Dickinson Language Reference

Vanessa McHale

October 11, 2020

Contents

0.1 Introduction

Dickinson is a language for generative literature targeting English. This reference specifies the syntax of the language.

0.2 Syntax

0.2.1 Lexical Structure

Dickinson programs have the following lexical structure:

$$
\begin{aligned}
\mathit{comment} &=: ;.^*\$ \\
\mathit{identifier} &=: [\texttt{a}-\texttt{z}][\texttt{a}-\texttt{zA}-\texttt{Z0}-\texttt{9}]^* \\
\mathit{typeIdentifier} &=: [\texttt{A}-\texttt{Z}][\texttt{a}-\texttt{zA}-\texttt{Z0}-\texttt{9}]^* \\
\mathit{moduleIdentifier} &=: (\mathit{identifier}.)^*\mathit{identifier} \\
\mathit{include} &=: :\texttt{include} \\
\mathit{def} &=: :\texttt{def} \\
\mathit{lambda} &=: :\texttt{lambda} \\
\mathit{tydecl} &=: \texttt{tydecl} \\
\mathit{probability} &=: ([\texttt{0}-\texttt{9}]^+|[\texttt{0}-\texttt{9}]^+.[\texttt{0}-\texttt{9}]^*)
\end{aligned}
$$

0.2.2 Syntax Tree

$$
\begin{aligned}
\langle \mathit{pattern} \rangle \quad ::= \quad & \_ \\
| \quad & \langle \mathit{identifier} \rangle \\
| \quad & \langle \mathit{typeIdentifier} \rangle
\end{aligned}
$$

| ⟨*pattern*⟩ (`|` ⟨*pattern*⟩)+
| (⟨*pattern*⟩ (`,` ⟨*pattern*⟩)+)

⟨*type*⟩ ::= `text`
| (→ ⟨*type*⟩ ⟨*type*⟩)
| (⟨*type*⟩ (, ⟨*type*⟩)*)
| ⟨*identifier*⟩

⟨*expression*⟩ ::= ⟨*string*⟩
| (`let:` [(⟨*identifier*⟩ ⟨*expression*⟩)+] ⟨*expression*⟩)
| (`bind:` [(⟨*identifier*⟩ ⟨*expression*⟩)+] ⟨*expression*⟩)
| (⟨*expression*⟩ (, ⟨*expression*⟩)*)
| (`:flatten` ⟨*expression*⟩)
| (⟨*expression*⟩ : ⟨*type*⟩)
| ⟨*typeIdentifier*⟩
| (`:pick` ⟨*identifier*⟩)
| (`>` ⟨*expression*⟩*)
| (`:oneof` (`|` ⟨*expression*⟩)+)
| (`:branch` (`|` ⟨*probability*⟩ ⟨*expression*⟩)+)
| (`$` ⟨*expression*⟩ ⟨*expression*⟩)
| (`:match` ⟨*expression*⟩ [(⟨*pattern*⟩ ⟨*expression*⟩)+])

⟨*declaration*⟩ ::= (`:def` ⟨*identifier*⟩ ⟨*expression*⟩)
| `tydecl` ⟨*identifier*⟩ `=` ⟨*typeIdentifier*⟩ (`|` ⟨*typeIdentifier*⟩)+

⟨*import*⟩ ::= (`:import` ⟨*moduleIdentifier*⟩)

⟨*module*⟩ ::= ⟨*import*⟩* `%-` ⟨*declaration*⟩*