Algorithms for Probabilistic and Stochastic Subsequential Failure Transducers

Diana Geneva, Georgi Shopov, and Stoyan Mihov(✉)

Institute of Information and Communication Technologies,
Bulgarian Academy of Sciences, 2, Acad. G. Bonchev Str., 1113 Sofia, Bulgaria
{dageneva,gshopov,stoyan}@lml.bas.bg

Abstract. This paper introduces a framework for building probabilistic models with subsequential failure transducers. We first show how various types of subsequential transducers commonly used in natural language processing are represented by probabilistic and conditional probabilistic subsequential failure transducers. Afterwards we introduce efficient algorithms for composition of conditional probabilistic subsequential transducers with probabilistic subsequential failure transducers and weight pushing (canonization) of probabilistic subsequential failure transducers. Those algorithms are applicable to many tasks for representing probabilistic models with subsequential failure transducers. One such task is the construction of the $HCLG$ weighted transducer used in speech recognition which we describe in detail. At the end, empirical results and comparison between the presented $HCLG$ failure weighted transducer and the standard $HCLG$ weighted transducer constructions are shown.

Keywords: Weighted transducers · Failure transducers · Probabilistic models

1 Introduction

Weighted finite-state transducers (WFST) are widely used for representing probability distributions over words such as language models, pronunciation lexicons, and hidden Markov models in automatic speech recognition (ASR) [8], and translation transformations in statistical machine translation [5]. In [1] it is shown that a back-off n-gram language model can be efficiently represented as a subsequential failure transducer. In practice however the failure transitions are often

Supported by the Bulgarian Ministry of Education and Science via grant DO1-200/2018 "Electronic healthcare in Bulgaria". We acknowledge the provided access to the e-infrastructure of the Centre for Advanced Computing and Data Processing, with the financial support by the Grant BG05M2OP001-1.001-0003, financed by the Science and Education for Smart Growth Operational Program and co-financed by the EU through the European structural and Investment funds.

S. Maneth (Ed.): CIAA 2021, LNCS 12803, pp. 127–139, 2021.
https://doi.org/10.1007/978-3-030-79121-6_11

substituted with ε-transitions [1,8]. The benefit is that general constructions for finite-state transducers can be applied. The drawback is twofold. First, since new non-valid paths are introduced, the output of the transducer is not equivalent to the original one. Second, the transducer is not deterministic anymore and there are many paths for one input. In [1] a procedure for eliminating some of the non-valid paths is presented. This procedure has the property that the maximal output probability for a given input in the resulting transducer will be equal to the output probability of this input in the original failure transducer. However, the resulting transducer might still contain more than one successful path for a given input and its size typically becomes 2 to 3 times the size of the input transducer.

In this paper we explore another approach. We introduce a framework for building probabilistic models with failure transducer constructions. Particularly, we show how to efficiently construct the $HCLG$ transducer [7] used in many modern ASR systems. We perform all constructions on failure transducers but we maintain the determinism on the input and thus obtain subsequential failure transducers as a result. The main advantages of our approach are:

- we directly construct deterministic devices – subsequential failure transducers, thus, avoiding additional determinizations,
- the resulting transducers represent correct probabilistic models.

Related work is conducted in [3] where specialized algorithms including intersection, failure transition removal and shortest distance for weighted failure automata are presented. As opposed to [3] here we focus our attention on subsequential failure transducers only. We present efficient, direct constructions which preserve the sequentiality of the failure transducers.

The formal definitions and detailed proofs of the constructions presented in this paper are provided separately in [4].

2 Preliminaries

We will use the standard notions of alphabet, word, language etc. from formal language theory. We will call a language L over the alphabet Σ *prefix-free* if $(\forall\alpha,\beta \in L)((\exists\gamma \in \Sigma^*)(\alpha\gamma = \beta) \implies \alpha = \beta)$. Throughout the paper we will distinguish between words as elements of the free monoid and *lexicon words* which occur in natural language. With $\mathcal{R}$ we will denote the monoid $\langle \mathbb{R}_+, \times, 1\rangle$ of non-negative real numbers with multiplication as the monoid operation. In this paper we consider only subsequential transduction devices introduced by Schützenberger [12]. For reasons of brevity, we will deliberately omit the word "subsequential".

A *transducer* is a tuple $\mathcal{T} := \langle \Sigma, \langle M, \otimes, \bar{1}\rangle, Q, s, F, \delta, \lambda, \iota, \rho\rangle$, where Σ is an alphabet, $\langle M, \otimes, \bar{1}\rangle$ is a monoid, Q is a finite set of states, $s \in Q$ is an initial state, $F \subseteq Q$ is a set of final states, $\delta\colon Q \times \Sigma \to Q$ is a partial transition function, $\lambda\colon Q \times \Sigma \to M$ is a partial output function, $\iota \in M$ is an initial output, $\rho\colon F \to M$ is a total final output function, and $\mathrm{Dom}(\delta) = \mathrm{Dom}(\lambda)$.

The *generalized transition function* $\delta^*\colon Q \times \Sigma^* \to Q$ and the *generalized output function* $\lambda^*\colon Q \times \Sigma^* \to M$ are the natural extensions of δ and λ over $Q \times \Sigma^*$.

For each $q \in Q$ the function $\mathcal{O}^q_{\mathcal{T}}\colon \Sigma^* \to M$ is defined for $\alpha \in \Sigma^*$ as $\mathcal{O}^q_{\mathcal{T}}(\alpha) := \lambda^*(q,\alpha) \otimes \rho(\delta^*(q,\alpha))$ if $\delta^*(q,\alpha) \in F$ and is undefined otherwise. The function $\mathcal{O}_{\mathcal{T}}\colon \Sigma^* \to M$, defined for $\alpha \in \Sigma^*$ as $\mathcal{O}_{\mathcal{T}}(\alpha) := \iota \otimes \mathcal{O}^s_{\mathcal{T}}(\alpha)$, is *the function represented by the transducer* $\mathcal{T}$. A state $q \in Q$ is called *co-accessible* in $\mathcal{T}$ if $\mathrm{Dom}(\mathcal{O}^q_{\mathcal{T}}) \neq \varnothing$.

A *failure transducer* is a tuple $\mathcal{F} := \langle \Sigma, \langle M, \otimes, \bar{1}\rangle, Q, s, F, \delta, \lambda, \iota, \rho, f, \varphi\rangle$ where $\langle \Sigma, \langle M, \otimes, \bar{1}\rangle, Q, s, F, \delta, \lambda, \iota, \rho\rangle$ is a transducer, $f\colon Q \to Q$ is a partial failure transition function, $\varphi\colon Q \to M$ is a partial failure output function, and $\mathrm{Dom}(f) = \mathrm{Dom}(\varphi)$. We define the *completed transition function* $\delta_f\colon Q \times \Sigma \to Q$ and the *completed output function* $\lambda_f\colon Q \times \Sigma \to M$ inductively:

$$\delta_f(q,\sigma) := \begin{cases} \delta(q,\sigma) & \text{if } !\,\delta(q,\sigma) \\ \delta_f(f(q),\sigma) & \text{otherwise} \end{cases} \qquad \lambda_f(q,\sigma) := \begin{cases} \lambda(q,\sigma) & \text{if } !\,\lambda(q,\sigma) \\ \varphi(q) \otimes \lambda_f(f(q),\sigma) & \text{otherwise} \end{cases}$$

where with $!\,\delta(q,\sigma)$ (resp. $!\,\lambda(q,\sigma)$) we denote that $(q,\sigma) \in \mathrm{Dom}(\delta)$ (resp. $(q,\sigma) \in \mathrm{Dom}(\lambda)$).

The *expanded transducer* of the failure transducer $\mathcal{F}$ is the transducer $\mathcal{T} := \langle \Sigma, \mathcal{M}, Q, s, F, \delta_f, \lambda_f, \iota, \rho\rangle$. For each $q \in Q$ we define $\mathcal{O}^q_{\mathcal{F}} := \mathcal{O}^q_{\mathcal{T}}$. The function $\mathcal{O}_{\mathcal{F}} := \mathcal{O}_{\mathcal{T}}$ is called *the function represented by the failure transducer* $\mathcal{F}$. A state $q \in Q$ is *co-accessible* in $\mathcal{F}$ if it is co-accessible in $\mathcal{T}$.

Definition 1. *A failure transducer is* monotonic *if for every* $q \in \mathrm{Dom}(f)$ *and every* $a \in \Sigma$ *it holds that* $q \in F \implies f(q) \in F$ *and* $!\,\delta(q,a) \implies !\,\delta(f(q),a)$.

The *signature* of a (failure) transducer state q is the set of labels for which q has an outgoing transition, i.e. $\mathrm{Sig}(q) := \{\sigma \in \Sigma \mid !\delta(q,\sigma)\}$. If the failure transducer is monotonic then $\mathrm{Sig}(q) \subseteq \mathrm{Sig}(f(q))$. Therefore the signatures of the states in every failure cycle are identical and thus the failure transitions in the cycle are redundant. In what follows, we will assume that every monotonic failure transducer that we consider has no failure cycles since they can be efficiently removed.

3 Probabilistic Transducers

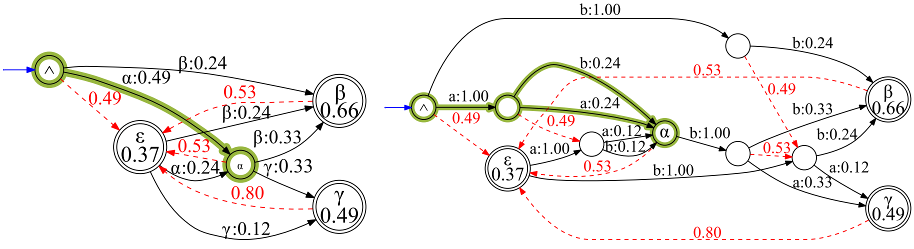
In speech recognition a widely used approach is to construct the so-called *HCLG* transducer [7,8]. This transducer is constructed by composing (from right to left) the language model transducer G with the lexicon transducer L, the context-dependency transducer C, and the HMM transducer H. We will show that the transducer G can be represented by a probabilistic failure transducer.

Definition 2. *A transducer* $\mathcal{T}$ *over the monoid* $\mathcal{R}$ *is* probabilistic *if* $\mathcal{O}_{\mathcal{T}}$ *is a probability distribution over* Σ^*, *i.e.* $(\forall\alpha \in \mathrm{Dom}(\mathcal{O}_{\mathcal{T}}))(\mathcal{O}_{\mathcal{T}}(\alpha) \in [0,1])$ *and* $\sum_{\alpha\in\mathrm{Dom}(\mathcal{O}_{\mathcal{T}})} \mathcal{O}_{\mathcal{T}}(\alpha) = 1$. *In order to emphasize that* $\mathcal{T}$ *represents a probability* *distribution, in what follows with $P_{\mathcal{T}}(\alpha)$ we shall denote $\mathcal{O}_{\mathcal{T}}(\alpha)$. We will use the expression $e(q)$ to mean $\rho(q)$ if $q \in F$ and 0 otherwise. We call $\mathcal{T}$* stochastic *if* $(\forall q \in Q)\left(e(q) + \sum_{a \in \Sigma:\ !\delta(q,a)} \lambda(q,a) = 1\right)$.

We call a failure transducer *probabilistic (stochastic)* if its corresponding expanded transducer is probabilistic (stochastic). Note that stochastic failure transducers may have failure outputs greater than 1.

Allauzen et al. have shown [1] that a smoothed n-gram language model can be represented by a failure transducer G. The transducer G maps a given sequence of lexicon words $w_1 w_2 \ldots w_n$ to the smoothed n-gram probability for the sequence $P(w_1 w_2 \ldots w_n)$. A typical back-off formulation of a smoothed n-gram language model is represented by the probability of a lexicon word w given a history h as follows: $P(w|h) := \hat{P}(w|h)$ if hw occurs, $P(w|h) := \alpha_h P(w|h')$ otherwise, where $\hat{P}$ is an empirical estimate of the probability that reserves probabilities for unseen n-grams, α_h is a normalizing back-off weight and h' is obtained by removing the earliest lexicon word from the history h. A failure transducer that represents the n-gram language model has states corresponding to the observed sequences of length $< n$. Its proper transitions (δ transitions) represent the case in which hw occurs and have weights equal to $\hat{P}(w|h)$. The failure transitions represent the other case and have weights α_h. It is assumed that every sentence in the corpus ends with the special lexicon word \$. A state h is final if $\hat{P}(\$|h)$ > 0 and $\rho(h) = \hat{P}(\$|h)$. Figure 1a depicts a failure transducer that represents a small bigram language model.

We can note that the failure transducer G is monotonic because the failure transitions enter states corresponding to shorter history. Also, the normalization constants α_h ensure that $\sum_w P(w|h) = 1$, which implies that $e(h) + \sum_w \lambda_f(h,w) = 1$. Therefore, G is monotonic and stochastic.



(a) Monotonic stochastic failure transducer for the smoothed bigram language model estimated from the corpus $\{\alpha\beta, \alpha\gamma, \beta\}$.

(b) Composition of the transducer for the phonetization lexicon $\alpha \mapsto \{\langle ab, 0.5\rangle, \langle aa, 0.5\rangle\}$, $\beta \mapsto \{\langle bb, 1\rangle\}$, and $\gamma \mapsto \{\langle ba, 1\rangle\}$ with the transducer from Figure 1a.

Fig. 1. The transducer G (left) and the composition $L \circ G$ (right).

4 Conditional Probabilistic Transducers

We next show that the transducers L, C, and H used to construct the $HCLG$ transducer can be represented by conditional probabilistic transducers.

Definition 3. *A* conditional probabilistic transducer *is a transducer* $\mathcal{T} := \langle \Sigma, \Omega^* \times \mathcal{R}, Q, s, F, \delta, \lambda, \iota, \rho\rangle$ *such that for every* $\beta \in \mathrm{Proj}_1(\mathrm{Range}(\mathcal{O}_\mathcal{T}))$ *it holds that* $\sum_{\alpha \in \mathrm{Dom}(\mathcal{O}_\mathcal{T}(\bullet|\beta))} \mathcal{O}_\mathcal{T}(\alpha \mid \beta) = 1$*, where the function* $\mathcal{O}_\mathcal{T}(\bullet \mid \beta) \colon \Sigma^* \to \mathbb{R}_+$ *for* $\alpha \in \Sigma^*$ *is defined as* $\mathcal{O}_\mathcal{T}(\alpha \mid \beta) := r$ *if* $\mathcal{O}_\mathcal{T}(\alpha) = \langle \beta, r\rangle$ *and is undefined otherwise. Again, in order to emphasize that* $\mathcal{T}$ *represents a conditional probability distribution, in what follows with* $P_\mathcal{T}(\alpha \mid \beta)$ *we shall denote* $\mathcal{O}_\mathcal{T}(\alpha \mid \beta)$*.*

The conditional probabilistic transducer L. The transducer L represents the pronunciation probabilities for the lexicon words. Here we assume that the set of lexicon words coincides with Ω. Given a sequence of lexicon words the transducer L represents the probability distribution over all phonetizations of the given sequence. In what follows the set of phones will be denoted with Σ. If the conditional probabilistic transducer $\mathcal{V} := \langle \Sigma, \Omega^* \times \mathcal{R}, Q_1, s_1, F_1, \delta_1, \lambda_1, \iota_1, \rho_1\rangle$ represents the phonetizations of single lexicon words then the transducer L is equal to the iteration (Kleene-Star) of $\mathcal{V}$. We have that $\mathrm{Proj}_1(\mathrm{Range}(\mathcal{O}_\mathcal{V})) = \Omega$ because $\mathcal{V}$ provides phonetizations for all lexicon words. Without loss of generality we can assume that $\mathrm{Dom}(\mathcal{O}_\mathcal{V})$ is prefix-free (this condition can easily be satisfied by adding new end word symbols to Σ), there are no transitions that enter s_1, $\mathrm{Range}(\rho_1) = \{\langle \varepsilon, 1\rangle\}$ and $\iota_1 = \langle \varepsilon, 1\rangle$.

Under these assumptions we construct the conditional probabilistic transducer L equal to the iteration (Kleene-Star) of $\mathcal{V}$ as

$$L = \mathcal{V}^* := \langle \Sigma, \Omega^* \times \mathcal{R}, Q_1 \setminus F_1, s_1, \{s_1\}, \delta_3, \lambda_1, \langle \varepsilon, 1\rangle, \{\langle s_1, \langle \varepsilon, 1\rangle\rangle\}\rangle,$$

where $\delta_3 := \delta_1 \restriction_{(Q_1 \setminus F_1)\times\Sigma\times(Q_1\setminus F_1)} \cup \{\langle p_1, a, s_1\rangle \mid \langle p_1, a, q_1\rangle \in \delta_1, q_1 \in F_1\}$. It follows that $\mathrm{Dom}(\mathcal{O}_{\mathcal{V}^*}) = \mathrm{Dom}(\mathcal{O}_\mathcal{V})^*$ and for every sequence of phonetizations $\bar{\alpha}_1, \bar{\alpha}_2, \ldots, \bar{\alpha}_n \in \Sigma^*$ of lexicon words $\beta_1, \beta_2, \ldots, \beta_n \in \Omega$ it follows that $\mathcal{O}_{\mathcal{V}^*}(\bar{\alpha}_1\bar{\alpha}_2\ldots\bar{\alpha}_n|\beta_1\beta_2\ldots\beta_n) = \prod_{i=1}^n P_\mathcal{V}(\bar{\alpha}_i|\beta_i)$. Since $\mathcal{V}$ is conditional probabilistic, summing in the above equation over all phonetizations of $\beta_1, \beta_2, \ldots, \beta_n$ we can observe that $L = \mathcal{V}^*$ is conditional probabilistic.

The conditional probabilistic transducer C. The context-dependency transducer C maps from context-dependent phones to context-independent phones. The context-dependent phones consist of l phones to the left (left context), central phone, and r phones to the right (right context). The states of the conditional probabilistic transducer C represent the last $l + r$ read context-independent phones. The transitions have context-dependent phones as labels and are of the form $p_1p_2\ldots p_{l+r} \xrightarrow{p_1p_2\ldots p_{l+r}q} p_2p_3\ldots p_{l+r}q$. The corresponding output of the transition is $\langle q, 1\rangle$. Insufficient contexts are padded with a special empty phone symbol. C is conditional probabilistic since for every sequence of context-independent phones β there exists a unique sequence of context-dependent phones α, such that $\mathcal{O}_C(\alpha) = \langle \beta, 1\rangle$.

The conditional probabilistic transducer H. The transducer H represents the HMM set, i.e. the Kleene-Star of the union of the HMMs for individual context-dependent phones. Each of the HMMs is a stochastic and conditional probabilistic transducer. The input symbols on the transitions are unique identifiers of the transitions in the HMM corresponding to the given context-dependent phone. Therefore, the union of the HMMs is prefix-free. Thus its Kleene-Star is conditional probabilistic and can be obtained using the same construction as the $\mathcal{V}^*$ transducer.

5 Composition of Conditional Probabilistic Transducers with Probabilistic Failure Transducers

In the previous section we showed how to construct efficiently the probabilistic failure transducer G and the conditional probabilistic transducers L, C and H. Here we introduce a general construction method for composing a conditional probabilistic transducer with a probabilistic failure transducer and obtaining a probabilistic failure transducer as a result. Using this general method we can construct the $HCLG$ transducer by the composition $H\circ(C\circ(L\circ G))$. In addition, we also introduce a more efficient construction for composition, applicable when certain conditions are satisfied by the conditional probabilistic transducer.

5.1 Generic Composition

In this subsection let $\mathcal{T} := \langle \Sigma, \Omega^* \times \mathcal{R}, Q_1, s_1, F_1, \delta_1, \lambda_1, \iota_1, \rho_1\rangle$ be a conditional probabilistic transducer and $\mathcal{F} := \langle \Omega, \mathcal{R}, Q_2, s_2, F_2, \delta_2, \lambda_2, \iota_2, \rho_2, f_2, \varphi_2\rangle$ be a probabilistic failure transducer. As a natural extension of the composition of transducers [8] and the intersection of weighted finite automata with failure transitions [3] we obtain a construction for the composition of $\mathcal{T}$ and $\mathcal{F}$.

Definition 4. *The* composition *of $\mathcal{T}$ and $\mathcal{F}$ is the failure transducer $\mathcal{T}\circ\mathcal{F} := \langle \Sigma, \mathcal{R}, Q_1 \times Q_2, s, F, \delta, \lambda, \iota, \rho, f, \varphi\rangle$, where*

$$
\begin{aligned}
s &:= \langle s_1, {\delta_2}^*_{f_2}(s_2, \mathrm{Proj}_1(\iota_1))\rangle,\\
F &:= \{\langle p_1, p_2\rangle \mid p_1 \in F_1, \langle p_2, \mathrm{Proj}_1(\rho_1(p_1)), q_2\rangle \in {\delta_2}^*_{f_2}, q_2 \in F_2\},\\
\delta &:= \{\langle\langle p_1, p_2\rangle, a, \langle q_1, p_2\rangle\rangle \mid \langle p_1, a, q_1\rangle \in \delta_1, \langle p_1, a, \langle \varepsilon, o_1\rangle\rangle \in \lambda_1, p_2 \in Q_2\} \cup\\
&\qquad \{\langle\langle p_1, p_2\rangle, a, \langle q_1, q_2\rangle\rangle \mid \langle p_1, a, q_1\rangle \in \delta_1, \langle p_1, a, \langle \omega\alpha, o_1\rangle\rangle \in \lambda_1, \langle p_2, \omega, r_2\rangle \in \delta_2,\\
&\qquad\qquad \langle p_2, \omega\alpha, q_2\rangle \in {\delta_2}^*_{f_2}\},\\
\lambda &:= \{\langle\langle p_1, p_2\rangle, a, o_1\rangle \mid \langle p_1, a, \langle \varepsilon, o_1\rangle\rangle \in \lambda_1, p_2 \in Q_2\} \cup\\
&\qquad \{\langle\langle p_1, p_2\rangle, a, o_1 o_2\rangle \mid \langle p_1, a, \langle \omega\alpha, o_1\rangle\rangle \in \lambda_1, \langle p_2, \omega, r_2\rangle \in \delta_2,\\
&\qquad\qquad \langle p_2, \omega\alpha, o_2\rangle \in {\lambda_2}^*_{f_2}\},\\
\iota &:= \mathrm{Proj}_2(\iota_1)\iota_2{\lambda_2}^*_{f_2}(s_2, \mathrm{Proj}_1(\iota_1)),\\
\rho &:= \{\langle\langle p_1, p_2\rangle, o_1 o_2 o_3\rangle \mid \langle p_1, p_2\rangle \in F, \langle p_1, \langle \alpha, o_1\rangle\rangle \in \rho_1, \langle p_2, \alpha, o_2\rangle \in {\lambda_2}^*_{f_2},
\end{aligned}
$$

$$\begin{aligned}&\langle {\delta_2}^*_{f_2}(p_2,\alpha), o_3\rangle \in \rho_2\},\\ f &:= \{\langle\langle p_1,p_2\rangle,\langle p_1,q_2\rangle\rangle \mid p_1 \in Q_1, \langle p_2,q_2\rangle \in f_2\},\\ \varphi &:= \{\langle\langle p_1,p_2\rangle,o_2\rangle \mid p_1 \in Q_1, \langle p_2,o_2\rangle \in \varphi_2\}.\end{aligned}$$

The above composition reflects the chain rule $P(\alpha,\beta) = P(\alpha \mid \beta)P(\beta)$. In our case β is uniquely identified by α, therefore $P_{\mathcal{T}\circ\mathcal{F}}(\alpha) = P_{\mathcal{T}}(\alpha \mid \beta)P_{\mathcal{F}}(\beta)$. The following proposition formalizes this idea. The proof can be found in [4].

Proposition 1. *Let* $\mathrm{Proj}_1(\mathrm{Range}(\mathcal{O}_{\mathcal{T}})) \supseteq \mathrm{Dom}(\mathcal{O}_{\mathcal{F}})$. *Then*

1. $(\forall\alpha \in \mathrm{Dom}(\mathcal{O}_{\mathcal{T}\circ\mathcal{F}}))(P_{\mathcal{T}\circ\mathcal{F}}(\alpha) = P_{\mathcal{T}}(\alpha \mid \beta)P_{\mathcal{F}}(\beta))$, *where* $\beta = \mathrm{Proj}_1(\mathcal{O}_{\mathcal{T}}(\alpha))$;
2. $\mathcal{T}\circ\mathcal{F}$ *is probabilistic;*
3. *if* $\mathcal{F}$ *is monotonic and for every* $p \in \mathrm{Dom}(f_2)$, $\alpha \in \mathrm{Proj}_1(\mathrm{Range}(\lambda_1))$ *and* $\beta \in \mathrm{Proj}_1(\mathrm{Range}(\rho_1))$ *we have that* $!\,{\delta_2}^*_{f_2}(p,\alpha) \implies !\,{\delta_2}^*_{f_2}(f_2(p),\alpha)$ *and* ${\delta_2}^*_{f_2}(p,\beta) \in F_2 \implies {\delta_2}^*_{f_2}(f_2(p),\beta) \in F_2$, *then* $\mathcal{T}\circ\mathcal{F}$ *is monotonic.*

In particular, when $\mathcal{T} = L$ and $\mathcal{F} = G$, the third statement of Proposition 1 follows from the monotonicity of G and $\mathrm{Proj}_1(\mathrm{Range}(\lambda_1 \cup \rho_1)) \subseteq \Omega \cup \{\varepsilon\}$. Also, since $\mathrm{Proj}_1(\mathrm{Range}(\mathcal{O}_L)) = \Omega^*$, we obtain that $L \circ G$ is monotonic and probabilistic. Figure 1a depicts a monotonic stochastic failure transducer for a simple language model. Therefore, its composition with a lexicon transducer will be monotonic and probabilistic (see Fig. 1b). Similarly the compositions $C \circ LG$ and $H \circ CLG$ yield monotonic and probabilistic failure transducers.

In practice the construction from Definition 4 might produce many states that are redundant. This particularly applies when composing L with G. We present a more efficient construction for this special case in Subsect. 5.2. This construction is also applicable for composing H with CLG more efficiently.

5.2 Special Case Composition

Let $\mathcal{V} := \langle \Sigma, \Omega^* \times \mathcal{R}, Q_1, s_1, F_1, \delta_1, \lambda_1, \iota_1, \rho_1\rangle$ be a trim (i.e. $(\forall q \in Q_1)(\exists\alpha,\beta \in \Sigma^*)(\delta_1^*(s_1,\alpha) = q \wedge \delta_1^*(q,\beta) \in F_1)$) conditional probabilistic transducer, which satisfies the conditions: $\mathrm{Proj}_1(\mathrm{Range}(\mathcal{O}_{\mathcal{V}})) = \Omega$ and $\mathrm{Dom}(\mathcal{O}_{\mathcal{V}})$ is prefix-free. Let $\mathcal{F} := \langle \Omega, \mathcal{R}, Q_2, s_2, F_2, \delta_2, \lambda_2, \iota_2, \rho_2, f_2, \varphi_2\rangle$ be a monotonic probabilistic failure transducer in which every state is co-accessible. Let us consider Definition 4 for the special case where $\mathcal{T} = \mathcal{V}^*$. We obtain $\mathcal{V}^*$ by redirecting the transitions ending in final states to the initial state. To express this we introduce the function $E : Q_1 \to Q_1$, such that $E(p) := s_1$ if $p \in F_1$ and $E(p) := p$ otherwise. Every successful path in $\mathcal{V}$ is of the form

$$p_1^0 \xrightarrow{a_1:\varepsilon/o_1} \dots \xrightarrow{a_{i-1}:\varepsilon/o_{i-1}} p_1^{i-1} \xrightarrow{a_i:\omega/o_i} p_1^i \xrightarrow{a_{i+1}:\varepsilon/o_{i+1}} \dots \xrightarrow{a_n:\varepsilon/o_n} p_1^n,$$

where $p_1^0 = s_1$ and $p_1^n \in F_1$. If we have a transition in $\mathcal{F}$ of the form $p_2 \to^{\omega:o'} q_2$, then we obtain the following path in the composition:

$$\langle p_1^0, p_2\rangle \xrightarrow{a_1:o_1} \dots \xrightarrow{a_{i-1}:o_{i-1}} \langle p_1^{i-1}, p_2\rangle \xrightarrow{a_i:o_i o'} \langle p_1^i, q_2\rangle \xrightarrow{a_{i+1}:o_{i+1}} \dots \xrightarrow{a_n:o_n} \langle E(p_1^n), q_2\rangle.$$

If $\omega \notin \mathrm{Sig}(p_2)$, then according to Definition 4 the states $\langle p_1^1, p_2\rangle, \ldots, \langle p_1^{i-1}, p_2\rangle$ are constructed but are redundant in $\mathcal{V}^* \circ \mathcal{F}$. In order to avoid constructing those states, we will restrict the states of $\mathcal{V}$, which we consider, to those on a successful path with label from the signature of the corresponding state in $\mathcal{F}$. Formally, we define the states to the left (Q^l_ω) and to the right (Q^r_ω) of transitions with output ω.

Definition 5. *Let $\omega \in \Omega$. We define*

$$\begin{aligned}
\Delta_\omega &:= \{\langle p_1, a, q_1\rangle \in \delta_1 \mid \mathrm{Proj}_1(\lambda_1(p_1, a)) = \omega\},\\
Q^l_\omega &:= \bigcup_{\langle p_1, a, q_1\rangle \in \Delta_\omega} \{l_1 \mid (\exists \alpha \in \Sigma^*)(\langle l_1, \alpha, p_1\rangle \in \delta_1^*)\},\\
Q^r_\omega &:= \bigcup_{\langle p_1, a, q_1\rangle \in \Delta_\omega} \{r_1 \mid (\exists \alpha \in \Sigma^*)(\langle q_1, \alpha, r_1\rangle \in \delta_1^*)\}.
\end{aligned}$$

For the example path in $\mathcal{V}$, the states $p_1^0, p_1^1, \ldots, p_1^{i-1}$ are from Q^l_ω and $p_1^i, p_1^{i+1}, \ldots, p_1^n$ are from Q^r_ω. Clearly the sets Q^l_ω and Q^r_ω are disjoint and any transition between two states in Q^l_ω or two states in Q^r_ω outputs ε and some probability. If for every $\omega \in \Omega$ we have Δ_ω (the transitions with output ω), the sets Q^l_ω and Q^r_ω can be computed in linear time with respect to their size with a simple traversal from respectively the source states of the transitions towards the initial state (having explicit backward transitions) and from the target states of the transitions towards the final states. We also observe that the failure transitions of the states with first coordinate in Q^r_ω are useless.

Proposition 2. *Let $\mathcal{W} := \langle \Sigma, \mathcal{R}, Q, \langle s_1, s_2\rangle, \{s_1\} \times F_2, \delta, \lambda, \iota_2, \rho, f, \varphi\rangle$, where*

$$\begin{aligned}
Q &:= \bigcup_{\langle p_2, \omega, q_2\rangle \in \delta_2} Q^l_\omega \times \{p_2\} \cup E(Q^r_\omega) \times \{q_2\},\\
\delta &:= \bigcup_{\langle p_2, \omega, q_2\rangle \in \delta_2} \begin{array}{l}\{\langle\langle p_1, p_2\rangle, a, \langle q_1, p_2\rangle\rangle \mid p_1, q_1 \in Q^l_\omega, \langle p_1, a, q_1\rangle \in \delta_1\} \cup\\
\{\langle\langle p_1, p_2\rangle, a, \langle E(q_1), q_2\rangle\rangle \mid \langle p_1, a, q_1\rangle \in \Delta_\omega\} \cup\\
\{\langle\langle p_1, q_2\rangle, a, \langle E(q_1), q_2\rangle\rangle \mid p_1, q_1 \in Q^r_\omega, \langle p_1, a, q_1\rangle \in \delta_1\},\end{array}\\
\lambda &:= \bigcup_{\langle p_2, \omega, q_2\rangle \in \delta_2} \begin{array}{l}\{\langle\langle p_1, p_2\rangle, a, o_1\rangle \mid p_1 \in Q^l_\omega, \langle p_1, a, \langle \varepsilon, o_1\rangle\rangle \in \lambda_1\} \cup\\
\{\langle\langle p_1, p_2\rangle, a, o_1 o_2\rangle \mid \langle p_1, a, \langle \omega, o_1\rangle\rangle \in \lambda_1, \langle p_2, \omega, o_2\rangle \in \lambda_2\} \cup\\
\{\langle\langle p_1, q_2\rangle, a, o_1\rangle \mid p_1 \in Q^r_\omega, \langle p_1, a, \langle \varepsilon, o_1\rangle\rangle \in \lambda_1\},\end{array}\\
\rho &:= \{\langle\langle s_1, p_2\rangle, o_2\rangle \mid \langle p_2, o_2\rangle \in \rho_2\},\\
f &:= \bigcup_{\langle p_2, \omega, q_2\rangle \in \delta_2} \{\langle\langle p_1, p_2\rangle, \langle p_1, r_2\rangle\rangle \mid p_1 \in Q^l_\omega, \langle p_2, r_2\rangle \in f_2\},\\
\varphi &:= \bigcup_{\langle p_2, \omega, q_2\rangle \in \delta_2} \{\langle\langle p_1, p_2\rangle, o_2\rangle \mid p_1 \in Q^l_\omega, \langle p_2, o_2\rangle \in \varphi_2\}.
\end{aligned}$$

Then $\mathcal{W}$ is trim and $\mathcal{O}_\mathcal{W} = \mathcal{O}_{\mathcal{V}^* \circ \mathcal{F}}$.

The construction from Proposition 2 can be implemented in linear time with respect to the transitions of the resulting transducer. The proof of the proposition can be found in [4].

6 Canonization

It has been shown that stochastic transducers are more effective than non-stochastic ones for speech recognition decoding [9]. As discussed the transducer G is stochastic. If LG is stochastic, then $C \circ LG$ is also stochastic because all weights in C are equal to 1. It can be easily observed that in this case $H \circ CLG$ will also be stochastic, since H is constructed from stochastic HMMs. However, the presented constructions for composition do not ensure that LG is stochastic.

It is easily shown that the stochasticity in the monoid $\mathcal{R}$ is equivalent to canonicity with respect to the semiring $\mathcal{R}^+ := \langle \mathbb{R}_+, +, \times, 0, 1 \rangle$.

Definition 6. *The transducer* $\mathcal{T} := \langle \Sigma, \langle K, \otimes, \bar{1} \rangle, Q, s, F, \delta, \lambda, \iota, \rho \rangle$ *is* canonical *with respect to the semiring* $\mathcal{K} := \langle K, \oplus, \otimes, \bar{0}, \bar{1} \rangle$ *if for every* $q \in Q$ *it holds that* $\bigoplus_{\alpha \in \mathrm{Dom}(\mathcal{O}^q_{\mathcal{T}})} \mathcal{O}^q_{\mathcal{T}}(\alpha) = \bar{1}$.

We call a failure transducer *canonical* if its expanded transducer is canonical. The standard canonization algorithm is based on modification of the weights of a given transducer so that the transition weights of every given state sum to $\bar{1}$. For a probabilistic (failure) transducer $\mathcal{T}$ and a state p with $S_{\mathcal{T}}(q)$ we denote the sum $\sum_{\alpha \in \mathrm{Dom}(\mathcal{O}^q_{\mathcal{T}})} \mathcal{O}^q_{\mathcal{T}}(\alpha)$. The weight-pushing is defined as updating the probability r of a given transition from the state p to the state q to $\frac{rS_{\mathcal{T}}(q)}{S_{\mathcal{T}}(p)}$. Next we extend the standard canonization algorithm for the case of probabilistic failure transducers.

Proposition 3. *Let* $\mathcal{W} := \langle \Sigma, \mathcal{R}, Q, s, F, \delta, \lambda, \iota, \rho, f, \varphi \rangle$ *be a probabilistic failure transducer and* $\mathcal{W}_{\mathcal{C}} := \langle \Sigma, \mathcal{R}, Q, s, F, \delta, \lambda_{\mathcal{C}}, \iota_{\mathcal{C}}, \rho_{\mathcal{C}}, f, \varphi_{\mathcal{C}} \rangle$*, where*

- $\lambda_{\mathcal{C}} := \{\langle p, a, \frac{rS_{\mathcal{W}}(\delta(p,a))}{S_{\mathcal{W}}(p)} \rangle \mid \langle p, a, r \rangle \in \lambda\}$;
- $\iota_{\mathcal{C}} := \iota S_{\mathcal{W}}(s)$;
- $\rho_{\mathcal{C}} := \{\langle p, \frac{r}{S_{\mathcal{W}}(p)} \rangle \mid \langle p, r \rangle \in \rho\}$;
- $\varphi_{\mathcal{C}} := \{\langle p, \frac{rS_{\mathcal{W}}(f(p))}{S_{\mathcal{W}}(p)} \rangle \mid \langle p, r \rangle \in \varphi\}$.

Then $\mathcal{O}_{\mathcal{W}_{\mathcal{C}}} = \mathcal{O}_{\mathcal{W}}$ *and* $\mathcal{W}_{\mathcal{C}}$ *is stochastic and canonical with respect to* $\mathcal{R}^+$.

The construction in Proposition 3 requires the computation of the sums $S_{\mathcal{W}}(q)$, which is computationally expensive. In this section we show how this can be achieved more efficiently in case $\mathcal{W}$ is obtained by composing a conditional probabilistic transducer $\mathcal{V}^*$ with a monotonic and stochastic failure transducer $\mathcal{F}$, where $\mathcal{V}$ is acyclic. This is the case for LG. The main idea of the specialized construction is to avoid the expansion of failure transitions by constructing an acyclic graph with nodes corresponding to transducer states with restricted signatures.

Let $\mathcal{V}$ and $\mathcal{F}$ be as in Subsect. 5.2. Also, let $\mathcal{F}$ be stochastic and $\mathcal{W} := \langle \Sigma, \mathcal{R}, Q, s, F, \delta, \lambda, \iota, \rho, f, \varphi \rangle$ be the failure transducer from Proposition 2, equivalent to the composition of $\mathcal{V}^*$ and $\mathcal{F}$. Since $\mathcal{W}$ is probabilistic, the sums $S_{\mathcal{W}}(q)$ exist for every $q \in Q$ and Proposition 3 can be used to obtain the canonical form of $\mathcal{W}$. Since $\mathcal{F}$ is stochastic, it is also canonical with respect to $\mathcal{R}^+$, i.e. for every state q of $\mathcal{F}$, $S_{\mathcal{F}}(q) = 1$. We show that the states of $\mathcal{W}$ with s_1 as first coordinate also satisfy this property.

Let $\omega \in \Omega$ and $\Gamma_\omega^{\mathcal{V}}$ be the set of all words from Σ^* for which $\mathcal{V}$ produces ω as output. i.e. $\Gamma_\omega^{\mathcal{V}} := \{\alpha \in \Sigma^* \mid \mathrm{Proj}_1(\mathcal{O}_{\mathcal{V}}(\alpha)) = \omega\}$. For example, in the case of the lexicon transducer L, Γ_ω^L represents the set of all phonetizations of ω. Then for every transition $\langle p, \omega, q \rangle \in \delta_2$ in $\mathcal{F}$ and every $\alpha \in \Gamma_\omega^{\mathcal{V}}$, there will be a path in $\mathcal{W}$ from $\langle s_1, p \rangle$ to $\langle s_1, q \rangle$ with input label α. Therefore, the sum of the outputs of the paths from $\langle s_1, p \rangle$ to $\langle s_1, q \rangle$ in $\mathcal{W}$ with input labels in $\Gamma_\omega^{\mathcal{V}}$ will be equal to the output of the transition $\langle p, \omega, q \rangle$ in $\mathcal{F}$. This is illustrated in Fig. 1. The highlighted transition in Fig. 1a with input α and output 0.49 is transformed into the two highlighted paths in Fig. 1b with inputs *aa* and *ab* (the phonetizations of α) and outputs 0.245. Using this observation and the fact that $\mathcal{F}$ is stochastic, the following proposition can be proved.

Proposition 4. $(\forall p \in Q_2)\,(S_{\mathcal{W}}(\langle s_1, p \rangle) = 1)$.

Therefore, it is sufficient to compute $S_{\mathcal{W}}(q)$ only for the states $q \in Q$ such that $\mathrm{Proj}_1(q) \neq s_1$. Thus, we consider the failure transducer $\widetilde{\mathcal{W}}$ in which the transitions from $\mathcal{W}$ that begin in such states are omitted and those states are made final with final output 1. Let $\widetilde{\mathcal{W}} := \langle \Sigma, \mathcal{R}, Q, s, Q_{s_1}, \widetilde{\delta}, \widetilde{\lambda}, \iota, Q_{s_1} \times \{1\}, \widetilde{f}, \widetilde{\varphi} \rangle$, where $Q_{s_1} = \{\langle p, q \rangle \in Q \mid p = s_1\}$, $\overline{Q}_{s_1}$ is $Q \setminus Q_{s_1}$, $\widetilde{\delta} := \delta \upharpoonright_{\overline{Q}_{s_1} \times \Sigma}$, $\widetilde{\lambda} := \lambda \upharpoonright_{\overline{Q}_{s_1} \times \Sigma}$, $\widetilde{f} := f \upharpoonright_{\overline{Q}_{s_1} \times \Sigma}$, and $\widetilde{\varphi} := \varphi \upharpoonright_{\overline{Q}_{s_1} \times \Sigma}$. From Proposition 4 it follows that $(\forall q \in Q)(S_{\widetilde{\mathcal{W}}}(q) = S_{\mathcal{W}}(q))$.

We reduce the problem of finding $S_{\widetilde{\mathcal{W}}}(q)$ to the single-source shortest distance problem with respect to the semiring $\mathcal{R}^+$ [6] in a special graph corresponding to $\widetilde{\mathcal{W}}$. The graph contains a node for each state in the transducer and an edge for each δ and failure transition. If there is a δ transition from p with label a then all δ_f transitions from p with a that begin with a failure transition are invalid. In order to avoid such paths in the graph we clone the target states of the failure transitions and allow from each cloned state only edges that correspond to δ-transitions that are not defined in the source of the failure transition (see Fig. 2).

We construct a labeled weighted acyclic graph, which in addition to the states from $\widetilde{\mathcal{W}}$ contains the cloned states, such that only the valid paths in $\widetilde{\mathcal{W}}$ are represented. Let $\mathcal{G} := (V, E)$, where

$$\begin{aligned}
V &:= Q \cup \{\langle q, \widetilde{f}(q) \rangle \mid q \in \mathrm{Dom}(\widetilde{f})\},\\
E &:= \{\langle p, \langle a, \widetilde{\lambda}(p,a) \rangle, q \rangle \mid \langle p, a, q \rangle \in \widetilde{\delta}\} \cup \{\langle p, \langle \varepsilon, \widetilde{\varphi}(p) \rangle, \langle p, q \rangle \rangle \mid \langle p, q \rangle \in \widetilde{f}\} \cup \\
&\quad \{\langle \langle p, q \rangle, \langle a, \widetilde{\lambda}(q,a) \rangle, r \rangle \mid \langle p, q \rangle \in \widetilde{f}, \langle q, a, r \rangle \in \widetilde{\delta}, \neg\, !\, \widetilde{\delta}(p,a)\} \cup \\
&\quad \{\langle \langle p, q \rangle, \langle \varepsilon, \widetilde{\varphi}(q) \rangle, \langle q, r \rangle \rangle \mid \langle p, q \rangle \in \widetilde{f}, \langle q, r \rangle \in \widetilde{f}\}.
\end{aligned}$$

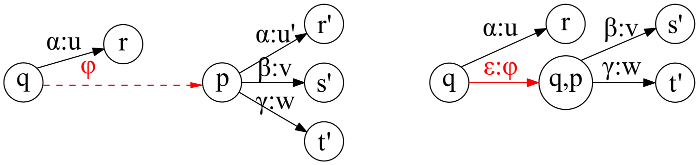


Fig. 2. $\langle q,p\rangle$ in the graph (right) is a clone of p in the failure transducer (left).

It then follows that $S_{\widetilde{\mathcal{W}}}(q)$ is the shortest distance from the new vertex x to the vertex q in the graph $\widetilde{\mathcal{G}} := (V \cup \{x\}, E^{rev} \cup \{x\} \times \{1\} \times Q_{s_1})$, which represents the reverse of the graph $\mathcal{G}$ extended with the initial vertex x. An important property of $\widetilde{\mathcal{G}}$ is that it is acyclic, since $\mathcal{V}$ is acyclic, there are no δ cycles because the transtions are restricted to those from $\overline{Q}_{s_1}$, and there are no failure cycles. This allows the sums $S_{\widetilde{\mathcal{W}}}(q)$ for $q \in \overline{Q}_{s_1}$ to be computed in linear time with respect to the size of $\widetilde{\mathcal{G}}$[1].

Thus, the procedure to construct the canonical form of $\mathcal{W}$ consists of first building the graph $\widetilde{\mathcal{G}}$, using $\widetilde{\mathcal{G}}$ to compute the values $S_{\mathcal{W}}(q)$, and applying Proposition 3 to push the weights of $\mathcal{W}$. Formal proofs of the above propositions are presented in [4].

7 Experimental Results and Conclusion

In our experiments we applied the presented constructions for building the $HCLG$ transducer for the LibriSpeech ASR language model [10] and compared them with the corresponding Kaldi [11] implementation which uses OpenFst [2]. All experiments are performed on a dual Intel Xeon Silver 4210 CPU at 2.20 GHz machine with 384 GB RAM running Debian Linux. The source code for the experiments can be requested from the authors by e-mail. For obtaining the $HCLG$ transducer we applied the following steps:

1. The stochastic failure transducer G is constructed using the construction from [1];
2. From the lexicon we construct $L \circ G$ using Subsect. 5.2, thus obtaining a probabilistic failure transducer;
3. Using the procedure from Sect. 6, we construct a stochastic failure transducer equivalent to $L \circ G$ which we additionally quasi minimize[2];
4. We construct the context-dependecy transducer C and compose it with the stochastic failure transducer LG, using Definition 4; The resulting failure transducer CLG is trim and stochastic;

[1] In comparison with the corresponding algorithm for ϕ-WFA presented in [3] we empirically observed that our algorithm introduces significantly less states and transitions.

[2] We use the standard automata minimization procedure considering failure transitions as proper transitions with special label.

5. From the definition of the HMMs using Proposition 2 we construct the failure transducer $HCLG$ which is stochastic, and which we additionally quasi minimize.

For constructing the standard WFST we used the standard Kaldi recipe . The main differences are that in the Kaldi recipe two intermediate deteminizations and additional optimization tricks are applied. The table below presents size and time comparison between the Kaldi construction of the HCLG WFST and our construction of the HCLG failure WFST.

Transducer	WFST			Failure WFST		
	States	Transitions	Time	States	Transitions	Time
G	7.6M	93.5M	4 m 24 s	7.6M	93.5M	1 m 47 s
LG	85.1M	230.5M	17 m 40 s	288.0M	717.9M	2 m 45 s
$min\&push(LG)$	72.7M	211.3M	1 h 10 m 30 s	254.7M	646.2M	20 m 34 s
CLG	73.6M	219.3M	1 h 20 m 14 s	255.1M	649.5M	27 m 13s
$min(HCLG)$	89.4M	316.7M	2 h 29 m 14 s	536.9M	1 213.1M	1h 22 m 34 s

The experiments show that the size of the failure WFST gets approximately 3–4 times bigger than the corresponding WFST but the construction time is around two times shorter. The main benefit of the presented approach is that the resulting failure WFST is deterministic, has no invalid paths, and represents a correct probability distribution.

References

1. Allauzen, C., Mohri, M., Roark, B.: Generalized algorithms for constructing statistical language models. In: ACL (2003)
2. Allauzen, C., Riley, M., Schalkwyk, J., Skut, W., Mohri, M.: OpenFst: a general and efficient weighted finite-state transducer library. In: Holub, J., Žd'árek, J. (eds.) CIAA 2007. LNCS, vol. 4783, pp. 11–23. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-76336-9_3
3. Allauzen, C., Riley, M.D.: Algorithms for weighted finite automata with failure transitions. In: Câmpeanu, C. (ed.) Implementation and Application of Automata, pp. 46–58. Springer International Publishing, Cham (2018)
4. Geneva, D., Shopov, G., Mihov, S.: Composition and Weight Pushing of Monotonic Subsequential Failure Transducers Representing Probabilistic Models. arXiv e-prints arXiv:2003.09364 (March 2020)
5. Kumar, S., Deng, Y., Byrne, W.: A weighted finite state transducer translation template model for statistical machine translation. Nat. Lang. Eng. **12**(1), 35–76 (2006)
6. Mohri, M.: Semiring frameworks and algorithms for shortest-distance problems. J. Automata Lang. Comb. **7**(3), 321–350 (2002)
7. Mohri, M., Pereira, F., Riley, M.: Weighted finite-state transducers in speech recognition. Comput. Speech Lang. **16**(1), 69–88 (2002)

8. Mohri, M., Pereira, F., Riley, M.: Speech recognition with weighted finite-state transducers. In: Benesty, J., Sondhi, M.M., Huang, Y.A. (eds.) Springer Handbook of Speech Processing. SH, pp. 559–584. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-49127-9_28
9. Mohri, M., Riley, M.: A weight pushing algorithm for large vocabulary speech recognition. In: Seventh European Conference on Speech Communication and Technology (2001)
10. Panayotov, V., Chen, G., Povey, D., Khudanpur, S.: LibriSpeech: an asr corpus based on public domain audio books. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5206–5210, April 2015
11. Povey, D., et al.: The Kaldi speech recognition toolkit. In: IEEE 2011 workshop (2011)
12. Schützenberger, M.: Sur une variante des fonctions sequentielles. Theor. Comput. Sci. **4**(1), 47–57 (1977)