

Text Correction Using Domain Dependent Bigram Models from Web Crawls

Christoph Ringlstetter¹ and Max Hadersbeck² and Klaus U. Schulz² and Stoyan Mihov³

¹ AICML, University of Alberta, Edmonton, Canada T6G 2E8, E-mail: kristof@cs.ualberta.ca

² CIS, University of Munich, Oettingenstr 67, D-80538 München, Germany

³ Central Laboratory for Parallel Processing, Bulgarian Academy of Sciences

Abstract

The quality of text correction systems can be improved when using complex language models and by taking peculiarities of the garbled input text into account. We report on a series of experiments where we crawl domain dependent web corpora for a given garbled input text. From crawled corpora we derive dictionaries and language models, which are used to correct the input text. We show that correction accuracy is improved when integrating word bigram frequency values from the crawls as a new score into a baseline correction strategy based on word similarity and word (unigram) frequencies. In a second series of experiments we compare the quality of distinct language models, measuring how closely these models reflect the frequencies observed in a given input text. It is shown that crawled language models are superior to language models obtained from standard corpora.

Keywords: error correction, language models, n-grams, adaptive techniques, web, crawling techniques.

1 Introduction

Typed manuscripts and texts obtained from optical character recognition (OCR) often come with a considerable number of orthographic errors. In order to (re)publish these texts, the number of errors must be reduced to a minimum. Even if texts are stored in an electronic archive where the high quality standards of publishing do not have to be met, access to texts via conventional keyword based search may be seriously disrupted if the number of erroneous tokens is too large [Taghva *et al.*, 1996; 2004]. Both scenarios explain why intelligent strategies for correcting orthographic errors in texts attract much attention. The experiments described in this paper are guided by two ambitious goals of intelligent text correction methods.

The first goal is the use of powerful language models as one component of the correction process. Language models (word frequencies, n-gram models, probabilistic grammars etc.) help to disambiguate between distinct correction candidates for ill formed tokens: obviously, an analysis of the sentence neighborhood can give very valuable hints on plausible correction suggestions. Given this background, a relatively small number of attempts have been made to integrate sentence context into the correction strategy [Keenan *et al.*, 1991; Srihari, 1993; Srihari and Baltus, 1993; Hong and Hull, 1994; Hwang *et al.*, 1999; Golding and Roth, 1999; Golding and Schabes, 1996]. Most of these studies are centered around a list of example problems and do not seriously discuss the problem of integrating methods into an efficient and fully-fledged text correction system. In this paper we report on the use of bigram counts for arbitrary word pairs over large dictionaries in a practical text correction system.

The second goal is the development of correction strategies that are “document centric” and “adaptive” in the sense that the peculiarities of the given input document, its thematic domain and language, are explicitly taken into account in the correction process [Taghva and Stofsky, 2001; Nartker *et al.*, 2003; Rong Jin, 2003]. There are many variants of this general idea. “Text centric” methods, which are mainly useful for long texts, typically prefer correction suggestions that occur elsewhere in the input document. Here we follow another line. In the experiments described below, the vocabulary and the thematic domain of the input document influence the selection of single word and bigram frequencies that are used for ranking distinct correction candidates.

The computation of a text/domain dependent language model is a nontrivial task. Until recently, language models have typically been derived from fixed standard corpora, such as the Brown Corpus or the British National Corpus BNC. However, for most input documents the number of texts on the same domain found in the background corpus is very limited. Hence, due to the sparseness of data, interesting docu-

ment centric language models cannot be obtained in this way. As a way out, various new approaches in corpus linguistics use the web as a corpus.

In earlier work [Strohmaier *et al.*, 2003a] it has been shown how to dynamically compute special dictionaries and word frequencies from web crawls, using terminological expressions of the input document in queries to web search engines. The correction accuracy obtained with these dictionaries and frequencies improves parallel values reached with conventional dictionaries and general word frequencies. In this paper, a new and sophisticated refinement of a previously used method [Strohmaier *et al.*, 2003a] is used to compute domain dependent web corpora and bigram models using the given input text. The method has been fully automated. A toolbox has been developed for storing all bigram values in main memory, using special techniques based on sparse matrices. In this way, bigram counts can be accessed immediately while efficiently processing arbitrary input texts. Our evaluation experiments are based on OCRed input texts from six special domains and cover the languages English and German. We describe two series of experiments.

1. In our *correction experiments* we test to which extend (i) domain dependent and (ii) general bigram values help to appropriately rank correction suggestions and to improve correction accuracy. In a first experiment we exclusively use bigram counts for a disambiguation of correction candidates, given a list of wrongly recognized words and their neighbors. Crawled domain dependent bigram counts turn out to have a much higher predictive power than bigram values from static standard corpora. In a second correction experiment, bigram values are used as a third score for candidate ranking besides word similarity and word frequencies. We show that using bigram values has a strictly positive effect. If the input texts are very accurate, then the gain is small.¹ For input texts that are more seriously garbled, a stronger effect can be observed. Perhaps surprisingly, the advantage of crawled bigram scores over standard bigram scores is only minimal. Given the above results on candidate disambiguation, this observation needs further analysis.

2. In the long term it seems interesting to receive a general picture of the differences that arise when building language models over distinct domains and genres. Our second series of experiments, where we *compare language models*, tries to contribute to this picture. For distinct language models we prove how closely the given word/bigram frequencies reflect the word/bigram frequencies observed in the input document

¹It is always difficult to improve highly accurate texts with fully automated methods for text correction. The use of special dictionaries and word frequencies already leads to higher accuracy, leaving little room for additional improvements.

Topic	Engl.	Ger.	Topic	Engl.	Ger.
Neur.	5,149	2,514	Holoc.	5,772	3,344
Fish	8,094	3,283	Rom.	5,980	4,178
Mushr.	5,943	2,884	Bot.	2,703	2,442

Table 1: Number of evaluation bigrams in primary evaluation collections for the six domains, English and German.

(ground truth version). Our results show that domain dependent language models obtained from we crawls are again clearly superior, the differences being significant.

The paper is structured as follows. In Section 2 we describe the experimental setup, background corpora and evaluation texts. Section 3 introduces our current strategy for crawling document centric corpora and deriving dictionaries, word frequencies and bigram frequencies. Since the number of bigrams to be taken into account is very large we also briefly comment on storage issues. Section 4 presents our experiments for text correction with domain dependent bigram models. The experiments in Section 5 show that domain dependent bigram models better reflect the language of the input text than general bigram models. Section 6 briefly comments on related work before we end with a short conclusion.

2 Experimental setup

In what follows, by an *evaluation collection*, we mean a sample of input documents as arising in text correction tasks. In contrast, large corpora that are used to derive language models are called *background corpora*.

Primary evaluation collections. For each language (English resp. German) we used six *primary evaluation collections* covering texts of the domains botany, fish, holocaust, Roman empire, mushrooms and neurology. Each collection contains 20 pages from real-life documents of the respective area. Documents were printed, scanned, and treated with two distinct OCR-engines: OCR1 is a high-quality commercial software, OCR2 is an experimental open-source software. An overview on the primary evaluation collections is given in Table 1. The primary evaluation collection of each area - as recognized by OCR1 - was used to crawl a *domain dependent background corpus*, using a “document centric” crawling strategy explained in Section 3.

Secondary evaluation collections. As we see in Section 3, the computation of a language model for a particular text needs some time. As an alternative we might use language models that take the thematic domain of the input text into account, but ignore details of the vocabulary. Language models for various domains can be computed once and offline. To test the suitability of precomputed domain dependent language models, for each thematic domain we also used a *secondary evaluation collection*. These texts are (from the

given domain but) not related to the crawling process. From another perspective, our crawling process is “document centric” for the primary evaluation collections and “domain centric” for the secondary evaluation collections.

Background corpora. As static general corpora we used the Brown corpus (1 million tokens) and the BNC (100 million tokens). In addition to the domain dependent background corpora (see Section 3) for each language we crawled a general corpus in the web. To this end we sent 200 disjunctive queries, each composed of five frequent non-stop-words of the given language (English or German) to the Yahoo/AlltheWeb search engine. We collected the first 30 documents of each answer set. The resulting general web corpora contain 46.8 (18.4) million tokens for English (German).

3 Crawling of domain/document centered corpora

Compound-based crawling strategy. Simple strategies for crawling domain specific corpora have been addressed in previous work [Strohmaier *et al.*, 2003a; Baroni and Bernardini, 2004]. Here we used a complex strategy which is based on three steps:

(1) *Extraction of open compound expressions/composite nouns.*

English: The OCRed input text is tagged and all bigrams of the form NN-NN or NN-NNS² are extracted. Only bigrams are used where both components are found in a standard dictionary.

German: All compound nouns of the input text are extracted. Components were checked to represent valid German words. Compound nouns in a dictionary of frequent compound nouns are erased.

Due to the selection strategy, most query atoms (i.e., the above bigrams/compound nouns) are terminological expressions such as “fish species” or “Gehirnzellen”. We received ca. 200 query atoms per text on average.

(2) *Construction of queries.* Each query atom is enriched by four other query atoms, randomly selecting four partners. Each quintuple of atoms built in this way represents a single query.

(3) *Retrieval of answer documents.* Each single query is sent as a disjunctive query to the Yahoo/AlltheWeb search engine. From each answer set we use a maximum of 30 top-ranked answers.³ Collecting these partial answer sets we obtain the

²NN stands for common noun, singular or mass, NNS stands for common noun, plural.

³Yahoo/AlltheWeb, unlike Google, uses a ranking mechanism for disjunctive queries where the number of distinct query atoms found in a web page strongly influences its ranking, a very valuable feature for domain corpus construction.

Lang.	Neur.	Fish	Mush.	Holo.	Rom.	Bot.
Engl.	19.7	49.4	19.4	19.6	7.4	7.5
Ger.	9.5	19.0	13.4	12.7	23.4	11.8

Table 2: Size of crawled thematic corpora for counting n-gram frequencies, in millions of tokens.

domain dependent corpus.

Due to small answer sets, crawled corpora in our experiments contain ca. 4,000 documents.⁴ Table 2 shows the size of the web corpora.

Automating corpus crawls: We implemented a Java-tool that fully automates the compound-based crawling strategy, using the API-interface provided by Yahoo. In an exclude file we maintain a list of URLs that are to be ignored, to avoid multiple downloads of the same source. We currently need approximately two hours for crawling a corpus for a given input text.

Storage of cooccurrence matrices. Given a corpus C and a dictionary D , we compute a triangular matrix assigning to each $(u, v) \in D \times D$ six cooccurrence counts in C . Usually D is the set of types of C . For the experiments described below, the relevant count is the number of consecutive pairs of the form uv . Other values address ordering vu and neighborhood in a given fixed distance/within the same sentence. To store the sparse matrix in compressed form we use cascades of hashfunctions [Jenkins, 1996]. We ran the program with 16 different dictionaries ranging up to 600,000 entries. Compressed matrices required between 320 MB and 2.6 GB main memory. For the largest corpus, the BNC, we need 10 hours, for the general English web corpus with 50 million tokens and a lexicon of 201,977 entries we need one hour processing time.

4 Correction experiments

Disambiguation of correction candidates. For a disambiguation experiment we scanned a printed version of the primary English evaluation collections, applied the commercial OCR1 engine and extracted all OCR errors. To each recognition error we assigned its predecessor token in the input text and a list of correction candidates with Levenshtein-distance ≤ 2 .⁵ Only candidate lists with length > 1 containing the correct candidate have been considered. Bigram values of candidates and left neighbors were used to disambiguate can-

⁴We used “fingerprints” of evaluation collections to avoid the inclusion of texts from these collections in the crawled background corpora.

⁵All errors made by OCR1 turned out to be isolated in the sense that predecessors and successors of wrongly recognized words were correct. Hence in order to disambiguate between several correction suggestions for a token it is realistic to assume that we know the neighboring words.

Model	Neur.	Fish	Mush.	Holo.	Rom.	Bot.
Crawl	64.5	43.6	54.8	59.5	48.2	56.5
BNC	46.8	34.7	41.8	40.9	37.5	28.5
Brown	38.2	30.5	36.4	40.2	37.0	25.5

Table 3: Percentage of correct candidate selections for different language models.

didates. If all bigrams counts are 0, unigram frequencies are used.

The results in Table 3 show that web crawled bigram counts offer a very good basis for disambiguation. On average, 54.52% of the disambiguation tasks were correctly solved. The predictive power of standard bigram counts is much weaker (38.38% for the BNC, 34.65% for Brown). The small difference between BNC and Brown is interesting, underpinning their inadequateness for representing the language found in thematic corpora.

When using the secondary evaluation collection (domain centric crawl) for domain fish we received 43.2% (crawl), 32.5% (BNC) and 30.1% (Brown) correct disambiguations, for the domain holocaust 55.79% (crawl), 44.42% (BNC), 42.0% (Brown). In Section 5 we see that the crawled corpus (BNC) covers 74.98% (76.41%) of all bigrams occurring in the secondary evaluation corpus for domain holocaust. These numbers show that even in situations where crawled corpora cover less bigrams, the counts are more reliable.

Improving correction accuracy. We integrated bigram frequencies obtained from various background corpora as an additional score for candidate ranking into an existing correction strategy [Strohmaier *et al.*, 2003b]. The baseline strategy can be described as follows: For each alphabetic token of the OCRred input text, V , we first retrieve the most similar words from a large background dictionary. In practice, we typically compute all dictionary entries W where the Levenshtein distance between V and W does not exceed 2. A weighted sum $\alpha s_{sim}(V, W) + (1 - \alpha) s_{freq}(W)$ is used to rank the resulting list of correction candidates for V : here $s_{sim}(V, W)$ represents the length-sensitive Levenshtein distance between the input token, V , and the given correction suggestion, W . Score $s_{freq}(W)$ denotes the frequency of W . Both scores are normalized to a value in $[0,1]$. The balance parameter α determines the relative weight of both scores. Non-lexical tokens V are only replaced by the best correction candidate if the score of this candidate exceeds a certain threshold, γ . Balance parameter and threshold parameter are optimized via training. Alternatively, standard values can be used.

For the experiments described below, we extended the toolbox. We now use a weighted sum of the form $\alpha s_{sim}(V, W) + \beta s_{freq}(W) + (1 - \alpha - \beta) s_{bigr}(U, W)$ where $s_{bigr}(U, W)$ is a normalized variant of the frequency of the bigram UW . Here U denotes the predecessor token of V . Other details of the

correction strategy were not modified. Results obtained for the primary English evaluation corpora are listed in Table 5. To simplify the comparison, parameter settings are optimized. For the input texts produced by the commercial OCR1, the additional use of bigram frequencies only leads to small improvements. This is due to the fact that already the accuracy of the OCRred input texts is very high. For the experimental OCR2, improvements are more significant.

5 Comparing language models

Comparing word frequencies. In order to compare frequencies obtained from general corpora with those retrieved from crawled and domain dependent corpora we computed the intersection D_{\cap} between our conventional static dictionary for English⁶, D^E , and the crawled dictionary for domain “Neurology_E^{OCR1}”. The resulting dictionary D_{\cap} was ordered, using two frequency counts:

1. word frequencies obtained from a huge collection of arbitrary web documents,
2. word frequencies obtained from the background corpus crawled for “Neurology_E^{OCR1}”.

In this way we received two orderings of the same list of words, D_{\cap} . Both orderings begin with the most frequent words. We then considered initial segments of the two ordered lists of various sizes and computed the coverage of these segments, using the ground truth corpus “Neurology_E”. Results are shown in Figure 1. On the x-axes we measure the size of the initial segment that is used. The y-axes gives the coverage. Lexical coverage grows much faster when using word frequency from the crawled and domain dependent corpus. The upper (lower) diagram refers to the percentage of covered tokens (types).

Table 4 illustrates the differences between distinct unigram models from another perspective. We consider important terminological expressions from neurology. Column 2 (R_{doc}) gives the rank of the word (type), counting frequencies of all tokens occurring in the ground truth input text “Neurology_E”. Column 3 (R_{crawl}) gives the rank with respect to the crawled dictionary, using frequencies in the crawled background corpus. Column 4 ($R_{standard}$) presents the rank with respect to the conventional dictionary for English, D_E , using general word counts. Obviously, ranks in R_{crawl} reflect R_{doc} much better than those from $R_{standard}$.

Comparing bigram coverage. In our evaluation collections we collected all bigrams composed of two alphabetic strings (called *evaluation bigrams*). We then counted the

⁶The dictionary D^E contains 315,300 entries.

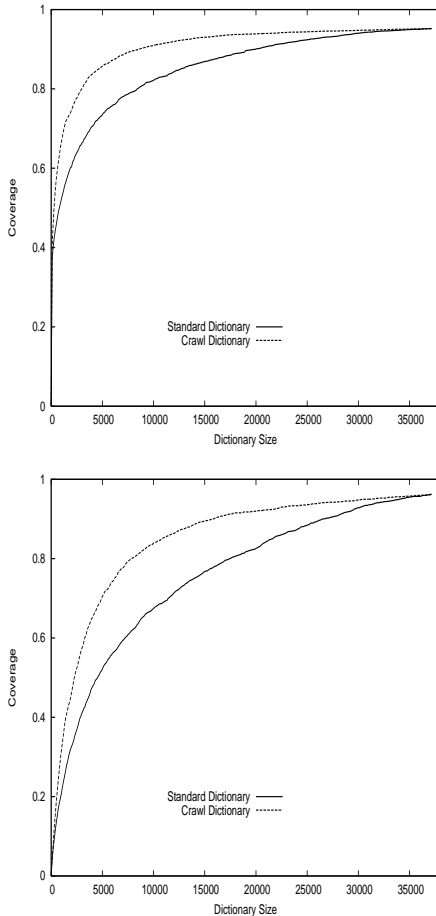


Figure 1: Standard word frequencies versus word frequencies from crawled domain dependent corpora. The growth of coverage for subdictionaries with most frequent tokens depends on the kind of frequency information used, even if the full dictionaries contain the same words. The upper (lower) diagram measures coverage of tokens (types) for the ground truth corpus “Neurology_E”.

Expression	R_{doc}	R_{crawl}	$R_{standard}$
<i>Neurology(E)</i>			
brain	1	7	510
symptoms	7	31	717
mental	6	14	505
disorder	5	5	914
inability	13	810	1,111
syndrome	32	123	842

Table 4: Comparing the rank of terminological expressions from domain “neurology” with respect to distinct frequency orderings.

frequency of each evaluation bigram in distinct large *background corpora*, thus comparing the adequateness of these corpora for deriving language models. Table 6 shows the percentage of evaluation bigrams from the primary English evaluation collections with a frequency ≥ 10 and ≥ 1 in the respective background corpus, comparing the Brown corpus, the BNC, the general web corpus and the crawled thematic background corpus (document centric corpus construction).

Using crawled thematic background corpora, 88.57% of all evaluation bigrams (average over all domains) are found at least once. For the BNC (Brown Corpus), the corresponding number is only 80.86% (56.52%). Remarkably, our domain dependent corpora are more than 5 times smaller than the BNC. Parallel results for German are given in Table 7.

For the secondary evaluation collections (domain centric corpus construction) we obtain the following percentages for seen evaluation bigrams with at least one occurrence in the thematic web corpus. Corresponding numbers for the BNC are given in brackets. 88.50% (82.28%), 86.83% (82.46%), 85.24% (66.33%), 87.99% (90.42%), 90.42% (85.11%), 74.98% (76.41%). With an average value of 85.66%, thematic web corpora again have a better coverage than the BNC (80.50%), despite of two domains where the BNC is better than the thematic corpus. Recall that the thematic corpora are much smaller.

Bigram coverage and corpus size. In [Williams and Zobel, 2005] it was shown that enlarging web corpora always leads to new and unseen words, regardless of the size of the inspected corpus. Figure 2 is meant to extend these results. For the domains Fish and Neurology we depict the evolution of the number of seen bigrams for fragments of the background corpora of distinct sizes. The diagrams show that no real saturation point is reached. Brown corpus, BNC and the general web corpus approximately lead to the same coverage if fragments of the same size are used. Domain specific web corpora are clearly superior.

6 Related Work

Text correction using sentence context. A survey on the subject can be found in [Dengel *et al.*, 1997]. In [Dengel *et al.*, 1997], statistical models (Markov models), grammar-based methods, methods based on collocations, and combinations are distinguished. The “word” n-grams used in Markov models only look at certain categories of words, to avoid an immense number of distinct transitions. Note that our n-gram counts are really based on word pairs. More recent literature [Golding and Roth, 1999] is mainly concerned with recognizing and correcting false friends (also called real word errors, or malapropisms). While these methods are more sophisti-

Corpus	OCR text	D_{crawl}	$D_{crawl}^{stand.ngram}$	$D_{crawl}^{crawl.ngram}$	Gain
“Neurology _E ^{OCR1} ”	98.74	99.39	99.44	99.44	0.05
“Fish _E ^{OCR1} ”	99.23	99.47	99.57	99.57	0.10
“Mushroom _E ^{OCR1} ”	99.01	99.50	99.54	99.55	0.05
“Holocaust _E ^{OCR1} ”	98.86	99.03	99.28	99.15	0.12
“RomanEmpire _E ^{OCR1} ”	98.73	98.90	98.92	99.00	0.10
“Botany _E ^{OCR1} ”	97.19	97.67	97.79	97.89	0.22
“Neurology _E ^{OCR2} ”	90.13	96.29	96.60	96.71	0.42
“Fish _E ^{OCR2} ”	93.36	96.71	97.68	98.02	1.31
“Mushroom _E ^{OCR2} ”	89.26	95.51	95.82	96.00	0.49
“Holocaust _E ^{OCR2} ”	88.77	94.23	94.51	94.61	0.38
“RomanEmpire _E ^{OCR2} ”	93.11	96.12	96.52	96.91	0.79
“Botany _E ^{OCR2} ”	91.71	95.41	95.80	96.09	0.68

Table 5: Improved correction accuracy after using a document centric bigram model as a third score in addition to word frequencies and similarity. Column 5 (4) shows values (percentage of correct words) reached with crawled domain specific bigram counts (bigram counts from standard corpora).

Eval. coll. \ Corpus	Brown		BNC		General web		Domain web	
“Neurology _E ”	18.14	26.98	62.46	67.66	56.06	61.93	68.96	73.90
“Fish _E ”	22.27	32.95	67.67	73.28	59.79	66.28	79.23	83.49
“Mushroom _E ”	20.32	26.42	64.05	67.44	57.85	60.98	66.11	72.98
“Holocaust _E ”	20.19	31.22	64.06	71.59	61.04	69.18	71.61	78.62
“RomanEmpire _E ”	25.29	36.61	67.61	72.96	61.07	67.58	69.33	74.80
“Botany _E ”	14.51	24.08	41.15	48.95	33.99	42.73	48.31	58.38
“Neurology _E ”	45.57	52.61	81.49	84.29	77.57	80.79	88.26	90.52
“Fish _E ”	50.20	57.76	84.03	86.56	78.14	81.74	90.90	92.60
“Mushroom _E ”	47.53	51.72	81.22	82.84	77.44	79.62	86.80	89.69
“Holocaust _E ”	49.26	59.20	80.80	84.84	79.15	83.84	85.30	88.88
“RomanEmpire _E ”	53.67	61.32	81.45	84.26	77.57	80.92	86.52	88.95
“Botany _E ”	31.67	40.44	56.62	62.38	50.39	56.94	75.77	80.80

Table 6: Percentage of evaluation bigrams (types/tokens) of the primary English evaluation collection with a frequency ≥ 10 (upper table) and ≥ 1 (lower table) in the background corpus, for distinct background corpora.

Eval. coll. \ Corpus	General Web		Domain web		General web		Domain web	
“Neurology _G ”	29.13	33.09	39.81	45.62	44.43	48.09	80.30	81.75
“Fish _G ”	25.87	29.00	37.65	43.90	42.30	44.72	63.53	67.57
“Mushroom _G ”	29.94	34.15	34.49	40.89	45.76	49.69	68.09	72.72
“Holocaust _G ”	45.17	49.58	46.31	51.59	66.75	69.92	80.88	83.11
“RomanEmpire _G ”	35.06	38.89	46.33	51.87	52.88	55.82	87.56	88.93
“Botany _G ”	27.95	32.06	37.59	43.37	43.96	47.87	64.33	68.27

Table 7: Percentage of evaluation bigrams (types/tokens) of the primary German evaluation collection with a frequency ≥ 10 (left) and ≥ 1 (right) in the background corpus, for distinct background corpora.

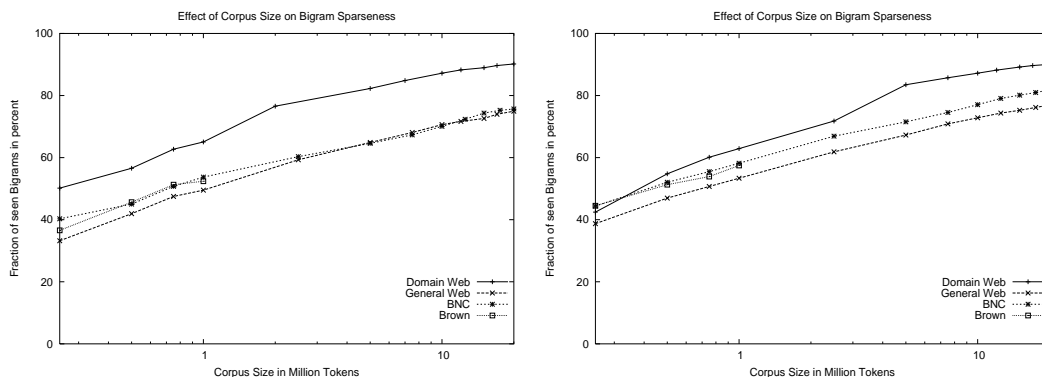


Figure 2: Evolution of seen bigrams for growing corpus size. The left-hand (right-hand) diagram presents the results obtained for the primary evaluation collection Neurology_E (Fish_E).

cated than the method described here, evaluations are centered around a small number of special examples. The problem of how to use the methods in a practical text correction system is not discussed.

The web as a corpus. The web by far represents the largest public repository for natural language texts; it is easily accessible by web search engines. Many recent experiments and technologies in corpus linguistics use the web as a corpus [Volk, 2001; Kilgarriff and Grefenstette, 2003; Resnik and Smith, 2003]. In [Zhu and Rosenfeld, 2001; Keller and Lapata, 2003] it has been shown that search engine hit counts can be successfully used to estimate frequencies of unseen bigrams. This work has been extended to different NLP tasks in [Lapata and Keller, 2005], contrasting web models with those obtained from standard corpora such as the BNC. Other recent disambiguation experiments on n -grams extracted from a topic diverse web corpus of 10 billion words [Liu and Curran, 2006] showed that corpus analysis leads to better results than counting hits of search engines. For IR relevance models this was confirmed by [Diaz and Metzler, 2006].

7 Conclusion

We described a series of experiments where word bigram counts from domain dependent web corpora are used to improve a practical text correction system. A sophisticated crawling strategy was introduced for computing suitable web corpora, taking the vocabulary of the input document into account. To guarantee efficiency, bigram counts for arbitrary bigrams over large dictionaries are kept in main memory, using special techniques. It was shown that bigram counts improve simpler scores based on word similarity and word frequency only. We also compared the reliability of general and domain specific language models retrieved from thematic web corpora. Our results underpin the superiority of the lat-

ter kind of language models over those obtained from static standard corpora. It seems interesting to see what improvements can be obtained using the large list of n -grams recently made available by Google [Brants and Franz, 2006]. It should be obvious that the techniques for computing domain specific language models can be used for various further application areas, such as speech recognition and word sense disambiguation.

Acknowledgments. This work was supported by AICML, iCORE, DFG and VolkswagenStiftung.

References

- [Baroni and Bernardini, 2004] M. Baroni and S. Bernardini. BootCaT: Bootstrapping corpora and terms from the web. In *Proceedings LREC 2004*, pages 1313–1316, Lisbon, 2004.
- [Brants and Franz, 2006] Thorsten Brants and Alex Franz. Web 1t 5-gram version 1. Linguistic Data Consortium, Philadelphia, 2006.
- [Dengel *et al.*, 1997] Andreas Dengel, Rainer Hoch, Frank Hönes, Thorsten Jäger, Michael Malburg, and Achim Weigel. Techniques for improving OCR results. In Horst Bunke and Patrick S.P. Wang, editors, *Handbook of Character Recognition and Document Image Analysis*, pages 227–258. World Scientific, 1997.
- [Diaz and Metzler, 2006] Fernando Diaz and Donald Metzler. Improving the estimation of relevance models using large external corpora. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 154–161, New York, NY, USA, 2006. ACM Press.
- [Golding and Roth, 1999] Andrew R. Golding and Dan Roth. A window-based approach to context-sensitive

- spelling correction. *Machine Learning*, 34(1-3):107–130, 1999.
- [Golding and Schabes, 1996] Andrew R. Golding and Yves Schabes. Combining trigram-based and feature-based methods for context-sensitive spelling correction. In *ACL*, pages 71–78, 1996.
- [Hong and Hull, 1994] Tao Hong and Jonathan J. Hull. Degraded text recognition using word collocation. In *Proceedings of the Conference on Document Recognition, 1994 SPIE Symposium*, pages 334–342, San Jose, CA, 1994.
- [Hwang *et al.*, 1999] Young-Sook Hwang, Bong-Rae Park, Bo-Hyun Yun, Hae-Chang Rim, and Seong-Whan Lee. A contextual post-processing model for Korean OCR using synthesized statistical information. In *Proc. of the 2nd International Conference on Multimodal Interface*, 1999.
- [Jenkins, 1996] Bob Jenkins. An in-memory hash table. <http://burtleburtle.net/bob/hash/hashtab.html>, 1996.
- [Keenan *et al.*, 1991] F.G. Keenan, L.J. Evett, and R.J. Withrow. A large vocabulary stochastic analyser for handwriting recognition. In *Proc. of the First International Conference on Document Analysis and Recognition (ICDAR 91)*, pages 794–802, 1991.
- [Keller and Lapata, 2003] F. Keller and M. Lapata. Using the web to obtain frequencies for unseen bigrams. *Computational Linguistics*, 29(3):459–484, 2003.
- [Kilgarriff and Grefenstette, 2003] Adam Kilgarriff and Gregory Grefenstette. Introduction. *Computational Linguistics - Special Issue on the Web as Corpus*, 29(3):333–348, 2003.
- [Lapata and Keller, 2005] M. Lapata and F. Keller. Web-based models for natural language processing. *ACM Transactions on Speech and Language Processing*, 2:1–31, 2005.
- [Liu and Curran, 2006] Vinci Liu and James R. Curran. Web text corpus for natural language processing. In *Proceedings of EACL-06, 11th Conference of the European Chapter of the Association for Computational Linguistics*, Trento, IT, 2006.
- [Nartker *et al.*, 2003] Thomas A. Nartker, Kazem Taghva, Ron Young, Julie Borsack, and Allen Condit. Ocr correction based on document level knowledge. In *Proceedings IS&T/SPIE 2003 Int. Symp. on Electronic Imaging Science and Technology*, volume 5010, pages 103–110, Santa Clara, CA, 2003.
- [Resnik and Smith, 2003] Philip Resnik and Noah A. Smith. The web as a parallel corpus. *Computational Linguistics - Special Issue on the Web as Corpus*, 29(3):349–380, 2003.
- [Rong Jin, 2003] Alex G. Hauptmann Rong Jin, Chengxiang Zhai. Information retrieval for OCR documents: A content-based probabilistic correction model. In *Proceedings of SPIE, Electronic Imaging '03, Document Recognition and Retrieval Conference DRR(X)*, Santa Clara, CA, 2003.
- [Srihari and Baltus, 1993] R.K. Srihari and C.M. Baltus. Incorporating syntactic constraints in recognizing handwritten sentences. In *Proc. of the 13th Int. Joint Conf. on Artificial Intelligence*, pages 1262–1267, Chambéry, France, 1993.
- [Srihari, 1993] S.N. Srihari. From pixels to paragraphs: the use of contextual models in text recognition. In *Proc. of the Second International Conference on Document Analysis and Recognition*, pages 416–423, Tsukuba Science City, Japan, 1993. IEEE Computer Society Press.
- [Strohmaier *et al.*, 2003a] C. Strohmaier, C. Ringlstetter, K. U. Schulz, and S. Mihov. Lexical postcorrection of OCR-results: The web as a dynamic secondary dictionary? In *Proc. 7th ICDAR 03*, pages 1133–1137, 2003.
- [Strohmaier *et al.*, 2003b] C. Strohmaier, C. Ringlstetter, K. U. Schulz, and S. Mihov. A visual and interactive tool for optimizing lexical postcorrection of OCR results. In *Proceedings of the IEEE Workshop on Document Image Analysis and Recognition, DIAR'03*, 2003.
- [Taghva and Stofsky, 2001] Kazem Taghva and Eric Stofsky. OCRSpell: an interactive spelling correction system for OCR errors in text. *International Journal of Document Analysis and Recognition*, 3:125–137, 2001.
- [Taghva *et al.*, 1996] Kazem Taghva, Julie Borsack, and Allen Condit. Effects of OCR errors on ranking and feedback using the vector space model. *Information Processing and Management*, 32(3):317–327, 1996.
- [Taghva *et al.*, 2004] Kazem Taghva, Thomas Nartker, and Julie Borsack. Information access in the presence of ocr errors. In *HDP '04: Proceedings of the 1st ACM workshop on Hardcopy document processing*, pages 1–8, New York, NY, USA, 2004. ACM Press.
- [Volk, 2001] M. Volk. Exploiting the www as a corpus to resolve pp attachment ambiguities. In *Proceedings of Corpus Linguistics*, Lancaster, UK, 2001.
- [Williams and Zobel, 2005] H.E. Williams and J. Zobel. Searchable words on the web. *International Journal of Digital Libraries*, 5(2):99–105, 2005.
- [Zhu and Rosenfeld, 2001] X. Zhu and R. Rosenfeld. Improving trigram language modeling with the world wide web. In *Proceedings of Int. Conf. on Acoustics, Speech, and Signal Processing*, 2001.