Comparing Canonicalizations of Historical German Text

Bryan Jurish
Berlin-Brandenburg Academy of Sciences
Berlin, Germany
jurish@bbaw.de

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 system requiring reference to a static lexicon accessed by orthographic form. In this paper, we present three methods for associating unknown historical word forms with synchronically active canonical cognates and evaluate their performance on an information retrieval task over a manually annotated corpus of historical German verse.

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 system requiring reference to a fixed lexicon accessed by orthographic form, such as document indexing systems (Sokirko, 2003; Cafarella and Cutting, 2004), part-of-speech taggers (DeRose, 1988; Brill, 1992; Schmid, 1994), simple word stemmers (Lovins, 1968; Porter, 1980), or more sophisticated morphological analyzers (Geyken and Hanneforth, 2006; Clematide, 2008).

When adopting historical text into such a system, one of the most crucial tasks is the association of one or more *extant equivalents* with each word of the input text: synchronically active types which best represent the relevant features of the input word. Which features are considered "relevant" here depends on the application in question: for a lemmatization task only the root lexeme is relevant, whereas syntactic parsing may require additional morphosyntactic features. For current purposes, extant equivalents are to be understood as *canonical cognates*, preserving both the root(s) and morphosyntactic features of the associated historical form(s), which should suffice (modulo major grammatical and/or lexical semantic shifts) for most natural language processing tasks.

In this paper, we present three methods for automatic discovery of extant canonical cognates for historical German text, and evaluate their performance on an information retrieval task over a small gold-standard corpus.

2 Canonicalization Methods

In this section, we present three methods for automatic discovery of extant canonical cognates for historical German input: *phonetic conflation* (Pho), *Levenshtein edit distance* (Lev), and a heuristic *rewrite transducer* (rw). The various methods are presented individually below, and characterized in terms of the linguistic resources required for their application. Formally, each canonicalization method R is defined by a characteristic *conflation relation* $\sim_R$, a binary relation on the set $\mathcal{A}^*$ of all strings over the finite grapheme alphabet $\mathcal{A}$. Prototypically, $\sim_R$ will be a true equivalence relation, inducing a partitioning of $\mathcal{A}^*$ into equivalence classes or "conflation sets" $[w]_R = \{v \in \mathcal{A}^* : v \sim_R w\}$.

2.1 Phonetic Conflation

If we assume despite the lack of consistent orthographic conventions that historical graphemic forms were constructed to reflect phonetic forms, and if the phonetic system of the target language is diachronically more stable than the graphematic system, then the phonetic form of a word should provide a better clue to its extant cognates (if any) than a historical graphemic form alone. Taken together, these assumptions lead to the canonicaliza-

Proceedings of the 11th Meeting of the ACL-SIGMORPHON, ACL 2010, pages 72–77,
Uppsala, Sweden, 15 July 2010.

tion technique referred to here as *phonetic conflation*.

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 an often-studied problem (Allen et al., 1987; Dutoit, 1997), the so-called "letter-to-sound" (LTS) conversion problem. The phonetic conversion module used here was adapted from the LTS rule-set distributed with the IMS German Festival package (Möhler et al., 2001), and compiled as a finite-state transducer (Jurish, 2008).

In general, the phonetic conflation strategy maps each (historical or extant) input word $w \in \mathcal{A}^*$ to a unique phonetic form $\mathrm{pho}(w)$ by means of a computable function $\mathrm{pho} : \mathcal{A}^* \rightarrow \mathcal{P}^*$,[1] conflating those strings which share a common phonetic form:

$$w \sim_{\mathrm{Pho}} v :\Leftrightarrow \mathrm{pho}(w) = \mathrm{pho}(v) \qquad (1)$$

2.2 Levenshtein Edit Distance

Although the phonetic conflation technique described in the previous section is capable of successfully identifying a number of common historical graphematic variation patterns such as *ey/ei*, *œ/ö*, *th/t*, and *tz/z*, it fails to conflate historical forms with any extant equivalent whenever the graphematic variation leads to non-identity of the respective phonetic forms, as determined by the LTS rule-set employed. In particular, whenever a historical variation would effect a pronunciation difference in synchronic forms, that variation will remain uncaptured by a phonetic conflation technique. Examples of such phonetically salient variations with respect to the simplified IMS German Festival rule-set include *guot/gut* "good", *liecht/licht* "light", *tiuvel/teufel* "devil", and *wolln/wollen* "want".

In order to accommodate graphematic variation phenomena beyond those for which strict phonetic identity of the variant forms obtains, we may employ an approximate search strategy based on the simple *Levenshtein edit distance* (Levenshtein, 1966; Navarro, 2001). Formally, let $\mathrm{Lex} \subseteq \mathcal{A}^*$ be the lexicon of all extant forms, and let $d_{\mathrm{Lev}} : \mathcal{A}^* \times \mathcal{A}^* \rightarrow \mathbb{N}$ represent the Levenshtein distance over grapheme strings, then define for every input word $w \in \mathcal{A}^*$ the "best" synchronic equivalent $\mathrm{best}_{\mathrm{Lev}}(w)$ as the unique extant word $v \in \mathrm{Lex}$ with minimal edit-distance to the input word:[2]

$$\mathrm{best}_{\mathrm{Lev}}(w) = \arg\min_{v \in \mathrm{Lex}} d_{\mathrm{Lev}}(w, v) \qquad (2)$$

Ideally, the image of a word w under $\mathrm{best}_{\mathrm{Lev}}$ will itself be the canonical cognate sought,[3] leading to conflation of all strings which share a common image under $\mathrm{best}_{\mathrm{Lev}}$:

$$w \sim_{\mathrm{Lev}} v :\Leftrightarrow \mathrm{best}_{\mathrm{Lev}}(w) = \mathrm{best}_{\mathrm{Lev}}(v) \qquad (3)$$

The function $\mathrm{best}_{\mathrm{Lev}}(w) : \mathcal{A}^* \rightarrow \mathrm{Lex}$ can be computed using a variant of the Dijkstra algorithm (Dijkstra, 1959) even when the lexicon is infinite (as in the case of productive nominal composition in German) whenever the set Lex can be represented by a finite-state acceptor (Mohri, 2002; Allauzen and Mohri, 2009; Jurish, 2010). For current purposes, we used the (infinite) input language of the TAGH morphology transducer (Geyken and Hanneforth, 2006) stripped of proper names, abbreviations, and foreign-language material to approximate Lex.

2.3 Rewrite Transducer

While the simple edit distance conflation technique from the previous section is quite powerful and requires for its implementation only a lexicon of extant forms, the Levenshtein distance itself appears in many cases too coarse to function as a reliable predictor of etymological relations, since each edit operation (deletion, insertion, or substitution) is assigned a cost independent of the characters operated on and of the immediate context in the strings under consideration. This operand-independence of the traditional Levenshtein distance results in a number of spurious conflations such as those given in Table 1.

In order to achieve a finer-grained and thus more precise mapping from historical forms to extant canonical cognates while preserving some degree of the robustness provided by the relaxation of the strict identity criterion implicit in the edit-distance conflation technique, a non-deterministic weighted finite-state "rewrite" transducer was developed to replace the simple Levenshtein metric. The rewrite transducer was compiled from a

[1] $\mathcal{P}$ is a finite phonetic alphabet.

[2] We assume that whenever multiple extant minimal-distance candidate forms exist, one is chosen randomly.

[3] Note here that every extant form is its own "best" equivalent: $w \in \mathrm{Lex}$ implies $\mathrm{best}_{\mathrm{Lev}}(w) = w$, since $d_{\mathrm{Lev}}(w, w) = 0 < d_{\mathrm{Lev}}(w, v)$ for all $v \neq w$.

w	$\text{best}_{\text{Lev}}(w)$		Extant Equivalent	
aug	*aus*	"out"	*auge*	"eye"
faszt	*fast*	"almost"	*fasst*	"grabs"
ouch	*buch*	"book"	*auch*	"also"
ram	*rat*	"advice"	*rahm*	"cream"
vol	*volk*	"people"	*voll*	"full"

Table 1: Example spurious Levenshtein distance conflations

heuristic two-level rule-set (Karttunen et al., 1987; Kaplan and Kay, 1994; Laporte, 1997) whose 306 rules were manually constructed to reflect linguistically plausible patterns of diachronic variation as observed in the lemma-instance pairs automatically extracted from the full 5.5 million word DWB verse corpus (Jurish, 2008). In particular, phonetic phenomena such as *schwa deletion*, *vowel shift*, *voicing alternation*, and *articulatory location shift* are easily captured by such rules.

Of the 306 heuristic rewrite rules, 131 manipulate consonant-like strings, 115 deal with vowel-like strings, and 14 operate directly on syllable-like units. The remaining 46 rules define expansions for explicitly marked elisions and unrecognized input. Some examples of rules used by the rewrite transducer are given in Table 2.

Formally, the rewrite transducer Δ_{rw} defines a pseudo-metric $[\![\Delta_{\text{rw}}]\!] : \mathcal{A}^* \times \mathcal{A}^* \to \mathbb{R}_\infty$ on all string pairs (Mohri, 2009). Assuming the non-negative tropical semiring (Simon, 1987) is used to represent transducer weights, analagous to the transducer representation of the Levenshtein metric (Allauzen and Mohri, 2009), the rewrite pseudo-metric can be used as a drop-in replacement for the Levenshtein distance in Equations (2) and (3), yielding Equations (4) and (5):

$$\text{best}_{\text{rw}}(w) = \arg\min_{v \in \text{Lex}} [\![\Delta_{\text{rw}}]\!](w, v) \quad (4)$$

$$w \sim_{\text{rw}} v :\Leftrightarrow \text{best}_{\text{rw}}(w) = \text{best}_{\text{rw}}(v) \quad (5)$$

3 Evaluation

3.1 Test Corpus

The conflation techniques described above were tested on a corpus of historical German verse extracted from the quotation evidence in a single volume of the digital first edition of the dictionary *Deutsches Wörterbuch* "DWB" (Bartz et al., 2004). The test corpus contained 11,242 tokens of 4157 distinct word types, discounting non-alphabetic types such as punctuation. Each corpus type was manually assigned one or more extant equivalents based on inspection of its occurrences in the whole 5.5 million word DWB verse corpus in addition to secondary sources. Only extinct roots, proper names, foreign and other non-lexical material were not explicitly assigned any extant equivalent at all; such types were flagged and treated as their own canonical cognates, *i.e.* identical to their respective "extant" equivalents. In all other cases, equivalence was determined by direct etymological relation of the root in addition to matching morphosyntactic features. Problematic types were marked as such and subjected to expert review. 296 test corpus types representing 585 tokens were ambiguously associated with more than one canonical cognate. In a second annotation pass, these remaining ambiguities were resolved on a per-token basis.

3.2 Evaluation Measures

The three conflation strategies from Section 2 were evaluated using the gold-standard test corpus to simulate a document indexing and query scenario. Formally, let $G \subset \mathcal{A}^* \times \mathcal{A}^*$ represent the finite set of all gold-standard pairs $(w, \widetilde{w})$ with $\widetilde{w}$ the manually determined canonical cognate for the corpus type w, and let $Q = \{\widetilde{w} : \exists (w, \widetilde{w}) \in G\}$ be the set of all canonical cognates represented in the corpus. Then define for a binary conflation relation $\sim_R$ on $\mathcal{A}^*$ and a query string $q \in Q$ the sets $\text{relevant}(q), \text{retrieved}_R(q) \subseteq G$ of *relevant* and *retrieved* gold-standard pairs as:

$$\begin{aligned} \text{relevant}(q) &= \{(w, \widetilde{w}) \in G : \widetilde{w} = q\} \\ \text{retrieved}_R(q) &= \{(w, \widetilde{w}) \in G : w \sim_R q\} \end{aligned}$$

Type-wise precision and recall can then be defined directly as:

$$\begin{aligned} \text{pr}_G &= \frac{\left|\bigcup_{q \in Q} \text{retrieved}_R(q) \cap \text{relevant}(q)\right|}{\left|\bigcup_{q \in Q} \text{retrieved}_R(q)\right|} \\ \text{rc}_G &= \frac{\left|\bigcup_{q \in Q} \text{retrieved}_R(q) \cap \text{relevant}(q)\right|}{\left|\bigcup_{q \in Q} \text{relevant}(q)\right|} \end{aligned}$$

If $\text{tp}_R(q) = \text{retrieved}_R(q) \cap \text{relevant}(q)$ represents the set of *true positives* for a query q, then token-wise precision and recall are defined in terms of the gold-standard frequency function

From → To /	Left __ Right	⟨Cost⟩	Example(s)	
$\varepsilon \rightarrow e$ /	$(\mathcal{A}\backslash\{e\})$ __ #	⟨ 5 ⟩	*aug* ⇝ *auge*	"eye"
$z \rightarrow s$ /	s __	⟨ 1 ⟩	*faszt* ⇝ *fasst*	"grabs"
$o \rightarrow a$ /	__ u	⟨ 1 ⟩	*ouch* ⇝ *auch*	"also"
$\varepsilon \rightarrow h$ /	V __ C	⟨ 5 ⟩	*ram* ⇝ *rahm*	"cream"
$l \rightarrow ll$ /	__	⟨ 8 ⟩	*vol* ⇝ *voll*	"full"

Table 2: Some example heuristics used by the rewrite transducer. Here, ε represents the empty string, # represents a word boundary, and $\mathrm{V}, \mathrm{C} \subset \mathcal{A}$ are sets of vowel-like and consonant-like characters, respectively.

$f_G : G \rightarrow \mathbb{N}$ as:

$$\mathrm{pr}_{f_G} = \frac{\sum_{q \in Q, g \in \mathrm{tp}_R(q)} f_G(g)}{\sum_{q \in Q, g \in \mathrm{retrieved}_R(q)} f_G(g)}$$

$$\mathrm{rc}_{f_G} = \frac{\sum_{q \in Q, g \in \mathrm{tp}_R(q)} f_G(g)}{\sum_{q \in Q, g \in \mathrm{relevant}(q)} f_G(g)}$$

We use the unweighted harmonic precision-recall average F (van Rijsbergen, 1979) as a composite measure for both type- and token-wise evaluation modes:

$$\mathrm{F}(pr, rc) = \frac{2 \cdot pr \cdot rc}{pr + rc}$$

3.3 Results

The elementary canonicalization function for each of the conflation techniques[4] was applied to the entire test corpus to simulate a corpus indexing run. Running times for the various methods on a 1.8GHz Linux workstation using the `gfsmxl` C library are given in Table 3. The Levenshtein edit-distance technique is at a clear disadvantage here, roughly 150 times slower than the phonetic technique and 40 times slower than the specialized heuristic rewrite transducer. This effect is assumedly due to the density of the search space (which is maximal for an unrestricted Levenshtein editor), since the `gfsmxl` greedy k-best search of a Levenshtein transducer cascade generates at least $|\mathcal{A}|$ configurations per character, and a single backtracking step requires an additional $3|\mathcal{A}|$ heap extractions (Jurish, 2010). Use of specialized lookup algorithms (Oflazer, 1996) might ameliorate such problems.

Qualitative results for several conflation techniques with respect to the DWB verse test corpus are given in Table 4. An additional conflation relation "Id" using strict identity of grapheme strings

[4] $\mathrm{pho}, \mathrm{best}_{\mathrm{Lev}}$ and $\mathrm{best}_{\mathrm{rw}}$ for the phonetic, Levenshtein, and heuristic rewrite transducer methods respectively

Method	Time	Throughput
Pho	1.82 sec	7322 tok/sec
Lev	278.03 sec	48 tok/sec
rw	7.02 sec	1898 tok/sec

Table 3: Processing time for elementary canonicalization functions

($w \sim_{\mathrm{Id}} v :\Leftrightarrow w = v$) was tested to provide a baseline for the methods described in Section 2.

As expected, the strict identity baseline relation was the most precise of all methods tested, achieving 99.9% type-wise and 99.1% token-wise precision. This is unsurprising, since the Id method yields false positives only when a historical form is indistinguishable from a non-equivalent extant form, as in the case of the mapping *wider* ⇝ *wieder* ("again") and the non-equivalent extant form *wider* ("against"). Despite its excellent precision, the baseline method's recall was the lowest of any tested method, which supports the claim that a synchronically-oriented lexicon cannot adequately account for a corpus of historical text. Type-wise recall was particularly low (70.8%), indicating that diachronic variation was more common in low-frequency types.

Surprisingly, the phonetic and Levenshtein edit-distance methods performed similarly for all measures except token-wise precision, in which Lev incurred 61.6% fewer errors than Pho. Given their near-identical type-wise precision, this difference can be attributed to a small number of phonetic misconflations involving high-frequency types, such as *wider*∼*wieder* ("against"∼"again"), *statt*∼*stadt*, ("instead"∼"city"), and *in*∼*ihn* ("in"∼"him"). Contrary to expectations, Lev did not yield any recall improvements over Pho, although the union of the two underlying conflation relations

	Type-wise %			Token-wise %		
R	pr_G	rc_G	F_G	pr_{f_G}	rc_{f_G}	F_{f_G}
Id	**99.9**	70.8	82.9	**99.1**	83.7	90.7
Pho	96.7	80.1	87.6	92.7	89.6	91.1
Lev	96.6	78.9	86.9	97.2	87.8	92.2
rw	98.5	88.4	**93.2**	98.2	93.4	**95.8**
Pho | Lev	94.1	84.3	88.9	91.3	91.6	91.5
Pho | rw	96.1	**89.8**	92.8	92.5	**94.5**	93.5

Table 4: Qualitative evaluation of various conflation techniques

($\sim_{\mathrm{Pho}\,|\,\mathrm{Lev}} = \sim_{\mathrm{Pho}} \cup \sim_{\mathrm{Lev}}$) achieved a type-wise recall of 84.3% (token-wise recall 91.6%), which suggests that these two methods complement one another when both an LTS module and a high-coverage lexicon of extant types are available.

Of the methods described in Section 2, the heuristic rewrite transducer Δ_{rw} performed best overall, with a type-wise harmonic mean F of 93.2% and a token-wise F of 95.8%. While Δ_{rw} incurred some additional precision errors compared to the naïve graphemic identity method Id, these were not as devastating as those incurred by the phonetic or Levenshtein distance methods, which supports the claim from Section 2.3 that a fine-grained context-sensitive pseudo-metric incorporating linguistic knowledge can more accurately model diachronic processes than an all-purpose metric like the Levenshtein distance.

Recall was highest for the composite phonetic-rewrite relation $\sim_{\mathrm{Pho}\,|\,\mathrm{rw}} = \sim_{\mathrm{Pho}} \cup \sim_{\mathrm{rw}}$, although the precision errors induced by the phonetic component outweighed the comparatively small gain in recall. The best overall performance is achieved by the heuristic rewrite transducer Δ_{rw} on its own, yielding a reduction of 60.3% in type-wise recall errors and of 59.5% in token-wise recall errors, while minimizing the number of newly introduced precision errors.

4 Conclusion & Outlook

We have presented three different methods for associating unknown historical word forms with synchronically active canonical cognates. The heuristic mapping of unknown forms to extant equivalents by means of linguistically motivated context-sensitive rewrite rules yielded the best results in an information retrieval task on a corpus of historical German verse, reducing type-wise recall errors by over 60% compared to a naïve text-matching strategy. Depending on the availability of linguistic resources (*e.g.* phonetization rule-sets, lexica), use of phonetic canonicalization and/or Levenshtein edit distance may provide a more immediately accessible route to improved recall for other languages or applications, at the expense of some additional loss of precision.

We are interested in verifying our results using larger corpora than the small test corpus used here, as well as extending the techniques described here to other languages and domains. In particular, we are interested in comparing the performance of the domain-specific rewrite transducer used here to other linguistically motivated language-independent metrics such as (Covington, 1996; Kondrak, 2000).

Acknowledgements

The work described above was funded by a *Deutsche Forschungsgemeinschaft* (DFG) grant to the project *Deutsches Textarchiv*. Additionally, the author would like to thank Jörg Didakowski, Oliver Duntze, Alexander Geyken, Thomas Hanneforth, Henriette Scharnhorst, Wolfgang Seeker, Kay-Michael Würzner, and this paper's anonymous reviewers for their helpful feedback and comments.

References

Cyril Allauzen and Mehryar Mohri. 2009. Linear-space computation of the edit-distance between a string and a finite automaton. In *London Algorithmics 2008: Theory and Practice*. College Publications.

Jonathan Allen, M. Sharon Hunnicutt, and Dennis Klatt. 1987. *From Text to Speech: the MITalk system*. Cambridge University Press.

Hans-Werner Bartz, Thomas Burch, Ruth Christmann, Kurt Gärtner, Vera Hildenbrandt, Thomas Schares, and Klaudia Wegge, editors. 2004. *Der Digitale*

Grimm. Deutsches Wörterbuch von Jacob und Wilhelm Grimm. Zweitausendeins, Frankfurt am Main.

Eric Brill. 1992. 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.

Mike Cafarella and Doug Cutting. 2004. Building Nutch: Open source search. *Queue*, 2(2):54–61.

Simon Clematide. 2008. An OLIF-based open inflection resource and yet another morphological system for German. In Storrer et al. (Storrer et al., 2008), pages 183–194.

Michael A. Covington. 1996. An algorithm to align words for historical comparison. *Computational Linguistics*, 22:481–496.

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

Edsger W. Dijkstra. 1959. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271.

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

Alexander Geyken and Thomas Hanneforth. 2006. TAGH: A complete morphology for German based on weighted finite state automata. In *Proceedings FSMNLP 2005*, pages 55–66, Berlin. Springer.

Bryan Jurish. 2008. Finding canonical forms for historical German text. In Storrer et al. (Storrer et al., 2008), pages 27–37.

Bryan Jurish. 2010. Efficient online *k*-best lookup in weighted finite-state cascades. To appear in *Studia Grammatica*.

Ronald M. Kaplan and Martin Kay. 1994. Regular models of phonological rule systems. *Computational Linguistics*, 20(3):331–378.

Lauri Karttunen, Ronald M. Kay, and Kimmo Koskenniemi. 1987. A compiler for two-level phonological rules. In M. Dalrymple, R. Kaplan, L. Karttunen, K. Koskenniemi, S. Shaio, and M. Wescoat, editors, *Tools for Morphological Analysis*, volume 87-108 of *CSLI Reports*, pages 1–61. CSLI, Stanford University, Palo Alto, CA.

Gregorz Kondrak. 2000. A new algorithm for the alignment of phonetic sequences. In *Proceedings NAACL*, pages 288–295.

Éric Laporte. 1997. Rational transductions for phonetic conversion and phonology. In Emmanuel Roche and Yves Schabes, editors, *Finite-State Language Processing*. MIT Press, Cambridge, MA.

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

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

Mehryar Mohri. 2002. Semiring frameworks and algorithms for shortest-distance problems. *Journal of Automata, Languages and Combinatorics*, 7(3):321–350.

Mehryar Mohri. 2009. Weighted automata algorithms. In *Handbook of Weighted Automata*, Monographs in Theoretical Computer Science, pages 213–254. Springer, Berlin.

Gregor Möhler, Antje Schweitzer, and Mark Breitenbücher, 2001. *IMS German Festival manual, version 1.2*. Institute for Natural Language Processing, University of Stuttgart.

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

Kemal Oflazer. 1996. Error-tolerant finite-state recognition with applications to morphological analysis and spelling correction. *Computational Linguistics*, 22(1):73–89.

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

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

Imre Simon. 1987. The nondeterministic complexity of finite automata. Technical Report RT-MAP-8073, Instituto de Matemática e Estatística da Universidade de São Paulo.

Alexey Sokirko. 2003. A technical overview of DWDS/dialing concordance. Talk delivered at the meeting *Computational linguistics and intellectual technologies*, Protvino, Russia.

Angelika Storrer, Alexander Geyken, Alexander Siebert, and Kay-Michael Würzner, editors. 2008. *Text Resources and Lexical Knowledge*. Mouton de Gruyter, Berlin.

C. J. van Rijsbergen. 1979. *Information Retrieval*. Butterworth-Heinemann, Newton, MA.