Finding canonical forms for historical German text

— DRAFT —

Bryan Jurish

jurish@bbaw.de

Berlin-Brandenburg Academy of Sciences · Jägerstrasse 22/23 · 10117 Berlin · Germany

July, 2008

Abstract

Historical text presents numerous challenges for contemporary natural language processing techniques. In particular, the absence of consistent orthographic conventions in historical text presents difficulties for any technique or system requiring reference to a fixed lexicon accessed by orthographic form. This paper presents two methods for mapping unknown historical text types to one or more synchronically active canonical types: conflation by phonetic form, and conflation by lemma instantiation heuristics. Implementation details and evaluation of both methods are provided for a corpus of historical German verse quotation evidence from the digital edition of the *Deutsches Wörterbuch*.

Contents

1 Introduction

Historical text presents numerous challenges for contemporary natural language processing techniques. In particular, the absence of consistent orthographic conventions in historical text presents difficulties for any technique or system requiring reference to a fixed lexicon accessed by orthographic form, such as document indexing systems (e.g. Sokirko, 2003), part-of-speech taggers (e.g. DeRose, 1988; Brill, 1992; Schmid, 1994; Jurish, 2003), simple word stemmers (e.g. Lovins, 1968; Porter, 1980), or more sophisticated morphological analyzers (e.g. Geyken and Hanneforth, 2006). When adopting historical text into such a system, one of the most important tasks is the discovery of one or more *canonical extant forms* for each word of the input text: synchronically active text types which best represent the historical input form.[1]

The process of collecting variant forms into equivalence classes represented by one or more canonical extant types is commonly referred to as *conflation*, and the equivalence classes themselves are referred to as *conflation sets*. Given a high-coverage analysis function for extant forms, an unknown (historical) form w can then be analyzed as the disjunction of analyses over (the extant members of) its conflation set $[w]$:

$$\text{analyses}(w) \quad := \quad \bigcup_{v \in [w]} \text{analyses}(v)$$

This paper describes two methods for finding conflation sets in a corpus of *circa* 5.5 million words of historical German verse extracted from quotation evidence in the digital edition of the *Deutsches Wörterbuch* (*DWB*, Bartz et al., 2004), and indexed with the the TAXI document indexing system. The conflation methods were implemented on the entire corpus as a TAXI plug-in module (TAXI/Grimm), and evaluated with respect to coverage by the TAGH morphology.

The rest of this paper is organized as follows: Section 2 describes the first conflation strategy, based on identity of phonetic forms. The second strategy making use of *a priori* assumptions regarding corpus structure and permitting "fuzzy" matching via phonetic edit distance is presented in Section 3. Finally, Section 4 contains a brief summary of the preceding sections and a sketch of the ongoing development process.

2 Conflation by Phonetic Form

Although the lack of consistent orthographic conventions for middle high German and early new high German texts led to great diversity in surface graphemic forms, we may assume that graphemic forms were constructed to reflect phonetic forms. Under this assumption, together with the assumption that the phonetic system of German is diachronically more

[1] As an anonymous reviewer pointed out, the absence of consistent orthographic conventions is not restricted to corpora of historical text. Various other types of text corpora – including transcriptions of spoken language, corpora containing transcription errors, and corpora for languages with non-standard orthography – might also benefit from a canonicalization strategy such as those presented here.

stable than the graphematic system, the phonetic form of a word type should provide a better clue to the extant lemma of a historical word than its graphemic form. This insight is the essence of the "conflation by phonetic form" strategy as implemented in the TAXI/Grimm index module.

In order to map graphemic forms to phonetic forms, we may avail ourselves of previous work in the realm of text-to-speech synthesis, a domain in which the discovery of phonetic forms for arbitrary text is a well-known and often-studied problem (cf. Allen et al., 1987; Liberman and Church, 1992; Dutoit, 1997), the so-called "grapheme-to-phoneme", "grapheme-to-phone", or "letter-to-sound" (LTS) conversion problem. Use of a full-fledged LTS conversion module to estimate phonetic forms provides a more flexible and finer-grained approach to canonicalization by phonetic form than strategies using language-specific phonetically motivated digest codes such as those described in Robertson and Willett (1993). The grapheme-to-phone conversion module in the TAXI/Grimm system uses the LTS rule-set distributed with the IMS German Festival package (Möhler et al., 2001), a German language module for the Festival text-to-speech system (Black and Taylor, 1997; Taylor et al., 1998).

2.1 Implementation

As a first step, the IMS German Festival letter-to-sound (LTS) rule-set was adapted to better accommodate both historical and contemporary forms; assumedly at the expense of precision for both historical and contemporary forms. In particular, the following changes were made:

1. By default, the grapheme "h" is ignored (considered silent).
2. A single additional rule maps the grapheme sequence "sz" to voiceless */s/*.
3. Vowel-length estimates output by the IMS German rule-set are ignored; thus */e/* and */e:/* are both mapped to the canonical phonetic form */e/*.
4. Schwas (*/ə/*) predicted by the IMS German rule-set are replaced by */e/* in the canonical phonetic form.
5. Adjacent occurrences of any single vowel predicted by the IMS German rule-set are replaced by a single occurrence, thus */aa/*, */aaa/*, and */aaaa/* are all mapped to */a/*.

The adapted rule-set was converted to a *deterministic finite state transducer* (Aho and Ullman, 1972; Roche and Schabes, 1997) using the GFSM finite state machine utility library. Formally, the finite state transducer (FST) used by the TAXI/Grimm LTS module is defined as the machine M_{LTS} arising from the composition of two *Aho-Corasick pattern matchers* (Aho and Corasick, 1975) M_L, M_R and an additional *output filter* M_O:

$$M_{LTS} = (M_L \circ M_R \circ M_O) : \mathcal{A}_g^* \rightarrow \mathcal{A}_p^* \tag{1}$$

where $\mathcal{A}_g$ is the finite *grapheme alphabet* and $\mathcal{A}_p$ is the finite *phone alphabet*. To define the individual component machines, let R be the (IMS German) Festival LTS rule-set source, a finite set of rules of the form $(\alpha[\beta]\gamma \rightarrow \pi) \in \mathcal{A}_g^* \times \mathcal{A}_g^+ \times \mathcal{A}_g^* \times \mathcal{A}_p^*$, read as: the *source grapheme string* β is to be mapped to the *target phonetic string* π if β occurs with *left graphemic context* α and *right graphemic context* γ; let $\prec$ be a linear *precedence order* on R which prevents multiple rules from applying to the same source substring (only the $\prec$-minimal rule is applied at each source position, proceeding from left to right); for a nonempty rule subset $S \subseteq R$, let $(\alpha_S[\beta_S]\gamma_S \rightarrow \pi_S) = \min_{\prec} S$; let $\text{AhoCorasick}(P) : \mathcal{A}^* \rightarrow \wp(P)^*$ be the Aho-Corasick pattern matcher for a set P of string patterns from a finite alphabet $\mathcal{A}$; let $|\cdot|$ denote string length or set cardinality, depending on context; let $\text{reverse}(\cdot)$ denote the transducer reversal operation, and let $\text{Concat}(\cdots)$ denote the string concatenation operation, then:

$$\begin{aligned} M_L &\approx \text{AhoCorasick}\left(\{\alpha : (\alpha[\beta]\gamma \rightarrow \pi) \in R\}\right) \\ &: \mathcal{A}_g^* \rightarrow (\mathcal{A}_g \times \wp(R))^* \\ &: w \mapsto \text{Concat}_{i=0}^{|w|} \left\langle w_i, \left\{(\alpha[\beta]\gamma \rightarrow \pi) \in R \mid w_{(i-|\alpha|)..i} = \alpha\right\}\right\rangle \end{aligned} \tag{2}$$

$$\begin{aligned} M_R &\approx \text{reverse}\left(\text{AhoCorasick}\left(\{(\beta\gamma)^{-1} : (\alpha[\beta]\gamma \rightarrow \pi) \in R\}\right)\right) \\ &: (\mathcal{A}_g \times \wp(R))^* \rightarrow \wp(R)^* \\ &: \langle w_i, S_i\rangle_I \mapsto \text{Concat}_{i\in I}\left(S_{i-1} \cap \left\{(\alpha[\beta]\gamma \rightarrow \pi) \in R : w_{i..(i+|\beta\gamma|)} = \beta\gamma\right\}\right) \end{aligned} \tag{3}$$

A similar construction also using a pair of Aho-Corasick pattern matchers (analagous to M_L and M_R) is employed by Laporte (1997) for compiling a single bimachine from a set of conflict-free hand-written phonetic conversion rules. Since `festival` LTS rule-sets are not conflict-free, Laporte's technique cannot be applied directly here, and the choice of which rule to apply must be delayed until application of the filter transducer M_O:

$$\begin{aligned} M_O &\approx \left(\bigcup_{S\in\wp(R)} \left[(S : \pi_S)\left(\wp(R) : \varepsilon\right)^{|\beta_S|-1}\right]\right)^* \\ &: \wp(R)^* \rightarrow \mathcal{A}_p^* \end{aligned} \tag{4}$$

In the interest of efficiency, the rule subsets $S \in \wp(R)$ on the lower tape of the filter transducer M_O can be restricted to those which actually occur on the upper tape of the right-context transducer M_R: such a restriction represents a considerable efficiency gain with respect to the "brute force" powerset construction given in Equation 4. Figure 1 shows an example of how the various machine components work together to map the graphemic form "sache" to the phonetic form */zaxə/*.

Finally, phonetic forms are used to conflate graphemic variants $w \in \mathcal{A}$ as equivalence classes $[w]_{\text{pho}}$ with respect to the *phonetic equivalence relation* $\equiv_{\text{pho}}$ on the corpus word-type alphabet $\mathcal{A} \subset \mathcal{A}_g^*$:

$$w \equiv_{\text{pho}} v \quad :\Leftrightarrow \quad M_{LTS}(w) = M_{LTS}(v) \tag{5}$$

$$[w]_{\text{pho}} = \{v \in \mathcal{A} : w \equiv_{\text{pho}} v\} \tag{6}$$

Note that the equivalence class generating function $[\cdot]_{\text{pho}} : \mathcal{A} \rightarrow \wp(\mathcal{A})$ can itself be characterized as a finite state transducer, defined as the composition of the LTS transducer with its inverse, and restricted to the alphabet $\mathcal{A}$ of actually occurring corpus word-types:

$$[\cdot]_{\text{pho}} \quad := \quad M_{LTS} \circ M_{LTS}^{-1} \circ \text{Id}\,(\mathcal{A}) \tag{7}$$

Input	#	s	a	c	h	e	#
$\mathbf{M_L}$ $\longrightarrow$	$\emptyset$	$[a]ch\rightarrow a$ $[a]\ \rightarrow a\text{ː}$, $[c]\ \rightarrow k$, $[e]\ \rightarrow \text{ə}$, $\#[s]a\ \rightarrow z$, $[s]\ \rightarrow s$	$[a]ch\rightarrow a$, $[a]\ \rightarrow a\text{ː}$, $[c]\ \rightarrow k$, $[e]\ \rightarrow \text{ə}$, $[s]\ \rightarrow s$	$[\ a\ ]ch\rightarrow a$, $[\ a\ ]\ \rightarrow a\text{ː}$, $a[ch]\ \rightarrow x$, $[\ c\ ]\ \rightarrow k$, $[\ e\ ]\ \rightarrow \text{ə}$, $[\ s\ ]\ \rightarrow s$	$\emptyset$	$[a]ch\rightarrow a$, $[a]\ \rightarrow a\text{ː}$, $[c]\ \rightarrow k$, $[e]\ \rightarrow \text{ə}$, $[s]\ \rightarrow s$	$\emptyset$
$\mathbf{M_R}$ $\longleftarrow$	$\emptyset$	$\#[s]a\rightarrow z$, $[s]\ \rightarrow s$	$[a]ch\rightarrow a$, $[a]\ \rightarrow a\text{ː}$	$a[ch]\rightarrow x$, $[\ c\ ]\rightarrow k$	$\emptyset$	$[e]\rightarrow \text{ə}$	$\emptyset$
$\mathbf{M_O}$ $\longrightarrow$	ε	**z**	**a**	**x**	ε	**ə**	ε

Figure 1: Example Letter-to-Sound Transduction from "sache" to/*zaxə*/. Here, italic "ε" indicates the empty (phonetic) string.

2.2 Performance

LTS Method	**Throughput (tok/sec)**	**Relative**
`festival` (TCP)	28.53	−4875.57 %
`festival` (pipe)	1391.45	± 0.00 %
FST (`libgfsm`)	9124.69	+ 555.77 %

Table 1: Performance results for LTS FST *vs.* direct communication with a `festival` process

A finite state LTS transducer M_{LTS} was compiled from the 396 rules of the adapted IMS German Festival rule-set using the procedure sketched above. The resulting transducer contained 131,440 arcs and 1,037 states, of which 292 were final states. The compilation

Extant Form w	**Phonetic Equivalence Class** $[w]_{\text{pho}}$
fröhlich	*frölich, fröhlich, vrœlich, frœlich, frȍlich, frȍhlich, vrölich, frölig, ...*
Herzenleid	*hertzenleid, herzenleid, herzenleit, hertzenleyd, hertzenleidt, herzenlaid, hertzenlaid, hertzenlaidt, hertzenlaydt, herzenleyd, ...*
Hochzeit	*hochtzeit, hochzeit, hochzeyt, hochzît, hôchzît, hochzeid, ...*
Schäfer	*schäfer, schäffer, scheffer, scheppher, schepher, schȁfer, schähffer, ...*

Table 2: Some words conflated by identity of phonetic form

lasted less than 30 seconds on a workstation with a 1.8GHz dual-core processor. Performance results for the transducer representation of the LTS rule-set and for two methods using `festival` directly are given in Table 1. As expected, the transducer implementation was considerably faster than either of the methods communicating directly with a `festival` process.

2.3 Coverage

The phonetic conflation strategy was tested on the full corpus of the verse quotation evidence extracted from the DWB, consisting of 6,581,501 tokens of 322,271 distinct graphemic word types. A preprocessing stage removed punctuation marks, numerals, and known foreign-language material from the corpus. Additionally, a rule-based graphemic normalization filter was applied which maps UTF-8 characters not occurring in contemporary German orthography onto the ISO-8859-1 (Latin-1) character set (e.g. *œ,* $\mathring{o}$, and *ô* are mapped to *oe,* $\ddot{o}$, and *o*, respectively). After preprocessing and filtering, the corpus contained 5,491,982 tokens of 318,383 distinct ISO-8859-1 encoded graphemic types.

Of these 318,383 Latin-1 word types occurring in the corpus, 135,070 (42.42%) were known to the TAGH morphology (Geyken and Hanneforth, 2006), representing a total coverage of 4,596,962 tokens (83.70%). By conflating those word types which share a phonetic form according to the LTS module, coverage was extended to a total of 173,877 (54.61%) types, representing 5,028,999 tokens (91.57%). Thus, conflation by phonetic form can be seen to provide a reduction of 21.17% in type-wise coverage errors, and of 48.27% in token-wise coverage errors. Some examples of word types conflated by the phonetic canonicalization strategy are given in Table 2.

3 Conflation by Lemma Instantiation Heuristics

Despite its encouragingly high coverage, conflation by identity of phonetic form is in many cases too strict a criterion for lemma-based canonicalization – many word pairs which intuitively should be considered instances of the same lemma are assigned to distinct phonetic equivalence classes. Examples of such desired conflations undiscovered by the phonetic conflation strategy include the pairs *(abbrechen, abprechen)*, *(geschickt, geschicket)*, *(gut,*

guot), *(Licht, liecht)*, *(Teufel, tiuvel)*, *(umgehen, umbgehn)*, *(voll, vol)*, and *(wollen, wolln)*. In an attempt to address these shortcomings of the phonetic conflation method, additional conflation heuristics were developed which make use of the dictionary structure of the TAXI/Grimm corpus in order to estimate and maximize a *lemma instantiation likelihood* function.

3.1 Implementation

The TAXI/Grimm corpus is comprised of *verse quotation evidence* drawn from a dictionary corpus (Bartz et al., 2004). It is plausible to assume that each of the quotations occurring in an article for a particular dictionary lemma contain some variant of that lemma – otherwise there would not be much sense including the quotation as "evidence" for the lemma in question.

Working from this assumption that each quotation contains at least one variant of the dictionary lemma for which that quotation appears as evidence, a lemma instantiation conflation heuristic has been developed which does not require strict identity of phonetic forms – instead, *string edit distance* (Levenshtein, 1966; Wagner and Fischer, 1974; Navarro, 2001) on phonetic forms is used to estimate similarity between each word in the corpus and each of the dictionary lemmata under which it occurs. Further, inspired by previous work in unsupervised approximation of semantics and morphology (Church and Hanks, 1990; Yarowsky and Wicentowski, 2000; Baroni et al., 2002), *pointwise mutual information* (McGill, 1955; Cover and Thomas, 1991; Manning and Schütze, 1999) between dictionary lemmata and their candidate instances is employed to detect and filter out "chance" similarities between rare lemmata and high-frequency words.

Formally, the lemma instantiation heuristics attempt to determine for each quotation q which phonetic type i occurring in q best instantiates the dictionary lemma ℓ associated with the article containing q. For $\mathcal{A}$ the set of all word types occurring in the corpus, $\mathcal{L} \subseteq \mathcal{A}$ the set of all dictionary lemmata, and $\mathcal{Q} \subseteq \wp(\mathcal{A}^*)$ the set of all quotations:

$$\begin{aligned} \text{bestInstance}(\cdot) &: \mathcal{Q} \to \mathcal{A} \\ &: q \mapsto \arg\max_{w \in q} \mathrm{L}(M_{LTS}(w), M_{LTS}(\text{lemma}(q))) \end{aligned} \tag{8}$$

where the probabilities $\mathrm{P}(\ell, i), \mathrm{P}(\ell)$, and $\mathrm{P}(i)$ used to compute pointwise mutual information are first instantiated by their maximum likelihood estimates over the entire corpus:

$$\mathrm{P}(\ell, i) = \frac{\sum_{w_i \in M_{LTS}^{-1}(i)} \sum_{w_\ell \in M_{LTS}^{-1}(\ell)} \mathrm{f}(Token = w_i, Lemma = w_\ell)}{|Corpus|} \tag{9}$$

$$\mathrm{P}(\ell) = \sum_i \mathrm{P}(\ell, i) \tag{10}$$

$$\mathrm{P}(i) = \sum_\ell \mathrm{P}(\ell, i) \tag{11}$$

Raw bit-length pointwise mutual information values $\tilde{\mathrm{I}}(\ell, i)$ are computed and normalized to the unit interval $[0, 1]$ for each lemma and candidate instance, defining $\tilde{\mathrm{I}}(i|\ell)$ and $\tilde{\mathrm{I}}(\ell|i)$

respectively:

$$\tilde{\mathrm{I}}(\ell, i) = \log_2 \frac{\mathrm{P}(\ell, i)}{\mathrm{P}(\ell)\mathrm{P}(i)} \tag{12}$$

$$\tilde{\mathrm{I}}(i|\ell) = \frac{\tilde{\mathrm{I}}(\ell, i) - \min \tilde{\mathrm{I}}(\ell, \mathcal{A})}{\max \tilde{\mathrm{I}}(\ell, \mathcal{A}) - \min \tilde{\mathrm{I}}(\ell, \mathcal{A})} \tag{13}$$

$$\tilde{\mathrm{I}}(\ell|i) = \frac{\tilde{\mathrm{I}}(\ell, i) - \min \tilde{\mathrm{I}}(\mathcal{L}, i)}{\max \tilde{\mathrm{I}}(\mathcal{L}, i) - \min \tilde{\mathrm{I}}(\mathcal{L}, i)} \tag{14}$$

The user-specified function $d_{\max}(\ell, i)$ serves a dual purpose: first as a normalization factor for the fuzzy phonetic similarity estimate $\mathrm{sim}(\ell, i)$, and second as a cutoff threshold for absolute phonetic edit distances $\mathrm{d}_{\mathrm{edit}}(\ell, i)$, blocking instantiation hypotheses when phonetic dissimilarity grows "too large":

$$d_{\max}(\ell, i) = \min\{|\ell|, |i|\} - 1 \tag{15}$$

The lemma instantiation likelihood function $\mathrm{L}(i, \ell)$ is defined as the product of the normalized phonetic similarity and the arithmetic average component-normalized mutual information score:

$$\mathrm{sim}(\ell, i) = \begin{cases} \frac{d_{\max}(\ell,i) - \mathrm{d}_{\mathrm{edit}}(\ell,i)}{d_{\max}(\ell,i)} & \text{if } \mathrm{d}_{\mathrm{edit}}(\ell, i) \leq d_{\max}(\ell, i) \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

$$\mathrm{L}(i, \ell) = \frac{\mathrm{sim}(\ell, i) \times (\tilde{\mathrm{I}}(\ell|i) + \tilde{\mathrm{I}}(i|\ell))}{2} \tag{17}$$

Finally, the edit-distance lemma instantiation heuristic conflates those word pairs which share either a phonetic form or appear as best instances of some common dictionary lemma:[2]

$$\begin{aligned} w \equiv_{\mathrm{li}} v \quad :\Leftrightarrow \quad & (w \equiv_{\mathrm{pho}} v) \text{ or} \\ & (\mathrm{lemma}(\mathrm{bestInstance}^{-1}(w)) \cap \mathrm{lemma}(\mathrm{bestInstance}^{-1}(v)) \neq \emptyset) \end{aligned} \tag{18}$$

3.2 Performance

A major advantage of this approach arises from the relatively small number of edit distance comparisons which must be performed. Since the Wagner-Fischer algorithm (Wagner and Fischer, 1974) used to compute phonetic edit distances has quadratic running time, $\mathbf{O}(\mathrm{d}_{\mathrm{edit}}(w, v)) = \mathbf{O}(|w||v|)$, the number of edit distance comparisons comprises the bulk of the heuristic's running time, and should be kept as small as possible. Restricting the comparisons to those pairs (ℓ, i) of dictionary lemmata and phonetic types occurring in quotation evidence for those lemmata requires that approximately 3.38 million comparisons be made

[2]Note that $\equiv_{\mathrm{li}}$ is not an equivalence relation in the strict sense, since although it is reflexive and symmetric, it is not transitive.

during analysis of the entire TAXI/Grimm quotation corpus. If instead every possible unordered pair of phonetic types were to be compared – as required by some morphology induction techniques – a total of *circa* 340 billion comparisons would be required, over ten thousand times as many! With restriction of comparisons to dictionary lemmata, the heuristic analysis completes in 28 minutes on a 1.8GHz dual-core processor workstation, which corresponds to a projected running time of about 5.35 years for a method comparing all unordered word pairs, which is clearly unacceptable.

3.3 Coverage

Using the verse quotation evidence corpus described above in Section 2.3, the lemma instantiation conflation heuristics discovered conflations with extant forms known to the TAGH morphology for 29,248 additional word types not discovered by phonetic conflation, including all of the example word pairs given in the introduction to this section. Additionally, 9,415 word types were identified as "best instances" for DWB lemmata unknown to the TAGH morphology. Together with phonetic conflation, the lemma instantiation heuristics achieve a total coverage of 212,540 types (66.76%), representing 5,185,858 tokens (94.43%). Thus, the lemma instantiation heuristic conflation method provides a reduction of 26.76% in type-wise coverage errors and of 33.88% in token-wise coverage errors with respect to the phonetic identity conflation method alone, resulting in a total reduction of 42.26% in type-wise coverage errors and of 65.80% in token-wise coverage errors with respect to the literal TAGH morphology.

4 Summary & Outlook

Two strategies were presented for discovering synchronically active canonical forms for unknown historical text forms. Together, the two methods achieve TAGH morphological analyses for 94.43% of tokens, reducing the number of unknown tokens by 65.8% in a corpus of *circa* 5.5 million words of historical German verse. In the interest of generalizing these strategies to arbitrary input texts, a robust system for lazy online best-path lookup operations in weighted finite state transducer cascades (such as phonetic equivalence classes or best-alignments with a target language in the form of a finite state acceptor) is currently under development.

While the high coverage rate of the conflation strategies presented here is encouraging, a number of important questions remain. Chief among these is the question of the canonicalization strategies' reliability: how many of the discovered extant "canonical" forms are in fact morphologically related to the source forms? Conversely, were all valid canonical forms for each covered source word indeed found, or were some missed? A small gold standard test corpus is currently under construction which should enable quantitative answers to these questions in terms of the information retrieval notions of *precision* and *recall*.

References

A. V. Aho and M. J. Corasick. Efficient string matching: an aid to bibliographic search. *Commun. ACM*, 18(6):333–340, 1975. ISSN 0001-0782. doi: http://doi.acm.org/10.1145/360825.360855.

A. V. Aho and J. D. Ullman. *The Theory of Parsing, Translation and Compiling*. Prentice-Hall, Englewood Cliffs, N.J., 1972.

J. Allen, S. Hunnicut, and D. Klatt. *From Text to Speech: the MITalk system*. Cambridge University Press, 1987.

M. Baroni, J. Matiasek, and H. Trost. Unsupervised discovery of morphologically related words based on orthographic and semantic similarity. In *Proceedings of the Workshop on Morphological and Phonological Learning of ACL-2002*, pages 48–57, 2002.

H.-W. Bartz, T. Burch, R. Christmann, K. Gärtner, V. Hildenbrandt, T. Schares, and K. Wegge, editors. *Der Digitale Grimm. Deutsches Wörterbuch von Jacob und Wilhelm Grimm*. Zweitausendeins, Frankfurt am Main, 2004. URL `http://www.dwb.uni-trier.de`.

A. W. Black and P. Taylor. Festival speech synthesis system: system documentation. Technical Report HCRC/TR-83, University of Edinburgh, Centre for Speech Technology Research, 1997. URL `http://www.cstr.ed.ac.uk/projects/festival`.

E. Brill. A simple rule-based part-of-speech tagger. In *Proceedings of ANLP-92, 3rd Conference on Applied Natural Language Processing*, pages 152–155, Trento, Italy, 1992.

K. W. Church and P. Hanks. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29, 1990.

T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, New York, 1991.

S. DeRose. Grammatical category disambiguation by statistical optimization. *Computational Linguistics*, 14(1):31–39, 1988.

T. Dutoit. *An Introduction to Text-to-Speech Synthesis*. Kluwer, Dordrecht, 1997.

A. Geyken and T. Hanneforth. TAGH: A Complete Morphology for German based on Weighted Finite State Automata. In *Finite State Methods and Natural Language Processing. 5th International Workshop, FSMNLP 2005, Helsinki, Finland, September 1-2, 2005. Revised Papers*, volume 4002, pages 55–66. Springer, 2006. doi: http://dx.doi.org/10.1007/11780885_7.

B. Jurish. A hybrid approach to part-of-speech tagging. Technical report, Project "Kollokationen im Wörterbuch", Berlin-Brandenburg Academy of Sciences, Berlin, 2003. URL `http://www.ling.uni-potsdam.de/~jurish/pubs/dwdst-report.pdf`.

É. Laporte. Rational transductions for phonetic conversion and phonology. In Roche and Schabes (1997).

V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(1966):707–710, 1966.

M. J. Liberman and K. W. Church. Text analysis and word pronunciation in text-to-speech synthesis. In S. Furui and M. M. Sondhi, editors, *Advances in Speech Signal Processing*. Dekker, New York, 1992.

J. B. Lovins. Development of a stemming algorithm. *Mechanical Translation and Computational Linguistics*, 11:22–31, 1968.

C. D. Manning and H. Schütze. *Foundations of statistical natural language processing*. MIT Press, Cambridge, MA, 1999.

W. J. McGill. Multivariate information transmission. *IEEE Trans. Inf. Theory*, 4(4): 93–111, 1955.

G. Möhler, A. Schweitzer, and M. Breitenbücher. *IMS German Festival manual, version 1.2*. Institute for Natural Language Processing, University of Stuttgart, 2001. URL `http://www.ims.uni-stuttgart.de/phonetik/synthesis`.

G. Navarro. A guided tour to approximate string matching. *ACM Computing Surveys*, 33 (1):31–88, 2001.

M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

A. M. Robertson and P. Willett. A comparison of spelling-correction methods for the identification of word forms in historical text databases. *Literary and Linguistic Computing*, 8(3):143–152, 1993.

E. Roche and Y. Schabes, editors. *Finite–State Language Processing*. MIT Press, Cambridege, Massachusetts, 1997.

H. Schmid. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK, 1994.

A. Sokirko. A technical overview of DWDS/dialing concordance. Talk delivered at the meeting *Computational linguistics and intellectual technologies*, Protvino, Russia, June 2003. URL `http://www.aot.ru/docs/OverviewOfConcordance.htm`.

P. Taylor, A. W. Black, and R. J. Caley. The architecture of the the Festival speech synthesis system. In *Third International Workshop on Speech Synthesis, Sydney, Australia, November, 1998*, 1998.

R. A. Wagner and M. J. Fischer. The string-to-string correction problem. *Journal of the ACM*, 21(1):168–173, 1974.

D. Yarowsky and R. Wicentowski. Minimally supervised morphological analysis by multimodal alignment. In K. Vijay-Shanker and C.-N. Huang, editors, *Proceedings of the 38th Meeting of the Association for Computational Linguistics*, pages 207–216, Hong Kong, October 2000.