Lab - 9

Matt Laws, Leah Williams

0.1 Minimal Formal Grammar

$\langle Expr\rangle := \langle Sentiment\rangle + "\backslash n" + \langle Keyword\rangle + "\backslash n" + \langle Line\rangle^{+}$
$\langle Keyword\rangle := \langle word\rangle^{*}$
$\langle Sentiment\rangle := \langle word\rangle$
$\quad\quad |\epsilon$
$\langle Line\rangle := \langle word\rangle^{+} + "\backslash n"$
$\langle word\rangle := \langle letter\rangle^{+} \in$ {CMU Pronunciation Dictionary}
$\langle Letter\rangle := \{a...z\}$

0.2 Minimal Semantics

Syntax	Abstract Syntax	Type	Prec./Assoc	Meaning
Word	word of string	string	N\A	Word is a primitive that represents a string of alpha characters that is contained in the CMU Pronunciation Dictionary
Line	Line of word list	string list	N\A	Line is a list of words that represents a line of a song that will be converted to new words each word is independent of each other
Keyword	Keywords of word list	string list	N\A	Keywords is a field that takes in a list of words and saves them as keywords to be added into the newly transformed song
Sentiment	string	string	N\A	Sentiment is a field that takes in a word and adds words to priority words list according to a given sentiment
Letter	char	char	N\A	Letter is a primitive. We represent chars using unicode character values F# data type
Line + $\backslash n$ + Line	Sequential lines of line list	string list list	first changed first	We parse each line in order when we transform them into new lines
CMU_dict	dict of records list	record list	N\A	This is the dict we query to do our word conversions, it