The Relevance of Frequency Lists for Error Correction and Robust Lemmatization

René Schneider and Ingrid Renz

DaimlerChrysler AG, Institute of Information Technology,
Department of Speech Understanding,
Ulm, Germany
{rene.schneider}{ingrid.renz}@daimlerchrysler.com

Abstract

In this paper we discuss the usefulness of frequency lists and the impact they have on a learning algorithm, named rank-and-similarity-based learning. The combination of frequency lists with a simple similarity measure leads to significant results that are useful for bootstrapping a frequency dictionary in which each lexical entry provides information about a stem and its well- and illformed variants. The modification of the frequency lists allows the construction of a collocation measure, determining the syntagmatic relationship of two or more words in a given domain. The results of the algorithm are applied to two different problems that arise in the area of information extraction from paperbound documents, namely error correction, and robust lemmatization. The paper finishes with some remarks concerning the validity and evaluation of the results.

Keywords: Exploratory Textual Data Analysis, Frequency Dictionaries, Lemmatization

1. Problem

A major reason for the development of information extraction (IE-)systems (Bayer et al., 1997) was the fact that quite often, especially in industrial applications, a deep text analysis may be abandoned in favor of the robust extraction and interpretation of relevant text segments. Nevertheless IE-systems still require knowledge bases that vary from application to application and are generally handcrafted. Furthermore, the employment of non-supervised learning algorithms is handicapped due to the very small set of training data available in industrial applications. Additionally, the majority of IE-systems are restricted to the analysis of electronical text input, and those working with paperbound information have to face a considerable amount of "noisy" output, produced by optical character recognition (OCR), together with an unexpected high number of mistakes that are produced during text production, consisting of typos, orthographical and grammatical mistakes.

That means that despite the fact that textual data are nowadays available in a vast amount (either as fixed corpora or with the WorldWideWeb as a source), some industrial applications still have to fight with the problem of sparseness and noisiness of linguistic data, especially in those cases where the textual data that has to be analyzed underlies very specific constraints and consists of nothing more than a sample of 70 - 100 letters, and/or whenever these letters are written by non-native speakers. Table 1 and figure 1 show examples for several distinct features of noisy textual data. All examples are taken from a corpus of requests for business letter reports with 4553 tokens and 1445 types and a text body length differing between 9 and 229 tokens.

Type	Example
Idio-syncrasies	up dating vs. up-dating vs. updating Annual Report vs. annual report
Typos	. . . to our workgroup coördinator . . . . . . send us twoo copies of your latest . . .
Mis-spellings	. . . an independant financial and research society . . . . . . the adress is printed below . . .
Grammatical Errors	. . . We are in need a lot of information about foreign . . . . . . I would like you please to send me a copy of . . .
Code Switching	. . . the main point of my interest is the auto industry . . . . . . würde ich gern aufgenommen werden in mailing-list. . . .

Table 1: Examples for utterances beyond strict wellformedness

2. Methods

2.1. Transformation of Rank-Frequency Lists

Frequency lists are a very common and widespread tool for the determination of general linguistic structures from corpora. In natural language processing they are generally used for information retrieval and text categorisation or for measuring corpus homogeneity and similarity between corpora (Kilgarriff, 1996). In small, domain-specific corpora frequency lists do not

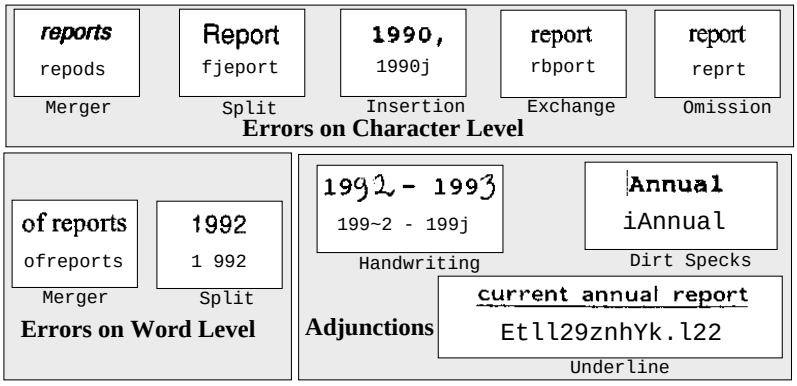

Figure 1: OCR–Errors

only give a rough overview of the preferred and essential words, (see list 1 in table 2), their lexical units also show a gratuitous phenomenon from the linguist's perspective, i.e. that even in very small text corpora, the frequency of wellformed words is much higher than those of illformed words. Furthermore, within the group of wellformed or regular words, those representing the stems or lemmata have a significantly higher frequency than their morphological inflections or derivations. As several empirical investigations have shown, illformed words rarely tend to appear a second or third time exactly in the same illformed shape. To strengthen this trend several transformations of the frequency list are possible: In a first step, we only consider the incremented frequency of a stem together with the frequencies of its variants (which are assigned automatically, see section 2.2). By computing this value and ordering the shrunken list, some stems receive a considerably higher value than other stems with the values of function words decreasing meantimewhile. For a new order of the list see list 2 in table 2.

	1		2		3		4	
rank	freq	word(s)	freq	stem(s)	var	word	weight	stem
1	210	to	211	your	10	report	1610	report
2	209	the	210	to	9	annual	1266	your
3	205	your	209	the	8	thank	1044	annual
4	201	of	201	of	6	your	676	would
5	169	you	169	would	6	information	472	thank
6	164	and	164	and	6	statements	330	please
7	145	in	161	report	6	economic	276	information
8	121	for	145	in	6	institute	258	send
9	120	would	121	for	6	please	255	mailing
10	111	a	116	annual	5	mailing	210	to
11	109	I, we	111	a	5	business	210	the
12	105	annual	109	I, we	5	receive	209	of
13	89	report	86	send	5	other	201	company
14	85	be	85	be	5	english	192	you
15	81	send	71	our	5	possible	169	list
16	71	our	69	if	5	international	168	and
17	69	if	66	please	5	publications	164	business
18	62	reports	64	company	5	collection	145	accounts
19	60	please, as, us	60	as, us	5	germany	144	address
20	54	is, me	59	thank	5	date	132	statements

Table 2: Rank-Frequency Lists (selected ranks)

In a second step the incremented frequencies of the stems are multiplicated with the number of their assigned variants to confirm the hypothesis that the more often a word appears in texts of a restricted category and the more morphological and graphemic variants (see list 3 in table 2) it has, the more probable the word will represent some domain-specific information. The multiplication of frequencies and the number of variants of a word $(freq_x \times var_x)$ enhances the significance of the lexical prototypes or lemmata and leads to a weighted frequency list (see the right column in table 2) whose first ranks comprise the most relevant lemmata and are helpful for the extraction of the salient syntactic patterns (see section 2.3).

2.2. Rank-and-Similarity-Based Learning

To compute the similarity relationship between the different elements of the frequency list (list 1 in table 2), we made use of the Levenshtein distance (Levenshtein, 1975; Nerbonne et al., 1996) or edit distance though it represents a useful string matching technique (Oakes, 1998), making use of three basic operations, namely the insertion, deletion and substitution of symbols, whereas substitution can be seen as the consecutive application of deletion and insertion. Table 3 gives the example for the transformation of the illformed string *nformati0ns* into the corresponding correct form *information.* The similarity or distance between the two strings is

String	**Operation**	**Cost**	**Sum**
nformati0ns	Insert i	1	1
informati0ns	Substitute 0/o	2	3
informations	Delete s	1	4
information			

Table 3: Edit distance between two strings

defined by the minimal number of operations that is necessary for transforming one symbol sequence into another. It may be expressed in a metric value by dividing the total number of transformations through the length of the longest string. In our case the distance between *nformati0ns* and *information* would be 4 in terms of operations or 0.36 as division of the operations through the word length 11.

The general cycle of the learning algorithm, is described as follows (for an illustrative example see figure 2): the first loop starts with the initial element of the ranked list, i.e. the element with the highest significance or degree–of–interest and compares this element with every succeeding element of the list. Each element, bearing a lower similarity to the top element as indicated by the threshold value is put into one class with the top element as class representative. Both list elements are simultaneously taken out of the list. In the second and all consecutive loops, the algorithm proceeds in the same way, comparing the respective top element with the remaining list elements until the list is shrunken to a group of elements that shows no significant similarity to all the other elements. The algorithm is deterministic insofar as it does not allow any ambi-

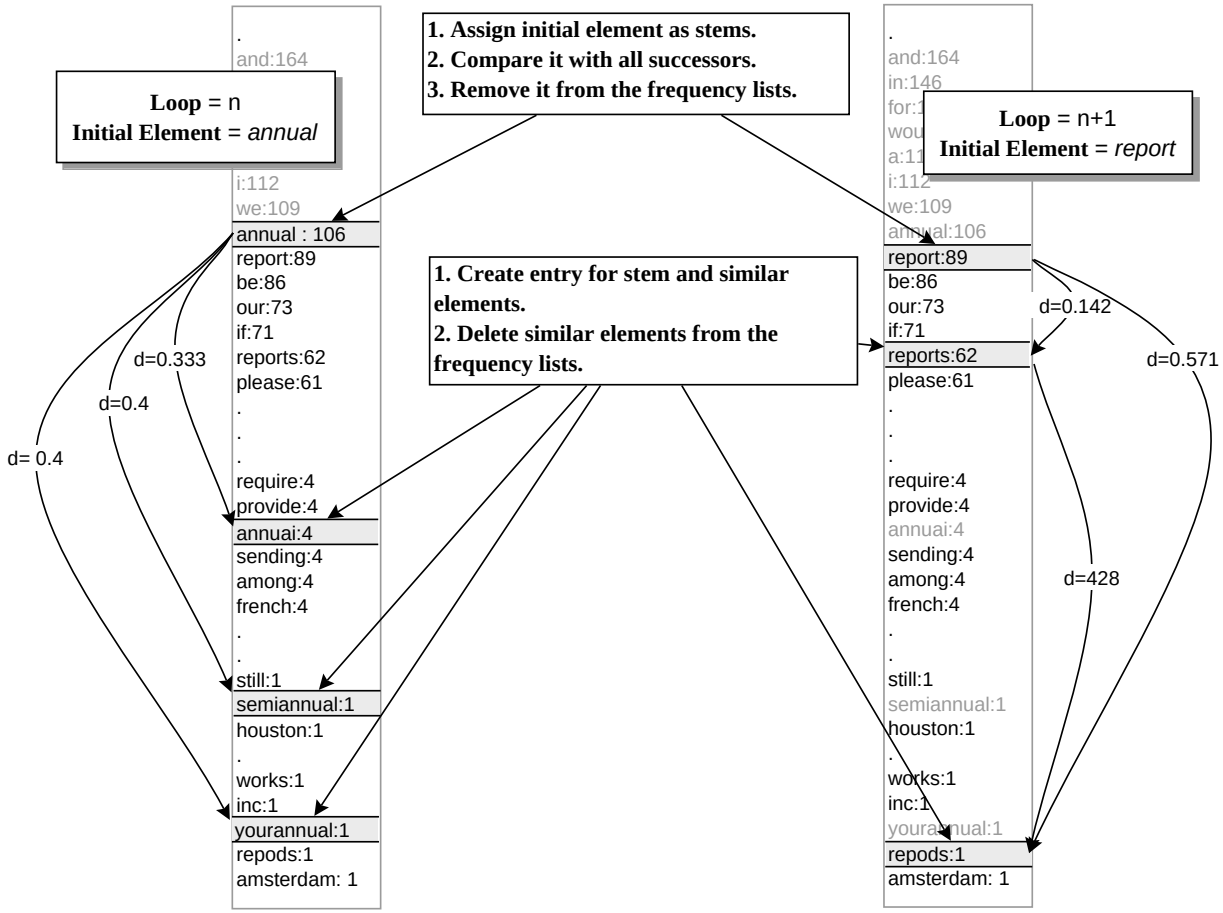

Figure 2: Automatic Acquisition of lexical entries

guity concerning the membership of the elements to a certain class, i.e. any variant, although it might have the same or even a lower similarity to one or several other elements is assigned to the respective initial list element. Thus, emphasis is given to the rank which is also deterministic for the differentiation between the class representative and the different class members.

A possible but optional extension of the algorithm might be achieved through the introduction

of a recursive procedure, as already indicated in the processing of the ranked list for the top element *report* in figure 2: in this case, the successive list elements are not only searched for the variants of the initial element, but additionally for strings that show the same similarity to the variants already found with a slightly lower similarity to the initial element. That means, we are no longer comparing the strings with the constraint of one single threshhold value for the direct similarity between the top element and its successors but also with a threshold value for indirect similarity; we thus consider the relationship between the top element and a successor via an already assigned variant that acts as a mediating element.

2.3. Definition of a Collocation Measure

As will be seen in section 3 the results of the rank-and-similarity based learning algorithm are useful to "clean" documents from noisy sequences but so far they do not bear any information about the syntagmatic relations or dependencies that exist in texts of a given domain. To reveal these dependencies, the original corpus was transformed into a lemmatized version (see figure 3), consisting only of the earlier derived prototypes together with their "weighted ranks". The concluding definition of a collocation measure follows the Firthian notion of "knowing a word by the company it keeps" (Firth, 1957), a postulate which emphasizes the fact that certain words have a strong tendency to be used together. Therefore, the texts are transformed parallely into a corpus of indices implying the ranks that are given to the lemmata after they have been weighted. With the help of the weighted ranks, it is possible to compute a probabilistic value similar to transition likelihoods. Looking at a pattern or window of several words w_i of a given pattern length n, we add up the ranks of the weighted frequency lists $\tilde{r}_{w_i}$ to $\tilde{r}_{w_n}$ and compute the average rank. This value is divided by the overall frequency $freq$ of the whole pattern $(w_1..w_n)$:

$$\tilde{C}_{(w_1..w_n)} = \frac{\sum_{i=1}^{n} \tilde{r}_{w_i}}{n \cdot freq(w_1..w_n)}$$

The resulting value represents the weighted likelihood for the co–occurrence $\tilde{C}_{(w_1..w_n)}$ of two (or more) words indicating how probable a word precedes or succeeds another word. To give an example for a word pattern of two words like *mailing list*, the equation is solved a follows:

$$\tilde{C}_{(mailing\ list)} = \frac{\tilde{r}_{mailing} + \tilde{r}_{list}}{2 \cdot freq(mailing\ list)} = \frac{9 + 15}{2 \cdot 35} = 0.343$$

or for longer patterns with a lower degree–of–interest, such as *as any interim*:

$$\tilde{C}_{(as\ any\ interim)} = \frac{\tilde{r}_{as} + \tilde{r}_{any} + \tilde{r}_{interim}}{3 \cdot freq(as\ any\ interim)} = \frac{53 + 87 + 28}{3 \cdot 1} = 56.0$$

Compared to other collocation measures, this value does not only take account of a word's frequencies and the collocation's frequencies (as e.g. Mutual Information (Church and Hanks, 1990) or transition likelihood but combines these two properties with a third one: the word's different modalities as indicated by their number of variants, i.e. their weighted ranks. This last value weakens the influence of both less frequent and functional words and supports the degree–of–interest of domain-specific and correct words.

The collocation values may be labeled to the arcs of the finite-state-automata that are built for the syntactic analysis to make the parsing process more effective since a low transition value reflects a high significance resp. a high degree–of–interest in texts of a certain domain.

3. Results

The results of the predecessing algorithms will be illustrated with the analysis of a document (see figures 3), which did not belong to the text collection that was used to calculate the lexical entries together with their syntagmatic relationships. Firstly, the paperbound document was processed by an OCR-component and the results stored in ASCII-format. In a second step, named tokenization, the lexical units were defined, superfluous characters were replaced by a single grapheme, and punctuation marks were isolated from the tokens. In the following

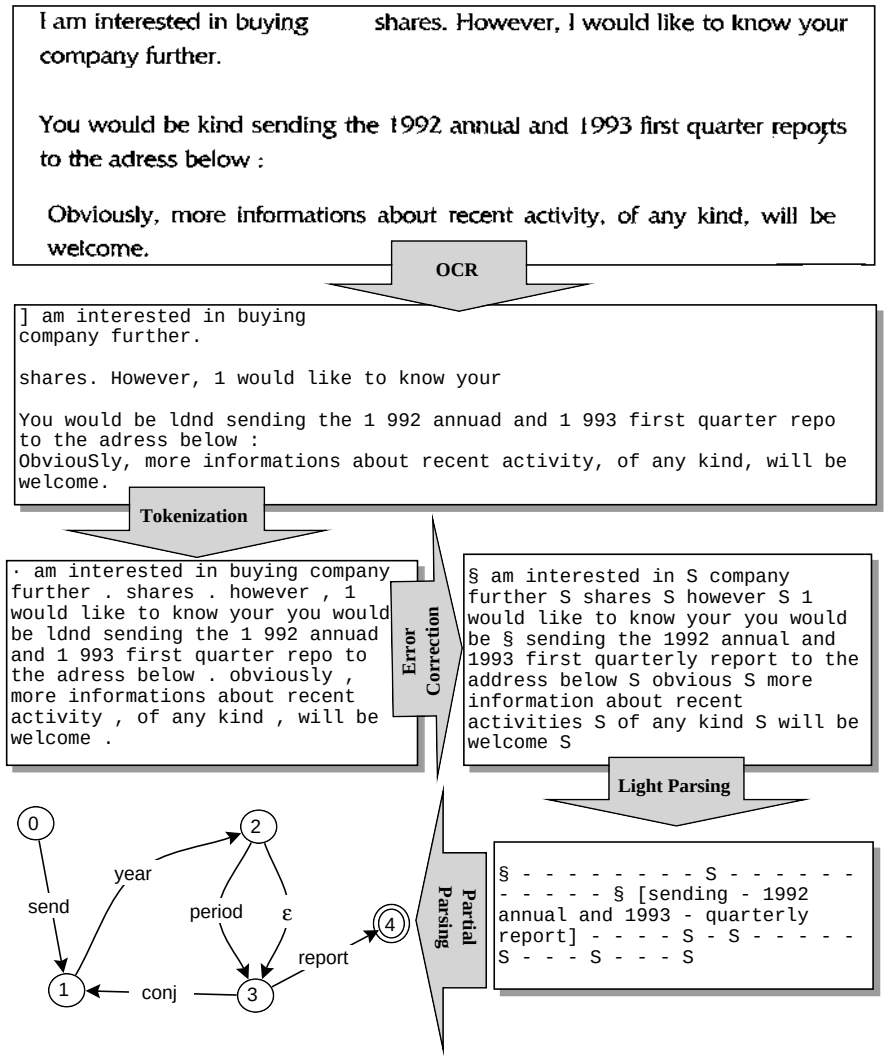

Figure 3: Processing an unknown document

step, the lexical entries were generated (as described in 2.2) to convert the several illformed and inflected word forms into their corresponding stems. Afterwards, word forms that play no importance for the syntactical analysis, such as words with a low degree–of–interest are replaced through a single symbol (in our case the paragraph symbol) with only the domain-specific and message-relevant stems remaining. The parsing starts with the analysis of the syntactic fragments with a high relevance for the extraction task as indicated by their collocation measures. In the final step the rests of the original text are analyzed with pre-defined finite-state-automata.

4. Evaluation

In order to examine the performance of our information extraction component, we evaluated the implementation. With most of the relevant errors being lemmatized correctly, the results are promising: since we modeled carefully the most typical linguistic expressions for the finite-state analysis, the error rate is below 5% (accepting a rejection rate of about 20%). As to the efficiency of the components, all results are immediately returned, independently of the length of the input. These evaluations indicate that robust lemmatization and the minimal definition of restricted linguistic resources are sufficient for a correct and efficient information extraction in the selected domain.

Similar studies on Information Extraction beyond wellformedness and automatic knowledge acquisition have been done in other domains, e.g. a German corpus with letters of cancellation of mobile telephone cards and another corpus with requests for indemnification of car damages. All applicatons showed similar results whereas a high domain specificity strenghtens the validity of the results received.

5. Conclusions and Further Applications

In this paper we presented a method for error correction and robust lemmatization with the ambition of finding an empirical learning technique for information extraction tasks (Cardie, 1997). To test the adaption of the learning algorithm to further applications, we integrated the learning algorithm to a information retrieval system for calculation and retrieval of similar text documents in intranets (Bohnacker et al., 1999). In this context error correction plays a minor role in favour of feature reduction (to reduce the number of text features and consequently computation time) and feature unification (i.e. different word forms are unified under one single stem, e.g. *download*, *downloads*, *downloaded* and *downloaden* are treated as one feature) to receive a higher similarity between text vectors. Due to the heterogenity of the lexical units, the threshold value had to be diminished for a document collection containing english and german texts. Besides that, a group of words like function words and words with an extremely high frequency were not considered. The results are encouraging and recommend the learning technique as a general technique in natural language processing tasks, esp. in the area of information retrieval, extraction and filtering.

References

Bayer T., Bohnacker U., and Renz I. (1997). Information extraction from paper documents. In Bunke H. and Wang P. editors, *Handbook on Optical Character Recognition and Document Image Analysis*, pages 653–677. World Scientific Publishing Company, Singapore.

Bohnacker U., Franke J., Mogg-Schneider H., Renz I., and Veltmann G. (1999). Restructuring intranets by computing text similarity. In *Proceedings of the Conference on Terminology and Knowledge Engineering (= TKE 99)*, pages 610–617.

Cardie C. (1997). Empirical methods in information extraction. *AI magazine*, 18(4):65–80.

Church K. W. and Hanks P. (1990). Word association norms, mutual information and lexicography. *Computational Linguistics*, 16(3):22–29.

Firth J. R. (1957). Modes of meaning. In *J.R. Firth: Papers in Linguistics*, pages 190–215, London. Oxford University Press.

Kilgarriff A. (1996). Using word frequency lists to measure corpus homogeneity and similarity between corpora. Technical report, Information Technology Research Institute, University of Brighton, UK.

Levenshtein V. I. (1975). On the minimal redundancy of binary error-correcting codes. *Information and Control*, 28(4):268–291.

Nerbonne J., Heeringa W., van den Hout E., van der Kooi P., Otten S., and van de Vis W. (1996). Phonetic distance between dutch dialects. In *Proceedings of Computational Linguistics in the Netherlands (CLIN-96)*, pages 185–202.

Oakes M. P. (1998). *Statistics for Corpus Linguistics*. Edinburgh University Press, Edinburgh.

Schneider R. (1998). *Maschineller Erwerb lexikalischen Wissens aus kleinen und verrauschten Textkorpora*. Utz Verlag, München.