KORPUSDRESS 1 – KORPUSBASIERTE KONKATENATIVE SPRACHSYNTHESESYSTEME

Hamurabi Gamboa Rosales, Oliver Jokisch
Technische Universität Dresden
Hamurabi.Gamboa@ias.et.tu-dresden.de

Abstract: Korpusbasierte Sprachsynthesesysteme haben eine hohe Qualität in der Sprachsynthese gezeigt. Dies ist möglich wegen der guten Leistung der Bausteinauswahl in den Sprachsynthesesystemen[1], in denen es das Hauptziel der Bausteinauswahl ist, die beste Kombination der Sprachbausteine zu finden, um den Wahrnehmungsunterschied zwischen dem erwarteten und synthetisierten Sprachsignal so minimal wie möglich zu erhalten. Aber die Bestimmung der Strategie der Bausteinauswahl und das Tuning der Parameter sind noch schwierige Herausforderungen. Dieses Paper beschreibt die Version 1 des KorpusDress Sprachsynthesesystems (TTS). Das System stellt die Bedingungen der Entwicklung für die korpusbasierte konkatenatenative Sprachsynthese mit nicht-gleichförmigen Sprachbausteinen zur Verfügung. Wir diskutieren die verschiedenen Aspekte der nicht-gleichförmigen Sprachbausteinauswahl und konzentrieren uns auf die Erforschung der Themen, die auf den Entwurf der Korpusdatenbank bezogen sind: die Suche der längsten Sprachbausteine und der Ziel- und Verkettungskosten in der Bausteinauswahl. Zusätzlich stellen wir eine Strategie dar, um die Leistungsfähigkeit der vollständig gewichteten Koeffizientensuche durch ein statistisches Training und Perzeptronnetze zu erhöhen. Das statistische Training, Techniken, die an einem großen Korpus verwenden wurden, wird verwendet, um die Entscheidung über vorausgesagte Sprachfälle und ausgewählte Sprachbausteine der Korpusdatenbank zu treffen. Am Ende beschleunigen das Design der Korpusdatenbank und die nicht-gleichförmige Sprachbausteinauswahl des TTS Systems die Entwicklungs- und Laufzeit des TTS und verbessern die Sprachqualität.

1 Einführung

In den letzten Jahren wurde die Korpussprachsynthese in TTS Systemen untersucht, analysiert und angewendet [1], [2], [3]. Bei dieser Methode soll die Datenbank so entworfen sein, dass sie möglichst viele phonetische und prosodische Eigenschaften der Sprache abdeckt. Deswegen ist es notwendig, ein TTS System zu entwickeln, das eine effiziente Methode der Bausteinauswahl enthält und uns mit einer Datenbank mit hohem Datenvolumen, Varietät und Diversität umzugehen erlaubt. Das Modul der Bausteinauswahl in einem TTS System soll fähig sein, die beste Bausteinfolge für die Synthese eines Texteingangs durch die Minimierung der totalen Kostenfunktion in der Datenbank zu finden. Die totale Kostenfunktion besteht aus der gewichteten Summe von Ziel- und Verkettungskosten, die die Dauer-, F0- und Energie als Merkmale der Prosodie (Ziel) und die Linear spectral frequencies (LSFs), Multiple centroid analysis (MCAs) und Mel frequency cepstral coefficients (MFCCs) als Merkmale der Verkettung enthalten. In diesem Paper werden wir die Hauptkomponenten des KorpusDress Systems und seiner Datenbank in Abschnitt 2 beschreiben. In Abschnitt 3 werden wir ein neues Modell für die Bausteinauswahl in einer Datenbank vorstellen. In Abschnitt 4 werden die Schlüsse und die Diskussion behandelt.

2 Das TTS System KorpusDress 1

In diesem Abschnitt werden wir einen allgemeinen Überblick über die Hauptmodule, die das TTS KospusDress bilden, geben. Dieses System ist eine Weiterentwicklung seines

Vorgängers TTS MicroDress[4], weil verschiedene Module des MicroDress angepasst und weiterentwickelt wurden, um damit das TTS KorpusDress, das Abbildung 1 zeigt, zu erhalten.

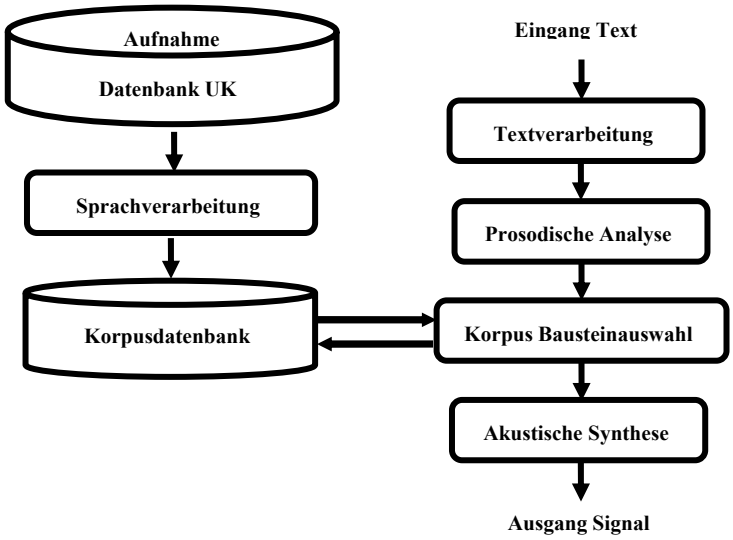

Abbildung 1 – Systemarchitektur des KorpusDress Systemes

2.1 Korpusdatenbank

Die Qualität der Sprachsynthese ist abhängig von der Qualität der Datenbank [6]. Dies bedeutet, dass die angemessene Bildung einer Sprachdatenbank schließlich eine gute Qualität in der Sprachsynthese liefert. Dafür beschreiben wir die drei nachfolgenden Module im linken Teil der Abbildung 1, die die Bildung der Korpusdatenbank gestalten.

2.1.1 Aufnahmen und Datenbank

Bei der Realisierung der Aufnahmen ist es sehr wichtig, die bestmöglichen Aufnahmebedingungen zu haben, um eine hohe Qualität der Sprachsynthese zu erhalten. Dafür haben wir eine Sprecherin von 5 verschiedenen Sprecherinnen englischer Muttersprache, die die notwendige Erfahrung in der Redewendung oder Interpretation der Sprache haben, ausgewählt. Die Haupteigenschaften der Aufnahmebedingungen sind in der folgenden Tabelle 1 zu sehen.

Abtastfrequenz	Bandbreite	SNR	Bits Precision
96 kHz (Downsampled 16 kHz)	40 Hz – 20 kHz	SNR > 40 dB	24 Bits

Tabelle 1 - Eigenschaften der Aufnahmen

Die erhaltene Sprachdatenbank hat eine Dauer von 20St. 18 min mit einer Gesamtzahl von 5558 Sätzen. Unter der Prämisse, dass die Qualität der Sprachsynthese abhängig von der Segmentierung und dem Labeln der Sprachdatenbank ist [7], ist es notwendig, die Qualität der handgelabelten Sätze zu bestimmen und so ein akzeptables Qualitätsniveau zu erhalten. Tabelle 2 zeigt die Anzahl und Verteilung der gelabelten Sätze..

	Handgelabelt – Dauer	Automatisch gelabelt - Dauer
N Sätze	1576 (28%) - 3 h 5 min (15%)	3982 (72%) - 17 h 13 min (85%)

Tabelle 2 - Eigenschaften des Labelns

Unsere Bausteinauswahlmethode von nicht-gleichförmigen Sprachbausteinen hat als grundsätzlichen Sprachbaustein das Diphonem. Die englische Sprache hat ungefähr 1702

Diphoneme, für diese kann man in der Abbildung 2 die Beziehung zwischen der Anzahl von Sätzen in unserer Sprachdatenbank und der Anzahl der Diphoneme, die die Sprachdatenbank abdeckt, sehen.

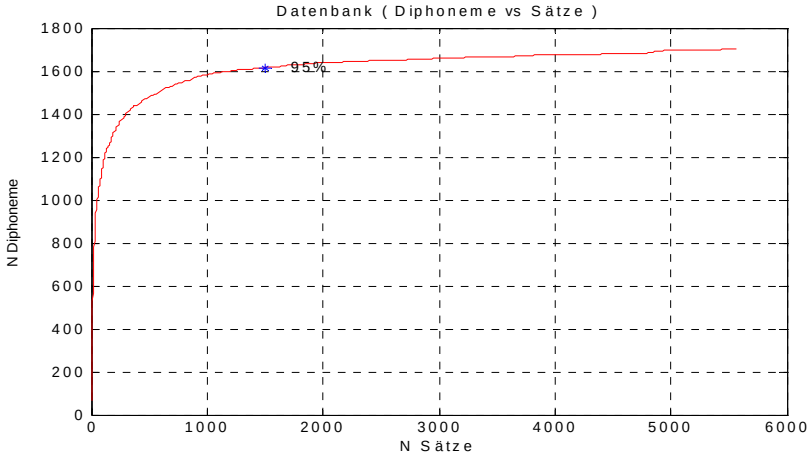

Abbildung 2 – Beschreibung der Datenbank

Aus der obigen Abbildung 2 können wir schließen, dass die Zahl der handgelabelten Sätze 95% der Gesamtzahl der Diphoneme der englischen Sprache abdeckt. Und daher erhalten wir die Anzahl der handgelabelten Sätze (1576), die uns eine minimale Qualität in der Sprachsynthese garantieren, weil 95% der Diphoneme fehlerfrei in ihrer Etikettierung sind.

2.1.2 Signalverarbeitung und Kontextverarbeitung

In diesem Ausschnitt der Verarbeitung unserer Sprachdatenbank findet die Offline-Berechnung der Eigenschaften, die zum Beispiel Dauer, F0, Energie, MCAs, LSFs und MFCCs sind, von jedem Sprachbaustein in unserer Sprachdatenbank statt. Die Berechnung der Dauer, F0 und Energie für jeden Sprachbaustein wurde auf [4]gegründet, damit werden wir später diese Information für die Berechnung der Zielkosten verwenden. Außerdem haben wir für beide Grenzen von jedem Sprachbaustein in unserer Sprachdatenbank die 9-MCA, 24-LSF y 26-MFCC-Koeffizienten [8][9] berechnet, die wir später für die Berechnung der Verkettungskosten zwischen zwei Sprachbausteinen, die deutlicher in der Abbildung 3 gezeigt ist, verwenden werden.

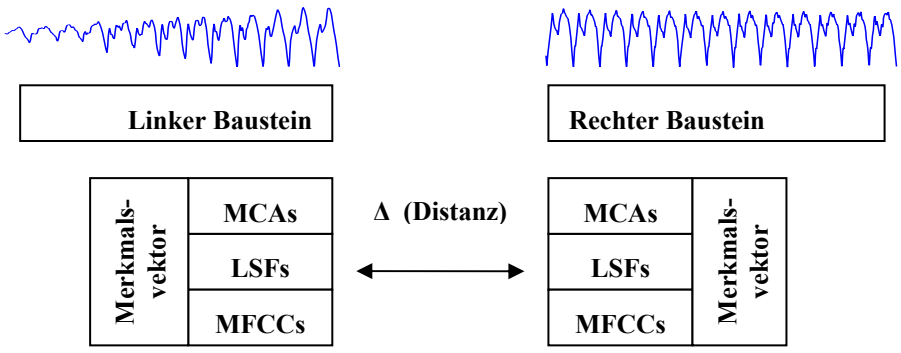

Abbildung 3 – Verkettungskosten

Zusätzlich verarbeiten wir die Informationen bezüglich des Kontextes des Sprachbausteins, um seine Position im Wort und im Satz zu bestimmen. Außerdem wurde die Betonung oder Nicht-Betonung des Sprachbausteines bestimmt.

2.1.3 Korpusdatenbank

Die Korpusdatenbank enthält alle Informationen bezüglich der Ziel- und Verkettungskosten. Zusätzlich enthält sie auch die Informationen bezüglich des Kontexts jedes Sprachbausteins. Damit ergänzt das Bausteinauswahlmodul [3] unsere Korpusdatenbank für ihre Anwendung im TTS System.

2.2 Textverarbeitung und prosodische Analyse

Diese zwei ersten Module setzen KorpusDress zusammen, deswegen ist es wichtig, ihr Funktionieren zu kennen. Mann kann diese zwei Module durch [4][5] tiefer analysieren, weil wir uns in diesem Paper nur mit dem Studium der Bausteinauswahl beschäftigen.

2.3 Akustische Synthese

In diesem Modul wurde die vorgerechnete prosodische Information mit der Sprachbausteinfolge, die vorher von der Bausteinauswahl gefunden wurde, verwendet. Mit dieser Information nutzen wir den TD-PSOLA Algorithmus [10], der sich durch eine Sprachsynthese mit Verständlichkeit und akzeptabler Natürlichkeit für den Benutzer bewährt hat.

3 Korpusbasierte Bausteinauswahl

Dieser Teil der TTS Systeme ist extrem wichtig, weil dieses Modul das Ergebnis in der Sprachsynthese bestimmt. Deshalb wurde dieses Modul aus vier Submodulen gebildet, die eine spezifizierte Funktion erfüllen, um von unserer Korpusdatenbank die Sprachbausteine, die die größte Qualität in der Sprachsynthese erreichen können, zu erhalten. Jedes der Submodule ist in Abbildung 4 zu sehen.

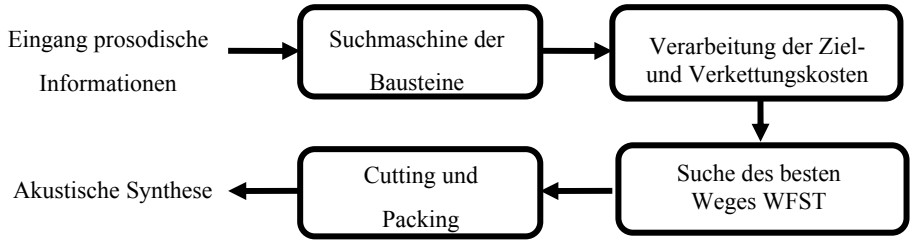

Abbildung 4 – Architektur der Korpusbausteinauswahl

3.1 Suchmaschine der Bausteine

Lange Sprachbausteine können besser die Natürlichkeit des Originalsprachsignals bewahren[11]. Deshalb haben wir eine Suchmaschine entwickelt, die die längste Sprachbausteinfolge in unserer Korpusprachdatenbank, die die erforderte Phonemsequenz vom TTS System abdeckt, findet.

3.2 Verarbeitung der Ziel- und Verkettungskosten

Sobald die verschiedenen Sprachbausteinfolgen in der Korpusdatenbank gefunden worden sind, die unseren Text zu einem synthetisierten Sprachsignal zusammensetzen können, ist es notwendig, die Distanz zwischen den prosodischen Eigenschaften von jedem Sprachbaustein und den erforderten prosodischen Eigenschaften des Prosodiemoduls zu berechnen. Dies wird in der folgenden Formel erläutert.

$$C(t^n, u^n) = \sum_{i=1}^{n} \sum_{j=1}^{p} w^t_j \square C^t(t^j_i, u^j_i) \qquad (1)$$

Wobei *p* die Zahl der prosodischen Eigenschaften (Dauer, F0 und Energie), *u* der Sprachbaustein und *n* die Zahl der Sprachbausteine ist. Dieser Prozess kann in der Abbildung 6 zwischen dem Zielmerkmalsvektor und dem Merkmalsvektorkandidat (Δ) veranschaulicht werden. Sobald die prosodische Distanz zwischen den erforderten Sprachbausteinen und den Kandidatensprachbausteinen berechnet worden ist, ist es auch notwendig, die Distorsion des Sprachsignals oder der Verkettungskosten, die auftreten können, wenn die Kandidatensprachbausteine verbunden wurden, zu berechnen. Die Berechnung der Verkettungskosten ist mathematisch durch die folgende Formel dargestellt.

$$C(u_{i-1}, u_i) = \sum_{i=2}^{n} \sum_{j=1}^{q} w_j^c \square C^c(u_{i-1}^j, u_i^j) \tag{2}$$

Wobei *q* die Zahl der spektralen Eigenschaften (MCAs, LSFs und MFCCs) ist, die am rechten Rand des Sprachbausteins u_{i-1} und am linken Rand des Sprachbausteins u_i durch einen Frame von 20 ms berechnet wurde, um so die Verkettungskosten einschätzen zu können. Der Prozess kann in der Abbildung 5 zwischen dem Vorgänger Merkmalsvektor u_{i-1} und dem Merkmalsvektorkandidat u_i veranschaulicht werden. Die Abbildung 5 stellt ausführlicher den Prozess der Verarbeitung der Ziel- und Verkettungskosten dar. Die Berechnung der Kostendistanz ist mit dem Symbol Δ erläutert und zusätzlich wurde eine Maskierungsfunktion hinzugefügt. Die Maskierungsfunktion hat das Ziel, einen Wert von 0 oder 1 in den Kosten für die Sprachbausteine zu erzielen, um die Sprachbausteine, die einen Wert in einer bestimmten Qualität zeigen, einzusetzen. Die Maskierungsfunktion ist in mehr Details im folgenden Abschnitt beschrieben.

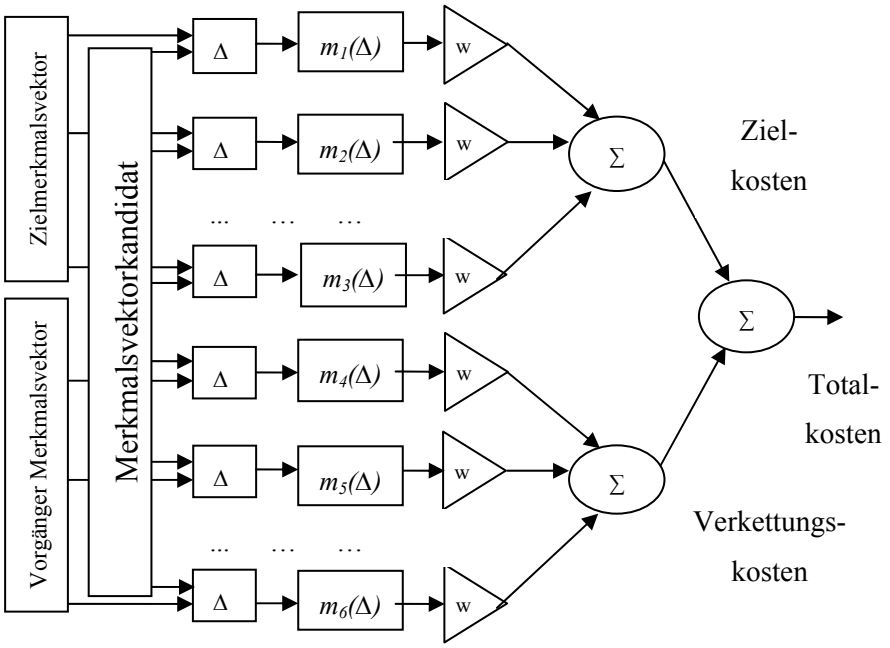

Abbildung 5 – Architektur der Korpusbausteinauswahl

3.2.1 Maskierungsfunktion und Deskriptive Statistik

Die Idee, eine Maskierungsfunktion zu formulieren, wird durch [14] vorgestellt, in der die besagte Funktion das Ziel erfüllt, die rechnerischen Kosten zu verringern, die die Berechnung der Ziel- und Verkettungskosten einbezieht. Dies wird mittels der Zuweisung des Wertes von 1 zu jenen Kosten, die außerhalb des Durchschnittsranges sind, und 0 zu jenen Werten, die innerhalb des Durchschnittsranges sind, erreicht. All dies wird in der Formel 3 erläutert.

$$m_k(\Delta) = \begin{cases} 0 & -\sigma \le \Delta \le \sigma \\ 1 & \text{andernfalls} \end{cases} \quad (3)$$

Wobei +/- den Rang darstellt, indem die Kosten 0 sind. Um die besagten Ränge zu bestimmen, haben wir die Verkettungskosten der kontinuierlichen Bausteine von unserer Korpusdatenbank berechnet. So erhielten wir die Verkettungskosten der Sprachbausteine ohne Distorsion, die wir die idealen Verkettungskosten nennen werden. Um dies alles zu realisieren, wurden die Sprachbausteine der Datenbank in Plosive, Afrikative, Frikative, Nasale, Liquide, Semivokale, Kurzvokale, Langvokale, Diphtonge, Sylcons und Pausen klassifiziert, mit denen die Verkettungskosten für jede Klassifikation berechnet wurden. Abbildung 6 zeigt die Verteilung der Verkettungskosten im Detail.

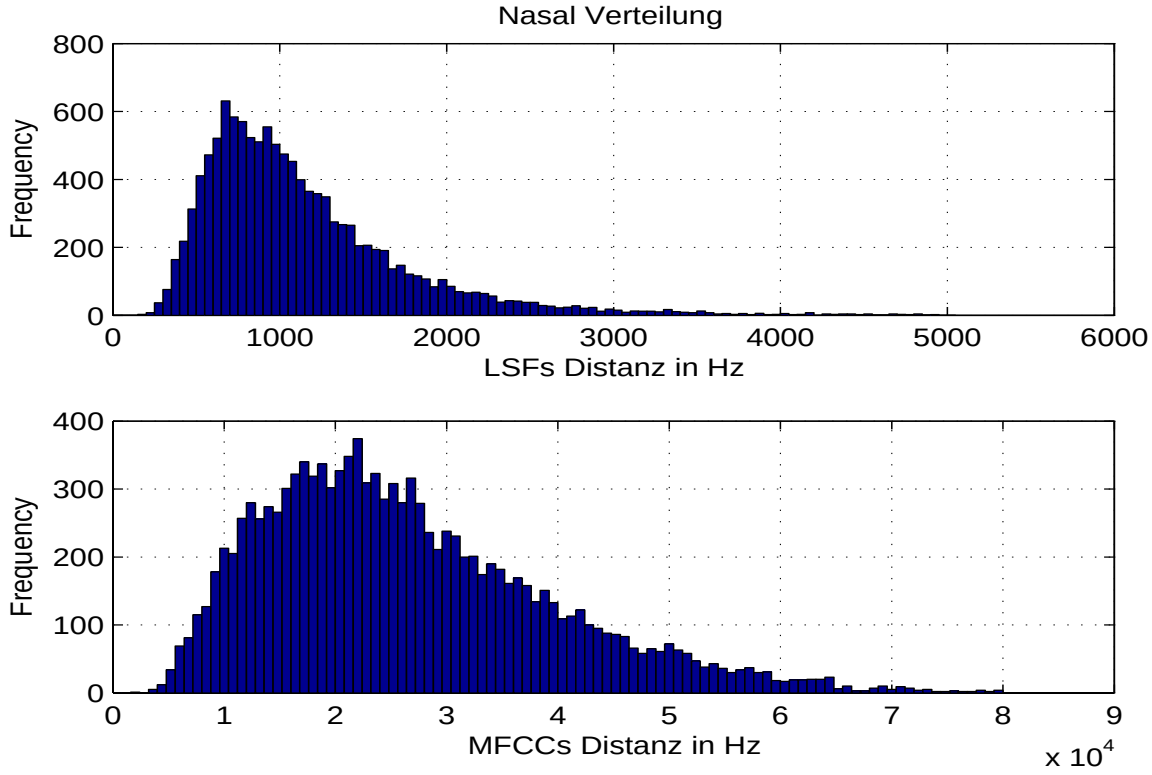

Abbildung 6 – Nasal Verteilung der LSF und MFCCs Verkettungskosten

Sobald der Durchschnitt und die Standardabweichung der Verteilungsfunktionen der Verkettungskosten berechnet worden sind, wurde der Rang der Maskierungsfunktionen ausgehend von +/- der Standardabweichung für jede Verteilung bestimmt. Am Ende des Prozesses haben wir einen Ausgang, der zwischen 1 und 0 variiert.

3.2.2 Gewichtete Koeffizientensuche

Das Tunen der gewichteten Koeffizientenparameter *w* in der Bausteinauswahl ist scheinbar der wichtigste Faktor für die Erhaltung der Konsistenz in der Sprachsynthese[6]. Deshalb verwenden wir das Neuronalnetz (RN) Perzeptron[12], das durch ein Training mit den Informationen der Ziel- und Verkettungskosten von nicht-kontinuierlichen und kontinuierlichen Sprachbausteinen der Korpusdatenbank angewandt wird. So trainieren wir das Perzeptron mit den Idealkosten der kontinuierlichen Sprachbausteine und mit den Kosten der nicht-idealen nicht–kontinuierlichen Bausteine. Das RN ist in der folgenden Abbildung 7 dargestellt.

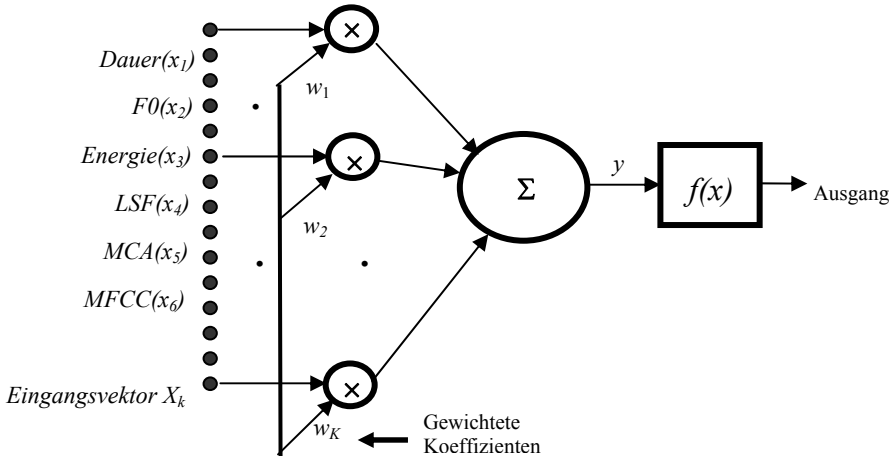

Abbildung 7 – Architektur des Perzeptron

Wobei *f(x)* eine Limitierungsfunktion (hardlims) ist, $x_k t$ sind die Abstände der Ziel- und Verkettungskosten nach der Maskierungsfunktion und w_k die gewichteten Koeffizienten. Durch diese Anwendung des RN bestimmen wir die gewichteten Koeffizienten der Ziel- und Verkettungskosten. Schließlich erhalten wir die zugehörigen Gewichte jener Eigenschaften in der Totalkostenfunktion, die prosodische oder spektrale (Verkettungs-)Eigenschaften sind.

3.3 Suche des besten Weges durch WFST

Sobald die frühere Koeffizientensuche und die Verarbeitung der Ziel- und Verkettungskosten beendet worden sind, können die gesagten Kosten für ihr besseres Verständnis und zur Veranschaulichung in einer Matrix wie in der Abbildung 8 dargestellt werden. Die Abbildung stellt die Matrix mit den Kosten dar, wobei die Zustände die Zielkosten und die Übergänge der Verkettungskosten darstellen.

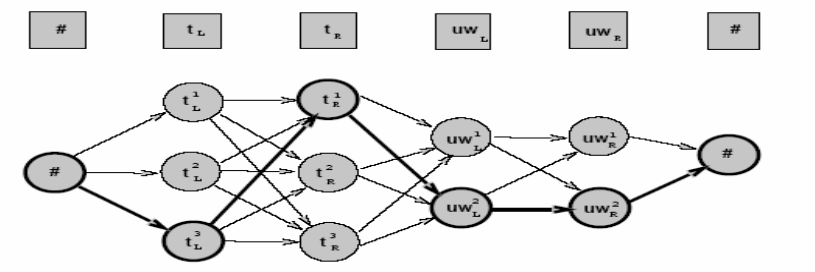

Abbildung 8 – Dartellung der Kostenmatrix

Die Funktion des Moduls ist, einen Weg durch die Matrix zu finden, der die kleinsten totalen Kosten hat und damit die Sprachbausteinfolge, die die beste Qualität in der Sprachsynthese erzeugt, zu erhalten. Der Suchprozess ist in der Abbildung 8 gezeigt und mathematisch in der folgenden Formel als Totalkosten dargestellt.

$$\vec{u}^{n} = \min_{u_1,\ldots,u_n} C(t^n, u^n) \tag{4}$$

Die Suche wurde durch Verwendung des Viterbi Algorithmus[9] durchgeführt.

3.4 Cutting und Packing

Dieser Prozess steht für die Extraktion der ausgewählten Sprachbausteine von der Korpusdatenbank. Das Cutting besteht darin [4][13], die Sprachbausteine der Sätze zu schneiden. Das Packing besteht in der Verarbeitung der Sprachbausteine für die Nutzung in der Akustiksynthese [4][10][13].

4 Schlussfolgerung

In diesem Paper haben wir das KorpusDress 1 Sprachsynthesesystem vorgestellt. Damit haben wir ein Framework mit einer flexiblen Plattform erschaffen, um eine gute Sprachsynthese zu erzeugen. Die Integration der Bausteinauswahl in dem TTS System erlaubte uns, die typischen Mismatches von der konkatenativen Sprachsynthese zu vermeiden. Zusätzlich hat sich die Prosodie verbessert, weil die Suche nach den Bausteinen in einer großen Korpusdatenbank uns ermöglicht hat, besser passende Bausteine für die geforderte Sprachsynthese zu finden. Deswegen können wir sagen, dass das KorpusDress System mit seinen Ergebnissen unsere Erwartungen erfüllt hat. In der Zukunft werden wir die Kostenfunktionen hinsichtlich neuer Aspekte weiter erforschen.

Literatur

[1] A. Hunt and A. Black, "Unit selection in a concatenative speech synthesis using a large speech database", in Proc. ICASSP ’96, pp. 373-376, 1996.

[2] M. Beutnagel, A. Conkie and A.K. Syrdal, "Diphone synthesis using unit selection”, Proc. 3rd ESCA/COCOSDA Int. Workshop on Speech Synthesis, Jenolan Caves, pp. 185-190, 1998.

[3] A.W. Black and N. Campbell, "Optimizing selection of units from speech databases for concatenatives synthesis.", Proc. Eurospeech ’95, Madrid, pp. 185-190, 1995.

[4] R. Hoffmann, O. Jokisch, H. Kruschke, G. Strecha, “microDRESS – a speech synthesis system with minimized footprint.”, Proc. 12th Czech-German Workshop Speech Processing, Prague, 2. – 4. pp. 9 – 12, September 2002,.

[5] R. Hoffmann, “Evaluation of a multilingual TTS system with respect to the prosodic quality”, Proc. Int. Cong. ICPhS, San Francisco, pp. 2307 – 2310, 1999.

[6] A. Black, "Perfect synthesis for all of the people all of the time.", Proc. IEEE Workshop on Speech Synthesis, 2002.

[7] A. Conkie, "A robust unit selection system for speech synthesis." In: Proc. 137 th meet. ASA/Forum Acusticum, Berlin, March 1999.

[8] A. Crowe and MA Jack., “Globally optimizing formant tracker using generalized centroids.”, Electronic Letters, Vol 23, No. 19, pp 1019-1020, 1987.

[9] R. Lawrence and J. Biing-Hwang “Fundamentals of speech recognition”, Prentice Hall 1993.

[10]E. Moulines , F. Charpentier, ”Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones”, Speech Communication, v.9 n.5-6, p.453-467, Dec. 1990.

[11] M. Lee, ”Perceptual cost functions for unit searching in large corpus-based concatenative text-to-speech.”, Proceedings of Eurospeech 2001. Aalborg,Denmark. 2001.

[12] N. Sing- Miller, C. Collins, “Trigger-Based language modeling using a loss-sensitive perceptron algorithm”, In proc. ICASSP , pp. IV-29-IV-32, Honolulu, Hawai, USA, 2007.

[13] R. Hoffmann, U. Kordon, S. Kürbis, K. Fellbaum, B. Ketzmerick, ”An interactive course on speech synthesis”. Proc. MATISSE, London, April 1999.

[14] G. Coorman, J. Fackrell, P. Rutten and B. Van Coile, “Segment selection in the L&H Realspeak laboratory TTS system.”, In Proc. ICSLP, 2:395-398, Beijing 2000.