Patentschrift

Verfasser: Prof. Dr. Günter Hotz

Thema:

Verfahren zur Verschlüsselung und Authentisierung des Nachrichtenaustausches zwischen Partnern.

Datum: 1. Mai 1997

1 Verfahren zur Verschlüsselung und Authentisierung des Nachrichtenaustausches zwischen Partnern

Es wird ein Verfahren beschrieben, das den Austausch von Nachrichten in Mailsystemen elektronischer Netze effizient kryptographisch zu verschlüsseln gestattet. Das Verfahren paßt sich dynamisch dem Nachrichtenaustausch zwischen den Partnern an. Der Kode ist approximativ optimal im Sinne der Informationstheorie unter der Voraussetzung, daß die Informationsquellen Markovprozesse sind. Indem der Kode sich automatisch an die Quellen anpaßt, wird er abhängig von der Geschichte des Nachrichtenaustausches. Partner in dem hier angesprochenen Sinn können zwei Personen sein oder auch jeweils zwei Gruppen von Personen, z.B. Behörden oder Firmen. Voraussetzung ist, daß der gesamte Nachrichtenaustausch zwischen den Partnern auf beiden Seiten zentral nach dem gleichen Schema verwaltet wird.

Ein Gegner, der mitlesen möchte, müßte nicht nur einmal den Zustand des Kodierungssystems stehlen, sondern müßte auch kontinuierlich mithören, da sich der Kode dynamisch dem Nachrichtenaustausch anpaßt.

Ein wichtiger Nebeneffekt besteht darin, daß das Verfahren auch die Daten beweisbar gut komprimiert. Darüber hinaus kann der Kode als Schlüsselsystem für eine effiziente Datenverwaltung genutzt werden.

Die Ideen gehen auf zwei Quellen zurück:

Die Arbeit von Shannon über Informationstheorie aus dem Jahre 1948 und die Arbeiten über Suchbäume aus den 70er Jahren von Bayer, Mehlhorn, Nievergelt und Tarjan. Während diese Arbeiten implizit voraussetzen, daß der Nachrichtenaustausch durch unabhängige Wahrscheinlichkeiten beschrieben wird, setzt die Arbeit des Antragstellers Markovprozesse voraus. Die für Suchbäume bekannten dynamischen Anpassungsverfahren lassen sich auf die von dem Antragsteller eingeführten Suchgraphen [1], [2] übertragen.

Das Verfahren:

Die Partner verwenden beide das gleiche Programm zur dynamischen Anpassung ihrer Datenstruktur an den Nachrichtenaustausch. Dieses Programm könnte z.B. in Mailsystemen (wie z.B. Netscape) fest integriert sein.

Die Partner P_1 und P_2 legen beide einen Suchbaum oder Suchgraphen $S(P_1, P_2)$ an. Das Programm liest jede von P_1 an P_2 oder von P_2 an P_1 abgesandte Nachricht und transformiert dabei die Datenstruktur im Sinne einer Anpassung an die Informationsquelle zur Datenkompression. Das Programm überträgt nicht die originale Nachricht, sondern den Weg, den das

Programm bei der Lokalisierung der in der Nachricht verwendeten Wörter und Zeichen nimmt. Verwendet diese Lokalisierungsprozedur binäre Abfragen, dann erhalten wir als Kode eine binäre Folge.

Die Theoreme der Informationstheorie garantieren, daß dieser Kode nahezu optimal ist, so daß die erzeugte Kommunikationsfolge approximativ zufällig ist. Der Empfänger ist nun in der Lage, in seiner Datenstruktur diese Wegbeschreibung zu verwenden, um die Wörter zu lokalisieren, die Anlaß für die Kodierung waren. Nach jedem dieser Dekodierungsschritte transformiert auch der Empfänger seine Datenstruktur, so daß beide Strukturen fortlaufend synchronisiert werden.

Kanalstörungen kann man dadurch begegnen, daß man ein Kommunikationsprotokoll vereinbart und die ursprüngliche Datenstruktur stets so lange aufbewahrt, bis das O.K. des Partners vorliegt.

Ein Beispiel soll das Verfahren anhand von Suchbäumen erläutern.

Wir betrachten einen Baum mit Knoten, die Wörter über dem deutschen Alphabet enthalten. Abbildung 1 zeigt die Knoten, die zu einem Wort gehören, am Beispiel des Wortes `genügt`.

Der oberste der drei Knoten entspricht der Abfrage des Eingabewortes u auf „$u =$ `genügt`?“. Falls die Abfrage erfüllt ist, folgt der Algorithmus der mit „=“ markierten Kante zu dem Blatt mit der Inschrift `genügt`. Ist die Abfrage nicht erfüllt, dann folgt der Algorithmus der mit „$\neq$“ markierten Kante zu dem Knoten mit der Abfrage „$u >$ `genügt`?“. Je nach Resultat der Abfrage folgt der Algorithmus der mit „>“ oder „<“ markierten Kante.

Der Suchbaum in Abbildung 2, der zu einer Menge deutscher Wörter gehört, besteht aus je drei Knoten in der Konfiguration von Abbildung 1a) für jedes Wort.

Am Endpunkt der mit „<“ markierten Kante hängt die Knotenkonfiguration eines Wortes, das dem Wort `genügt` lexikographisch vorausgeht, am Endpunkt der mit „>“ markierten Kante hängt die Knotenkonfiguration eines Wortes, das lexikographisch hinter `genügt` steht. Die hier für das Wort `genügt` beschriebene lokale Situation des Baumes gilt für jedes im Baum repräsentierte Wort. Kommt das Wort u, das gesucht wird, in dem Baum nicht vor, dann führt uns der Algorithmus zu dem Endknoten e einer Kante, an dem keine Wortkonfiguration hängt. Wir bringen in diesem Fall an e die Markierung „nicht vorhanden“ an. Wie mit diesem Fall verfahren wird erläutern wir später. Wir gehen zunächst davon aus, daß der Baum bezüglich aller Anfragen vollständig ist. Abbildung 2 stellt einen solchen Baum dar. Die Knoten, an denen keine Wortkonfiguration hängt, sind mit „$\emptyset$“ markiert. In dem Baum in Abbildung 2 haben wir die in Abbildung 1 beschriebene

Konfiguration durch die einfachere in Abbildung 1a) beschriebene ersetzt, die aber die gleiche Funktion hat. Im übrigen sei auf [3] verwiesen.

Wir wollen nun den Satz

`Für große Blöcke genügt es, die typischen Quellworte zu kodieren`

anhand des Baumes in Abbildung 2 binär verschlüsseln.

Wir setzen $u :=$`Für` und erhalten, wenn wir den Baum von seiner Wurzel zu dem Blatt mit dem Eintrag `Für` durchlaufen, als Markierung des Weges die Folge 0001001 als Kode für das Wort `Für`. Entsprechend erhalten wir für `große` den Kode 01001. Indem man das Verfahren für den gesamten Satz anwendet, erhält man als Kode

$$\overbrace{0001001}\,\overbrace{01001}\,\overbrace{0000001}\,\overbrace{\;1\;}\,\overbrace{001}\,\overbrace{0000011}\,\overbrace{0101011}\,\overbrace{01011}\,\overbrace{010101011}\,\overbrace{011}$$

Wir erhalten die Dekodierung dieses Satzes, indem wir in dem Baum die entsprechend markierten Pfade bis zu einem Blatt verfolgen. In unserem Beispiel also beginnen wir von links den Kode abzuarbeiten. Indem man den mit 0001001 markierten Weg verfolgt, gelangt man zu `Für`. Nun streicht man diesen Präfix des Kodes und verfährt mit dem Rest ebenso. In unserem Fall beginnt dieser Rest mit 01001. Diese Folge führt uns in dem Baum zu `große`. Man sieht, daß der Kode eindeutig entschlüsselbar ist, wenn der Empfänger im Besitz des gleichen Baumes ist.

Schauen wir unser Beispiel an, dann entdecken wir einen offensichtlichen Mangel, nämlich die Unsymmetrie in der Anzahl der im Kode vorkommenden Elemente 0 und 1. Diese Unsymmetrie beheben wir wie folgt: Wir ordnen jedem Wort, das in unserem Baum vorkommt, durch einen Zufallsgenerator genau eine der Abfragen $=$`?` oder $\neq$`?` zu. So erhalten wir entweder den in Abbildung 1a) oder den in Abbildung 1b) repräsentierten Knoten.

Wir haben bis hierher einen *nicht dynamischen* Suchbaum beschrieben. Die besondere Sicherheit gewinnt unser Kodierverfahren aber gerade durch die Verwendung dynamischer Suchbäume. Diese Suchbäume sind dynamisch in zweierlei Hinsicht:

1. Im Suchbaum nicht vorhandene Wörter werden bei entsprechender Suchanfrage automatisch eingefügt.

2. Die Knoten, die zu nachgefragten Wörten gehören, werden durch *Rotation* an die Wurzel des Baumes gebracht.

Es ist bekannt, daß diese dynamischen Bäume eine mittlere Zugriffszeit von höchstens $t \cdot (4 \cdot \ln 2 \cdot \mathrm{H(A)} + 4)$ haben, worin H(A) die Entropie der Quelle ist. Hieraus ergibt sich, daß das Verfahren zu einer guten Datenkompression und guten Randomisierung der Nachrichten führt. Wir erläutern das Verfahren anhand unseres Beispiels und verweisen ansonst auf die Literatur.

Das Wort `das` ist in dem Baum in Abbildung 2 nicht vorhanden. Die Nachfrage nach `das` führt zu dem Suchpfad, der durch 00000001 gekennzeichnet ist. Der Pfad endet mit einem durch $\emptyset$ markierten Knoten. Wir ersetzen nun diesen Knoten durch eine `das` nach Vorbild von Abbildung 1a) oder 1b) zugeordnete Konfiguration, deren Endknoten mit $\emptyset$ markiert werden.

Wir erläutern die Rotation durch das Beispiel der Suchanfrage nach `Fuß`. Der mit `Fuß` markierte Knoten wird durch Operationen, wie sie in Abbildung 3 dargestellt sind, zur Wurzel des Baumes gemacht. Die beiden erforderlichen Transformationsschritte sind in Abbildung 4 dargestellt.

Unser Verfahren arbeitet also nicht mit einem konstanten Kode, vielmehr transformiert sich der Kode in Abhängigkeit von jedem gerade kodierten Element. Es bleibt zu erläutern, wie der Empfänger seinen Dekodierbaum synchron zu dem Baum des Empfängers halten kann.

Im Falle der Rotation geschieht das nach Empfang und Dekodierung des entsprechenden Wortes. Die Neuaufnahme eines Wortes geschieht dadurch, daß ein spezieller Kode übertragen wird, nämlich der durch die Aufnahme des neuen Wortes u in den Suchbaum definierte Kode $c(u)$ des Wortes u und die anschließend übertragene buchstabenweise Kodierung $c'(u)$ von u. Letztere setzt voraus, daß die Alphabetelemente als „Elementarwörter“ mit in den Baum aufgenommen wurden. Auf die oben beschriebene Weise werden die Bäume der beiden Kommunikationspartner stets synchron gehalten, so daß dieses Kodierungsverfahren in beiden Richtungen verwendet werden kann.

Der Start der Kommunikation braucht an sich keine besondere Regelung. Es ist aber im Interesse einer höheren Sicherheit vorteilhaft, zugleich mit einem größeren Wortschatz zu beginnen. Das kann dadurch geschehen, daß dem Kodierprogramm ein fester Kernwortschatz beigegeben wird. Eine von einem Partner gestartete Übertragung von zufällig aus dem Kernwortschatz ausgewählten Wörtern „verrauscht“ die Bäume, so daß sie sich mit hoher Wahrscheinlichkeit von den Bäumen aller Dritter unterscheiden.

Eine Authentifikation der Partner kann durch den Austausch von Informationen über den aktuellen Zustand des dynamischen Baumes oder Graphen stattfinden. Eine Möglichkeit besteht darin, den "Ort“ einiger zufällig ausgewählter Wörter mitzuteilen: A eröffnet die Kommunikation, B fragt A nach der Position einiger Wörter im Baum.

2 Patentansprüche

1. Einrichtung zur kryptographischen und datenkomprimierenden Verschlüsselung der zwischen Partnern auszutauschenden Nachrichten, wobei jeder Partner jede Nachricht erhält und die Verschlüsselung darauf beruht, daß

   (a) jeder Partner einen dynamischen Suchbaum oder Suchgraphen in gleichem Anfangszustand besitzt,

   (b) der Suchbaum bzw. Suchgraph dynamisch an die Nachrichtenquelle zur Erzielung guter mittlerer Zugriffszeiten angepaßt wird,

   (c) der Kode durch die Markierungen des Suchpfades beschrieben wird,

   (d) die Dekodierung durch Verfolgung des Pfades mit der Kodemarkierung bis zu einem Blatt des Suchgraphen bzw. Suchbaumes erfolgt,

   (e) Suchbaum und Suchgraph isomorph gehalten werden, indem sie auf die gleiche Weise an die aktuelle Nachricht angepaßt werden.

2. Verschlüsselungseinrichtung nach Anspruch 1
   dadurch **gekennzeichnet**,
   daß Suchbaum und Suchgraph bei der dynamischen Anpassung das Verfahren der „Rotation zur Wurzel“ verwenden.

3. Verschlüsselungseinrichtung nach Anspruch 1 oder 2
   dadurch **gekennzeichnet**,
   daß in den Nachrichten auftretende Worte, die im Suchgraph bzw. Suchbaum nicht vorhanden sind, an dem Knoten eingefügt werden, an dem die Ergebnislosigkeit der Suche entschieden wird.

4. Verschlüsselungseinrichtung nach Anspruch 1 oder 2 oder 3
   dadurch **gekennzeichnet**,
   daß die Einrichtung im Anfangszustand einen Kernwortschatz und das Alphabet sowie Sonderzeichen enthält.

5. Verschlüsselungseinrichtung nach Anspruch 4
   dadurch **gekennzeichnet**,
   daß im Suchbaum oder Suchgraphen des Senders neu aufgenommene Wörter bei ihrer Erstübertragung durch die Markierung des Pfades von Wurzel zu Neueintrag und anschließend zeichenweise Kodierung übertragen wird.

6. Verschlüsselungseinrichtung nach Anspruch 1 bis 5
   dadurch **gekennzeichnet**,

daß jedem Wort u, das in dem System vorhanden ist, „Elementarkonfigurationen" konfig(u) aus drei Knoten zugeordnet sind, die die Suche in dem Suchbaum bzw. Suchgraphen verzweigen und die Parameter zur Randomisierung der Markierungen der von konfig(u) ausgehenden Kanten enthalten.

7. Verschlüsselungseinrichtung nach Anspruch 6
dadurch **gekennzeichnet**,
daß die Randomisierung von beim Sender neu aufgenommenen Konfigurationen konfig(u) den Empfängern verschlüsselt im Anschluß an die Verschlüsselung des Wortes mitgeteilt wird.

8. Verschlüsselungseinrichtung nach Anspruch 1
dadurch **gekennzeichnet**,
daß die dynamische Anpassung des Suchbaumes bzw. Suchgraphen nicht nach der Übertragung jedes Wortes oder Zeichens vorgenommen wird, sondern erst nachdem die gesamte Sendung übertragen wurde.

9. Verschlüsselungseinrichtung nach Anspruch 1 bis 8
dadurch **gekennzeichnet**,
daß eine Kopie des alten Suchbaumes so lange aufbewahrt wird, bis die gesamte Übertragung einer Sendung abgeschlossen ist und durch eine Empfangsbestätigung quittiert wurde.

10. Verschlüsselungseinrichtung nach Anspruch 1 bis 9
dadurch **gekennzeichnet**,
daß sie in ein Emailsystem oder Navigationssystem wie Mosaic oder Netscape als Tool integriert ist.

11. Verschlüsselungseinrichtung nach Anspruch 1 bis 10
dadurch **gekennzeichnet**,
daß zur Authentifikation Nachrichten über den aktuellen Zustand des Suchgraphen ausgetauscht werden.

12. Verschlüsselungseinrichtung nach Anspruch 1 bis 11
dadurch **gekennzeichnet**,
daß der Empfänger von dem Sender die Mitteilung des Ortes von bestimmten Wörtern in seinem Suchbaum verlangt.

Literatur

[1] Günter Hotz. *Algorithmische Informationstheorie.* Vorlesung WS 1996/97, Universität des Saarlandes, Saarbrücken.

[2] Günter Hotz. *Search Trees and Search Graphs for Markov Sources.* J. Inform. Process. Cybernet. EIK 29 (1993) 5, 283–292.

[3] Kurt Mehlhorn. *Effiziente Algorithmen*. Teubner, 1977.

Abbildungen

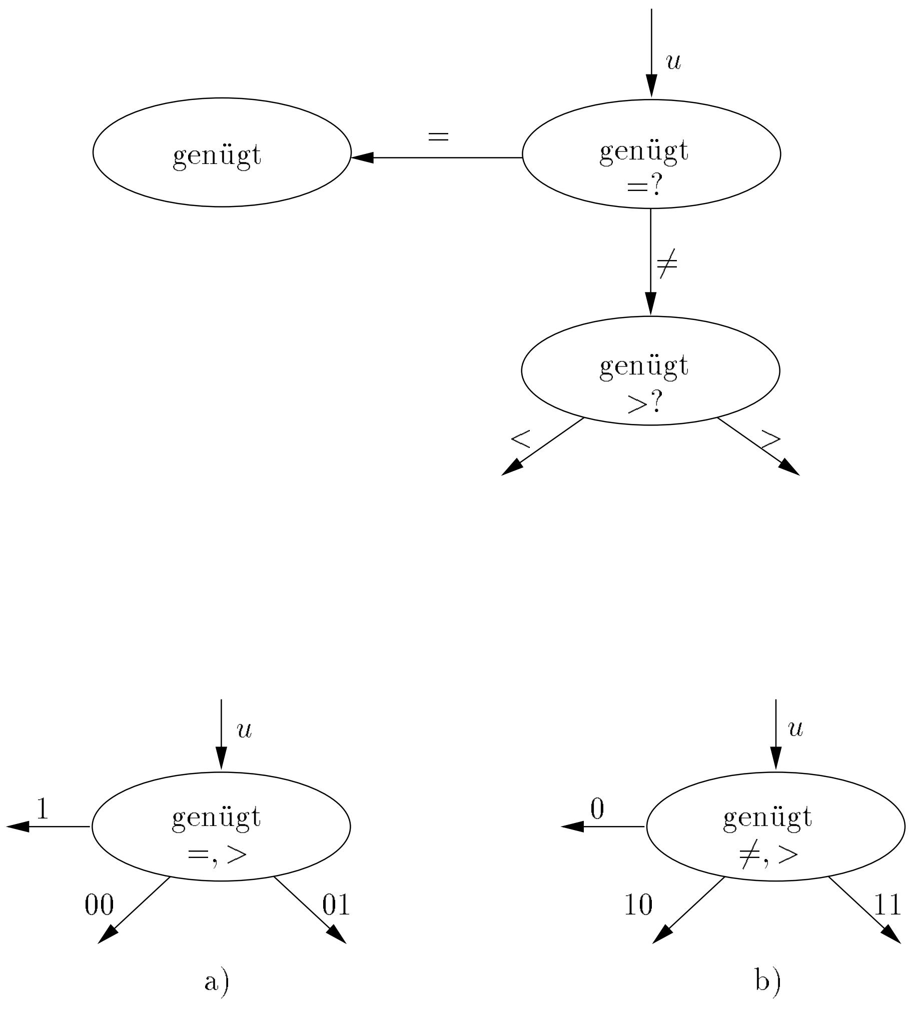

Abbildung 1: Knoten im Suchbaum

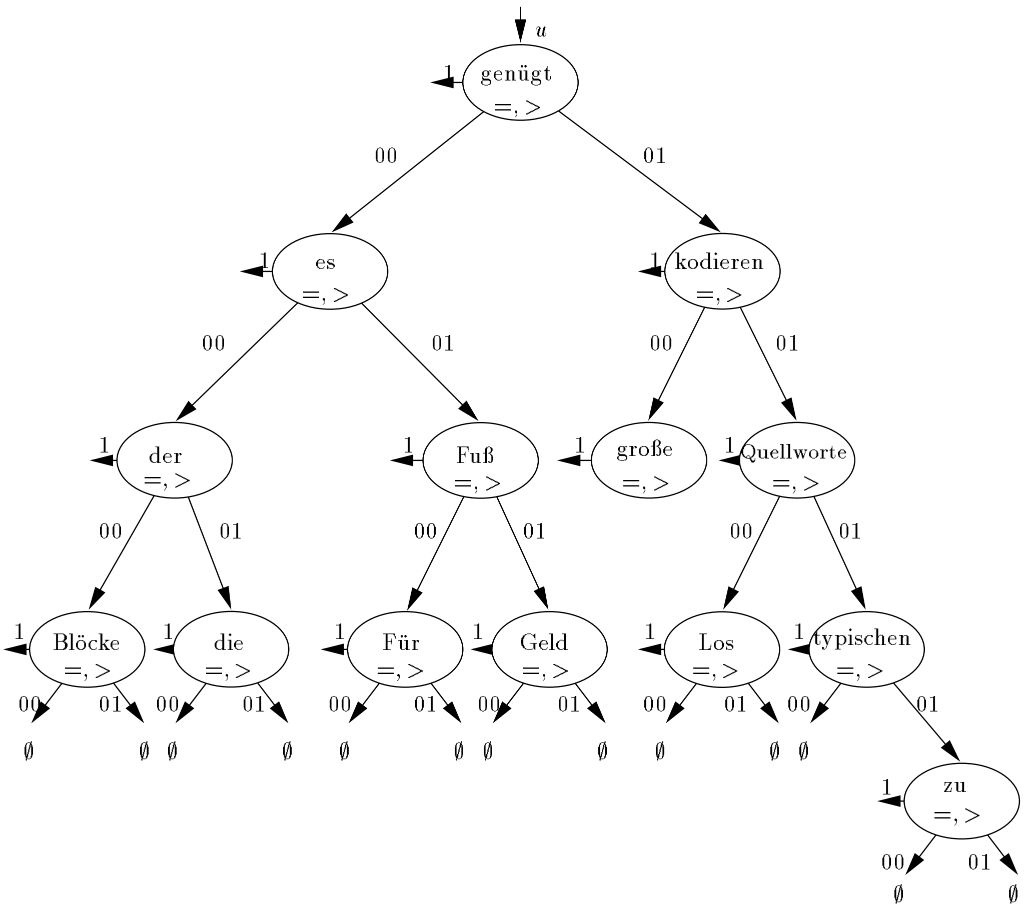

Abbildung 2: Suchbaum

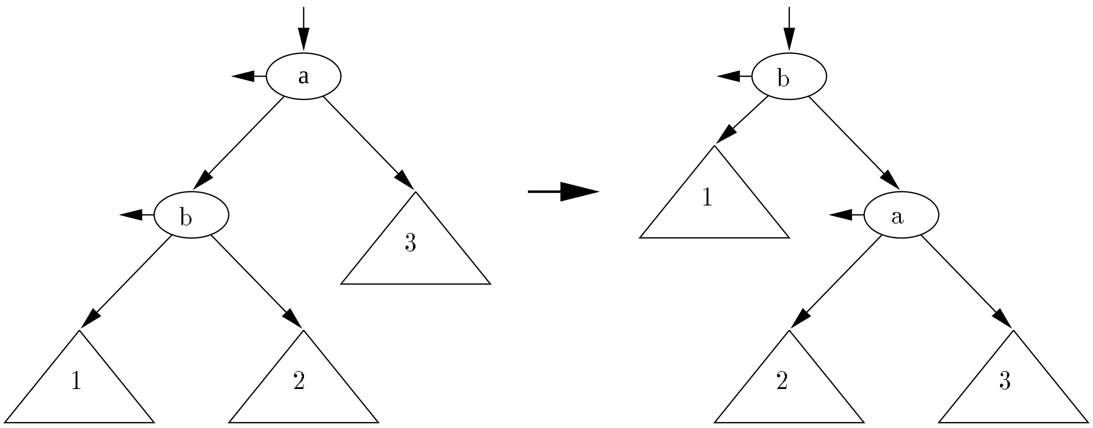

Abbildung 3: Schema der Rotation

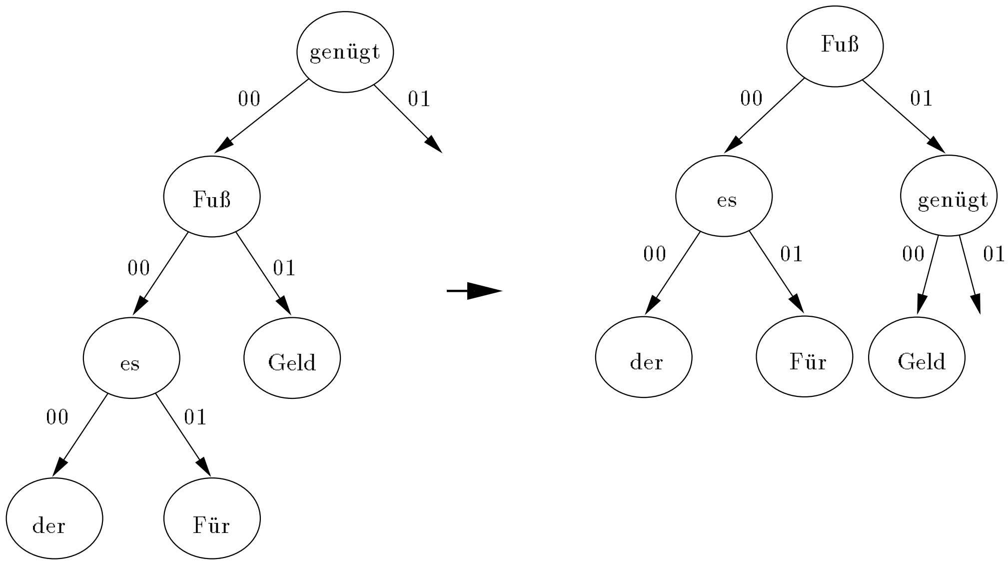

Abbildung 4: Beispiel einer Rotation