AUFGABENBLATT

Im Praktikum betrachten wir grundlegende Ansätze und Verfahren der automatischen Spracherkennung.

Dabei lassen wir allerdings einen wesentlichen Aspekt außer Acht: Wir werden die Merkmalsextraktion, also die Bestimmung von geeigneten Merkmalen aus den Audio-Signalen der Sprache, im Praktikum nicht weiter betrachten[1]. Wir werden uns vor allem den Informatik-Aspekten widmen.

TERMIN 1

AUFGABE P1 – AUSWAHL DES THEMAS

Ein Ziel des Praktikums kann beispielsweise die Realisierung eines „Buchstabenerkenners“ sein. Anhand eines solchen Erkenners erläutern wir den ersten Praktikumsteil:

Mobil-Telefone der Vor-Smartphone-Zeit realisierten die Texteingabe mittels der Zifferntastatur. Die Kombination

43556#96753

kann beispielsweise auf den Text „hello world“ abgebildet werden, wenn die „übliche“ Abbildung von Zahlen auf Buchstaben zugrunde gelegt wird:

0	→	0
1	→	.,1
2	→	aAbBcC2
3	→	dDeEfF3
4	→	gGhHiI4
5	→	jJkKlL5
6	→	mMnNoO6
7	→	pPqQrRsS7
8	→	tTuUvV8
9	→	wWxXyYzZ9
*	→	<keine Funktion>
#	→	<Leerzeichen>.

Eine verbreitete, in Telefonen eingesetzte Technik ist „text on 9 keys“, kurz T9, von Tegic. Ihre Version eines Buchstabenerkenners wird auf einem trainierten stochastischen Modell basieren und soll letztendlich folgendes leisten:

- Die Umwandlung einer Sequenz von Symbolen ‚0‘,…,‚9‘, ‚*‘ und ‚#‘ in eine zugehörige Text-Sequenz.
- Die Verwendung eines deutschen oder englischen Modells zur Generierung deutsch- oder englisch-sprachlicher Texte aus einer Eingabesequenz.
- Die Beachtung von Groß- und Kleinschreibung.
- Die Verwendung von ‚.‘ und ‚,‘ als einzige Satzzeichen.

Wesentliche Beiträge zur Realisierung des Buchstabenerkenners werden sein:

1. Die Erstellung von Trainingskorpora in Deutsch bzw. Englisch.
2. Die geeignete Wahl eines Modells.
3. Das Training des Modells aus den Trainingskorpora.
4. Die Suche des besten Textes für die gegebene Eingabe.

Die Realisierung dieses Projektes wird über mehrere Aufgaben verteilt sein.

[1] Für den Aufbau eines Spracherkenners kann auf die Merkmalsextraktion nicht verzichtet werden – die Erkennungsqualität hängt bedeutend von der Qualität der Merkmalsextraktion und von den gewählten Merkmalen ab.

Anstelle eines T9-Erkenner können Sie auch ein alternatives Thema wählen, z.B. eine Swype-artige Tastatur, ein eigenes Tastatur-Layout (z.B. linear) oder eine Gesteneingabe (etwa Winkeralphabet, entsprechende Sensoren stehen im Labor zur Verfügung). Beachten Sie, dass die alternativen Projekte in den meisten Fällen einen Mehraufwand bedeuten.

Legen Sie fest, welches Thema Sie bearbeiten werden.

Im Folgenden werden wir die Aufgaben für einen T9-Erkenner formulieren, Sie müssen die Aufgabenstellung ggf. auf ihr gewähltes Thema übertragen.

AUFGABE P2 – ERSTELLEN EINES DATENKORPUS

Erzeugen Sie einen deutschen oder englischen Korpus, den Sie später zum Training bzw. Test verwenden.

1. Identifizieren Sie, welche Symbole in der Eingabe behandelt werden müssen und welche Symbole im generierten Text möglich sein sollen. Die Symbole, die in der Ausgabe vorkommen können, bilden das sogenannte Lexikon.
2. Geben Sie eine Abbildungsvorschrift zwischen den Eingabe- und Ausgabesymbolen an.
3. Identifizieren Sie Quellen, aus denen Sie Material für Ihren Korpus erhalten, insbes. im WWW.
4. Sammeln Sie Daten für ihren Korpus. Ihre Korpus sollte mindestens 10 000 Zeichen der Ausgabesymbole enthalten, gerne auch 100 000 Zeichen oder mehr. Unterteilen Sie den Korpus in einen Trainings- und Testkorpus. Welche Aufteilung erscheint Ihnen sinnvoll? Zwei disjunkte Teilmengen. 70% Training und 30% Test, zufällig ausgewählt -> hold out p = 30% Oder cross-validation z.B. k-fold
5. Überlegen Sie sich ein Vorgehen, um mit Symbolen im Korpus umgehen zu können, die nicht im Lexikon zu finden sind. Dies wir bei neuen (unbekannten) Eingabedaten relevant.
6. Erzeugen Sie aus Ihrem Korpus einen weiteren Korpus, in dem anstelle der Lexikon Symbole die entsprechenden Eingabesymbole ‚0',…,‚9', ‚*' und ‚#' verwendet werden.

AUFGABE P3 – GRUNDGERÜST DES SUCHBAUMES

und * auslassen?

Erstellen Sie ein Tool, welches aus einer Folge von Tastendrücken einer Handytastatur (0-9#*) alle möglichen resultierenden Buchstabenfolgen generiert. Die Folge „23" generiert beispielsweise „23", „ad", „ae", „af", ..., „cd", „ce", cf", ...

Speichern Sie die Möglichkeiten in Form eines Baumes. Erstellen Sie dazu eine geeignete Struktur und wählen Sie geeignete Informationen, die in den Knoten und Blättern gespeichert werden. Später im Praktikum werden wir für jeden Pfad durch den Baum eine Wahrscheinlichkeit berechnen, welche die Güte des Pfades wiedergibt.

Sehen Sie zur Erstellung des Baumes folgendes vor:

1. Die Tastenfolgen wird aus einer Datei gelesen und der zugehörige Baum erstellt.
2. Die Tastenfolge wird nach und nach per Tastatur eingeben. Der Baum wird nach jeder Tasteneingabe aktualisiert. Sie können dazu eine GUI erstellen, dies ist jedoch optional.

Dieser Baum wird die Grundlage für die weiteren Praktikumsaufgaben sein. Der Baum kann sehr schnell wachsen. Ein gutes Design und eine sorgfältige Implementierung dieser Aufgabe erleichtert die Durchführung der folgenden Aufgaben.

Darüberhinaus wird noch an einer weiteren Stelle eine Baum-artige Struktur benötigt. Es bietet sich daher an, eine gemeinsame Grundlage für Bäume zu schaffen.

TERMIN 2

AUFGABE P4 – RECHNEN MIT WAHRSCHEINLICHKEITEN

Es seien 3 farbige Kisten in den Farben rot (r), blau (b) und grün (g) gegeben. Die Kisten beinhalten Äpfel, Orangen und Limetten:

- Kiste rot: 3 Äpfel, 4 Orangen und 3 Limetten
- Kiste blau: 1 Apfel, 1 Orange und keine Limetten
- Kiste grün: 3 Äpfel, 3 Orangen und 4 Limetten.

Die Kisten sind unterschiedlich gestaltet. Die Auswahl der einzelnen Kisten genügt der Verteilung

$$p(r) = 0{,}2$$
$$p(b) = 0{,}2$$
$$p(g) = 0{,}6.$$

Nach der Auswahl einer Kiste wird eine Frucht aus dieser Kiste gezogen. Jede Frucht besitzt die gleiche Wahrscheinlichkeit, gezogen zu werden.

1. Berechnen Sie die Wahrscheinlichkeit, einen Apfel zu ziehen.
2. Wie groß ist die Wahrscheinlichkeit, dass eine Frucht aus der grünen Kiste stammt unter der Voraussetzung, dass es sich um eine Orange handelt?

Es sei nun P eine Wahrscheinlichkeitsverteilung auf der Menge $\mathcal{X}$ und $A \subset \mathcal{X}$.

3. Beweisen Sie, dass für das Komplement A^C von A die Beziehung $P(A^C) = 1 - P(A)$ gilt.

Notieren Sie ihre Lösung mit Zwischenschritten und nachvollziehbar.

AUFGABE P5 – BUCHSTABENWAHRSCHEINLICHKEITEN

Verwenden Sie den Trainingskorpus, um Buchstabenwahrscheinlichkeiten zu bestimmen. Schreiben Sie ein Programm, das den Trainingskorpus lädt und folgende Wahrscheinlichkeitsverteilungen mit Hilfe der relativen Häufigkeiten berechnet:

1. Die Wahrscheinlichkeit für das Auftreten eines einzelnen Buchstabens b_1 [2]:
   $P(B_1 = b_1)$
2. Die Wahrscheinlichkeit für das Auftreten eines Buchstaben-Zweierfolge $b_1 b_2$:
   $P(B_1 = b_1, B_2 = b_2)$
3. *Allgemein:* Die Wahrscheinlichkeit für das Auftreten eines Buchstaben-Folge der Länge n, $b_1 b_2 \dots b_n$:
   $P(B_1 = b_1, B_2 = b_2, \dots, B_n = b_n)$
4. Die bedingte Wahrscheinlichkeit für das Auftreten des Buchstabens b_3, wenn zuvor die Buchstaben-Zweierfolge $b_1 b_2$ aufgetreten ist:

$$P(B_3 = b_3 \mid B_1 = b_1, B_2 = b_2) = \frac{P(B_1 = b_1, B_2 = b_2, B_3 = b_3)}{P(B_1 = b_1, B_2 = b_2)}$$
$$\approx \frac{\text{Häufigkeit der Folge } b_1 b_2 b_3 \text{ im Trainingstext}}{\text{Häufigkeit der Folge } b_1 b_2 \text{ im Trainingstext}}$$

[2] Beachten Sie, dass b_1 hier eine Variable darstellt, die einen Buchstaben enthält, und nicht den Buchstaben „b“. Für die Wahrscheinlichkeit des kleinen Buchstabens „a“ bedeutet dies also $B_1 =$ “ a “ und die zugehörige Wahrscheinlichkeit ist $P(B_1 = \text{"a"})$.

5. *Allgemein:* Die bedingte Wahrscheinlichkeit für das Auftreten des Buchstabens b_n, wenn zuvor die Buchstaben-Folge $b_1 b_2 \dots b_{n-1}$ aufgetreten ist:

$$P(B_n = b_n \mid B_1 = b_1, B_2 = b_2, \dots, B_{n-1} = b_{n-1})$$

Wählen Sie eine geeignete Datenstruktur, um eine schnelle Berechnung der bedingten Wahrscheinlichkeit zu ermöglichen. Es bietet sich auch hier eine baum-artige Struktur an. Da der Trainingsprozess zeitintensiv sein kann, ist es ggf. sinnvoll, die Counts der Wortsequenzen in einer Datei zwischen zu speichern.

Kapseln Sie die Funktionalität, um die spätere Verwendung einfach zu gestalten.

AUFGABE P6 – WAHRSCHEINLICHKEIT EINER BUCHSTABENFOLGEN

Es gilt die Beziehung

$$P(B = b) = P\big((B_1, \dots, B_n) = b_1, \dots, b_n)\big) = \prod_{j=1}^{n} P\big(B_j = b_j \big| B_1 = b_1, B_2 = b_2, \dots, B_{j-1} = b_{j-1}\big).$$

1. Veranschaulichen Sie sich die Fälle $n = 1$, $n = 2$ und $n = 3$ und überlegen Sie sich induktiv, was dies für „größere n“ bedeutet. Hierzu reichen geeignete Überlegungen auf einem Blatt Papier.
2. Verwenden Sie nachfolgend die Näherung

$$P(B = b) = P\big((B_1, \dots, B_n) = (b_1, \dots, b_n)\big) \approx \prod_{j=1}^{n} P\big(B_j = b_j \big| B_{j-2} = b_{j-2}, B_{j-1} = b_{j-1}\big).$$

   Was bedeutet diese Näherung?
3. Realisieren Sie auf der Basis ihres Trainingskorpus die Berechnung der Wahrscheinlichkeit einer Buchstabenfolge $P(B = b)$ gemäß Aufgabenteil 1.
   Berechnen Sie die Wahrscheinlichkeit des Satzes „In Steinfurt regnet es fast nie.“ bzw. „Did it ever rain in Steinfurt?“.
   Berechnen Sie die Wahrscheinlichkeit für einen oder mehrere Sätze ihres Testkorpus, die ca. 100 Buchstaben umfasst.
4. Die Wahrscheinlichkeit einer langen Sequenz liefert häufig sehr kleine Zahlenwerte. Verwenden Sie die Beziehung $\ln a \cdot b = \ln a + \ln b$, um

$$-\ln P(B = b)$$

   mit Hilfe von Additionen (anstelle der Multiplikationen bei $P(B = b)$) zu berechnen[3]. Berechnen Sie ausgehend von $-\ln P(B = b)$ ebenfalls die Wahrscheinlichkeit $P(B = b)$.

TERMIN 3

In dieser und den nachfolgenden Aufgaben bezeichne $T = (T_1, \dots, T_n)$ die Tastendrücke und $B = (B_1, \dots, B_n)$ die resultierende Buchstabenfolge. Das Ziel des T9-Erkenners ist es, für eine Folge von Tastendrücken $t = (t_1, t_2, t_3, \dots, t_n)$ eine „passende“ Buchstabenfolge $b = (b_1, b_2, b_3, \dots, b_n)$ abzuleiten.

AUFGABE P7 – TASTENWAHRSCHEINLICHKEIT FÜR VORGEGEBENE BUCHSTABEN

Diese Aufgabe formalisiert einen Aspekt des T9-Erkenners für das Bayessche Modell, welcher intuitiv vollkommen klar ist:

> Drücken wir z.B. die Taste „5“, so können die Symbole „jJkKlL5“ aus diesem Tastendruck erzeugt werden.

[3] Damit vermeiden wir die sehr kleinen Zahlen. Wieso?

Beachten Sie, dass dies nur der Fall ist, wenn auch stets die gewünschte (die „richtige“) Taste getroffen wird. D.h., wenn der Benutzer beispielsweise das Symbol „k“ für ein Wort erzeugen möchte, wählt er die korrekte Taste „5“ und nicht versehentlich etwa die benachbarte Taste „6“.

Betrachten wir nun $P(T = t|B = b)$.

1. Gegeben sei $t = (8,3,7,8)$ und $b = (T, e, s, t)$. Geben Sie eine möglichst sinnvolle Wahrscheinlichkeit $P\big(T = (8,3,7,8)\big|B = (T, e, s, t)\big)$ an. In diesem Aufgabenteil überlegen Sie, wie eine geeignete Verteilung aussieht. Die Verteilung müssen Sie nicht berechnen oder aus Daten schätzen.
   Nun zu $t = (8,3,7,8)$ und $b = (T, e, x, t)$. Wie sollte $P\big(T = (8,3,7,8)\big|B = (T, e, x, t)\big)$ in diesem Fall aussehen?
   Verallgemeinern Sie die vorherigen Überlegungen und geben Sie eine sinnvolle Wahrscheinlichkeit $P(T = t|B = b)$ für beliebige t und b an[4].
2. Machen Sie sich klar, was das Ergebnis aus Teilaufgabe 1 für die Auswahl von Symbolen nach einem Tastendruck bedeutet.

AUFGABE P8 – BAYESSCHES MODELL

Begründen Sie, warum das Problem als Suche nach der maximierenden Buchstabenfolge $\hat{b}$ für eine gegebene Tastensequenz t formuliert werden kann:

$$\hat{b} = \arg\max_b P(T = t|B = b)P(B = b).$$

Wieso lautet die Funktion, die maximiert wird, $P(T = t|B = b)P(B = b)$? Was hätten Sie eigentlich erwartet?

AUFGABE P9 – BEWERTUNG DER EINZELNEN BAUMKNOTEN

Schreiben Sie ein Programm, bei dem als Eingabe die 10er-Tastatur eines „alten“ Mobiltelefons dient. Nach jedem Tastendruck suchen Sie die beste zugehörige Buchstabenfolge. Hier werden noch alle Möglichkeiten betrachtet und schlecht bewertete Möglichkeiten nicht entfernt.

Hier kombinieren Sie nun die Lösungen der vorherigen Aufgaben. Die nachfolgenden Komponenten sollten nun zur Verfügung stehen:

---

[4] Zur Information, ein möglicher formaler Ansatz: Es werde davon ausgegangen, dass die Tastendrücke T voneinander unabhängig sind und das ferner für den Tastendruck T_k nur der Buchstabe B_k von Bedeutung ist. Im Teil zur Mathematik haben wir Unabhängigkeit kennengelernt, das können Sie hier anwenden. Die zweite Annahme besagt, dass

$$P(T_k = t_k|B = b) = P\big(T_k = t_k|\,(B_1, \dots, B_n) = (b_1, \dots, b_n)\big) \approx P(T_k = t_k|B_k = b_k).$$

Die erste Annahme liefert

$$P(T = t|B = b) = \prod_{k=1}^{n} P(T_k = t_k|B = b)$$

Diese Annahme ist praktikabel. Allerdings trifft diese Annahme i.A. nicht zu und der resultierende Fehler wird in Kauf genommen.
Zusammen ergibt sich

$$P(T = t|B = b) \approx \prod_{k=1}^{n} P(T_k = t_k|B_k = b_k)$$

Die Wahrscheinlichkeit für die Tastendrücke zu einer gegebenen Buchstabenfolge ergibt sich also aus den Einzelwahrscheinlichkeiten jeder einzelnen Taste zu einem einzelnen Buchstaben.

i. Training der Wahrscheinlichkeiten.
Indem Sie die Auftretenshäufigkeiten von 1er, 2er, 3er usw. Tupeln im Trainingstext zählen, können Sie (bedingte) Wahrscheinlichkeiten für Buchstaben und Buchstabenfolgen berechnen.

ii. Berechnung von Wahrscheinlichkeiten.
Ausgehend von den trainierten Wahrscheinlichkeiten sind Sie in der Lage, für eine Buchstabenfolge eine Wahrscheinlichkeit zu berechnen. Außerdem können Sie bei einer vorgegebenen Buchstabenfolge $(b_1, \dots, b_{k-1})$ berechnen, wie wahrscheinlich das Auftreten eines ‚a', ‚b', ... als nächster Buchstabe b_k ist.

iii. Aufbau eines Suchbaums.
Zu einer gegebenen Folge von Tastendrücken werden alle zulässigen Symbolfolgen bestimmt. Um dies effizient speichern zu können, werden die Möglichkeiten in Form eines Baums gespeichert.

Gehen Sie nun wie folgt vor:

Nutzen Sie ihre Komponenten zur Wahrscheinlichkeitsberechnung, um im Suchbaum für jeden Knoten eine Wahrscheinlichkeit zu bestimmen. Für den ersten Tastendruck t_1 ist dies einfach. Für jeden möglichen Buchstaben b_1 berechnet sich die zugehörigen Wahrscheinlichkeit[5], die später im Bayesschen Modell verwendet wird, als:

$$P(t_1|b_1) \cdot P(b_1)$$

Um sehr kleine Zahlen zu vermeiden, berechnen die den $\ln\cdot$ dieses Ausdrucks

$$\ln P(t_1|b_1) + \ln P(b_1)$$

Diese $\ln\cdot$-Wahrscheinlichkeit speichern Sie in jedem Knoten der ersten Baumebene ab. Damit können wir diese Berechnung später weiterverwenden.
Nach dem zweiten Tastendruck t_2 benötigen wir die Wahrscheinlichkeiten der zweiten Baumebene:

$$P(t_2|b_2) \cdot P(t_1|b_1) \cdot P(b_1 b_2) = P(t_1|b_1) \cdot P(t_2|b_2) \cdot P(b_1) \cdot P(b_2|b_1)$$

bzw.

$$\ln P(t_1|b_1) + \ln P(t_2|b_2) + \ln P(b_1) + \ln P(b_2|b_1)$$

Allgemeiner gilt, wenn die Historie $t_1 \dots t_{n-1}$ bereits betrachtet wurde und nun ein neuer Tastendruck t_n hinzukommt:

$$\prod_{k=1}^{n} P(t_k|b_k) \cdot P(b_1 \dots b_{n-1} b_n) = \prod_{k=1}^{n-1} P(t_k|b_k) \cdot P(t_n|b_n) \cdot P(b_1 \dots b_{n-1}) \cdot P(b_n|b_1 \dots b_{n-1})$$

und erneut wieder nach Übergang zu logarithmierten Wahrscheinlichkeiten:

$$\sum_{k=1}^{n-1} \ln P(t_k|b_k) + \ln P(t_n|b_n) + \ln P(b_1 \dots b_{n-1}) + \ln P(b_n|b_1 \dots b_{n-1})$$

Die Berechnung der nötigen $\ln\cdot$-Wahrscheinlichkeiten für die Baumebene n basiert also auf der Bewertung der vorherigen $n-1$-ten Baumebene.

Außerdem wird üblicherweise nicht die gesamte vorherige Buchstabenfolge berücksichtigt, sondern nur ein Teil davon, z.B.

Unigram: $\ln P(b_n|b_1 \dots b_{n-1}) \approx \ln P(b_n)$
Bigram: $\ln P(b_n|b_1 \dots b_{n-1}) \approx \ln P(b_n|b_{n-1})$

oder auch

Trigram: $\ln P(b_n|b_1 \dots b_{n-1}) \approx \ln P(b_n|b_{n-2} b_{n-1})$

Sehen Sie in ihrem Programm vor, dass die Zahl der zu berücksichtigen vorherigen Buchstaben konfigurierbar ist.
Warum ist die Beschränkung der Historienlänge sinnvoll?

TERMIN 4

AUFGABE P10 – SUCHE NACH DEM BESTEN PFAD

[5] Wir verwenden ab jetzt die Kurzschreibweise und unterlassen die explizite Notation von $T = (T_1, \dots, T_n)$ und $B = (B_1, \dots, B_n)$

Jeder Knotenpunkt besitzt nun eine Bewertung gemäß der zuvor erlernten Wahrscheinlichkeiten. Suchen Sie nach jedem Tastendruck die 10 Pfade mit der besten Bewertung bis zur zugehörigen Baumebene für diesen Tastendruck.

Geben Sie die besten 10 Pfade inkl. der zugehörigen ($\ln$-)Wahrscheinlichkeit aus. Der beste Pfad stellt das gesuchte $\hat{b}$ in

$$\hat{b} = \arg\max_b P(T = t|B = b)P(B = b)$$

dar.

Hinweis: Besitzt ein Pfad von der Wurzel zum Knoten die Wahrscheinlichkeit 0, so gilt dies auch für alle weiteren Pfade, die diesen Pfad als Teil-Pfad umfassen. Dies ergibt sich sofort aus den Überlegungen des letzten Aufgabenblattes. Ein solcher Pfad muss damit nicht weiter expandiert werden. (Warum?)

AUFGABE P11 – DEKODIERUNG

Optimieren Sie die Brute-Force Suche aus Aufgabe P10:

Stellen Sie sicher, dass Pfade nicht weiter expandiert werden, sobald ein Pfad die Wahrscheinlichkeit 0 besitzt.

Prunen Sie den Suchbaum gemäß dem folgenden sub-optimalen Verfahren:

Nach jedem Tastendruck suchen Sie für jedes mögliche Symbol in einem Blatt-Knoten den besten Blatt-Knoten heraus. Ist beispielsweise die letzte Eingabe die Taste „6“ mit den zugehörigen Symbolen „mMnNoO6“, so suchen Sie für alle Pfade von der Wurzel zu einem Blatt-Knoten mit Symbol „m“ den Besten aller solcher Pfade heraus. Das wird ebenfalls für alle Pfade vom Wurzel-Knoten zu Blatt-Knoten „M“ durchgeführt, usw. Schließlich wurde für jedes Blatt-Knoten-Symbol aus der Menge der Symbole „mMnNoO6“ der jeweils beste Pfade bestimmt, in diesem Beispiel somit 7 beste unterschiedliche Pfade.

Für eine Konstante K suchen Sie nun außerdem die K besten Pfade im Suchbaum heraus, in Aufgabe P10 haben Sie dies bereits für $K = 10$ durchgeführt.

Die beiden „Beste-Pfad-Mengen“ werden nun verwendet, um aktive und inaktive Pfade festzulegen. Ist ein Pfad in einer der beiden „Beste-Pfad-Mengen“ enthalten, so bleibt der zugehörige Blatt-Knoten aktiv. Ist ein Pfad nicht in einer der beiden Mengen enthalten, so wird der Pfad als inaktiv markiert.

Beim nächsten Tastendruck werden nur noch Pfade mit aktiven Blatt-Knoten weiter expandiert. Inaktive Blatt-Knoten werden ignoriert.

Optional können Sie inaktive Knoten, und ggf. rekursiv deren Elternknoten, aus dem Baum entfernen. Elternknoten dürfen allerdings nur dann entfernt werden, wenn kein aktiver Pfad einen solchen Knoten nutzt.

AUFGABE P12 – FEHLERZÄHLER

Ausgangspunkt dieser Aufgabe sind die Tastendrücke der Mobiltelefon-Tastatur, die zu ihrem Test-Korpus gehören.

Wenden Sie ihren T9-Erkenner auf die Tastendrücke zum Test-Korpus an.

Entwickeln Sie ein Programm, dass das Erkennungsergebnis des T9-Erkenners mit dem „wahren“, ursprünglichen Text vergleicht. Zählen Sie die Abweichungen.

Besitzt ihr Test-Korpus beispielsweise 10 000 Zeichen und das Erkennungsergebnis weicht an 345 Stellen vom ursprünglichen Text ab, so beträgt die Fehlerquote $\frac{345}{10\,000} = 3{,}45\%$.

Berechnen Sie die Fehlerrate für verschiedenen N-gramm Längen. Welche Läge ist für ihren Erkenner optimal?

AUFGABE P13 – ALTERNATIVES EXAKTES PRUNING-VERFAHREN (ANSPRUCHSVOLL)

Verbesserung des Prunings aus Aufgabe P11:

Prunen Sie den Suchbaum analog zum DNA-Bigramm-Beispiel aus der Vorlesung. Nach Möglichkeit verallgemeinern Sie das Verfahren für die Verwendung von N-grammen.

Hinweis: Die Annahmen unseres T9-Erkenners erlauben es, dass Sie sich bei der Lösung dieser Aufgabe am DNA-Beispiel des Abschnitts zur Dekodierung und den HMMs orientieren.