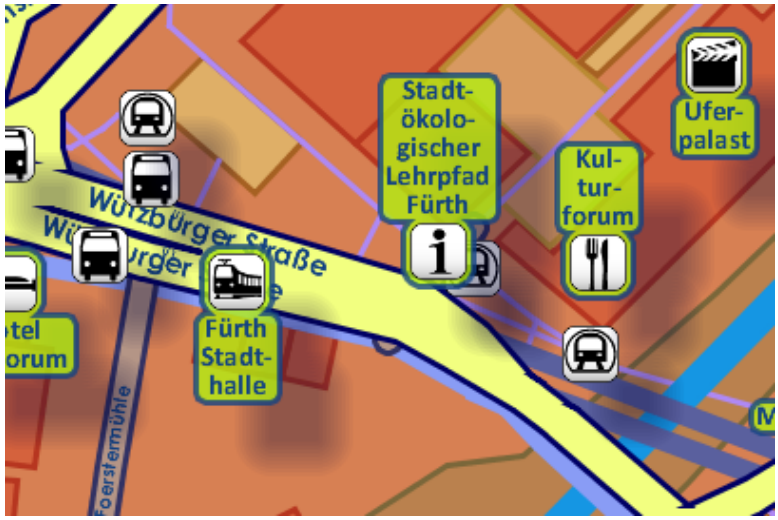
Vorschlag für eine Bachelor-Abschlussarbeit (Prof. Jörg Roth):

Konzeption und Realisierung eines Verfahrens für eine automatische Silbentrennung deutscher Worte

Ausgangslage

Eine automatische Silbentrennung ist mittlerweile eine Standardfunktion in Textverarbeitungssystemen. Durch Regelwerke und Wörterbücher wird eine hohe Trefferquote erreicht. Dennoch hat in Textverarbeitungssystemen der Autor die letzte Kontrolle und kann falsche Trennungen korrigieren.

Es gibt aber Fälle, in denen kein Benutzer eine Endkontrolle über die Silbentrennung durchführen kann. Ein typisches Beispiel ist die automatische Kartenbeschriftung. Hier werden leicht einige Millionen Beschriftungen durchgeführt – viele davon müssen über eine Silbentrennung optimiert dargestellt werden um beispielsweise Textblöcke möglichst überlappungsfrei darstellen zu können. Ein Benutzereingriff oder eine manuelle Überprüfung ist bei dieser großen Zahl nicht sinnvoll.



In einem solchen Fall hat man im Vergleich zur Textverarbeitung andere Ansprüche an die Silbentrennung:

- In den meisten Fällen soll eine korrekte Trennung durchgeführt werden.
- In geringem Maße kann eine falsche Silbentrennung toleriert werden, wenn sie nicht zu "absurd" ist.
- Im Falle einer Unsicherheit ist es besser, überhaupt keine Trennung durchzuführen als eine falsche.
- Die Laufzeit zur Berechnungen eines einzelnen Trennungsvorschlags muss sehr klein sein. Auch darf die Funktion zur Silbentrennung keine nennenswerten Speicherressourcen beanspruchen.

Das Problem

Die Silbentrennung im Deutschen ist nur vermeintlich einfach. Man kann Worte über Regeln in Phoneme umwandeln (bei denen beispielsweise "sch" ein einzelner Laut ist). Allerdings ist diese Zerlegung schon mehrdeutig. Z.B.

- diese Worte enden alle auf "schen", werden aber unterschiedlich getrennt:
  Häuschen → Häus-chen
  täuschen → täu-schen
- diese Worte enden alle auf "klein" und werden sehr unterschiedlich getrennt:
  Fünklein → fünk-lein
  klitzeklein → klitze-klein
  nuklein → nu-kle-in

Phoneme kann man schließlich zu Silben überführen – die Grundlage der Silbentrennung. Hierzu gibt es Muster, wie Silben aus Phonemen aufgebaut werden können, beispielsweise eine Unterteilung in Silbenkopf – Silbenkern – Silbenschwanz wie im Wort *Strand*:
- Silbenkopf: *Str*
- Silbenkern: *a*
- Silbenschwanz: *nd*

Problematisch bei einer Zerlegung sind aber beispielsweise schon zusammengesetzte Worte, da diese immer zwischen den einzelnen Teilworten getrennt werden. Das erfordert aber, dass die entsprechenden Teilworte identifizier werden. Als Beispiele:
- *Oberstrand*: könnte *Ober-strand* oder *Oberst-rand* (was immer das sein könnte) sein
- *Eispanne*: *Ei-spanne* oder *Eis-panne*?
- *Kursaal*: *Kur-saal* oder *Kurs-aal*?
- *Handelsturm*: *Handel-sturm* oder *Handels-turm*?

Solche Mehrdeutigkeiten werden derzeit über umfangreiche Wörterbücher behandelt. Zusätzlich wäre denkbar, dass bei Mehrdeutigkeiten *alle* Varianten mit einer Gewichtung zurückgegeben werden. Die Gewichtung könnte eine "Wahrscheinlichkeit" dafür repräsentieren, dass ein Vorschlag richtig ist. So gibt es beispielsweise bestimmte Wortendungen, bei denen *meistens* eine bestimmte Trennung vorliegt, z.B. Worte auf "lich", "bar", "keit", "ion" und "ung". Die benutzende Anwendung könnte dann auf der Basis der Gewichtung entscheiden, ob ein Vorschlag verwendet wird. Liegen beispielsweise mehreren Vorschläge mit ähnlichen Gewichten oder eine einzelne Trennung mit einem geringen Gewicht vor, könnte auf eine Trennung komplett verzichtet werden.
Weiter ist denkbar, dass jede einzelne Trennungsposition in einem Wort mit einem Verlässlichkeitsfaktor bewertet wird. Die Anwendung könnte dann beispielsweise nur solche Trennungspositionen verwenden, die ein bestimmtes Niveau überschreiten.

Ziel

Ziel der Arbeit ist, ein Rahmenwerk zu schaffen, das die Silbentrennung in der deutschen Sprache durchführt. Die Realisierung soll ohne fremde Rahmenwerke oder Bibliotheken auskommen. Das Resultat soll eine Software-Bibliothek sein, die man einfach in eigene Projekte integrieren kann.

Die Arbeitspunkte im Einzelnen:
- Recherche über existierende automatische Ansätze zur Silbentrennung,
- Recherchieren des Regelwerks für die Silbentrennung im Deutschen,
- Konzeption eines Formalismus, der dieses Regelwerk abbildet,
- Konzeption eines Wörterbuchverfahrens, Aufbau eines prototypischen Wörterbuchs,
- Realisierung, Test und Bewertung des Ansatzes.

Vorraussetzungen

Diese Arbeit ist für jemanden geeignet, der sich mit einer linguistischen Fragestellung in der deutschen Sprache auseinandersetzen möchte.

Die Implementierung erfolgt in Java. Es handelt sich aber um eine Software-Lösung, die keine graphische Benutzungsoberfläche, Netzwerk- oder Datenbankprogrammierung erfordert. Damit sind Basiskenntnisse oder Kenntnisse in einer ähnlichen Programmiersprache ausreichend.