P}(\alpha) \sum_{\beta\in\Sigma^{|\alpha|}} \prod_{i=1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)},$$

which can be rearranged as

$$\mathbb{P}(\alpha) = \frac{\mathbb{P}(\Sigma^{|\alpha|})}{\sum_{\beta\in\Sigma^{|\alpha|}} \prod_{i=1}^{|\alpha|} \frac{\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)}{\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)}} = \mathbb{P}_L\big(|\alpha|\big)\Phi_\alpha.$$

Furthermore, if $\mathbb{P}_L\big(|\alpha|\big) \neq 0$, then $\mathbb{P}(\Sigma^{|\alpha|}) \neq 0$,

$$\Phi_\alpha = \frac{\mathbb{P}(\alpha)}{\mathbb{P}(\Sigma^{|\alpha|})} = \mathbb{P}(\alpha \mid \Sigma^{|\alpha|}) \quad \text{and} \quad \Phi_{|\alpha|} = \sum_{\alpha\in\Sigma^{|\alpha|}} \Phi_\alpha = \sum_{\alpha\in\Sigma^{|\alpha|}} \mathbb{P}(\alpha \mid \Sigma^{|\alpha|}) = 1.$$

□

Theorem B.2. *Let $(\Phi, \mathbb{P}_L)$ be a consistent complete confix factorisation over Σ. Then, the confix model generated by $(\Phi, \mathbb{P}_L)$ is the only language model over Σ that is compatible with $(\Phi, \mathbb{P}_L)$.*

Proof. Let $\mathbb{P}$ be a language model over Σ that is compatible with $(\Phi, \mathbb{P}_L)$ (such a language model exists since $(\Phi, \mathbb{P}_L)$ is consistent). Now, Proposition B.7 implies that $\mathbb{P}$ coincides with the confix model M generated by $(\Phi, \mathbb{P}_L)$. Therefore, M is the only language model that is compatible with $(\Phi, \mathbb{P}_L)$. □

Next, we show that, unlike prefix factorisations, there exist inconsistent complete confix factorisations whose confix models are language models. The main reason for this deficiency is that, as we shall show, Φ_n can take on values that are less than and greater than one when $n \geq 2$. We begin by noting that $\Phi_0 = \Phi_1 = 1$ for every positive confix factorisation Φ over Σ.

Proposition B.8. *Let Φ be a positive confix factorisation over Σ. Then, $\Phi_0 = \Phi_1 = 1$.*

Proof. Let $\Phi =: (\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$. Then,

$$\Phi_0 = \sum_{\alpha\in\Sigma^0} \frac{1}{\sum_{\beta\in\Sigma^0} \frac{\phi^*_{\alpha,\beta}(\beta)}{\phi^*_{\alpha,\beta}(\alpha)}} = \frac{1}{\frac{\cancel{\phi^*_{\epsilon,\epsilon}(\epsilon)}}{\cancel{\phi^*_{\epsilon,\epsilon}(\epsilon)}}} = 1,$$

$$\Phi_1 = \sum_{\alpha\in\Sigma^1} \frac{1}{\sum_{\beta\in\Sigma^1} \frac{\phi^*_{\alpha,\beta}(\beta)}{\phi^*_{\alpha,\beta}(\alpha)}} = \sum_{a\in\Sigma} \frac{1}{\sum_{b\in\Sigma} \frac{\phi_{\epsilon,\epsilon}(b)}{\phi_{\epsilon,\epsilon}(a)}} = \sum_{a\in\Sigma} \frac{\phi_{\epsilon,\epsilon}(a)}{\sum_{b\in\Sigma} \phi_{\epsilon,\epsilon}(b)} = 1. \qquad \square$$

Now, we consider the particular case where $|\Sigma| = 2$ and $n \geq 2$. We note that the following result can be further extended to alphabets with more than two letters. However, in this work, we do not pursue this direction.

Proposition B.9. *Let $\lessgtr \in \{<, =, >\}$, Σ be an alphabet such that $|\Sigma| = 2$, and $n \in \mathbb{N}$ be such that $n \geq 2$. Then, there exists a positive confix factorisation Φ over Σ such that $\Phi_n \lessgtr 1$.*

Proof. First, we consider the case where $n = 2$. Let $\Sigma =: \{a, b\}$ and $\Phi := (\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$ be a positive confix factorisation over Σ. We begin by observing that

$$\begin{aligned}
\Phi_2 &= \sum_{\alpha\in\Sigma^2} \frac{1}{\sum_{\beta\in\Sigma^2} \frac{\phi^*_{\alpha,\beta}(\beta)}{\phi^*_{\alpha,\beta}(\alpha)}} = \sum_{\alpha_1,\alpha_2\in\Sigma} \frac{1}{\sum_{\beta_1,\beta_2\in\Sigma} \frac{\phi_{\epsilon,\beta_2}(\beta_1)\phi_{\alpha_1,\epsilon}(\beta_2)}{\phi_{\epsilon,\beta_2}(\alpha_1)\phi_{\alpha_1,\epsilon}(\alpha_2)}} \\
&= \sum_{\alpha_1,\alpha_2\in\Sigma} \frac{1}{\sum_{\beta_2\in\Sigma} \frac{\phi_{\alpha_1,\epsilon}(\beta_2)}{\phi_{\epsilon,\beta_2}(\alpha_1)\phi_{\alpha_1,\epsilon}(\alpha_2)} \cancel{\sum_{\beta_1\in\Sigma} \phi_{\epsilon,\beta_2}(\beta_1)}^{\,1}} = \sum_{\alpha_1,\alpha_2\in\Sigma} \frac{\phi_{\alpha_1,\epsilon}(\alpha_2)}{\sum_{\beta_2\in\Sigma} \frac{\phi_{\alpha_1,\epsilon}(\beta_2)}{\phi_{\epsilon,\beta_2}(\alpha_1)}} \\
&= \sum_{\alpha_1\in\Sigma} \frac{\cancel{\sum_{\alpha_2\in\Sigma} \phi_{\alpha_1,\epsilon}(\alpha_2)}^{\,1}}{\sum_{\beta_2\in\Sigma} \frac{\phi_{\alpha_1,\epsilon}(\beta_2)}{\phi_{\epsilon,\beta_2}(\alpha_1)}} = \sum_{\alpha_1\in\Sigma} \frac{1}{\sum_{\beta_2\in\Sigma} \frac{\phi_{\alpha_1,\epsilon}(\beta_2)}{\phi_{\epsilon,\beta_2}(\alpha_1)}}.
\end{aligned}$$

Then, by letting $\phi_{\epsilon,a} = \phi_{a,\epsilon}$, we obtain that

$$\begin{aligned}
\sum_{\alpha_1\in\Sigma} \frac{1}{\sum_{\beta_2\in\Sigma} \frac{\phi_{\alpha_1,\epsilon}(\beta_2)}{\phi_{\epsilon,\beta_2}(\alpha_1)}} &= \frac{1}{\frac{\phi_{a,\epsilon}(a)}{\phi_{\epsilon,a}(a)} + \frac{\phi_{a,\epsilon}(b)}{\phi_{\epsilon,b}(a)}} + \frac{1}{\frac{\phi_{b,\epsilon}(a)}{\phi_{\epsilon,a}(b)} + \frac{\phi_{b,\epsilon}(b)}{\phi_{\epsilon,b}(b)}} \\
&= \frac{\phi_{\epsilon,a}(a)\phi_{\epsilon,b}(a)}{\phi_{a,\epsilon}(a)\phi_{\epsilon,b}(a) + \phi_{a,\epsilon}(b)\phi_{\epsilon,a}(a)} + \frac{\phi_{\epsilon,a}(b)\phi_{\epsilon,b}(b)}{\phi_{b,\epsilon}(a)\phi_{\epsilon,b}(b) + \phi_{b,\epsilon}(b)\phi_{\epsilon,a}(b)} \\
&= \frac{\cancel{\phi_{\epsilon,a}(a)}\phi_{\epsilon,b}(a)}{\cancel{\phi_{\epsilon,a}(a)}\phi_{\epsilon,b}(a) + \phi_{\epsilon,a}(b)\cancel{\phi_{\epsilon,a}(a)}} + \frac{\phi_{\epsilon,a}(b)\phi_{\epsilon,b}(b)}{\phi_{b,\epsilon}(a)\phi_{\epsilon,b}(b) + \phi_{b,\epsilon}(b)\phi_{\epsilon,a}(b)} \\
&= \frac{\phi_{\epsilon,b}(a)\big(\phi_{b,\epsilon}(a)\phi_{\epsilon,b}(b) + \phi_{b,\epsilon}(b)\phi_{\epsilon,a}(b)\big) + \phi_{\epsilon,a}(b)\phi_{\epsilon,b}(b)\big(\phi_{\epsilon,b}(a) + \phi_{\epsilon,a}(b)\big)}{\big(\phi_{\epsilon,b}(a) + \phi_{\epsilon,a}(b)\big)\big(\phi_{b,\epsilon}(a)\phi_{\epsilon,b}(b) + \phi_{b,\epsilon}(b)\phi_{\epsilon,a}(b)\big)}.
\end{aligned}$$

Finally, we note that $\Phi_2 \lesseqgtr 1$ if and only if

$$\cancel{\phi_{\epsilon,b}(a)\big(\phi_{b,\epsilon}(a)\phi_{\epsilon,b}(b)+\phi_{b,\epsilon}(b)\phi_{\epsilon,a}(b)\big)} + \cancel{\phi_{\epsilon,a}(b)}\phi_{\epsilon,b}(b)\big(\phi_{\epsilon,b}(a)+\phi_{\epsilon,a}(b)\big) \lesseqgtr \big(\cancel{\phi_{\epsilon,b}(a)}+\cancel{\phi_{\epsilon,a}(b)}\big)\big(\phi_{b,\epsilon}(a)\phi_{\epsilon,b}(b)+\phi_{b,\epsilon}(b)\phi_{\epsilon,a}(b)\big),$$

which is equivalent to

$$\big(\phi_{\epsilon,a}(a)-\phi_{\epsilon,b}(a)\big)\big(\phi_{\epsilon,b}(a)-\phi_{b,\epsilon}(a)\big) \lesseqgtr 0.$$

Thus, it is obvious that Φ can be chosen so that it satisfies the statement of the proposition for $n=2$.

Next, we consider the case where $n>2$. Let $\Phi \coloneqq (\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$ be a positive confix factorisation over Σ such that, for $\alpha,\beta\in\Sigma^n$ and $\sigma\in\Sigma$,

$$(\forall 2\le i\le n-1)\Big(\phi_{\alpha_{<i},\beta_{>i}}(\sigma)=\tfrac{1}{|\Sigma|}\Big) \quad\text{and}\quad (\forall\gamma\in\Sigma^{n-2})(\phi_{\epsilon,\gamma\sigma}=\phi_{\epsilon,\sigma}\wedge\phi_{\sigma\gamma,\epsilon}=\phi_{\sigma,\epsilon}).$$

Then, we have that

$$\Phi_n = \sum_{\alpha\in\Sigma^n}\frac{1}{\sum_{\beta\in\Sigma^n}\frac{\phi^*_{\alpha,\beta}(\beta)}{\phi^*_{\alpha,\beta}(\alpha)}} = \sum_{\alpha\in\Sigma^n}\frac{1}{\sum_{\beta\in\Sigma^n}\frac{\phi_{\alpha_{<1},\beta_{>1}}(\beta_1)\cancel{\big(\prod_{i=2}^{n-1}\phi_{\alpha_{<i},\beta_{>i}}(\beta_i)\big)}\phi_{\alpha_{<n},\beta_{>n}}(\beta_n)}{\phi_{\alpha_{<1},\beta_{>1}}(\alpha_1)\cancel{\big(\prod_{i=2}^{n-1}\phi_{\alpha_{<i},\beta_{>i}}(\alpha_i)\big)}\phi_{\alpha_{<n},\beta_{>n}}(\alpha_n)}}$$

$$= \sum_{\alpha\in\Sigma^n}\frac{1}{\sum_{\beta\in\Sigma^n}\frac{\phi_{\epsilon,\beta_n}(\beta_1)\phi_{\alpha_1,\epsilon}(\beta_n)}{\phi_{\epsilon,\beta_n}(\alpha_1)\phi_{\alpha_1,\epsilon}(\alpha_n)}} = \sum_{\alpha_1,\alpha_2\in\Sigma}\frac{\cancel{|\Sigma|^{n-2}}}{\sum_{\beta_1,\beta_2\in\Sigma}\cancel{|\Sigma|^{n-2}}\frac{\phi_{\epsilon,\beta_2}(\beta_1)\phi_{\alpha_1,\epsilon}(\beta_2)}{\phi_{\epsilon,\beta_2}(\alpha_1)\phi_{\alpha_1,\epsilon}(\alpha_2)}} = \Phi_2.$$

Now, from the case where $n=2$, it follows that Φ can be chosen so that $\Phi_n \lesseqgtr 1$. □

Proposition B.10. *Let $(\lesseqgtr_n)_{n=2}^{\infty}$ be a sequence of elements of $\{<,=,>\}$ and Σ be an alphabet such that $|\Sigma|=2$. Then, there exists a positive confix factorisation Φ such that*

$$(\forall n\ge 2)(\Phi_n \lesseqgtr_n 1). \tag{8}$$

Proof. If $\Phi \coloneqq (\phi_{\alpha,\beta})_{\alpha,\beta\in\Sigma^*}$ is a positive confix factorisation over Σ, then we know that Φ_n, for $n\ge 2$, is defined only in terms of the probability distributions $\phi_{\alpha,\beta}$ for $\alpha,\beta\in\Sigma^*$ such that $|\alpha\beta|=n-1$. Now, from Proposition B.9, it follows that there is a positive confix factorisation Φ over Σ that satisfies (8). □

Theorem B.3. *There exists a complete confix factorisation $(\Phi,\mathbb{P}_L)$ over Σ that is inconsistent and the confix model generated by $(\Phi,\mathbb{P}_L)$ is a language model over Σ.*

Proof. Let $\Sigma \coloneqq \{a,b\}$. We will show that there exists a complete confix factorisation $(\Phi,\mathbb{P}_L)$ over Σ such that the confix model generated by $(\Phi,\mathbb{P}_L)$ is a language model over Σ that is not compatible with $(\Phi,\mathbb{P}_L)$. Thus, from Theorem B.2, it would follow that $(\Phi,\mathbb{P}_L)$ is inconsistent.

Let Φ be a positive confix factorisation over Σ such that

$$\Phi_2<1, \quad \Phi_3>1 \quad\text{and}\quad \big(\forall n\in\mathbb{N}\setminus\{2,3\}\big)(\Phi_n=1).$$

The existence of such a confix factorisation follows from Proposition B.10.

Let $\mathbb{P}_L$ be a length distribution such that $\mathbb{P}_L(2),\mathbb{P}_L(3)\in(0,1]$ and

$$\frac{\mathbb{P}_L(2)}{\mathbb{P}_L(3)}=\frac{\Phi_3-1}{1-\Phi_2} \quad\text{or equivalently}\quad \mathbb{P}_L(2)\Phi_2+\mathbb{P}_L(3)\Phi_3=\mathbb{P}_L(2)+\mathbb{P}_L(3).$$

It is straightforward to verify that such a length distribution exists.

Now, consider the confix model M that is generated by $(\Phi,\mathbb{P}_L)$ and observe that

$$\sum_{\alpha\in\Sigma^*}M(\alpha)=\sum_{n\in\mathbb{N}}\sum_{\alpha\in\Sigma^n}M(\alpha)=\sum_{n\in\mathbb{N}}\sum_{\alpha\in\Sigma^n}\mathbb{P}_L(n)\Phi_\alpha=\sum_{n\in\mathbb{N}}\mathbb{P}_L(n)\Phi_n$$

$$=\underbrace{\mathbb{P}_L(2)\Phi_2+\mathbb{P}_L(3)\Phi_3}_{\mathbb{P}_L(2)+\mathbb{P}_L(3)}+\sum_{n\in\mathbb{N}\setminus\{2,3\}}\mathbb{P}_L(n)\cancel{\Phi_n}^{1}=\sum_{n\in\mathbb{N}}\mathbb{P}_L(n)=1.$$

Consequently, M is a language model over Σ. However, from Proposition B.7, it follows that M is not compatible with $(\Phi,\mathbb{P}_L)$ because $\mathbb{P}_L(2)\ne 0$ and $\Phi_2\ne 1$ (also, $\mathbb{P}_L(3)\ne 0$ and $\Phi_3\ne 1$). □

C Sequential Language Models

C.1 Real-Time Transducers

In the literature, transducers are typically allowed to have ϵ transitions and transducers that do not have ϵ transitions are called *real-time* (Mihov and Schulz, 2019). Additionally, a relation is called *rational* if it can be realised by a transducer. When it comes to representing functions, transducers and real-time transducers have the same expressive power (Mihov and Schulz, 2019, Proposition 4.4.8). However, real-time transducers cannot realise every rational relation. Indeed, they can represent only those rational relations that are not *infinitely ambiguous*.[23] In this work, our focus is on representing language models; that is, we are primarily interested in the class of rational functions and not the class of rational relations. Thus, we shall consider only real-time transducers and call them simply 'transducers'.

C.2 Representational Capacity of Stochastic Sequential Transducers

In this appendix, we describe in more details the representational capacity of stochastic sequential transducers. More precisely, we prove that the behaviours of stochastic sequential transducers correspond to a subclass of sequential prefix models.

Definition C.1. Let $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{R}_{[0,1]}, Q, (i,1), \mathbb{F}, \delta, \lambda\big)$ be a stochastic sequential transducer. *The prefix factorisation* $(\phi_\alpha)_{\alpha\in\Sigma^*}$ *associated with* $\mathcal{T}$ is defined as

$$\phi_\alpha(a) \coloneqq \begin{cases} \lambda\big(\delta^*(i,\alpha), a\big) & \text{if } a \in \Sigma \\ \mathbb{F}\big(\delta^*(i,\alpha)\big) & \text{if } a = \$ \end{cases}.$$

Proposition C.1. *Let* $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{R}_{[0,1]}, Q, (i,1), \mathbb{F}, \delta, \lambda\big)$ *be a stochastic sequential transducer and* $(\phi_\alpha)_{\alpha\in\Sigma^*}$ *be the prefix factorisation associated with* $\mathcal{T}$*. Then,*

$$(\forall \alpha, \beta \in \Sigma^*)\Big(\phi^*_\alpha(\beta) = \lambda^*\big(\delta^*(i,\alpha),\beta\big)\Big).$$

Proof. For every $\alpha \in \Sigma^*$, the statement follows by a straightforward induction on $|\beta|$. □

Proposition C.2. *Let* $\mathcal{T}$ *be a stochastic sequential transducer and* Φ *be the prefix factorisation associated with* $\mathcal{T}$*. Then,* $[\![\mathcal{T}]\!]$ *coincides with the prefix model generated by* Φ*.*

Proof. Let $\mathcal{T} =: \big(\Sigma, \mathcal{R}_{[0,1]}, Q, (i,1), \mathbb{F}, \delta, \lambda\big)$, $\Phi =: (\phi_\alpha)_{\alpha\in\Sigma^*}$ and M be the prefix model generated by Φ. Then, Proposition C.1 implies that, for $\alpha \in \Sigma^*$,

$$M(\alpha) = \phi^*_\epsilon(\alpha)\phi_\alpha(\$) = \lambda^*\big(\delta^*(i,\epsilon),\alpha\big)\mathbb{F}\big(\delta^*(i,\alpha)\big) = [\![\mathcal{T}]\!](\alpha).$$ □

Remark C.1. It should be noted that the converse statement does not hold. That is, there exist sequential prefix models that cannot be represented by a stochastic sequential transducer. Nevertheless, as we shall see in Appendix C.4, the sequential prefix models that we care about (that is, the sequential language models) can all be realised by stochastic sequential transducers.

Proposition C.3. *Let* $\Phi \coloneqq (\phi_{a^n})_{n\in\mathbb{N}}$ *be a prefix factorisation defined as*

$$\phi_{a^n}(a) \coloneqq \frac{1+2^{n+1}}{2+2^{n+1}} \quad \textit{and} \quad \phi_{a^n}(\$) \coloneqq \frac{1}{2+2^{n+1}}.$$

Then, the prefix model generated by Φ *is a sequential function that cannot be represented by a stochastic sequential transducer.*

Proof. Let M be the prefix model generated by Φ. We can verify by induction on n that

$$(\forall n \in \mathbb{N})\left(M(a^n) = \frac{1}{2^{n+2}}\right).$$

[23] A relation $R \subseteq X \times Y$ is *infinitely ambiguous* if $\big\{y \in Y \mid (x,y) \in R\big\}$ is infinite for some $x \in X$.

Indeed, for $n = 0$,

$$M(\epsilon) = \phi_\epsilon(\$) = \frac{1}{2 + 2^{0+1}} = \frac{1}{2^2}.$$

Now, suppose that the statement holds for $n \in \mathbb{N}$. Then,

$$\begin{aligned} M(a^{n+1}) &= \phi_\epsilon^*(a^n)\phi_{a^n}(a)\phi_{a^{n+1}}(\$) = \frac{M(a^n)}{\phi_{a^n}(\$)}\phi_{a^n}(a)\phi_{a^{n+1}}(\$) \\ &= \frac{\cancel{2 + 2^{n+1}}}{2^{n+2}} \frac{\cancel{1 + 2^{n+1}}}{\cancel{2 + 2^{n+1}}} \frac{1}{\cancel{2(1 + 2^{n+1})}} = \frac{1}{2^{n+3}}. \end{aligned}$$

Thus, M is not a language model because

$$\sum_{n\in\mathbb{N}} M(a^n) = \sum_{n\in\mathbb{N}} \frac{1}{2^{n+2}} = \frac{1}{2}. \tag{9}$$

Next, we note that M can be realised by the sequential transducer

$$\Big(a, \mathcal{R}_{[0,1]}, q, \big(q, \tfrac{1}{2}\big), \big(q, \tfrac{1}{2}\big), \big((q, a), q\big), \big((q, a), \tfrac{1}{2}\big)\Big).$$

However, if we suppose that M can be represented by a stochastic sequential $(a, \mathcal{R}_{[0,1]})$-transducer $\mathcal{T}$, then every accessible state of $\mathcal{T}$ should be co-accessible because $\mathrm{Supp}(M) = a^*$. Therefore, by Theorem C.4, M should be a language model, which leads to a contradiction with (9). Thus, M cannot be represented by a stochastic sequential $(a, \mathcal{R}_{[0,1]})$-transducer. □

C.3 Canonisation of Sequential Transducers

In this appendix, we review a construction by Mohri et al. (2008), known in the literature as *weight-pushing*, that builds from a sequential transducer an equivalent canonical one. In essence, the construction consists of pushing the outputs of the transitions and the outputs of the final states 'towards the initial state as much a possible'.

Definition C.2. Let $\mathcal{M} \coloneqq (M, \circ, e)$ be a monoid. $\mathcal{M}$ is called *commutative* if

$$(\forall a, b \in M)(a \circ b = b \circ a).$$

An element $z \in M$ is called an *absorbing element of* $\mathcal{M}$ if

$$(\forall a \in M)(a \circ z = z \circ a = z).$$

Definition C.3. A *semiring* is a tuple $\mathcal{K} \coloneqq (K, \oplus, \odot, \overline{0}, \overline{1})$, where

(i) K is a set, called *the carrier of* $\mathcal{K}$;

(ii) $(K, \oplus, \overline{0})$ is a commutative monoid, denoted $\mathcal{K}_\oplus$;

(iii) $(K, \odot, \overline{1})$ is a monoid, denoted $\mathcal{K}_\odot$, with an absorbing element $\overline{0}$;

(iv) $\odot$ distributes over $\oplus$; that is, for any $a, b, c \in K$, it holds that

$$\begin{aligned} (a \oplus b) \odot c &= (a \odot c) \oplus (b \odot c), \\ a \odot (b \oplus c) &= (a \odot b) \oplus (a \odot c). \end{aligned}$$

Remark C.2. Let $\mathcal{K} \coloneqq (K, \oplus, \odot, \overline{0}, \overline{1})$ be a semiring. Whenever we write $\bigoplus_{i\in I} k_i$ for some family $(k_i)_{i\in I}$ of elements of K, we will implicitly assume that $\mathcal{K}$ is equipped with a partial infinitary sum operation, written $\bigoplus$, such that

(i) $\bigoplus$ is consistent with the finitary sum of $\mathcal{K}$; that is, if $I = \{i_1, i_2, \ldots, i_n\}$, then

$$\bigoplus_{i \in I} k_i = k_{i_1} \oplus k_{i_2} \oplus \cdots \oplus k_{i_n};$$

(ii) $\bigoplus$ is associative; that is, for any partition $(I_j)_{j \in J}$ of I,

$$\bigoplus_{j \in J} \bigoplus_{i \in I_j} k_i = \bigoplus_{i \in I} k_i;$$

(iii) $\odot$ distributes over $\bigoplus$; that is, for any $l \in K$,

$$l \odot \Big(\bigoplus_{i \in I} k_i\Big) = \bigoplus_{i \in I} (l \odot k_i) \quad \text{and} \quad \Big(\bigoplus_{i \in I} k_i\Big) \odot l = \bigoplus_{i \in I} (k_i \odot l).$$

Note that we do not require the infinitary sum operation to be total (that is, for $\mathcal{K}$ to be *complete*). Hence, all equalities in the equations above are conditional; that is, the left and right hand sides are either both defined and equal or are both undefined.

Definition C.4. A semiring $(K, \oplus, \odot, \overline{0}, \overline{1})$ is called *weakly left-divisible*[24] if for every family $(k_i)_{i \in I}$ of elements of K such that $\bigoplus_{i \in I} k_i \in K \setminus \overline{0}$ and every $j \in I$, there exists a unique element of K, denoted $\big(\bigoplus_{i \in I} k_i\big)^{-1} \odot k_j$, such that

$$\Big(\bigoplus_{i \in I} k_i\Big) \odot \Big(\Big(\bigoplus_{i \in I} k_i\Big)^{-1} \odot k_j\Big) = k_j.$$

Remark C.3. For convenience, in what follows, we shall assume that the $\mathbb{F}$, δ and λ functions of sequential transducers are total.

Definition C.5. Let $\mathcal{K} \coloneqq (K, \oplus, \odot, \overline{0}, \overline{1})$ be a semiring and $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{K}_\odot, Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$ be a sequential transducer. *The sum of $\mathcal{T}$ with respect to $\mathcal{K}$*, written $[\![\mathcal{T}]\!]_\oplus$, is defined as

$$[\![\mathcal{T}]\!]_\oplus \coloneqq \bigoplus_{\alpha \in \Sigma^*} [\![\mathcal{T}]\!](\alpha)$$

whenever the infinitary sum exists.

Definition C.6. Let $\mathcal{T} \coloneqq \big(\Sigma, (M, \circ, e), Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$ be a sequential transducer. For every $q \in Q$, we define the sequential transducer

$$\mathcal{T}_q \coloneqq \big(\Sigma, (M, \circ, e), Q, (q, e), \mathbb{F}, \delta, \lambda\big).$$

Proposition C.4. *Let $\mathcal{K} \coloneqq (K, \oplus, \odot, \overline{0}, \overline{1})$ be a semiring and $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{K}_\odot, Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$ be a sequential transducer. Then,*

$$[\![\mathcal{T}]\!]_\oplus = \iota \odot [\![\mathcal{T}_i]\!]_\oplus \quad \textit{and} \quad (\forall q \in Q)\Big([\![\mathcal{T}_q]\!]_\oplus = \bigoplus_{a \in \Sigma} \lambda(q, a) \odot [\![\mathcal{T}_{\delta(q,a)}]\!]_\oplus\Big).$$

Definition C.7. Let $\mathcal{K} \coloneqq (K, \oplus, \odot, \overline{0}, \overline{1})$ be a semiring and $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{K}_\odot, Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$ be a sequential transducer. We say that $\mathcal{T}$ is *summable with respect to* $\mathcal{K}$ if the sums $[\![\mathcal{T}_q]\!]_\oplus$ exist for all $q \in Q$. Moreover, $\mathcal{T}$ is *strictly summable with respect to* $\mathcal{K}$ if it is summable with respect to $\mathcal{K}$ and

$$(\forall q \in Q)\big([\![\mathcal{T}_q]\!]_\oplus \neq \overline{0}\big).$$

[24] Mohri et al. (2008) call those semirings *weakly left-divisible and cancellative*. Here, for the sake of conciseness, we call them simply *weakly left-divisible*.

Definition C.8. Let $\mathcal{K} \coloneqq (K, \oplus, \odot, \overline{0}, \overline{1})$ be a semiring and $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{K}_\odot, Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$ be a sequential transducer. We say that $\mathcal{T}$ is *canonical with respect to* $\mathcal{K}$ if

$$(\forall q \in Q)\big([\![\mathcal{T}_q]\!]_\oplus = \overline{1}\big).$$

Proposition C.5. *Let $\mathcal{K} \coloneqq (K, \oplus, \odot, \overline{0}, \overline{1})$ be a semiring and $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{K}_\odot, Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$ be a canonical with respect to $\mathcal{K}$ sequential transducer. Then,*

$$(\forall q \in Q)\Big(\mathbb{F}(q) \oplus \bigoplus_{a \in \Sigma} \lambda(q, a) = \overline{1}\Big).$$

Definition C.9. Let $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{K}_\odot, Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$ be a sequential transducer that is strictly summable with respect to the weakly left-divisible semiring $\mathcal{K} \coloneqq (K, \oplus, \odot, \overline{0}, \overline{1})$. *The canonical form of $\mathcal{T}$ with respect to $\mathcal{K}$* is defined as the sequential transducer

$$\big(\Sigma, \mathcal{K}_\odot, Q, (i, \iota'), \mathbb{F}', \delta, \lambda'\big), \quad \text{where}$$

(i) $\iota' \coloneqq \iota \odot [\![\mathcal{T}_i]\!]_\oplus$;

(ii) $\mathbb{F}' \coloneqq \Big\{\big(q, [\![\mathcal{T}_q]\!]_\oplus^{-1} \odot \mathbb{F}(q)\big) \mid q \in Q\Big\}$;

(iii) $\lambda' \coloneqq \Big\{\Big((q, a), [\![\mathcal{T}_q]\!]_\oplus^{-1} \odot \big(\lambda(q, a) \odot [\![\mathcal{T}_{\delta(q,a)}]\!]_\oplus\big)\Big) \;\Big|\; (q, a) \in Q \times \Sigma\Big\}$.

Theorem C.1. *Let $\mathcal{T}$ be a sequential transducer that is strictly summable with respect to the weakly left-divisible semiring $\mathcal{K}$ and $\mathcal{T}'$ be the canonical form of $\mathcal{T}$ with respect to $\mathcal{K}$. Then, $\mathcal{T}'$ is equivalent to $\mathcal{T}$ and canonical with respect to $\mathcal{K}$.*

C.4 From Probabilistic to Stochastic Sequential Transducers

In this appendix, we show that every probabilistic sequential transducer is equivalent to a stochastic sequential transducer. Hence, every sequential language model is a sequential prefix model. The proof is based on an application of the canonisation construction from Appendix C.3 with respect to the semiring $\mathcal{R}^+_{[0,\infty)} \coloneqq \big([0, \infty), +, \cdot, 0, 1\big)$. Note that $\mathcal{R}_{[0,1]}$ is a submonoid of $\mathcal{R}_{[0,\infty)} \coloneqq \big([0, \infty), \cdot, 1\big)$. Thus, in what follows, we shall also view probabilistic and stochastic transducers as $(\Sigma, \mathcal{R}_{[0,\infty)})$-transducers.

Proposition C.6. *$\mathcal{R}^+_{[0,\infty)}$ is a weakly left-divisible semiring.*

Proof. For every family $(x_i)_{i \in I}$ of elements of $[0, \infty)$ such that $\sum_{i \in I} x_i \neq 0$, we know that $\sum_{i \in I} x_i$ has a multiplicative inverse y and

$$(\forall j \in I)\Big(\Big(\sum_{i \in I} x_i\Big) \cdot (y \cdot x_j) = x_j\Big). \qquad \square$$

Proposition C.7. *Every probabilistic sequential transducer that is canonical with respect to $\mathcal{R}^+_{[0,\infty)}$ is stochastic.*

Proof. Let $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{R}_{[0,\infty)}, Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$ be a probabilistic sequential transducer that is canonical with respect to $\mathcal{R}^+_{[0,\infty)}$. Then,

$$\iota \overset{\text{canonical}}{=} \iota [\![\mathcal{T}_i]\!]_+ \overset{\text{Proposition C.4}}{=} [\![\mathcal{T}]\!]_+ \overset{\text{probabilistic}}{=} 1.$$

Additionally, Proposition C.5 implies that

$$(\forall q \in Q)\Big(\mathbb{F}(q) + \sum_{a \in \Sigma} \lambda(q, a) = 1\Big). \qquad \square$$

Remark C.4. Let $\mathcal{T}$ be a probabilistic sequential transducer. Consider the sequential transducer $\mathcal{T}'$ obtained from $\mathcal{T}$ by removing all states that are not co-accessible, adding a new state q_c with final output 1 and then completing the transition functions with transitions to q_c with output 0. Now, it is not hard to verify that $\mathcal{T}'$ is equivalent to $\mathcal{T}$ and every state of $\mathcal{T}'$ is co-accessible. Thus, if $\mathcal{T}$ is probabilistic, then $\mathcal{T}'$ is also probabilistic. Furthermore, if $\mathcal{T}$ is stochastic and every accessible state of $\mathcal{T}$ is co-accessible, then $\mathcal{T}'$ is also stochastic.

Theorem C.2. *Every probabilistic sequential transducer is equivalent to a stochastic sequential transducer.*

Proof. Let $\mathcal{T}$ be a probabilistic sequential transducer. Without loss of generality, we can assume that every state of $\mathcal{T}$ is co-accessible (see Remark C.4). Then, since $\mathcal{R}^{+}_{[0,\infty)}$ is positive (that is, $a + b = 0$ if and only if $a = b = 0$), it follows that $\mathcal{T}$ is strictly summable. Therefore, $\mathcal{T}$ has a canonical form $\mathcal{T}'$ (see Definition C.9) that is equivalent to $\mathcal{T}$ and canonical with respect to $\mathcal{R}^{+}_{[0,\infty)}$ (see Theorem C.1). Finally, Proposition C.7 implies that $\mathcal{T}'$ is a stochastic sequential transducer. □

C.5 Characterisation of the Probabilistic Stochastic Sequential Transducers

In this appendix, we describe a simple condition that characterises the stochastic sequential transducers that are probabilistic. The condition is a consequence of a classical result from the theory of Markov chains (Norris, 1997). Thus, we proceed by illustrating the correspondence between stochastic sequential transducers and Markov chains.

Definition C.10. A *Markov chain* is a tuple (S, μ, P), where

(i) S is a finite set of *states*;

(ii) $\mu \in [0,1]^S$ is a stochastic vector; that is, $\sum_{i \in S} \mu_i = 1$;

(iii) $P \in [0,1]^{S \times S}$ is a stochastic matrix; that is, $\sum_{j \in S} P_{ij} = 1$ for every $i \in S$.

Definition C.11. Let $\mathcal{C} \coloneqq (S, \mu, P)$ be a Markov chain. We say that $i \in S$ *leads to* $j \in S$ if

$$\sum_{n \in \mathbb{N}} (P^n)_{ij} \neq 0.$$

Moreover, we say that $i \in S$ is *absorbing* if $P_{ii} = 1$. Finally, we say that the Markov chain $\mathcal{C}$ is *absorbing* if every state leads to some absorbing state.

Definition C.12. Let $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{R}_{[0,1]}, Q, (i,1), \mathbb{F}, \delta, \lambda\big)$ be a stochastic sequential transducer. *The Markov chain* (S, μ, P) *associated with* $\mathcal{T}$ is defined as

$$S \coloneqq Q \cup q_{\$}, \quad \mu_q \coloneqq \begin{cases} 1 & \text{if } q = i \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad P_{pq} \coloneqq \begin{cases} \sum_{a \in \Sigma_{pq}} \lambda(p,a) & \text{if } p \in Q \ \wedge q \in Q \\ \mathbb{F}(p) & \text{if } p \in Q \ \wedge q = q_{\$} \\ 0 & \text{if } p = q_{\$} \wedge q \in Q \\ 1 & \text{if } p = q_{\$} \wedge q = q_{\$} \end{cases},$$

where $q_{\$} \notin Q$ and $\Sigma_{pq} \coloneqq \Big\{ a \in \Sigma \;\Big|\; \big((p,a),q\big) \in \delta \Big\}$ for $p, q \in Q$.

In the following proposition, we state several straightforward correspondences between stochastic sequential transducers and the Markov chains associated with them.

Proposition C.8. *Let $\mathcal{C} \coloneqq (S, \mu, P)$ be the Markov chain associated with the stochastic sequential transducer $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{R}_{[0,1]}, Q, (i,1), \mathbb{F}, \delta, \lambda\big)$. Then,*

(i) $q_{\$}$ is an absorbing state in $\mathcal{C}$;

(ii) a state $q \in Q$ is co-accessible in $\mathcal{T}$ if and only if it leads to $q_{\$}$ in $\mathcal{C}$;

(iii) if a state $q \in Q$ is co-accessible in $\mathcal{T}$, then it is not absorbing in $\mathcal{C}$;

(iv) if every state $q \in Q$ is co-accessible in $\mathcal{T}$, then $\mathcal{C}$ is an absorbing Markov chain;

(v) if $\Sigma^*_{pq} \coloneqq \left\{\alpha \in \Sigma^* \mid \big((p, \alpha), q\big) \in \delta^*\right\}$ for $p, q \in Q$, then

$$(\forall q \in Q)\Big(\sum_{n\in\mathbb{N}}(\mu P^n)_q = \sum_{\alpha\in\Sigma^*_{iq}} \lambda^*(i,\alpha)\Big) \quad \textit{and} \quad \sum_{n\in\mathbb{N}}(\mu P^n)_{q_\$} = \sum_{\alpha\in\Sigma^*} [\![\mathcal{T}]\!](\alpha).$$

Next, we refer to a result from the theory of Markov chains that states that in an absorbing Markov chain the probability of reaching an absorbing state is 1 (Grinstead and Snell, 1997, Theorem 11.3).

Theorem C.3. *Let (S, μ, P) be an absorbing Markov chain and A be the set of its absorbing states. Then,*

$$\sum_{i\in A}\sum_{n\in\mathbb{N}}(\mu P^n)_i = 1.$$

Now, using the established correspondences between stochastic sequential transducers and their associated prefix models (see Appendix C.2) and Markov chains, we can characterise the stochastic sequential transducers that are probabilistic.

Theorem C.4. *A stochastic sequential transducer $\mathcal{T}$ is probabilistic if and only if every accessible state of $\mathcal{T}$ is co-accessible.*

Proof. Let $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{R}_{[0,1]}, Q, (i, 1), \mathbb{F}, \delta, \lambda\big)$ be a stochastic sequential transducer. First, assume that $\mathcal{T}$ is probabilistic and consider the prefix factorisation $\Phi \coloneqq (\phi_\alpha)_{\alpha\in\Sigma^*}$ that is associated with it. Let $q \in Q$ be an accessible state. Then, there exists $\alpha \in \Sigma^*$ such that $\delta^*(i, \alpha) = q$ and $\lambda^*(i, \alpha) \neq 0$. From the correspondence between $\mathcal{T}$ and Φ (see Proposition C.1), it follows that

$$\phi^*_\epsilon(\alpha) = \lambda^*(i, \alpha) \neq 0.$$

Therefore, since the prefix model generated by Φ is the language model $[\![\mathcal{T}]\!]$ (see Proposition C.2), Proposition B.4 implies that

$$\sum_{\beta\in\Sigma^*} \lambda^*(q,\beta)\mathbb{F}\big(\delta^*(q,\beta)\big) = \sum_{\beta\in\Sigma^*} \phi^*_\alpha(\beta)\phi_{\alpha\beta}(\$) = \sum_{\beta\in\Sigma^*} \phi^*_\alpha(\beta\$) = \phi^*_\alpha(\Sigma^*\$) = 1.$$

Thus, there exists $\beta \in \Sigma^*$ such that $\lambda^*(q,\beta)\mathbb{F}\big(\delta^*(q,\beta)\big) \neq 0$; that is, q is co-accessible.

Now, assume that every accessible state of $\mathcal{T}$ is co-accessible. As noted in Remark C.4, we can assume, without loss of generality, that every state of $\mathcal{T}$ is co-accessible. Now, consider the Markov chain $\mathcal{C} \coloneqq (S, \mu, P)$ associated with $\mathcal{T}$. From Proposition C.8, it follows that $\mathcal{C}$ is an absorbing Markov chain and $q_\$$ is its unique absorbing state. Furthermore,

$$\sum_{\alpha\in\Sigma^*} [\![\mathcal{T}]\!](\alpha) \overset{\text{Proposition C.8}}{=} \sum_{n\in\mathbb{N}}(\mu P^n)_{q_\$} \overset{\text{Theorem C.3}}{=} 1;$$

that is, $\mathcal{T}$ is probabilistic. □

C.6 Modelling of State Distributions with Softmax

In this appendix, we continue the discussion of the fact that, in practice, all stochastic sequential language models that use the $\mathrm{softmax}$ activation function to define the transition and final output functions are probabilistic because all of their accessible states are co-accessible.

As already mentioned, unidirectional language models based on saturated RNNs, RNNs using the Heaviside activation function or Transformers with bounded context length are, in fact, stochastic sequential transducers. In practice, the state of such a model at time step t (that is, after processing the prefix $\alpha_{\leq t}$ of the input α) is represented by a d-dimensional vector $h_t \in \mathbb{R}^d$. In order to obtain the probability

distribution p_t over $\Sigma_\$$ that defines the transition and final outputs of h_t, a transformation $\phi\colon \mathbb{R}^d \to \mathbb{R}^{|\Sigma_\$|}$ and the $\mathrm{softmax}$ activation function are applied to h_t; that is,

$$p_t \coloneqq \mathrm{softmax}\big(\phi(h_t)\big).$$

The $\mathrm{softmax}$ activation function is a function from $\mathbb{R}^{|\Sigma_\$|}$ to the probability simplex

$$\left\{x \in [0,1]^{|\Sigma_\$|} \mid \sum_{i=1}^{|\Sigma_\$|} x_i = 1\right\}$$

and is defined, for $x \in \mathbb{R}^{|\Sigma_\$|}$ and $1 \leq i \leq |\Sigma_\$|$, as

$$\mathrm{softmax}(x)_i \coloneqq \frac{\exp(x_i)}{\sum_{j=1}^{|\Sigma_\$|} \exp(x_j)}. \tag{10}$$

From (10), it is obvious that p_t is a positive probability distribution over $\Sigma_\$$. Thus, Theorem 3.2 implies that every stochastic sequential transducer that is implemented in such a way is probabilistic.

C.7 Characterisation of Sequential Language Models

In this appendix, we provide a detailed proof of the characterisation of sequential language models. We consider the more general case of sequential functions from Σ^* to $\mathcal{R}_{[0,1]}$ and show that threy are characterised by several different properties; namely, *uniform finiteness*, *uniform boundedness* and *Lipschitzness*. We begin by recalling a result by Mohri (1997, Theorem 9) that characterises the sequential functions from Σ^* to $\mathcal{S}_{[0,\infty)} \coloneqq \big([0,\infty), +, 0\big)$.[25]

Definition C.13. Let d be a metric on M. A function $f\colon \Sigma^* \to M$ is called *uniformly bounded*[26] *with respect to* d if

$$(\forall n \in \mathbb{N})(\exists N \in \mathbb{N})\big(\forall \alpha, \beta \in \mathrm{Dom}(f)\big)\Big(d_p(\alpha,\beta) \leq n \implies d\big(f(\alpha), f(\beta)\big) \leq N\Big).$$

Theorem C.5. *Let f be a rational function from Σ^* to $\mathcal{S}_{[0,\infty)}$. Then, f is sequential if and only if it is uniformly bounded with respect to the metric*

$$d_{\mathcal{S}}\colon (x,y) \mapsto |x - y|.$$

To transfer this characterisation to the probability monoid $\mathcal{R}_{[0,1]}$, we make several observations. First, we note that it is sufficient to consider only functions from Σ^* to $\mathcal{R}_{(0,1]} \coloneqq \big((0,1], \cdot, 1\big)$.

Proposition C.9. *Let f be a rational function from Σ^* to $\mathcal{R}_{[0,1]}$. Then, f is sequential if and only if $f\restriction_{\mathrm{Supp}(f)}$ if sequential.*

Proof. If f is realised by a sequential transducer, then, by removing the transitions with zero output and making the initial (final) states with zero initial (final) output non-initial (non-final), we obtain a sequential transducer that represents $f\restriction_{\mathrm{Supp}(f)}$. Conversely, if $f\restriction_{\mathrm{Supp}(f)}$ is sequential, we can complete any sequential transducer that realises it in order to obtain a sequential transducer that represents f. □

Next, we note that the negative logarithm is an *isomorphism* from $\mathcal{R}_{(0,1]}$ to $\mathcal{S}_{[0,\infty)}$.

Definition C.14. Let $\mathcal{M}_1 \coloneqq (M_1, \circ_1, e_1)$ and $\mathcal{M}_2 \coloneqq (M_2, \circ_2, e_2)$ be monoids. A function $h\colon M_1 \to M_2$ is a *homomorphism from $\mathcal{M}_1$ to $\mathcal{M}_2$* if

$$h(e_1) = e_2 \quad \text{and} \quad (\forall a, b \in M_1)\big(h(a \circ_1 b) = h(a) \circ_2 h(b)\big).$$

An *isomorphism from $\mathcal{M}_1$ and $\mathcal{M}_2$* is a bijective homomorphism from $\mathcal{M}_1$ to $\mathcal{M}_2$.

[25] We write $\mathcal{S}$ instead of $\mathcal{R}$ in order to emphasise that the monoid operation is addition and not multiplication.

[26] Choffrut (1977), Mohri (1997) and Mihov and Schulz (2019) call such functions 'of bounded variation'. We instead use the terminology of Reutenauer and Schützenberger (1991).

An important property of rational and sequential functions is that they are closed with respect to composition with homomorphisms.

Proposition C.10. *Let f be a rational function from Σ^* to $\mathcal{M}$ and h be a homomorphism from $\mathcal{M}$ to $\mathcal{M}'$. Then, the composition $f \circ h$ is a rational function from Σ^* to $\mathcal{M}'$. Furthermore, $f \circ h$ is sequential whenever f is sequential.*

Proof. Let $\mathcal{T} \coloneqq (\Sigma, \mathcal{M}, Q, \mathbb{I}, \mathbb{F}, \Delta)$ be a transducer that realises f. Consider the transducer

$$h(\mathcal{T}) \coloneqq \Big(\Sigma, \mathcal{M}', Q, \mathbb{I} \circ h, \mathbb{F} \circ h, \Big\{\big(p, a, h(m), q\big) \mid (p, a, m, q) \in \Delta\Big\}\Big).$$

It is easy to verify that $[\![h(\mathcal{T})]\!] = [\![\mathcal{T}]\!] \circ h$. Thus, $f \circ h$ is rational. Moreover, if $\mathcal{T}$ is sequential, $h(\mathcal{T})$ is also sequential. Therefore, $f \circ h$ is sequential whenever f is sequential. □

Additionally, we note that, apart from being and isomorphism, $-\log$ is also an *isometry* from the metric space $(\mathcal{R}_{(0,1]}, d_{\mathcal{R}})$, where

$$d_{\mathcal{R}} \colon (x, y) \mapsto \big|\log(x) - \log(y)\big|,$$

to the metric space $(\mathcal{S}_{[0,\infty)}, d_{\mathcal{S}})$. Thus, uniform boundedness can be transferred between functions from Σ^* to $\mathcal{R}_{(0,1]}$ and functions from Σ^* to $\mathcal{S}_{[0,\infty)}$.

Proposition C.11. *Let f be a rational function from Σ^* to $\mathcal{R}_{(0,1]}$. Then, f is uniformly bounded with respect to $d_{\mathcal{R}}$ if and only if $f \circ (-\log)$ is uniformly bounded with respect to $d_{\mathcal{S}}$.*

Proof. Let $g \coloneqq f \circ (-\log)$; that is, $f = g \circ (-\log)^{-1}$, where $(-\log)^{-1}(x) = \exp(-x)$. It is sufficient to note that, for $\alpha, \beta \in \mathrm{Dom}(f)$,

$$\begin{aligned}
d_{\mathcal{R}}\big(f(\alpha), f(\beta)\big) &= d_{\mathcal{R}}\Big(\big(g \circ (-\log)^{-1}\big)(\alpha), \big(g \circ (-\log)^{-1}\big)(\beta)\Big) \\
&= d_{\mathcal{R}}\Big(\exp\big(-g(\alpha)\big), \exp\big(-g(\beta)\big)\Big) \\
&= \Big|\log\Big(\exp\big(-g(\alpha)\big)\Big) - \log\Big(\exp\big(-g(\beta)\big)\Big)\Big| \\
&= \big|-g(\alpha) + g(\beta)\big| \\
&= d_{\mathcal{S}}\big(g(\alpha), g(\beta)\big).
\end{aligned}$$

□

Now, we can state a characterisation of the sequential functions from Σ^* to $\mathcal{R}_{(0,1]}$.

Theorem C.6. *Let f be a rational function from Σ^* to $\mathcal{R}_{(0,1]}$. Then, f is sequential if and only if it is uniformly bounded with respect to $d_{\mathcal{R}}$.*

Proof. Let $g \coloneqq f \circ (-\log)$. Now, since $-\log$ is an isomorphism, we can conclude that

$$\begin{aligned}
f \text{ is sequential from } \Sigma^* \text{ to } \mathcal{R}_{(0,1]} &\xLeftrightarrow{\textit{Proposition C.10}} g \text{ is sequential from } \Sigma^* \text{ to } \mathcal{S}_{[0,\infty)} \\
&\xLeftrightarrow{\textit{Theorem C.5}} g \text{ is uniformly bounded with respect to } d_{\mathcal{S}} \\
&\xLeftrightarrow{\textit{Proposition C.11}} f \text{ is uniformly bounded with respect to } d_{\mathcal{R}}.
\end{aligned}$$

□

Finally, we note that in Theorem C.6 one can replace the *uniform boundedness* with *Lipschitzness* or *uniform finiteness*.

Definition C.15. Let d be a metric on M. A function $f \colon \Sigma^* \to M$ is called *Lipschitz with respect to d* if and only if

$$(\exists L \in \mathbb{N})\big(\forall \alpha, \beta \in \mathrm{Dom}(f)\big)\Big(d\big(f(\alpha), f(\beta)\big) \leq L \cdot d_p(\alpha, \beta)\Big).$$

Definition C.16. A function $f\colon \Sigma^* \to (0,1]$ is called *uniformly finite* if and only if

$$\left\{ \frac{f(\alpha)}{f(\beta)} \;\middle|\; \alpha,\beta \in \mathrm{Dom}(f) \wedge d_p(\alpha,\beta) \leq n \right\}$$

is finite for all $n \in \mathbb{N}$.

It is obvious that every function that is Lipschitz with respect to d is also uniformly bounded with respect to d. Furthermore, every function that is uniformly finite is also uniformly bounded with respect to $d_{\mathcal{R}}$. In the class of rational functions, the opposite directions also hold.

Theorem C.7. *Let f be a rational function from Σ^* to $\mathcal{R}_{(0,1]}$. Then, the following are equivalent:*

(i) f is sequential;

(ii) f is uniformly bounded with respect to $d_{\mathcal{R}}$;

(iii) f is Lipschitz with respect to $d_{\mathcal{R}}$;

(iv) f is uniformly finite.

Proof. It remains to show that (i) implies (iii) and (iv). To this end, let $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{R}_{(0,1]}, Q, (i,\iota), \mathbb{F}, \delta, \lambda\big)$ be a sequential transducer that realises f.

Let $\alpha,\beta \in \mathrm{Dom}(f)$ and $\gamma,\alpha',\beta' \in \Sigma^*$ be such that $\gamma = \alpha \wedge \beta$, $\alpha = \gamma\alpha'$ and $\beta = \gamma\beta'$. Then,

$$\frac{f(\alpha)}{f(\beta)} = \frac{\cancel{\iota\lambda^*(i,\gamma)}\lambda^*\big(\delta^*(i,\gamma),\alpha'\big)\mathbb{F}\Big(\delta^*\big(\delta^*(i,\gamma),\alpha'\big)\Big)}{\cancel{\iota\lambda^*(i,\gamma)}\lambda^*\big(\delta^*(i,\gamma),\beta'\big)\mathbb{F}\Big(\delta^*\big(\delta^*(i,\gamma),\beta'\big)\Big)}.$$

Now, if $q \coloneqq \delta^*(i,\gamma)$, $N \coloneqq \min\big\{\lambda(q,a) \mid (q,a) \in Q \times \Sigma\big\}$ and $M \coloneqq \min\big\{\mathbb{F}(q) \mid q \in Q\big\}$, we can conclude that

$$\begin{aligned}
d_{\mathcal{R}}\big(f(\alpha),f(\beta)\big) &= \Big|\log\big(f(\alpha)\big) - \log\big(f(\beta)\big)\Big| \\
&= \left|\log \frac{\lambda^*(q,\alpha')\mathbb{F}\big(\delta^*(q,\alpha')\big)}{\lambda^*(q,\beta')\mathbb{F}\big(\delta^*(q,\beta')\big)}\right| \\
&\leq \left|\log \frac{1}{\lambda^*(q,\beta')\mathbb{F}\big(\delta^*(q,\beta')\big)}\right| \\
&\leq |\beta'| \log\frac{1}{N} + \log\frac{1}{M} \\
&\leq d_p(\alpha,\beta)\log\frac{1}{NM};
\end{aligned}$$

that is, f is Lipschitz with respect to $d_{\mathcal{R}}$.

Furthermore, for $n \in \mathbb{N}$, we have that

$$\begin{aligned}
&\left\{ \frac{f(\alpha)}{f(\beta)} \;\middle|\; \alpha,\beta \in \mathrm{Dom}(f) \wedge d_p(\alpha,\beta) \leq n \right\} \\
&\qquad\subseteq \left\{ \frac{f(\gamma\alpha)}{f(\gamma\beta)} \;\middle|\; \gamma \in \Sigma^* \wedge \alpha,\beta \in \Sigma^{\leq n} \wedge \gamma\alpha,\gamma\beta \in \mathrm{Dom}(f) \right\} \\
&\qquad= \left\{ \frac{\lambda^*\big(\delta^*(i,\gamma),\alpha\big)\mathbb{F}\Big(\delta^*\big(\delta^*(i,\gamma),\alpha\big)\Big)}{\lambda^*\big(\delta^*(i,\gamma),\beta\big)\mathbb{F}\Big(\delta^*\big(\delta^*(i,\gamma),\beta\big)\Big)} \;\middle|\; \gamma \in \Sigma^* \wedge \alpha,\beta \in \Sigma^{\leq n} \wedge \gamma\alpha,\gamma\beta \in \mathrm{Dom}(f) \right\} \\
&\qquad\subseteq \left\{ \frac{\lambda^*(q,\alpha)\mathbb{F}\big(\delta^*(q,\alpha)\big)}{\lambda^*(q,\beta)\mathbb{F}\big(\delta^*(q,\beta)\big)} \;\middle|\; q \in Q \wedge \alpha,\beta \in \Sigma^{\leq n} \wedge \delta^*(q,\alpha),\delta^*(q,\beta) \in F \right\}
\end{aligned}$$

is finite because it is contained in a finite set; that is, f is uniformly finite. □

Lastly, we note that Theorem 3.3 is a special case of Theorem C.7 when Proposition C.9 is taken into consideration.

D Rational Language Models

D.1 Examples of Rational Language Models

In this appendix, we give formal proofs of the statements from Section 4.1. In particular,

(i) in Proposition D.1, we prove that, when p_0 and p_1 are distinct, $\overline{\mathbb{P}}(\alpha)$ is a co-sequential but not a sequential language model;

(ii) in Figure 3, we depict a representation of a bisequential decomposition of the language model $\widetilde{\mathbb{P}}(\alpha)$;

(iii) in Figure 4, we depict a stochastic sequential transducer that is equivalent to the sequential transducer $\mathcal{T}_g$ from Figure 3;

(iv) in Proposition D.2 we prove that, when $(p_{ij})_{i,j\in\{0,1\}}$ are pairwise distinct, $\widetilde{\mathbb{P}}(\alpha)$ is neither a sequential nor a co-sequential language model.

Remark D.1. We utilise the following standard graphical representation in order the visualise transducers (Sakarovitch, 2009): states are depicted as circles (inside of which the name of the state may be written), each transition (p, α, m, q) is represented by an arrow from p to q with label $\alpha \mid m$, initial states are identified by an incoming arrow labelled with the corresponding initial output and final states are identified by an outgoing arrow labelled with the corresponding final output.

Proposition D.1. *For $i \in \{0,1\}$, let $p_i \in (0, 0.5)$ and $\mathbb{P}_i$ be a language model over $\{0,1\}$ defined as*

$$\mathbb{P}_i(\alpha) \coloneqq \begin{cases} (1-2p_i)p_i^{|\alpha'|} & \text{if } \alpha = \alpha' i \\ 0 & \text{otherwise} \end{cases}.$$

Let $w \in \mathbb{R} \setminus 0$ and $\overline{\mathbb{P}}\colon \Sigma^* \to \mathbb{R}$ be defined as

$$\overline{\mathbb{P}}(\alpha) \coloneqq w\mathbb{P}_0(\alpha) + (1-w)\mathbb{P}_1(\alpha).$$

Then, if $p_0 \neq p_1$, $\overline{\mathbb{P}}$ is a co-sequential function that is not sequential.

Proof. It is obvious that $\overline{\mathbb{P}}$ is a co-sequential function (see Figure 2). Assume that $p_0 \neq p_1$. Then, for every $\alpha \in \{0,1\}^*$, we have that $d_p(\alpha 0, \alpha 1) = 2$ and

$$\frac{\overline{\mathbb{P}}(\alpha 0)}{\overline{\mathbb{P}}(\alpha 1)} = \frac{w\mathbb{P}_0(\alpha)}{(1-w)\mathbb{P}_1(\alpha)} = \frac{w(1-2p_0)p_0^{|\alpha|}}{(1-w)(1-2p_1)p_1^{|\alpha|}} = \frac{w(1-2p_0)}{(1-w)(1-2p_1)}\left(\frac{p_0}{p_1}\right)^{|\alpha|}.$$

Therefore,

$$\left\{ \frac{w(1-2p_0)}{(1-w)(1-2p_1)}\left(\frac{p_0}{p_1}\right)^n \;\middle|\; n \in \mathbb{N} \right\} \subseteq \left\{ \frac{\overline{\mathbb{P}}(\alpha)}{\overline{\mathbb{P}}(\beta)} \;\middle|\; \alpha, \beta \in \mathrm{Supp}(\overline{\mathbb{P}}) \wedge d_p(\alpha, \beta) \leq 2 \right\}.$$

Since $p_0 \neq p_1$, the former set is infinite and consequently the latter is also infinite. By Theorem C.7, we conclude that $\overline{\mathbb{P}}$ is not sequential. □

Remark D.2. Note that, if $w \in (0,1)$, then $\overline{\mathbb{P}}$ is a language model. Additionally, if the language models $\mathbb{P}_i$ are defined so that they are discriminative with respect to the first instead of the last letter, then $\overline{\mathbb{P}}$ would be sequential but not co-sequential when $p_0 \neq p_1$.

Proposition D.2. *For $i, j \in \{0,1\}$, let $p_{ij} \in (0, 0.5)$ and $\mathbb{P}_{ij}$ be a language model over $\{0,1\}$ defined as*

$$\mathbb{P}_{ij}(\alpha) \coloneqq \begin{cases} (1-2p_{ij})p_{ij}^{|\alpha'|} & \text{if } \alpha = i\alpha' j \\ 0 & \text{otherwise} \end{cases}.$$

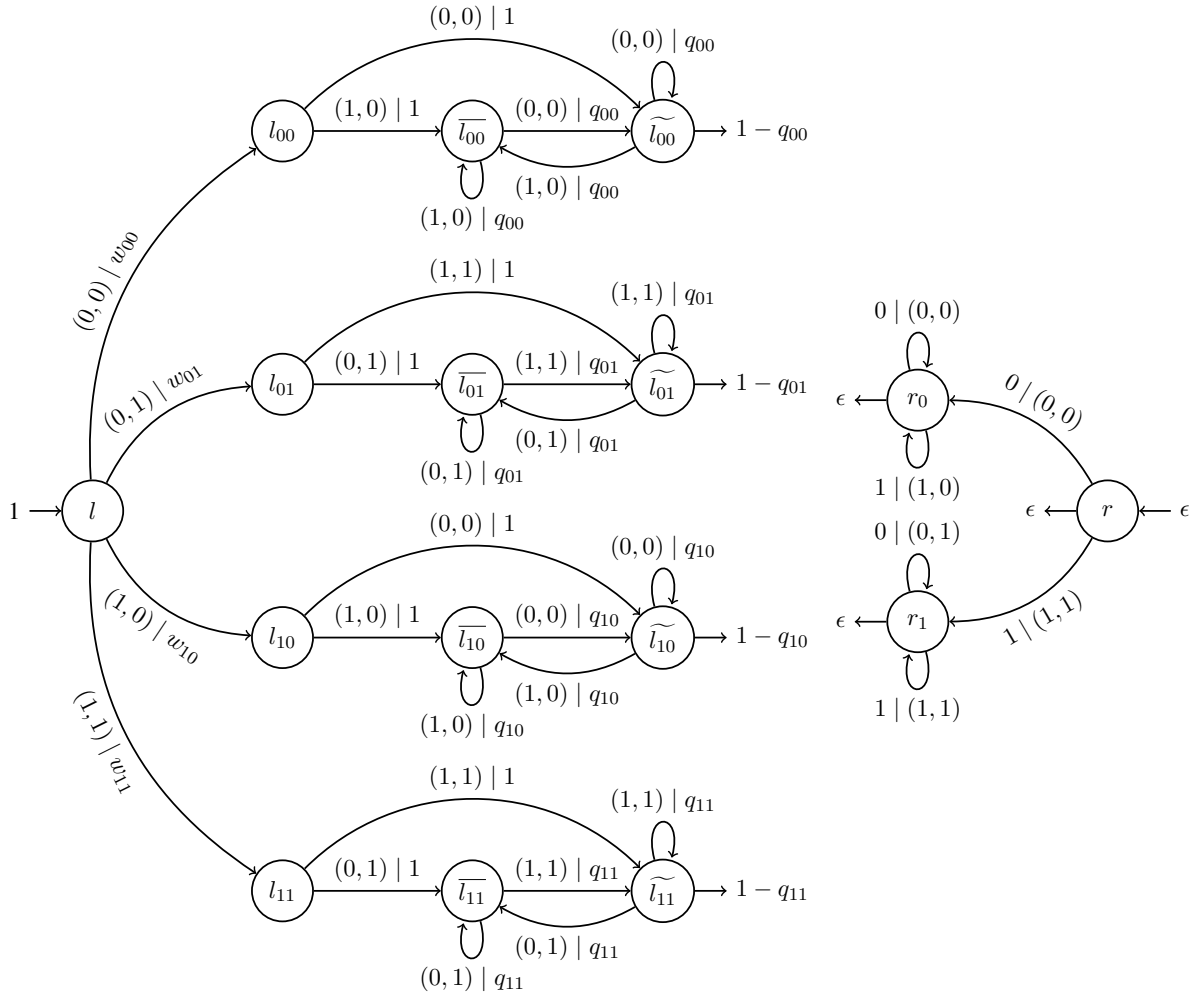

Figure 3: A representation of a standard bisequential decomposition $\big(\{0,1\}^2, \eta, g\big)$ of the language model $\widetilde{\mathbb{P}}$ from Section 4.1. On the right hand side is the sequential transducer $\mathcal{T}_\eta$ that represents the co-sequential function $\eta\colon \{0,1\} \to \{0,1\}^2$ defined as $\eta(\epsilon) \coloneqq \epsilon$ and $\eta(\beta) \coloneqq (\beta_1, j)(\beta_2, j) \cdots (\beta_{|\beta|}, j)$ for $\beta = \alpha j$. On the left hand side is the sequential transducer $\mathcal{T}_g$ that realises the sequential language model g over $\{0,1\}^2$ such that $\eta \circ g = \widetilde{\mathbb{P}}$. Note that $\mathcal{T}_g$ has intentionally not been completed to avoid clutter. Additionally, the sequential transducer $\mathcal{T}_g$ is probabilistic but not stochastic since the transition and final outputs of states l_{ij} sum to two instead of one.

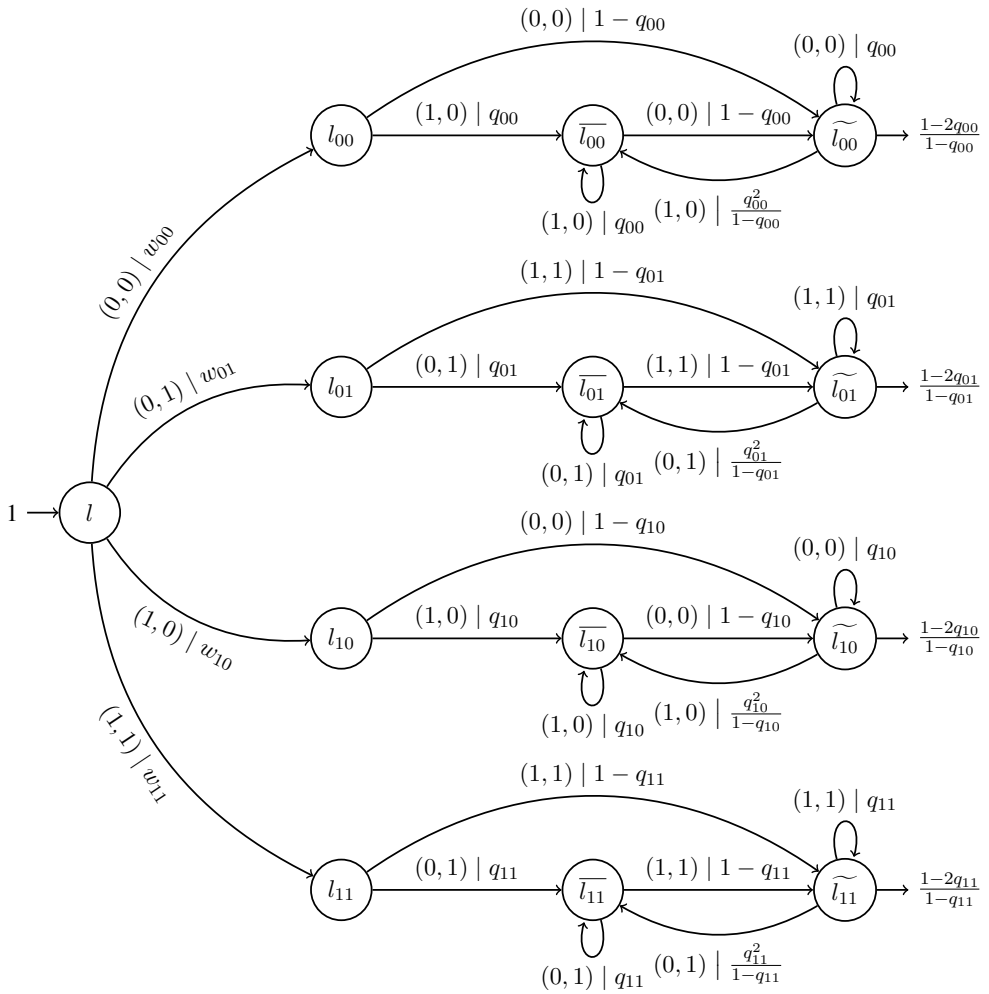

Figure 4: A stochastic version of the transducer $\mathcal{T}_g$ from Figure 3. One can efficiently sample from it and then project onto Σ^*. As explained in Section 4.2, this corresponds to sampling from the language model $\widetilde{\mathbb{P}}$.

Let $w_{ij} \in \mathbb{R} \setminus 0$*, for* $i, j \in \{0,1\}$*, and* $\widetilde{\mathbb{P}} \colon \Sigma^* \to \mathbb{R}$ *be defined as*

$$\widetilde{\mathbb{P}}(\alpha) \coloneqq \sum_{i,j \in \{0,1\}} w_{ij} \mathbb{P}_{ij}(\alpha).$$

Then, $\widetilde{\mathbb{P}}$ *is a rational function that is neither sequential nor co-sequential whenever* $(p_{ij})_{i,j \in \{0,1\}}$ *are pairwise distinct.*

Proof. $\widetilde{\mathbb{P}}$ is obviously a rational function. Assume that $(p_{ij})_{i,j \in \{0,1\}}$ are pairwise distinct. Now, Proposition D.1 implies that

$$\widetilde{\mathbb{P}}{\restriction}_{0\{0,1\}^*\{0,1\}} = w_{00}\mathbb{P}_{00} + w_{01}\mathbb{P}_{01}$$

is not sequential. Similarly, $\widetilde{\mathbb{P}}{\restriction}_{\{0,1\}\{0,1\}^*0}$ is not co-sequential (see Remark D.2). Since sequential and co-sequential functions are closed with respect to regular restrictions, it follows that $\widetilde{\mathbb{P}}$ is neither sequential nor co-sequential. □

Remark D.3. Note that, if $w_{ij} \in (0,1)$, for $i, j \in \{0,1\}$, and $\sum_{i,j \in \{0,1\}} w_{ij} = 1$, then $\widetilde{\mathbb{P}}$ is a language model.

D.2 Conciseness of the Representations of Bisequential Decompositions

In this appendix, we give formal proofs of the statements from Section 4.3. Recall that, for an alphabet Σ and $n \in \mathbb{N}$, we use $\mathcal{P}_{\Sigma,n}$ to denote the class of language models $\mathbb{P}$ over Σ such that

$$\operatorname{Supp}(\mathbb{P}) = \bigcup_{a,b \in \Sigma} a\Sigma^n a \Sigma^* b \Sigma^n b.$$

Theorem D.1. *Every sequential transducer that represents (either sequentially or co-sequentially) a language model from* $\mathcal{P}_{\Sigma,n}$ *has* $\Omega\big(|\Sigma|^n\big)$ *states.*

Proof. Let $\mathbb{P} \in \mathcal{P}_{\Sigma,n}$ and $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{R}_{[0,1]}, Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$ be a sequential transducer that represents $\mathbb{P}$ sequentially (by symmetry, a similar argument can be applied when $\mathcal{T}$ represents $\mathbb{P}$ co-sequentially) . Suppose that $|Q| < |\Sigma|^{n+1}$. Then, for $\pi \in \bigcup_{a \in \Sigma} a\Sigma^n a$, there exist $\alpha, \beta \in \Sigma^{n+1}$ such that

$$\alpha \neq \beta \quad \text{and} \quad \delta^*(i, \pi\alpha) = \delta^*(i, \pi\beta) =: q.$$

Let $\gamma, \alpha', \beta' \in \Sigma^*$ and $a, b \in \Sigma$ be such that

$$\alpha = \gamma a \alpha', \quad \beta = \gamma b \beta' \quad \text{and} \quad a \neq b.$$

Now, note that, for every $\xi \in \Sigma^{|\gamma|}$,

$$\pi\alpha\xi a = \pi\gamma a \alpha' \xi a \in \operatorname{Supp}(\mathbb{P}) \quad \text{and} \quad \pi\beta\xi b = \pi\gamma b \beta' \xi b \in \operatorname{Supp}(\mathbb{P})$$

because $|\alpha'\xi| = |\alpha'\gamma| = n$ and $|\beta'\xi| = |\beta'\gamma| = n$. Similarly, $\pi\alpha\xi b, \pi\beta\xi a \notin \operatorname{Supp}(\mathbb{P})$. From

$$\begin{aligned}
\mathbb{P}(\pi\alpha\xi a) = [\![\mathcal{T}]\!](\pi\alpha\xi a) &= \iota\lambda^*(i, \pi\alpha)\lambda^*(q, \xi a)\mathbb{F}\big(\delta^*(q, \xi a)\big) > 0, \\
\mathbb{P}(\pi\beta\xi b) = [\![\mathcal{T}]\!](\pi\beta\xi b) &= \iota\lambda^*(i, \pi\beta)\lambda^*(q, \xi b)\mathbb{F}\big(\delta^*(q, \xi b)\big) > 0,
\end{aligned}$$

it follows that all the terms above are non-zero. Consequently,

$$\mathbb{P}(\pi\alpha\xi b) = [\![\mathcal{T}]\!](\pi\alpha\xi b) = \iota\lambda^*(i, \pi\alpha)\lambda^*(q, \xi b)\mathbb{F}\big(\delta^*(q, \xi b)\big) > 0,$$

which contradicts with $\pi\alpha\xi b \notin \operatorname{Supp}(\mathbb{P})$. Thus, $|Q| \geq |\Sigma|^{n+1}$. □

Theorem D.2. *There exist (co-)sequential language models in* $\mathcal{P}_{\Sigma,n}$ *that admit a bisequential decomposition with a representation* $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ *such that* $\mathcal{T}_\eta$ *and* $\mathcal{T}_g$ *have* $O\big(n|\Sigma|\big)$ *states.*

Proof. Let $\mathbb{P}$ be a language model over Σ defined, for $a, b \in \Sigma$, $\alpha, \beta \in \Sigma^n$ and $\gamma \in \Sigma^*$, as

$$\mathbb{P}(a\alpha a\gamma b\beta b) \coloneqq \frac{1 - |\Sigma|p}{|\Sigma|^{2(n+1)}} p^{|\gamma|},$$

where $p \in \left(0, \frac{1}{|\Sigma|}\right)$. It is obvious that $\mathbb{P} \in \mathcal{P}_{\Sigma,n}$. Now, consider the sequential transducers $\mathcal{T}_\eta$ and $\mathcal{T}_g$ defined as[27]

$$\mathcal{T}_\eta \coloneqq \Big(\Sigma, \Sigma^*, Q, (i, \epsilon), (f, \epsilon), \delta, \mathrm{id}_{\mathrm{Dom}(\delta)} \circ \pi_\Sigma\Big),$$
$$\mathcal{T}_g \coloneqq \Big(\Sigma, \mathcal{R}_{[0,1]}, Q, (i, 1), \big(f, \tfrac{1-|\Sigma|p}{|\Sigma|^{2(n+1)}}\big), \delta, \mathrm{Dom}(\delta) \times p\Big),$$

where $Q \coloneqq \big(\Sigma \times \{1, 2, \ldots, n+1\}\big) \cup \{i, f\}$ and $\delta \colon Q \times \Sigma \to Q$ is defined as

$$\delta(i, a) \coloneqq (a, 1), \quad \delta(f, a) \coloneqq f \quad \text{and} \quad \delta\big((b, j), a\big) \coloneqq \begin{cases} (b, j+1) & \text{if } j \neq n+1 \\ f & \text{if } (b, j) = (a, n+1) \end{cases}.$$

It is straightforward to verify that $(\Sigma, \mathcal{T}_\eta, \mathcal{T}_g)$ is a representation of a bisequential decomposition of $\mathbb{P}$ such that both $\mathcal{T}_\eta$ and $\mathcal{T}_g$ have $O\big(n|\Sigma|\big)$ states. □

D.3 Closure of Rational Language Models with Respect to Mixing and Regular Conditioning

In this appendix, we prove the closure properties of rational language models stated in Theorem 4.5.

Theorem D.3. *Let $w \in (0, 1)$, $L \subseteq \Sigma^*$ be a regular language and $\mathbb{P}_1, \mathbb{P}_2$ be language models over Σ. Then,*

(i) if $\mathbb{P}_1$ is sequential or co-sequential, then $\mathbb{P}_1$ is rational;

(ii) if $\mathbb{P}_1$ is rational and $\mathbb{P}_1(L) \neq 0$, then the conditional language model $\mathbb{P}_1(\bullet \mid L)$ is rational;

(iii) if $\mathbb{P}_1$ and $\mathbb{P}_2$ are rational with disjoint supports, then so is the mixture $w\mathbb{P}_1 + (1-w)\mathbb{P}_2$.

Proof. (i) Follows by definition.

(ii) Since L is regular, $\mathbb{P}_1 \restriction_L$ is a rational function. Furthermore, $\mathbb{P}_1(L)$ is non-zero and therefore

$$\mathbb{P}_1(\bullet \mid L) = \frac{1}{\mathbb{P}_1(L)} \mathbb{P}_1 \restriction_L$$

is a well-defined rational function.[28]

(iii) It is obvious that $w\mathbb{P}_1$ and $(1-w)\mathbb{P}_2$ are rational functions. Since, $\mathrm{Supp}(\mathbb{P}_1)$ and $\mathrm{Supp}(\mathbb{P}_2)$ are disjoint regular languages,[29] it follows that

$$w\mathbb{P}_1 \restriction_{\mathrm{Supp}(\mathbb{P}_1)}, \quad (1-w)\mathbb{P}_2 \restriction_{\mathrm{Supp}(\mathbb{P}_2)} \quad \text{and} \quad \Big(\Sigma^* \setminus \big(\mathrm{Supp}(\mathbb{P}_1) \cup \mathrm{Supp}(\mathbb{P}_2)\big)\Big) \times 0$$

are rational functions with disjoint domains. Therefore, their union, which coincides with their mixture with parameter w, is a rational function. □

[27] Below, we use π_Σ to denote the projection from $Q \times \Sigma$ to Σ.

[28] Note that, given a transducer for $\mathbb{P}_1$, we can effectively transform it into an unambiguous transducer and use standard matrix operations in order to effectively compute $\mathbb{P}_1(L)$.

[29] Indeed, given a transducer realising $\mathbb{P}_i$, we can remove all transitions with output 0 along them and make all initial (final) states with initial (final) output 0 non-initial (non-final). This would not change the outputs along the runs for words $\alpha \in \mathrm{Supp}(\mathbb{P}_i)$ but the domain of the behaviour of the resulting transducer would be exactly $\mathrm{Supp}(\mathbb{P}_i)$; thus, proving that $\mathrm{Supp}(\mathbb{P}_i)$ is regular.

D.4 Characterisation of Rational Language Models

In this appendix, we prove Theorem 4.6. To this end, we use the notion of a *bimachine* (Schützenberger, 1961; Eilenberg, 1974; Mihov and Schulz, 2019). Similarly to representations of bisequential decompositions, bimachines are deterministic devices that can represent any rational function. Every bimachine consists of two deterministic automata – a *left* and a *right* one – and an *output function*. Just like the encoder of a representation of a bisequential decomposition, the right automaton scans the input from right to left. However, as opposed to the encoder, it does not output any information. Correspondingly, the left automaton scans the input from left to right. Based on the runs of the two automata, the output function produces the output. Thus, the main difference between bimachines and representations of bisequential decompositions is the fact that bimachines treat the left-to-right and right-to-left scans independently and thus symmetrically, which allows for more transparent arguments.

We start by recalling the formal definition of a bimachine and behaviour of a bimachine. Then, we state the equivalence of the expressive power of bimachines and bisequential decompositions. Finally, we focus on the main topic of this section; that is, the proof of Theorem 4.6.

Definition D.1. A $(\Sigma, \mathcal{M})$-*bimachine* is a tuple $(\mathcal{M}, \mathcal{A}_L, \mathcal{A}_R, \psi, \iota)$, where

(i) $\mathcal{M} \coloneqq (M, \circ, e)$ is a monoid;

(ii) $\mathcal{A}_L \coloneqq (\Sigma, Q_L, i_L, \delta_L, Q_L)$ is a deterministic automaton, called *the left automaton*;[30]

(iii) $\mathcal{A}_R \coloneqq (\Sigma, Q_R, i_R, \delta_R, Q_R)$ is a deterministic automaton, called *the right automaton*;

(iv) $\psi \colon Q_L \times \Sigma \times Q_R \to M$ is *the output function*;

(v) $\iota \in M$ is *the initial output*.

Definition D.2. Let $\mathcal{B} \coloneqq (\mathcal{M}, \mathcal{A}_L, \mathcal{A}_R, \psi, \iota)$ be a $(\Sigma, \mathcal{M})$-bimachine. We extend ψ to a function $\psi^* \colon Q_L \times \Sigma^* \times Q_R \to M$ as follows. For states $l \in Q_L$, $r \in Q_R$, word $\alpha \in \Sigma^*$ and $0 \leq i \leq |\alpha|$, let

$$l_i \coloneqq \delta_L^*(l, \alpha_{\leq i}) \quad \text{and} \quad r_{i+1} \coloneqq \delta_R^*\big(r, (\alpha_{\geq i+1})^\top\big).$$

Then, we define

$$\psi^*(l, \alpha, r) \coloneqq \prod_{i=1}^{|\alpha|} \psi(l_{i-1}, \alpha_i, r_{i+1}).$$

Finally, we define the *behaviour of* $\mathcal{B}$ as the function $[\![\mathcal{B}]\!] \colon \Sigma^* \to M$ such that

$$[\![\mathcal{B}]\!](\alpha) \coloneqq \iota \psi^*(i_L, \alpha, i_R).$$

We also say that $\mathcal{B}$ *represents* (or *realises*) $[\![\mathcal{B}]\!]$.

Remark D.4. In the notation of Definition D.2, a simple inductive argument shows that, for $0 \leq i \leq |\alpha|$,

$$\psi^*(l, \alpha, r) = \psi^*(l, \alpha_{\leq i}, r_{i+1}) \psi^*(l_i, \alpha_{\geq i+1}, r).$$

Bimachines represent exactly the set of rational functions (Schützenberger, 1961; Eilenberg, 1974; Mihov and Schulz, 2019). In particular, for language models, we obtain the following.

Theorem D.4. *A language model over* Σ *is rational if and only if it can be represented by a* $(\Sigma, \mathcal{R}_{[0,1]})$*-bimachine.*

Now, we have the necessary formal background to prove Theorem 4.6. We begin with a simple auxiliary proposition.

[30] This notation signifies that Σ is the alphabet, Q_L is the set of states, i_L is the initial state, δ_L is the transition function and all states in Q_L are final.

Definition D.3. Let $\mathbb{P}$ be a language model over Σ. The relation $\mathcal{C}_{\mathbb{P}} \subseteq \mathfrak{P}(\Sigma^*) \times \Sigma^* \times \Sigma^*$ is defined as $(A, \alpha, \beta) \in \mathcal{C}_{\mathbb{P}}$ if and only if

$$\mathbb{P}(A\alpha) = 0 \iff \mathbb{P}(A\beta) = 0 \quad \text{and} \quad \mathbb{P}(A\alpha) \neq 0 \implies \mathbb{P}(\bullet\,\alpha \mid A\alpha) = \mathbb{P}(\bullet\,\beta \mid A\beta).$$

Remark D.5. Note that, for every $A \subseteq \Sigma^*$, the relation $\big\{(\alpha, \beta) \in \Sigma^* \times \Sigma^* \mid (A, \alpha, \beta) \in \mathcal{C}_{\mathbb{P}}\big\}$ is an equivalence relation.

Proposition D.3. *Let $\mathbb{P}$ be a language model over Σ. Then, for every $(A, \alpha, \beta) \in \mathcal{C}_{\mathbb{P}}$, it follows that*

$$(\forall B \subseteq A)\big((B, \alpha, \beta) \in \mathcal{C}_{\mathbb{P}}\big).$$

Proof. Let $(A, \alpha, \beta) \in \mathcal{C}_{\mathbb{P}}$ and $B \subseteq A$. If $\mathbb{P}(B\alpha) \neq 0$ and $\mathbb{P}(B\beta) \neq 0$, then $\mathbb{P}(A\alpha) \neq 0$, $\mathbb{P}(A\beta) \neq 0$ and, for $\gamma \in B$,

$$\begin{aligned}\mathbb{P}(\gamma\alpha \mid B\alpha) &= \frac{\mathbb{P}(\gamma\alpha)}{\mathbb{P}(B\alpha)} = \frac{\mathbb{P}(\gamma\alpha)\mathbb{P}(A\alpha)}{\mathbb{P}(A\alpha)\mathbb{P}(B\alpha)} = \frac{\mathbb{P}(\gamma\alpha \mid A\alpha)}{\mathbb{P}(B\alpha \mid A\alpha)} = \frac{\mathbb{P}(\gamma\alpha \mid A\alpha)}{\sum_{\delta \in \Sigma^*} \mathbb{P}(\delta\alpha \mid A\alpha)} \\ &= \frac{\mathbb{P}(\gamma\beta \mid A\beta)}{\sum_{\delta \in \Sigma^*} \mathbb{P}(\delta\beta \mid A\beta)} = \frac{\mathbb{P}(\gamma\beta \mid A\beta)}{\mathbb{P}(B\beta \mid A\beta)} = \frac{\mathbb{P}(\gamma\beta)\mathbb{P}(A\beta)}{\mathbb{P}(A\beta)\mathbb{P}(B\beta)} = \mathbb{P}(\gamma\beta \mid B\beta).\end{aligned}$$

Furthermore,

$$\begin{aligned}\mathbb{P}(B\alpha) \neq 0 &\iff \mathbb{P}(A\alpha) \neq 0 \wedge \mathbb{P}(B\alpha \mid A\alpha) \neq 0 \\ &\iff \mathbb{P}(A\alpha) \neq 0 \wedge \sum_{\gamma \in B} \mathbb{P}(\gamma\alpha \mid A\alpha) \neq 0 \\ &\iff \mathbb{P}(A\beta) \neq 0 \wedge \sum_{\gamma \in B} \mathbb{P}(\gamma\beta \mid A\beta) \neq 0 \\ &\iff \mathbb{P}(A\beta) \neq 0 \wedge \mathbb{P}(B\beta \mid A\beta) \neq 0 \\ &\iff \mathbb{P}(B\beta) \neq 0.\end{aligned}$$

□

Theorem D.5. *Let $\mathbb{P}$ be a language model over Σ. Then, the following are equivalent:*

(i) *$\mathbb{P}$ is rational;*

(ii) *there is a finite cover of Σ^* with regular languages $\{L_i\}_{i=1}^n$ such that, for $1 \leq i \leq n$, the number of conditional distributions $\{\mathbb{P}(\bullet\,\alpha \mid L_i\alpha)\}_{\alpha \in \Sigma^*}$ is finite;*

(iii) *there is a finite partition of Σ^* into regular languages $\{L_i\}_{i=1}^n$ such that, for $1 \leq i \leq n$, the number of conditional distributions $\{\mathbb{P}(\bullet\,\alpha \mid L_i\alpha)\}_{\alpha \in \Sigma^*}$ is finite;*

(iv) *there is a finite partition of Σ^* into regular languages $\{L_i\}_{i=1}^n$ such that, for $1 \leq i \leq n$, the number of conditional distributions $\{\mathbb{P}(\bullet\,\alpha \mid L_i\alpha)\}_{\alpha \in \Sigma^*}$ is finite and, for every $a \in \Sigma$, there is a unique $1 \leq j \leq n$ such that $L_i a \subseteq L_j$.*

Proof. *(i)* $\implies$ *(ii)* Assume that $\mathbb{P}$ is rational. Then, by Theorem D.4, we have that there exists a bimachine $\mathcal{B} \coloneqq (\mathcal{R}_{[0,1]}, \mathcal{A}_L, \mathcal{A}_R, \psi, \iota)$ that represents $\mathbb{P}$. Let

$$\begin{aligned}\mathcal{A}_L &\coloneqq (\Sigma, Q_L, i_L, \delta_L, Q_L), \\ \mathcal{A}_R &\coloneqq (\Sigma, Q_R, i_R, \delta_R, Q_R).\end{aligned}$$

Without loss of generality, we assume that all states in Q_L and Q_R are accessible. Since the domain of $\mathbb{P}$ is Σ^*, it follows that $\mathcal{A}_L$ and $\mathcal{A}_R$ are complete.

For a state $l \in Q_L$, let L_l be defined as the left language of l with respect to $\mathcal{A}_L$; that is,

$$L_l \coloneqq \big\{\alpha \in \Sigma^* \mid \delta_L^*(i_L, \alpha) = l\big\}.$$

Since $\mathcal{A}_L$ is a complete deterministic automaton over Σ, it follows that $\{L_l\}_{l \in Q_L}$ is a finite partition of Σ^* ($L_l \neq \emptyset$, for $l \in Q_L$, because l is accessible). Furthermore, by Kleene's Theorem, each of the languages L_l is regular. Therefore, $\{L_l\}_{l \in Q_L}$ is a partition and thus a cover of Σ^* with regular languages. To complete the proof of this part of the theorem, it suffices to show that the conditional distributions

$$\big\{\mathbb{P}(\,\bullet\,\alpha \mid L_l\alpha)\big\}_{\alpha \in \Sigma^*}$$

are finitely many for all $l \in Q_L$.

Let $\alpha \in \Sigma^*$ be such that $\mathbb{P}(L_l\alpha) \neq 0$, and let $r \coloneqq \delta_R^*(i_R, \alpha^\top)$. Now, for every $\beta \in L_l$, we have that

$$\mathbb{P}(\beta\alpha) = [\![\mathcal{B}]\!](\beta\alpha) = \iota\psi^*(i_L, \beta\alpha, i_R).$$

Since $\beta \in L_l$, we have that $\delta^*(i_L, \beta) = l$. Therefore, by Remark D.4, we conclude that

$$\mathbb{P}(\beta\alpha) = \iota\psi^*(i_L, \beta\alpha, i_R) = \iota\psi^*(i_L, \beta, r)\psi^*(l, \alpha, i_R).$$

Now it is straightforward to note that, for every $\beta \in L_l$,

$$\mathbb{P}(\beta\alpha \mid L_l\alpha) = \frac{\mathbb{P}(\beta\alpha)}{\mathbb{P}(L_l\alpha)} = \frac{\mathbb{P}(\beta\alpha)}{\sum_{\gamma \in L_l} \mathbb{P}(\gamma\alpha)} = \frac{\cancel{\iota}\psi^*(i_L, \beta, r)\cancel{\psi^*(l, \alpha, i_R)}}{\sum_{\gamma \in L_l} \cancel{\iota}\psi^*(i_L, \gamma, r)\cancel{\psi^*(l, \alpha, i_R)}} = \frac{\psi^*(i_L, \beta, r)}{\sum_{\gamma \in L_l} \psi^*(i_L, \gamma, r)}.$$

Observe that the final expression depends on α only through $r = \delta_R^*(i_R, \alpha^\top)$. It follows that, for every state $l \in Q_L$, the number of distinct distributions $\mathbb{P}(\,\bullet\,\alpha \mid L_l\alpha)$ is at most $|Q_R|$ when α ranges over Σ^*.

(ii) $\implies$ (iii) Assume that $\{L_i\}_{i=1}^n$ is a cover of Σ^* with regular languages such that, for every i, the number of distributions $\mathbb{P}(\,\bullet\,\alpha \mid L_i\alpha)$ is finite when α ranges over Σ^*. We need to show that there is a partition of Σ^* with the same property.

To this end, for a subset $I \subseteq \{1, 2, \ldots, n\}$, we define

$$\mathcal{L}_I \coloneqq \Big(\bigcap_{i \in I} L_i\Big) \cap \Big(\bigcap_{i \notin I} \Sigma^* \setminus L_i\Big).$$

Since the class of regular languages is closed under complement and intersection, it follows that each of the languages $\mathcal{L}_I$ is regular. Next, it should be also clear that, if I and J are distinct subsets of $\{1, 2, \ldots, n\}$, then $\mathcal{L}_I \cap \mathcal{L}_J = \emptyset$. Indeed, if $i \in \mathcal{L}_I \setminus \mathcal{L}_J$, then $\mathcal{L}_I \subseteq L_i$ whereas $\mathcal{L}_J \subseteq \Sigma^* \setminus L_i$. The case, where there is $j \in \mathcal{L}_J \setminus \mathcal{L}_I$, is symmetric; thus, the conclusion follows. Finally, since

$$\bigcup_{I \subseteq \{1,2,\ldots,n\}\colon\, i \in I} \mathcal{L}_I = L_i,$$

it follows that the union of all the languages $\mathcal{L}_I$ is the same as the union of $\{L_i\}_{i=1}^n$.

So far we have proven that the set

$$\mathcal{P} \coloneqq \big\{\mathcal{L}_I \mid I \subseteq \{1, 2, \ldots, n\} \wedge \mathcal{L}_I \neq \emptyset\big\}$$

obeys the following properties:

- $\mathcal{L}_I$ is regular for every $I \subseteq \{1, 2, \ldots, n\}$;
- $\mathcal{L}_I \cap \mathcal{L}_J = \emptyset$ if and only if $I \neq J$;
- $\bigcup \mathcal{P} = \bigcup_{i=1}^n L_i = \Sigma^*$.

Therefore $\mathcal{P}$ is a partition of Σ^* into regular languages.

Let $\mathcal{L}_I \in \mathcal{P}$. Then, $\mathcal{L}_I \neq \emptyset$ and since $\bigcap_{i=1}^n \Sigma^* \setminus L_i = \emptyset$, it follows that there is at least one $1 \leq i \leq n$ such that $i \in I$. Let i be a fixed index with this property. Then, $\mathcal{L}_I \subseteq L_i$ and, for every $\alpha \in \Sigma^*$, $\mathcal{L}_I\alpha \subseteq L_i\alpha$. Now, Proposition D.3 implies that the number of distinct distributions of the form $\big\{\mathbb{P}(\,\bullet\,\alpha \mid \mathcal{L}_I\alpha)\big\}_{\alpha \in \Sigma^*}$ is bounded from above by the number of distinct distributions $\big\{\mathbb{P}(\,\bullet\,\alpha \mid L_i\alpha)\big\}_{\alpha \in \Sigma^*}$.

(iii) $\implies$ *(iv)* Let $\{L_j\}_{j=1}^n$ be a finite partition of Σ^* into regular languages such that, for $1 \leq j \leq n$, there are finitely many distributions $\mathbb{P}(\,\cdot\,\alpha \mid L_j\alpha)$ for $\alpha \in \Sigma^*$. We prove that $\{L_j\}_{j=1}^n$ can be chosen such that, for every $a \in \Sigma$ and $1 \leq j \leq n$, there is a unique $1 \leq k \leq n$ with $L_j a \subseteq L_k$.

To this end, we first use that L_j is regular and thus, by Kleene's Theorem, there is a complete deterministic automaton $\mathcal{A}_j \coloneqq (\Sigma, Q_j, i_j, \delta_j, F_j)$ that recognises L_j. Next, using the cartesian product construction, we obtain a complete deterministic automaton $\mathcal{A} \coloneqq (\Sigma, Q, i, \delta, Q)$, where

(i) $Q \coloneqq \prod_{j=1}^n Q_j$;

(ii) $i \coloneqq (i_j)_{j=1}^n$;

(iii) $\delta \coloneqq \left\{ \left(\left((q_j)_{j=1}^n, a\right), \left(\delta_j(q_j, a)\right)_{j=1}^n \right) \;\middle|\; q \in Q \wedge a \in \Sigma \right\}$.

We consider the set of all accessible states Q' of $\mathcal{A}$. For each such state $q \in Q'$, we define

$$L'_q \coloneqq \left\{\alpha \in \Sigma^* \mid \delta^*(i, \alpha) = q\right\}.$$

As above, $\{L'_q\}_{q \in Q'}$ forms a finite partition of Σ^* into regular languages. Furthermore, it is straightforward that, for every $q \in Q'$ and $a \in \Sigma$,

$$L'_q a = \left\{\alpha a \mid \alpha \in \Sigma^* \wedge \delta^*(i, \alpha) = q\right\} \subseteq \left\{\alpha \in \Sigma^* \mid \delta^*(i, \alpha) = \delta(q, a)\right\} = L'_{\delta(q,a)}.$$

Now, $\{L'_q\}_{q \in Q'}$ is a partition and thus there is no other state $p \in Q'$ such that $L'_q a \subseteq L'_p$.

Finally, by the construction of $\mathcal{A}$, we have that, for every $\gamma \in \Sigma^*$,

$$\delta^*(i, \gamma) = \left(\delta_j^*(i_j, \gamma)\right)_{j=1}^n.$$

Since $\{L_j\}_{j=1}^n$ is a partition of Σ^* and $\mathcal{A}_j$ recognises L_j for $1 \leq j \leq n$, it follows that, for every $q \in Q'$, there is a unique $1 \leq j \leq n$ such that $q_j \in F_j$ and $L'_q \subseteq L_j$. Now, Proposition D.3 implies that the number of distinct distributions of the form $\left\{\mathbb{P}(\,\cdot\,\alpha \mid L'_q\alpha)\right\}_{\alpha \in \Sigma^*}$ is bounded from above by the number of distinct distributions $\left\{\mathbb{P}(\,\cdot\,\alpha \mid L_j\alpha)\right\}_{\alpha \in \Sigma^*}$.

(iv) $\implies$ *(i)* Let $\{L_i\}_{i=1}^n$ be a partition of Σ^* into regular languages with the property that, for every $1 \leq i \leq n$ and $a \in \Sigma$, there is a unique $1 \leq j \leq n$ with $L_i a \subseteq L_j$. Assume further that, for $1 \leq i \leq n$, the number of distributions $\mathbb{P}(\,\cdot\,\alpha \mid L_i\alpha)$ is finite when α ranges over Σ^*. In what follows, we set out to construct a representation $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ of a bisequential decomposition (Γ, η, g) of $\mathbb{P}$.

We start by constructing the encoding transducer $\mathcal{T}_\eta$. To this end, we study the relation $\sim \subseteq \Sigma^* \times \Sigma^*$ defined as

$$\alpha \sim \beta \iff (\forall 1 \leq i \leq n)\big((L_i, \alpha, \beta) \in \mathcal{C}_{\mathbb{P}}\big).$$

In other words, $\alpha \sim \beta$ expresses the property that α and β have the same conditional distributions with respect to every L_i.

It is straightforward to note that $\sim$ is an equivalence relation (see Remark D.5) and, since, for every $1 \leq i \leq n$, there are finitely many distributions of the form $\mathbb{P}(\,\cdot\,\alpha \mid L_i\alpha)$, it follows that $\sim$ has a finite index. By Proposition D.3 and the fact that, for every $1 \leq i \leq n$ and $a \in \Sigma$, there is a unique $1 \leq j \leq n$ with $L_i a \subseteq L_j$, it follows that $\sim$ is a left congruence (see Definition D.5). Thus, we can encode the equivalence classes of $\sim$ as a right-to-left scanning deterministic automaton.

We define the sequential transducer

$$\mathcal{T}_\eta \coloneqq \Big(\Sigma, \big(\Sigma \times (\Sigma^*/\!\sim)\big)^*, \Sigma^*/\!\sim, \big([\epsilon]_\sim, \epsilon\big), (\Sigma^*/\!\sim) \times \epsilon, \delta_\eta, \lambda_\eta\Big), \quad \text{where}$$

(i) $\delta_\eta \coloneqq \left\{ \Big(\big([\alpha]_\sim, a\big), [a\alpha]_\sim \Big) \;\middle|\; \alpha \in \Sigma^* \wedge a \in \Sigma \right\}$;

(ii) $\lambda_\eta \coloneqq \left\{ \Big(\big([\alpha]_\sim, a\big), \big(a, [\alpha]_\sim\big) \Big) \;\middle|\; \alpha \in \Sigma^* \wedge a \in \Sigma \right\}$.

A simple inductive argument reveals that, for every word $\alpha \in \Sigma^*$,

$$\delta_\eta^*\big([\epsilon]_\sim, \alpha^\top\big) = [\alpha]_\sim.$$

Consequently, using that $\lambda_\eta\big([\alpha]_\sim, a\big) = \big(a, [a\alpha]_\sim\big)$ and taking into account that initial and final outputs of $\mathcal{T}_\eta$ are ϵ, we obtain that

$$\eta(\alpha) = [\![\mathcal{T}_\eta]\!](\alpha^\top)^\top = \Big(\big(\alpha_i, [\alpha_{>i}]_\sim\big)\Big)_{i=1}^{|\alpha|}.$$

In other words, η preserves the input word in the first coordinate by outputting the letters α_i, whereas in the second coordinate it encodes the equivalence class $[\alpha_{>i}]_\sim$. This knowledge, along with the properties of the languages $\{L_i\}_{i=1}^n$, enables the construction of the generator $\mathcal{T}_g$.

Let L_ϵ is be element of $\{L_i\}_{i=1}^n$ that contains ϵ. Then, we define

$$\mathcal{T}_g \coloneqq \big(\Sigma \times (\Sigma^*/\!\sim), \mathcal{R}_{[0,1]}, i_g \cup \{L_i\}_{i=1}^n, (i_g, 1), \mathbb{F}_g, \delta_g, \lambda_g\big), \quad \text{where}$$

$$\mathbb{F}(q) \coloneqq \begin{cases} \mathbb{P}(\epsilon) & \text{if } q = i_g \\ \mathbb{P}(L_i) & \text{if } q = L_i \end{cases},$$

$$\delta_g\Big(q, \big(a, [\alpha]_\sim\big)\Big) \coloneqq \begin{cases} \delta_g\Big(L_\epsilon, \big(a, [\alpha]_\sim\big)\Big) & \text{if } q = i_g \\ L_j & \text{if } q = L_i \wedge L_j a \subseteq L_j \end{cases},$$

$$\lambda_g\Big(q, \big(a, [\alpha]_\sim\big)\Big) \coloneqq \begin{cases} 0 & \text{if } q = i_g \ \wedge L_\epsilon a \subseteq L_j \wedge \mathbb{P}(L_j\alpha) = 0 \\ \mathbb{P}(a\alpha \mid L_j\alpha) & \text{if } q = i_g \ \wedge L_\epsilon a \subseteq L_j \wedge \mathbb{P}(L_j\alpha) \neq 0 \\ 0 & \text{if } q = L_i \wedge L_i a \subseteq L_j \wedge \mathbb{P}(L_j\alpha) = 0 \\ \mathbb{P}(L_i a\alpha \mid L_j\alpha) & \text{if } q = L_i \wedge L_i a \subseteq L_j \wedge \mathbb{P}(L_j\alpha) \neq 0 \end{cases}.$$

Since, for every $1 \leq i \leq n$ and every letter $a \in \Sigma$, there is unique $1 \leq j \leq n$ such that $L_i a \subseteq L_j$, it follows that δ_g and λ_g are well-defined total functions. Let $\alpha \in \Sigma^* \setminus \epsilon$ and denote by $\rho(i)$ the unique index such that $\alpha_{\leq i} \in L_{\rho(i)}$ for $1 \leq i \leq |\alpha|$. Now, a straightforward induction on $1 \leq i \leq |\alpha|$ shows that

$$\delta_g^*\Big(i_g, \big(\eta(\alpha)\big)_{\leq i}\Big) = \delta_g^*\Big(i_g, \big(\alpha_1, [\alpha_{>1}]_\sim\big)\big(\alpha_2, [\alpha_{>2}]_\sim\big)\cdots\big(\alpha_i, [\alpha_{>i}]_\sim\big)\Big) = L_{\rho(i)}.$$

Therefore,

$$\lambda_g^*\big(i_g, \eta(\alpha)\big) = \lambda\Big(i_g, \big(\alpha_1, [\alpha_{>1}]_\sim\big)\Big)\prod_{i=1}^{|\alpha|-1}\lambda\Big(L_{\rho(i)}, \big(\alpha_{i+1}, [\alpha_{>i+1}]_\sim\big)\Big). \tag{11}$$

Note that, if $\mathbb{P}(L_{\rho(j)}\alpha_{>i}) = 0$ for some $1 \leq i \leq |\alpha|$, then, since $\alpha \in L_{\rho(i)}\alpha_{>i}$, it follows that $\mathbb{P}(\alpha) = 0$. These considerations show that, if some of the values in (11) is zero due to the case $\mathbb{P}(L_{\rho(i)}\alpha_{>i}) = 0$, then $[\![\mathcal{T}_g]\!]\big(\eta(\alpha)\big) = 0 = \mathbb{P}(\alpha)$ as required.

Next, we assume that $\mathbb{P}(L_{\rho(i)}\alpha_{>i}) \neq 0$ for all $1 \leq i \leq |\alpha|$. Therefore, for $1 \leq i \leq |\alpha| - 1$,

$$\lambda\Big(i_g, \big(\alpha_1, [\alpha_{>1}]_\sim\big)\Big) = \mathbb{P}(\alpha_1\alpha_{>1} \mid L_{\rho(1)}\alpha_{>1}) = \frac{\mathbb{P}(\alpha_1\alpha_{>1})}{\mathbb{P}(L_{\rho(1)}\alpha_{>1})} = \frac{\mathbb{P}(\alpha)}{\mathbb{P}(L_{\rho(1)}\alpha_{>1})},$$

$$\lambda\Big(L_{\rho(i)}, \big(\alpha_{i+1}, [\alpha_{>i+1}]_\sim\big)\Big) = \mathbb{P}(L_{\rho(i)}\alpha_{i+1}\alpha_{>i+1} \mid L_{\rho(i+1)}\alpha_{>i+1}) = \frac{\mathbb{P}(L_{\rho(i)}\alpha_{>i})}{\mathbb{P}(L_{\rho(i+1)}\alpha_{>i+1})}.$$

Now it is straightforward to verify that

$$\begin{aligned}\lambda_g^*\big(i_g, \eta(\alpha)\big) &= \lambda\Big(i_g, \big(\alpha_1, [\alpha_{>1}]_\sim\big)\Big)\prod_{i=1}^{|\alpha|-1}\lambda\Big(L_{\rho(i)}, \big(\alpha_{i+1}, [\alpha_{>i+1}]_\sim\big)\Big) \\ &= \frac{\mathbb{P}(\alpha)}{\mathbb{P}(L_{\rho(1)}\alpha_{>1})}\prod_{i=1}^{|\alpha|-1}\frac{\mathbb{P}(L_{\rho(i)}\alpha_{>i})}{\mathbb{P}(L_{\rho(i+1)}\alpha_{>i+1})} = \frac{\mathbb{P}(\alpha)}{\mathbb{P}(L_{\rho(|\alpha|)}\alpha_{>|\alpha|})} = \frac{\mathbb{P}(\alpha)}{\mathbb{P}(L_{\rho(|\alpha|)})}.\end{aligned}$$

Finally, since $\mathbb{F}_g(L_{\rho(|\alpha|)}) = \mathbb{P}(L_{\rho(|\alpha|)})$, we conclude that

$$[\![\mathcal{T}_g]\!]\big(\eta(\alpha)\big) = \lambda_g^*\big(i_g, \eta(\alpha)\big)\mathbb{F}_g(L_{\rho(|\alpha|)}) = \frac{\mathbb{P}(\alpha)}{\mathbb{P}(L_{\rho(|\alpha|)})}\mathbb{P}(L_{\rho(|\alpha|)}) = \mathbb{P}(\alpha).$$

So far we considered the case where $|\alpha| > 0$. In the case where $\alpha = \epsilon$, we have

$$[\![\mathcal{T}_g]\!]\big(\eta(\epsilon)\big) = [\![\mathcal{T}_g]\!](\epsilon) = \mathbb{F}_g(L_\epsilon) = \mathbb{P}(\epsilon).$$

This readily shows that $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ is a representation of a bisequential decomposition (Γ, g, η) for $\mathbb{P}$, which completes the proof. □

D.5 Minimal Co-sequential Lookahead of Rational Language Models

In this appendix, we describe the minimal co-sequential lookahead that is needed in order to represent a rational language model. It should be noted that the results in this appendix are stated more generally; that is, for functions from Σ^* to $[0, 1]$ and not specifically for language models. However, the results hold only for positive-valued functions $f\colon \Sigma^* \to (0, 1]$ (in particular, positive language models) and representations $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ of bisequential decompositions of f such that $\mathrm{Dom}\big([\![\mathcal{T}_\eta]\!]\big) = \mathrm{Dom}(f)^\top$. The condition placed on the representations is non-restrictive; thus, in what follows, we shall implicitly assume that every representation of a bisequential decomposition satisfies the above-mentioned property. Furthermore, given a representation $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ of a bisequential decomposition, we shall assume that

$$\mathcal{T}_\eta \coloneqq \big(\Sigma, \Gamma^*, Q_\eta, (i_\eta, \iota_\eta), \mathbb{F}_\eta, \delta_\eta, \lambda_\eta\big) \quad \text{and} \quad \mathcal{T}_g \coloneqq \big(\Gamma, \mathcal{R}_{(0,1]}, Q_g, (i_g, \iota_g), \mathbb{F}_g, \delta_g, \lambda_g\big).$$

We begin by reviewing the notions of quotient and congruence from the theory of automata and formal languages (Eilenberg, 1974; Sakarovitch, 2009).

Definition D.4. Let $\alpha \in \Sigma^*$ and $L \subseteq \Sigma^*$. Then, *the left quotient of L by α*, denoted $\alpha^{-1}L$, is defined as

$$\alpha^{-1}L \coloneqq \{\beta \in \Sigma^* \mid \alpha\beta \in L\}.$$

Similarly, *the right quotient of L by α*, denoted $L\alpha^{-1}$, is defined as

$$L\alpha^{-1} \coloneqq \{\beta \in \Sigma^* \mid \beta\alpha \in L\}.$$

Proposition D.4. *Let $\alpha, \beta \in \Sigma^*$ and $L \subseteq \Sigma^*$. Then,*

$$(\alpha\beta)^{-1}L = \beta^{-1}(\alpha^{-1}L) \quad \textit{and} \quad L(\alpha\beta)^{-1} = (L\beta^{-1})\alpha^{-1}.$$

Definition D.5. Let $\mathcal{M} \coloneqq (M, \circ, e)$ be a monoid and $\equiv\, \subseteq M \times M$ be an equivalence relation on M. Then, $\equiv$ is called a

(i) *left congruence on $\mathcal{M}$* if $a \equiv b$ implies $(\forall c \in M)(c \circ a \equiv c \circ b)$;

(ii) *right congruence on $\mathcal{M}$* if $a \equiv b$ implies $(\forall c \in M)(a \circ c \equiv b \circ c)$.

Next, we recall that the transition function of a sequential transducer (or a deterministic automaton) defines a left and a right congruence (Eilenberg, 1974; Sakarovitch, 2009).

Definition D.6. Let $\mathcal{T} \coloneqq \big(\Sigma, \mathcal{M}, Q, (i, \iota), \mathbb{F}, \delta, \lambda\big)$ be a sequential transducer. *The right transition congruence of $\mathcal{T}$* is the relation $\hookrightarrow_{\mathcal{T}} \subseteq \Sigma^* \times \Sigma^*$ defined as

$$\alpha \hookrightarrow_{\mathcal{T}} \beta \iff \delta^*(i, \alpha) = \delta^*(i, \beta).$$

The left transition congruence of $\mathcal{T}$ is the relation $\hookleftarrow_{\mathcal{T}} \subseteq \Sigma^* \times \Sigma^*$ defined as

$$\alpha \hookleftarrow_{\mathcal{T}} \beta \iff \delta^*(i, \alpha^\top) = \delta^*(i, \beta^\top).$$

Proposition D.5. *Let $\mathcal{T}$ be a sequential $(\Sigma, \mathcal{M})$-transducer. Then, $\hookrightarrow_{\mathcal{T}}$ is a right congruence on Σ^* and $\hookleftarrow_{\mathcal{T}}$ is a left congruence on Σ^*.*

Proof. Follows directly from Definition D.6. □

Now, we can view the minimal co-sequential lookahead of a positive rational language model $\mathbb{P}$ as the index of a left congruence that is canonically associated with $\mathbb{P}$ (Reutenauer and Schützenberger, 1991).

Definition D.7. Let $f\colon \Sigma^* \to (0, 1]$. *The syntactic left congruence of f* is the relation $\equiv_f \subseteq \Sigma^* \times \Sigma^*$ defined as[31]

$$\alpha \equiv_f \beta \iff \operatorname{Dom}(f)\alpha^{-1} = \operatorname{Dom}(f)\beta^{-1} \wedge \left\{ \frac{f(\gamma\alpha)}{f(\gamma\beta)} \;\middle|\; \gamma \in \operatorname{Dom}(f)\alpha^{-1} \right\} \text{ is finite.}$$

Proposition D.6. *Let $f\colon \Sigma^* \to (0, 1]$. Then, $\equiv_f$ is a left congruence on Σ^*.*

Proof. $\equiv_f$ is obviously reflexive and symmetric. To see that it is also transitive, let $\alpha \equiv_f \beta$ and $\beta \equiv_f \delta$. Then, we observe that

$$\operatorname{Dom}(f)\alpha^{-1} = \operatorname{Dom}(f)\beta^{-1} = \operatorname{Dom}(f)\delta^{-1}$$

and the set

$$\begin{aligned}
&\left\{ \frac{f(\gamma\alpha)}{f(\gamma\beta)} \;\middle|\; \gamma \in \operatorname{Dom}(f)\alpha^{-1} \right\} \\
&\qquad = \left\{ \frac{f(\gamma\alpha)}{f(\gamma\beta)} \frac{f(\gamma\beta)}{f(\gamma\delta)} \;\middle|\; \gamma \in \operatorname{Dom}(f)\alpha^{-1} \right\} \\
&\qquad \subseteq \left\{ x \cdot y \;\middle|\; x \in \left\{ \frac{f(\gamma\alpha)}{f(\gamma\beta)} \;\middle|\; \gamma \in \operatorname{Dom}(f)\alpha^{-1} \right\} \wedge y \in \left\{ \frac{f(\gamma\beta)}{f(\gamma\delta)} \;\middle|\; \gamma \in \operatorname{Dom}(f)\beta^{-1} \right\} \right\}
\end{aligned}$$

is finite because it is contained in a finite set. Finally, to prove that $\equiv_f$ is a left congruence, let $\alpha \equiv_f \beta$ and $\delta \in \Sigma^*$. Then, we have that

$$\operatorname{Dom}(f)(\delta\alpha)^{-1} = \big(\operatorname{Dom}(f)\alpha^{-1}\big)\delta^{-1} = \big(\operatorname{Dom}(f)\beta^{-1}\big)\delta^{-1} = \operatorname{Dom}(f)(\delta\beta)^{-1}$$

and the set

$$\left\{ \frac{f(\gamma\delta\alpha)}{f(\gamma\delta\beta)} \;\middle|\; \gamma \in \operatorname{Dom}(f)(\delta\alpha)^{-1} \right\} \subseteq \left\{ \frac{f(\gamma\alpha)}{f(\gamma\beta)} \;\middle|\; \gamma \in \operatorname{Dom}(f)\alpha^{-1} \right\}$$

is finite since it is contained in a finite set. □

Proposition D.7. *Let $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ be a realisation of a bisequential decomposition of $f\colon \Sigma^* \to (0, 1]$. Then, $\hookleftarrow_{\mathcal{T}_\eta} \subseteq \equiv_f$.*

Proof. Let $\alpha \hookleftarrow_{\mathcal{T}_\eta} \beta$ and $q_\eta \coloneqq \delta_\eta^*(i_\eta, \alpha^\top)$. Then, for $\gamma \in \Sigma^*$,

$$\begin{aligned}
\gamma\alpha \in \operatorname{Dom}(f) &\iff \alpha^\top\gamma^\top \in \operatorname{Dom}\big([\![\mathcal{T}_\eta]\!]\big) \\
&\iff \delta_\eta^*\big(q_\eta, \gamma^\top\big) \in F_\eta \\
&\iff \beta^\top\gamma^\top \in \operatorname{Dom}\big([\![\mathcal{T}_\eta]\!]\big) \\
&\iff \gamma\beta \in \operatorname{Dom}(f).
\end{aligned}$$

Now, let $\gamma \in \operatorname{Dom}(f)\alpha^{-1}$. Then, if

$$\phi \coloneqq \iota_\eta\lambda_\eta^*(i_\eta, \alpha^\top), \quad \psi \coloneqq \iota_\eta\lambda_\eta^*(i_\eta, \beta^\top), \quad \tau \coloneqq \lambda_\eta^*(q_\eta, \gamma^\top)\mathbb{F}_\eta\big(\delta_\eta^*(q_\eta, \gamma^\top)\big) \quad \text{and} \quad q_g \coloneqq \delta_g^*(i_g, \tau^\top),$$

it follows that

$$\frac{f(\gamma\alpha)}{f(\gamma\beta)} = \frac{[\![\mathcal{T}_g]\!]\big([\![\mathcal{T}_\eta]\!](\alpha^\top\gamma^\top)^\top\big)}{[\![\mathcal{T}_g]\!]\big([\![\mathcal{T}_\eta]\!](\beta^\top\gamma^\top)^\top\big)} = \frac{[\![\mathcal{T}_g]\!]\big((\phi\tau)^\top\big)}{[\![\mathcal{T}_g]\!]\big((\psi\tau)^\top\big)} = \frac{\cancel{\iota_g\lambda_g^*(i_g, \tau^\top)}\lambda_g^*(q_g, \phi^\top)\mathbb{F}\big(\delta_g^*(q_g, \phi^\top)\big)}{\cancel{\iota_g\lambda_g^*(i_g, \tau^\top)}\lambda_g^*(q_g, \psi^\top)\mathbb{F}\big(\delta_g^*(q_g, \psi^\top)\big)}$$

[31] Note that, if f is a language model (i.e., a total function), then $\operatorname{Dom}(f)\alpha^{-1} = \Sigma^*$ for every $\alpha \in \Sigma^*$.

and therefore the set

$$\left\{ \frac{f(\gamma\alpha)}{f(\gamma\beta)} \;\middle|\; \gamma \in \mathrm{Dom}(f)\alpha^{-1} \right\} \subseteq \left\{ \frac{\lambda_g^*(q_g, \phi^\top)\mathbb{F}\big(\delta_g^*(q_g, \phi^\top)\big)}{\lambda_g^*(q_g, \psi^\top)\mathbb{F}\big(\delta_g^*(q_g, \psi^\top)\big)} \;\middle|\; q_g \in Q_g \right\}$$

is finite because it is contained in a finite set. □

Proposition D.8. *Let f be a rational function from Σ^* to $\mathcal{R}_{(0,1]}$. Then, for every left congruence $\approx \subseteq \equiv_f$, there exists a realisation $(\Gamma, \mathcal{T}_\eta, \mathcal{T}_g)$ of a bisequential decomposition of f such that $\leftrightarrow_{\mathcal{T}_\eta} = \approx$.*

Proof. Since f is rational, it admits a bisequential decomposition and thus, by Proposition D.7, $\equiv_f$ has a finite index. Now, consider the sequential transducer

$$\mathcal{T}_\eta \coloneqq \Big(\Sigma, (\Sigma \times Q_\eta)^*, Q_\eta, \big([\epsilon]_\approx, \epsilon\big), \mathbb{F}_\eta, \delta_\eta, \lambda_\eta\Big), \quad \text{where}$$

(i) $Q_\eta \coloneqq \Sigma^*/\!\approx$;

(ii) $\mathbb{F}_\eta \coloneqq \Big\{ \big([\alpha]_\approx, \epsilon\big) \mid \alpha \in \mathrm{Dom}(f) \Big\}$;

(iii) $\delta_\eta \coloneqq \Big\{ \Big(\big([\alpha]_\approx, a\big), [a\alpha]_\approx \Big) \;\Big|\; \alpha \in \Sigma^* \wedge a \in \Sigma \Big\}$;

(iv) $\lambda_\eta \coloneqq \Big\{ \Big(\big([\alpha]_\approx, a\big), \big(a, [\alpha]_\approx\big) \Big) \;\Big|\; \alpha \in \Sigma^* \wedge a \in \Sigma \Big\}$.

It is obvious that $\leftrightarrow_{\mathcal{T}_\eta} = \approx$ and, for $\alpha \in \Sigma^*$,

$$[\![\mathcal{T}_\eta]\!](\alpha^\top)^\top = \Big(\big(\alpha_i, [\alpha_{>i}]_\approx\big)\Big)_{i=1}^{|\alpha|}.$$

Let g be the co-sequential function

$$\Sigma^* \to (\Sigma \times Q_\eta)^* \colon \alpha \mapsto [\![\mathcal{T}_\eta]\!](\alpha^\top)^\top$$

and $h \coloneqq g^{-1} \circ f$. Then, $f = g \circ h$ and, since g is injective, h is a function from $(\Sigma \times Q_\eta)^*$ to $(0, 1]$. It remains to show that h is sequential. However, h is rational because it is the composition of two rational functions. Thus, it is sufficient to demonstrate that h is uniformly finite.

Let $n \in \mathbb{N}$ and define

$$A \coloneqq \left\{ \frac{h(\alpha)}{h(\beta)} \;\middle|\; \alpha, \beta \in \mathrm{Dom}(h) \wedge d_p(\alpha, \beta) \leq n \wedge |\alpha| + |\beta| \leq n \right\},$$
$$B \coloneqq \left\{ \frac{h(\alpha)}{h(\beta)} \;\middle|\; \alpha, \beta \in \mathrm{Dom}(h) \wedge d_p(\alpha, \beta) \leq n \wedge |\alpha| + |\beta| > n \right\}.$$

Then, we have that

$$\left\{ \frac{h(\alpha)}{h(\beta)} \;\middle|\; \alpha, \beta \in \mathrm{Dom}(h) \wedge d_p(\alpha, \beta) \leq n \right\} = A \cup B.$$

The set A is obviously finite. To show that B is finite, consider $\alpha, \beta \in \mathrm{Dom}(h)$ such that $d_p(\alpha, \beta) \leq n$ and $|\alpha| + |\beta| > n$. Let $\gamma \coloneqq \alpha \wedge \beta$, $\alpha' \coloneqq \gamma^{-1}\alpha$ and $\beta' \coloneqq \gamma^{-1}\beta$. Then, $d_p(\alpha, \beta) = |\alpha'| + |\beta'| \leq n$ and $\gamma \neq \epsilon$, which means that $(\alpha', \beta') \in \approx \subseteq \equiv_f$. Therefore,

$$B \subseteq \bigcup_{\substack{\alpha, \beta \in \Sigma^n \colon \\ \alpha \equiv_f \beta}} \left\{ \frac{h(\gamma\alpha)}{h(\gamma\beta)} \;\middle|\; \gamma \in \mathrm{Dom}(h)\alpha^{-1} \right\}$$

is finite because it is contained in a finite union of finite sets. □

We summarise the obtained results in the following theorem, which is a generalisation of Theorem 4.7.

Theorem D.6. *Let f be a rational function from Σ^* to $\mathcal{R}_{(0,1]}$. Then, $\equiv_f$ is of finite index. Furthermore, if $(\Gamma, \mathcal{T}_g, \mathcal{T}_\eta)$ is a representation of a bisequential decomposition of f, then $\mathcal{T}_\eta$ has at least $|\Sigma^*/\equiv_f|$ states and this bound is tight.*

Proof. Let (Γ, η, g) be a bisequential decomposition of f. From Proposition D.5, it follows that $\leftrightarrows_{\mathcal{T}_\eta}$ is a left congruence on Σ^*. Furthermore, Proposition D.7 implies that $\equiv_f$ is of finite index and $\mathcal{T}_\eta$ has at least $|\Sigma^*/\equiv_f|$ states. Lastly, by Proposition D.8, there exists a bisequential decomposition of f with an encoder that has exactly $|\Sigma^*/\equiv_f|$ states. □

Lastly, we verify formally that sequential language models require no information from the future in order to be represented; that is, the syntactic left congruence of a sequential language model has a single equivalence class. By Theorem D.6, this means that every sequential language model admits a bisequential decomposition with an encoder that produces information that is constant and does not change throughout time.

Theorem D.7. *Let f be a total rational function from Σ^* to $\mathcal{R}_{(0,1]}$. Then, f is sequential if and only if $\equiv_f = \Sigma^* \times \Sigma^*$.*

Proof. First, assume that f is sequential. Then, by Theorem C.7,

$$\left\{ \frac{f(\alpha)}{f(\beta)} \;\middle|\; \alpha, \beta \in \Sigma^* \wedge d_p(\alpha, \beta) \leq n \right\}$$

is finite for all $n \in \mathbb{N}$. Therefore, for $\alpha, \beta \in \Sigma^*$, we have that

$$\left\{ \frac{f(\gamma\alpha)}{f(\gamma\beta)} \;\middle|\; \gamma \in \Sigma^* \right\} \subseteq \left\{ \frac{f(\alpha')}{f(\beta')} \;\middle|\; \alpha', \beta' \in \Sigma^* \wedge d_p(\alpha', \beta') \leq d_p(\alpha, \beta) \right\}$$

is finite since it is contained in a finite set. In other words, $\alpha \equiv_f \beta$ for all $\alpha, \beta \in \Sigma^*$; that is, $\equiv_f = \Sigma^* \times \Sigma^*$.

Next, assume that $\equiv_f = \Sigma^* \times \Sigma^*$ and let $n \in \mathbb{N}$. Then,

$$\left\{ \frac{f(\alpha)}{f(\beta)} \;\middle|\; \alpha, \beta \in \Sigma^* \wedge d_p(\alpha, \beta) \leq n \right\} \subseteq \bigcup_{\alpha,\beta \in \Sigma^n} \left\{ \frac{f(\gamma\alpha)}{f(\gamma\beta)} \;\middle|\; \gamma \in \Sigma^* \right\}$$

is finite because it is contained in a finite union of finite sets; that is, f is sequential. □

E Latent Language Models

E.1 Language Modelling with Latent Decompositions

In this appendix, we give a formal proof of the statements that

(i) standard bisequential decompositions of functions from Σ^* to $[0, 1]$ are a special type of latent decompositions;

(ii) latent language models are exactly the functions that admit a latent decomposition with a generator that is a language model.

As direct corollaries, we obtain Theorems 4.2 and 5.1.

Theorem E.1. *Every standard bisequential decomposition of a function $f\colon \Sigma^* \to [0, 1]$ is a latent decomposition of f.*

Proof. Let (Γ, η, g) be a standard bisequential decomposition of $f\colon \Sigma^* \to [0, 1]$. Then, η is injective on $\Sigma^* \supseteq \eta^{-1}\big(\mathrm{Supp}(g)\big)$ and $\mathrm{Supp}(g\mathbf{1}_{\mathrm{Im}(\eta)}) \subseteq \mathrm{Im}(\eta)$. Furthermore,

$$\eta \circ g\mathbf{1}_{\mathrm{Im}(\eta)} = \eta \circ g = f.$$

Thus, $(\Gamma, \eta, g\mathbf{1}_{\mathrm{Im}(\eta)})$ is a latent decomposition of f. □

Theorem E.2. *Let (Γ, η, g) be a latent decomposition of $\mathbb{P}\colon \Sigma^* \to [0,1]$. Then, $\mathbb{P}$ is a language model over Σ if and only if g is a sequential language model over Γ.*

Proof. From the definition of a latent decomposition, it follows that η is a bijection from $\eta^{-1}\big(\mathrm{Supp}(g)\big)$ to $\mathrm{Supp}(g)$. Therefore,

$$\sum_{\alpha \in \Sigma^*} \mathbb{P}(\alpha) = \sum_{\alpha \in \Sigma^*} g\big(\eta(\alpha)\big) = \sum_{\alpha \in \eta^{-1}(\mathrm{Supp}(g))} g\big(\eta(\alpha)\big) = \sum_{\gamma \in \mathrm{Supp}(g)} g(\gamma) = \sum_{\gamma \in \Gamma^*} g(\gamma).$$

Thus, it is obvious that $\mathbb{P}$ is a language model over Σ if and only if g is a language model over Γ. □

E.2 Expressivemess of Latent Decompositions

In this appendix, we provide more detailed proofs of the claims that

(i) standard latent decompositions can represent only rational language models;

(ii) non-standard latent decompositions can represent non-rational languages.

Theorem E.3. *Every latent language model that admits a standard latent decomposition is rational.*

Proof. Let (Γ, η, g) be a standard latent decomposition of a language model $\mathbb{P}$ over Σ. Then, Γ is a Cartesian product $\Sigma \times \Gamma'$ and $\eta \circ \pi_{\Sigma^*} = \mathrm{id}_{\Sigma^*}$. Therefore, $\mathrm{Im}(\eta)$ is the graph of the function $\eta \circ \pi_{\Gamma'^*}$. Furthermore, we known that $\mathrm{Supp}(g)$ is a regular language, $\mathrm{Supp}(g) \subseteq \mathrm{Im}(\eta)$ and $\mathrm{Supp}(g)$ is the graph of the function $\eta \circ \mathrm{id}_{\mathrm{Supp}(g)} \circ \pi_{\Gamma'^*}$. This means that $\eta \circ \mathrm{id}_{\mathrm{Supp}(g)} \circ \pi_{\Gamma'^*}$ is a rational function and thus $\eta \circ \mathrm{id}_{\mathrm{Supp}(g)}$ is also rational. Now, if $\mathrm{Supp}(g) = \Gamma^*$, then $\eta = \eta \circ \mathrm{id}_{\mathrm{Supp}(g)}$ is rational and $\mathbb{P}$ is a rational language model as the composition of two rational functions. Next, suppose that $\mathrm{Supp}(g) \subsetneq \Gamma^*$. Let $\gamma \in \Gamma^* \setminus \mathrm{Supp}(g)$ and $\eta'\colon \Sigma^* \to \Gamma^*$ be defined as

$$\eta'(\alpha) \coloneqq \begin{cases} \eta(\alpha) & \text{if } \alpha \in \eta^{-1}\big(\mathrm{Supp}(g)\big) \\ \gamma & \text{otherwise} \end{cases}.$$

Obviously, η' is rational and $\eta' \circ g = \mathbb{P}$. Therefore, $\mathbb{P}$ is a rational language model. □

Theorem E.4. *Latent language models are strictly more expressive than rational language models.*

Proof. Let $\mathbb{P}$ be a language model over $\{a, b\}$ defined as

$$\mathbb{P}(\alpha) \coloneqq \begin{cases} \frac{1}{2^{n+1}} & \text{if } \alpha = a^n b^n \\ 0 & \text{otherwise} \end{cases}.$$

Obviously, $\mathbb{P}$ is not rational because its support $\{a^n b^n \mid n \in \mathbb{N}\}$ is not a regular language. However, $\mathbb{P}$ admits a latent decomposition $\big(\{a, b\}, \eta, g\big)$, where the encoder η simplifies the non-regular support of $\mathbb{P}$.

Indeed, let $\eta\colon \{a, b\}^* \to \{a, b\}^*$ and $g\colon \{a, b\}^* \to [0, 1]$ be defined as

$$\eta(\alpha) \coloneqq \begin{cases} (ab)^n & \text{if } \alpha = a^n b^n \\ a & \text{otherwise} \end{cases} \quad \text{and} \quad g(\alpha) \coloneqq \begin{cases} \frac{1}{2^{n+1}} & \text{if } \alpha = (ab)^n \\ 0 & \text{otherwise} \end{cases}.$$

Then, it is straightforward to verify that

(i) $\mathrm{Supp}(g) = \big\{(ab)^n \mid n \in \mathbb{N}\big\} \subseteq \big\{(ab)^n \mid n \in \mathbb{N}\big\} \cup a = \mathrm{Im}(\eta)$;

(ii) η is injective on $\eta^{-1}\big(\mathrm{Supp}(g)\big) = \big\{a^n b^n \mid n \in \mathbb{N}\big\}$;

(iii) g is a sequential language model over $\{a, b\}$;

(iv) $\mathbb{P} = \eta \circ g$;

that is, $\mathbb{P}$ is a latent language model. □

E.3 Comparison of Latent Language Models with D3PM

In this appendix, we compare latent language models with discrete diffusion language models and more specifically with D3PM (Austin et al., 2021).

Essentially, D3PM is a variational autoencoder (Kingma and Welling, 2014) that consists of a fixed stochastic encoder ϕ that, given a word α from the input space Σ^*, defines a probability distribution $\phi(\bullet \mid \alpha)$ over a latent space Γ^*, and a stochastic decoder ψ that, given an element γ of the latent space Γ^*, defines a probability distribution $\psi(\bullet \mid \gamma)$ over the input space Σ^*. Importantly, the encoder is constructed so that, regardless of the given word from the input space, it induces approximately the same probability distribution $\phi(\bullet)$ over the latent space; that is, $\phi(\bullet) \approx \phi(\bullet \mid \alpha)$ for all $\alpha \in \Sigma^*$.

When compared to a latent decomposition (Γ, η, g) of a language model over Σ, the encoder ϕ and the decoder ψ of a D3PM model correspond to stochastic equivalents of η and η^{-1}, respectively. Sampling from a D3PM model can be done efficiently by first sampling a latent element γ from the fixed encoder ϕ and then sampling a word over Σ from the decoder $\psi(\bullet \mid \gamma)$. A latent language model achieves the same result by first sampling a latent element γ from the generator g and then mapping it into the input space via η^{-1}.

A major drawback of D3PM models, when compared to latent language models, is the fact that their encoders and decoders are stochastic; that is, they are not exact inverses of each other. This deficiency leads to the inability of D3PM models to efficiently calculate exact word probabilities and necessitates the use of estimates such as the evidence lower bound (ELBO). Furthermore, the computation of the ELBO is often intractable and typically requires the use of Monte-Carlo based approximation methods. On the other hand, latent language models can exactly and efficiently score words by first encoding them via η and then computing the probabilities of the resulting latent elements via the sequential generator g.