

Rheinisch-Westfälische Technische Hochschule Aachen
Lehrstuhl für Informatik 6
Prof. Dr.-Ing. Hermann Ney

Seminar Selected Topics in Human Language Technology and Pattern Recognition im SS
2016

Attention-based Neural Machine Translation

Derui ZHU

Matrikelnummer 352273

Betreuer: Parnia Bahar

<i>INHALTSVERZEICHNIS</i>	3
---------------------------	---

Inhaltsverzeichnis

1 Introduction	5
1.1 Rule-based Translation	5
1.2 Stastical-based Translation	5
2 Neural Nerwork	6
3 Neural Machine Translation	6
4 Attention-based Neural Machine Translation	6
5 Evaluation	6
6 Introduction	6
7 Neural Network	7
7.1 Unterabschnitt	7
8 Neural Machine Translation	7
9 Attention-based Neural Machine Translation	7
9.0.1 Unter-Unterabschnitt: Beispiel für Tabellen	7
9.1 Unterabschnitt: Beispiel für Grafiken	7
10 Evaluation	8
Literaturverzeichnis	8

Tabellenverzeichnis

1 Beispiel Tabelle	7
------------------------------	---

Abbildungsverzeichnis

1 Example of rule-based translation process	5
2 string-based and tree-based translation models	6
3 Architektur eines Spracherkennungssystems	8

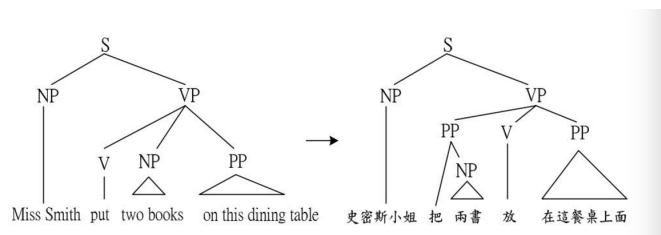


Abbildung 1: Example of rule-based translation process

1 Introduction

Machine translation is one important research field of natural language processing. In general, machine translation is the process of transferring one natural language into another natural language. From early lexicon translation to corpus-based statistical translation, machine translation technologies experience a huge improvement.

Recalling the history of machine translation development, before 1980s, rule-based translation were the most important approaches. In the end of 1980s, statistical translation approaches were proposed, the situation changed. It rapidly becomes mainstream approaches. Recently, neural network-based translation has shown its huge power.

1.1 Rule-based Translation

Rule-based translation is a process that generate target language by analysing source language's structure and building a meaningful representation. Rule-based approaches are mainly divided into transfer-based and interlingua-based translation process. Interlingua-based approach generate intermediate language first after analysing source language. Then the target language is produced by the intermediate language. Most of the rule-based translation system uses transfer-based translation process. It is composed of three steps. Firstly, the system generates an abstract representation of source language. Then, the abstract representation of source language is transferred into the abstraction representation of target language. In the end, the abstraction representation of target language generates target language.

For example, we consider to translate the english sentence "Miss Smith put two books on this dining table" into chinese. First of all, the system praser its morphological syntax to get the syntax tree(below [2]). During the transfer phase, lexical and syntax transform are excuted at the same time. After that, we get the target words, which locate in the nodes of syntax tree generated.

1.2 Stastical-based Translation

The basic idea of statistical approach is to obtain translation rules and probability parameters automatically from bilingual corpus by employing statistical machine learning technologies. Graphic 1 shows string-based and tree-based translation model, which are used in statistical translation widely. Phrase-based model is a mature tranlation approach. The phrase means successive word sequences rather than the phrase defined in linguistics.

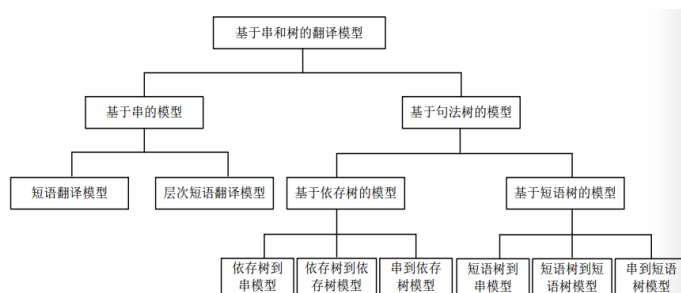


Abbildung 2: string-based and tree-based translation models

2 Neural Network

3 Neural Machine Translation

4 Attention-based Neural Machine Translation

5 Evaluation

6 Introduction

Dies ist eine Referenz auf Abschnitt 6

Erzeugen wir erst etwas Text, um uns mit der Schreibweise in \LaTeX vertraut zu machen. \LaTeX bricht selbständig Zeilen und Seiten um. Selbst in einem Neogloismus wie dem ziemlich langen Wort aus Michael Endes zuletzt geschriebenem Buch „Der satan-archäolügenialkohöllische Wunschpunsch“ findet \LaTeX eine gute Trennstelle. Manchmal ist es aber sinnvoll, selber eine Trennstelle vorzugeben. Dies gibt man mit `\-` im Wort im \TeX -File selbst an, also z.B. Trenn\–stelle.

Lässt man eine Zeile im \TeX -File frei so wird automatisch die nächste Zeile eingerückt. Wem das nicht gefällt, der kann im \TeX -File den Befehl `\parindent0pt` auskommentieren und das damit unterdrücken. Text mit ä, ö, Ü, Straße, Café.

Dies ist ein laufender-Text mit Einsteins Formel $E = m \cdot c^2$.

Eine etwas kompliziertere Formel:

$$a \frac{1 - q^{n+1}}{1 - q} = \sum_{i=0}^n a q^i \quad \text{mit} \quad a, q \in \mathcal{R}, q \neq 1$$

Nun folgt die Bayessche Entscheidungsregel:

Tabelle 1: Beispiel Tabelle

links	zentriert	rechts
links2	zentriert2	rechts2
XXXXXXXXXXXXX	XXXXXXXXXXXXXXXXXXXXXXXXXXXXX	XXXXXXXXXXXXX

$$r(X) = \operatorname{argmax}_{w_1 \dots w_N} \left\{ Pr(w_1^N | x_1^T) \right\} \quad \text{mit} \quad Pr(w_1^N | x_1^T) = \frac{Pr(x_1^T | w_1^N) \cdot Pr(w_1^N)}{Pr(x_1^T)} \quad . \quad (1)$$

Beispiel für ein lineares Gleichungssystem:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= y_1 \\ a_{21}x_1 + a_{22}x_2 &= y_2 \\ a_{31}x_1 &= y_3 \end{aligned}$$

7 Neural Network

7.1 Unterabschnitt

8 Neural Machine Translation

9 Attention-based Neural Machine Translation

9.0.1 Unter-Unterabschnitt: Beispiel für Tabellen

In der Umgebung zum Plazieren von Tabellen (und auch Grafiken) kann man optional die Werte t, b, h und H verwenden. Damit lassen sich Tabellen am oberen Rand (top), unten (bottom) und an der aktuellen Position (here) plazieren. Je nach Seitenlayout funktioniert die Angabe h nicht immer. Mittels H lässt sich dann die aktuelle Position erzwingen.

9.1 Unterabschnitt: Beispiel für Grafiken

Grafiken werden üblicherweise mit dem (frei verfügbaren) Programm tgif erstellt. Alternativ hierzu lässt sich auch das Programm xfig verwenden. tgif ist zwar nicht ganz einfach zu bedienen, stellt dem (erfahrenen) Anwender aber wesentlich mehr Möglichkeiten zur Gestaltung von Grafiken zur Verfügung. Die Konvertierung der *.obj Dateien nach encapsulated PostScript erfolgt über die Kommandozeile mit

```
tgif -print -eps <Datei>.obj
```

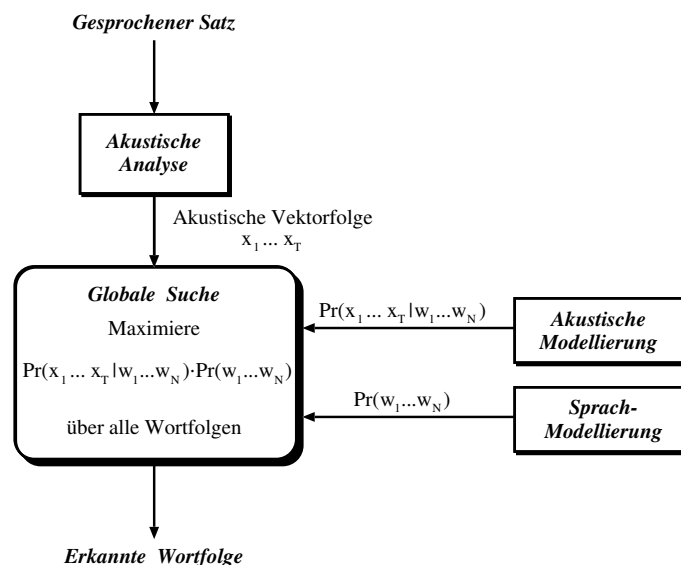


Abbildung 3: Architektur eines Spracherkennungssystems

Nun erzeugen wir mittels $\text{BIB}\text{T}_{\text{E}}\text{X}$ eine Literaturliste. Dazu erzeugen wir uns ein File namens `<Dateiname>.bib`, das denselben Dateinamen wie unser $\text{T}_{\text{E}}\text{X}$ -File hat. Dieses wird dann mit

```
bibtex <Dateiname>
```

compiliert. Die im laufenden Text vorkommenden symbolischen Literaturverweise zeigen dann auf den entsprechenden Eintrag im Literaturverzeichnis. Dies geschieht mit dem Befehl `\cite{}`. Das sieht dann z.B. wie folgt aus.

Zitat aus [1], Zitat aus [3], Zitat aus [2]. Wichtig ist, dass zuerst das $\text{T}_{\text{E}}\text{X}$ -File compiliert wird und dann das $\text{BIB}\text{T}_{\text{E}}\text{X}$ -File. Da sich dann die Referenzen meist ändern, ist ein erneutes Compilieren des $\text{T}_{\text{E}}\text{X}$ -Files notwendig.

Es gibt zahlreiche style-files, mit denen man das Layout eines Literaturverzeichnisses ändern und die Darstellung der Literatur-Referenzen modifizieren kann.

Letzter Hinweis: $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ -Neulinge sollten auch die Kommentare im $\text{T}_{\text{E}}\text{X}$ -File lesen.

10 Evaluation

Literatur

- [1] Autor1, Autor2, and Autor3. Titel des Papers. In *Name des Bandes, in dem Artikel veröffentlicht wurde*, volume Band, pages xxx–yyy, Monat 1991.
- [2] Autor1, Autor2, and Autor3. Titel des Reports. Technical report, Name des Instituts, Monat 1996.
- [3] M. Ende. *Der Wunschpunsch*. Tiemann Verlag, 1989.