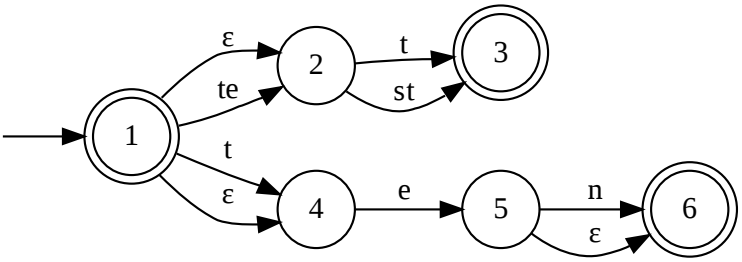
3. Übungsblatt - Abgabe: 17.11.2015

Aufgabe 3.1

Im Folgenden ist ein möglicher NEA angegeben, der die finiten Endungen eines schwachen Verbs wie „*kaufen*“ akzeptiert.



a) Konstruieren Sie einen zum NEA äquivalenten buchstabierenden Automaten.

b) Erzeugen sie aus diesem Automaten durch Potenzautomatenkonstruktion einen äquivalenten DEA. Geben Sie auch die Übergangstabelle an.

c) In der Datei tiger2.txt ist – wie in tiger.txt – in jeder Zeile ein Wort angegeben; zusätzlich finden Sie in jeder Zeile Informationen über das Lemma des Wortes. Benutzen Sie grep, um alle Wörter mit dem Lemma „kaufen“ im Tiger-Corpus zu finden. Wie viele sind es?

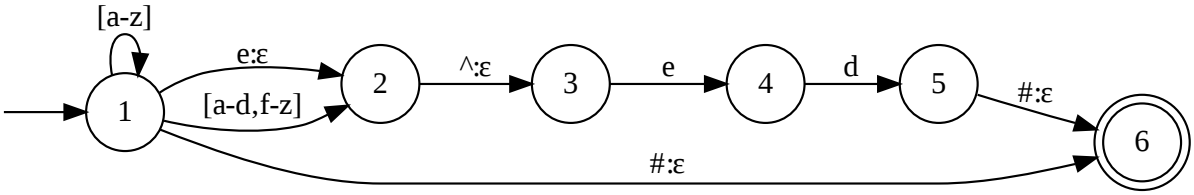
d) Finden Sie jetzt einen regulären Ausdruck, der für das Verb “kaufen” dieselben Wortformen wie Ihr Automat erkennt. (Hinweis: Auf Blatt 2 haben Sie schon etwas sehr ähnliches gemacht) Suchen Sie alle Vorkommen im Tiger-Corpus, bei denen das Wort auf Ihren regulären Ausdruck passt. Wie viele Wörter finden Sie hier?

e) Wie erklären Sie sich den Unterschied zwischen den Zahlen? Suchen Sie Beispiele für Wörter, die von Ihrer Suchanfrage in Aufgabe c aber nicht von der in Aufgabe d gefunden wurden (oder umgekehrt, falls sie für d mehr Ergebnisse als für c bekommen haben).

Aufgabe 3.2

Ein endlicher Transduktor ist ein EA der jeder akzeptierten Eingabesequenz eine Menge von Ausgabesequenzen zuordnet. Zu jeder Kante des Zustandsdiagramms gehört deshalb

neben dem Eingabe- ein zusätzliches Ausgabesymbol (notationell durch „:“ von einander getrennt).



Der abgebildete Transduktor übersetzt vorläufige Repräsentationen von Vergangenheitsformen englischer Verben, wie zum Beispiel

`bake^ed#`[1]

in die orthographisch korrekte Form:

`baked`

Wie muss der Transduktor erweitert werden, damit auch die Formen der Verben *fry*, *envy* und *panic* richtig erzeugt werden (aus `fry^ed#`, `envy^ed#` und `panic^ed#`)?

Folgende Notation kann verwendet werden: Wird ein Symbol auf sich selbst abgebildet, muss es nur einmal an die Kante geschrieben werden; Mengen von Kanten, die Symbole auf sich selbst abbilden, können durch eine Kante dargestellt werden, an der die Menge der Symbole steht (z.B. [a-z]).

Aufgabe 3.3

Der Unix-Befehl *sed 's/alt/NEU/g' inputfile* ersetzt in der Datei *inputfile* alle Vorkommen des regulären Ausdrucks *alt* durch *NEU*. Benutzen Sie *sed* um die Basisversion des Transducers aus der vorherigen Aufgabe nachzubauen. fry, envy und panic müssen *nicht* funktionieren.

Testen Sie Ihren Befehl auf der auf der Vorlesungshomepage verlinkten Datei Verben.txt und geben Sie Ihren Befehl und den Output, den Sie bekommen, an.

Aufgabe 3.4

a) Geben Sie die folgenden zwei Automaten über dem Alphabet $\Sigma = \{a, b\}$ an:
(1) den Automaten A_1, der alle Worte mit gerader Anzahl von a's (und beliebig vielen b's) akzeptiert,
(2) den Automaten A_2 ,der alle Worte mit gerader Anzahl von b's (und beliebig vielen a's) akzeptiert.

[1] „`^`“ markiert eine Morphemgrenze und „`#`“ markiert das Wortende.

b) Zu zwei (entweder nicht-deterministischen oder deterministischen) Automaten $A_1 = \langle K_1, \Sigma, \Delta_1, s_1, F_1 \rangle$ und $A_2 = \langle K_2, \Sigma, \Delta_2, s_2, F_2 \rangle$ über dem gleichen Alphabet Σ lässt sich auf sehr einfache Weise ein NEA $A = \langle K, \Sigma, \Delta, s, F \rangle$ konstruieren, der genau die Vereinigung der beiden von A_1 und A_2 spezifizierten Sprachen akzeptiert, also: $L(A) = L(A_1) \cup L(A_2)$. Beschreiben Sie die allgemeine Konstruktion in Worten (oder durch eine Zeichnung) und formal, indem Sie die verschiedenen Komponenten des Automaten A unter Rückgriff auf A_1 und A_2 explizit spezifizieren.

c) Konstruieren Sie einen NEA, der alle Worte über dem Alphabet $\Sigma = \{a, b\}$ akzeptiert, die eine gerade Anzahl von a's oder eine gerade Anzahl von b's enthalten, indem Sie die beiden Automaten aus (a) durch das Verfahren aus (b) zu einem neuen Automaten verknüpfen.

Aufgabe 3.5

Lesen Sie im Handbuch von Carstensen et al. den Abschnitt 5.1 (Korrekturprogramme), und zwar bis einschließlich 5.1.2 (Kontextabhängige Korrektur); das Papier befindet sich im Vorlesungsordner.

a) Eine ältere Version der MS Word-Rechtschreibprüfung hat vor Jahren als Alternativen für „semantisch“ die Wörter „seemännisch“ und „romantisch“ angegeben. Bestimmen Sie die im Text genannte Levenshtein-Distanz zwischen den drei Wörtern (also auch zwischen „seemännisch“ und „romantisch“).
Hinweis: Eine ausführlichere Beschreibung des Levenshtein-Algorithmus finden sie z.B. bei wikipedia.

b) Im Standard-Levenshtein-Algorithmus gibt es drei Operationen, die alle mit den gleiche Kosten verbunden sind. Auch innerhalb einer Operation kosten z.B. alle Ersetzungen gleich viel, egal, um welche Buchstaben es sich handelt.
Überlegen Sie sich mindestens 2 Möglichkeiten, wie man den Levenshtein-Algorithmus so abändern könnte, dass er für verschiedene Nutzergruppen (z.B nicht-Muttersprachler oder „flüchtige Tipper“) bessere Ergebnisse liefert, d.h. dass der Algorithmus besonders wahrscheinlichen Korrekturen einen niedrigeren Fehler-Score zuweist als unwahrscheinlichen Alternativen. Erklären Sie Ihre Verbesserungen anhand von Beispielen: Für welche Nutzergruppe sind welche Fehler wahrscheinlich und wieso weist Ihr Algorithmus diesem Fehler einen niedrigeren Score zu als anderen potenziellen Alternativen (die im Standardalgorithmus den gleichen Score hätten)?

c) Erklären Sie in anhand von 2 Beispielen, wozu man bei der Rechtschreibprüfung kontextabhängige Korrektur benötigt.

Abgabe in Gruppen von bis zu drei Studierenden am **17.11.2015** vor der Vorlesung.