Document Highlighting —
Message Classification in Printed Business Letters

Rainer Hoch, Andreas Dengel
German Research Center for Artificial Intelligence (DFKI)
P.O. Box 20 80, D-6750 Kaiserslautern, Germany
E-mail: {hoch, dengel}@dfki.uni-kl.de

Abstract

This paper presents the INFOCLAS system applying statistical methods of information retrieval primarily for the classification of German business letters into corresponding message types such as order, offer, confirmation, etc. INFOCLAS is a first step towards understanding of documents. Actually, it is composed of three modules: the central indexer (extraction and weighting of indexing terms), the classifier (classification of business letters into given types) and the focuser (highlighting relevant letter parts). The system employs several knowledge sources including a database of about 100 letters, word frequency statistics for German, message type specific words, morphological knowledge as well as the underlying document model. As output, the system evaluates a set of weighted hypotheses about the type of letter at hand, or highlights relevant text (text focus), respectively. Classification of documents allows the automatic distribution or archiving of letters and is also an excellent starting point for higher-level document analysis. [1]

CONTENTS:

[1] This work has been supported by the Germany Ministry for Research and Technology (BMFT).

1 Introduction

There exist various models for representing the contents of a document. With respect to the application-specific requirements, a document may be represented by entire sets of words, by word signatures, by its physical organization (layout structure) and/or its composition of meaningful constituents (logical objects), or further by complex semantic structures. Exemplary work is given by [1] for a word-based representation, by [2, 3, 4] for a descriptor-based representation, by [5, 6] for structural representation, and by [7, 8, 9, 10] for frame- and script-based representations. However, most of the techniques addressing document representation are driven either from database requirements or the perspective of text categorization or document analysis.

While document analysis concentrates on a transformation of printed information into a symbolic equivalent representation accessible by electronic services, aspects traditionally associated with databases focus on efficient techniques for filing and retrieval of electronically available data. In this way, both research fields have a common superior goal: to make printed information available for electronic access. In other words, after analyzing and symbolically reproducing the unstructured printed document data, it should be fit into a predefined structured scheme or pattern capturing those aspects of information relevant for an individual user, a specific task or application.

To do so, the printed document must be *understood* by computer up to a degree where efficient techniques for electronic information management may be initiated. Understanding a document does not necessarily imply a full comprehension of the contained text, establishing unlimited relations and associations w.r.t. common sense (ref. [11]). Instead, it is sufficient to step into a document to get more value out of it. This limited strategy—also designated as *information refining* [12]—seems to provide enough support for applications in restricted domains, such as automatic indexing, task or form processing, and library systems (cf. [13, 14, 15, 16).

In this paper, we describe our prototypical system INFOCLAS for indexing and classifying printed business letters. The system has been developed as a tool to provide expectations to the message possibly conveyed in business letters, such as offer or invoice. Thus, it serves as an aid to guide further text analysis procedures enabling application-oriented document understanding. In Section 2, we first give some background information about our efforts in document analysis to get a feeling regarding how the INFOCLAS system is used. While Section 3 illustrates the overall system architecture, Section 4 explains the system components of INFOCLAS in more detail. Then Section 5 presents the first results of using INFOCLAS. Section 6 concludes the paper outlining current research activities.

2 Background

In the project called ALV—a German acronym for *Automatic Reading and Understanding*—we focus on an analysis of printed business letters stepwise enhancing the information about the document by refining its structure and its contents. Fig. 1 shows the *document analysis staircase* of ALV illuminating the individual processing steps and the corresponding enhancement in information.

The ultimate goal of our research activities is an attempt to discern the central message conveyed in printed business letters. For example, this includes verification of the persons involved, such as *sender* and *recipient*, the type of message, such as *offer*, *order*, or *receipt*, but also aspects of information related to the message, such as *date*, *name of product*, *number* and *price*.

Starting with a binary image of 200 dpi, the physical organization of a document is extracted and subsequently mapped into a complementary logical structure only using geometric knowledge [17]. As a consequence, logical constituents or objects, such as *recipient*, *date*, *letter-body*, etc. provide contextual views to parts of the letter. Corresponding logical dictionaries may be accessed to verify fragmentary strings resulting from OCR as legal words [18]. Each of the logical objets may be characterized by a typical structure, by a specific complexity as well as by typical contents. Up to these characteristics, different strategies are required to solve the problem of verifying the captured information.

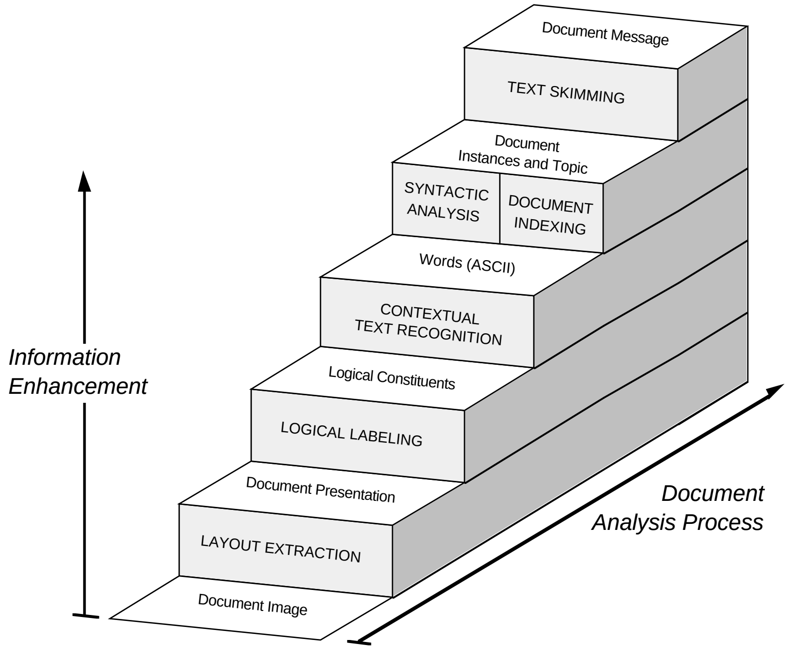

Figure 1: Stepping into a document (document analysis staircase of ALV).

Considering simple logical objects (e.g., *date* or *sender*) it is obvious that single words are arranged side by side using very strong syntactic conventions and therefore may be analyzed by applying syntactic knowledge [19]. More complex objects (e.g., *subject* or *letter-body*) mainly follow a natural language dialogue. For these reasons, other techniques are required to understand the message of business letters.

INFOCLAS is a first step towards such message understanding. Using statistical methods of conventional information retrieval (IR) we differentiate the letters into different classes which are called *message types* according to the EDIFACT standard [20]. Actually we are able to analyze five distinct message types, that are *order*, *offer*, *inquiry*, *confirmation*, and *advertisement*. In the near future, other message types (e.g., invoice) will also be incorporated.

A classification of electronic documents is advantageous for several reasons. Primarily, automatic distribution, further processing and archiving of letters is simplified. By this way, possible application scenarios are inhouse mail or fax distribution, knowledge based indexing of documents as well as automatic task processing [14]. Second, a hypothesis about the type of letter is an excellent starting point for higher-level document analysis.

In our domain, each message type is represented as a frame-like structure consisting of a collection of real world entities being related to each other. These entities may describe individuals, such as persons, physical objects, abstract terms, or events which are characteristic of a certain message type. For example, an offer may be composed of entities *supplier, recipient, subject, price, number, date,* etc. Additionally, the entities may be hierarchically organized in an inheritance network (see Fig. 2).

For message identification, these entities have to be instantiated during text analysis. To achieve this goal, we use the INFOCLAS system which is based on classical IR techniques to extract significant words of a letter on a purely statistical basis [2]. These words, or index terms, are also weighted according to their likelihood of relevance. Our experience shows that some words are the most characteristic of particular message types. For instance, typical offers include word inflections of the German verb “anbieten” (infinitive), “angeboten” (past participle), “boten <*text*> an” (simple past) or their synonyms.

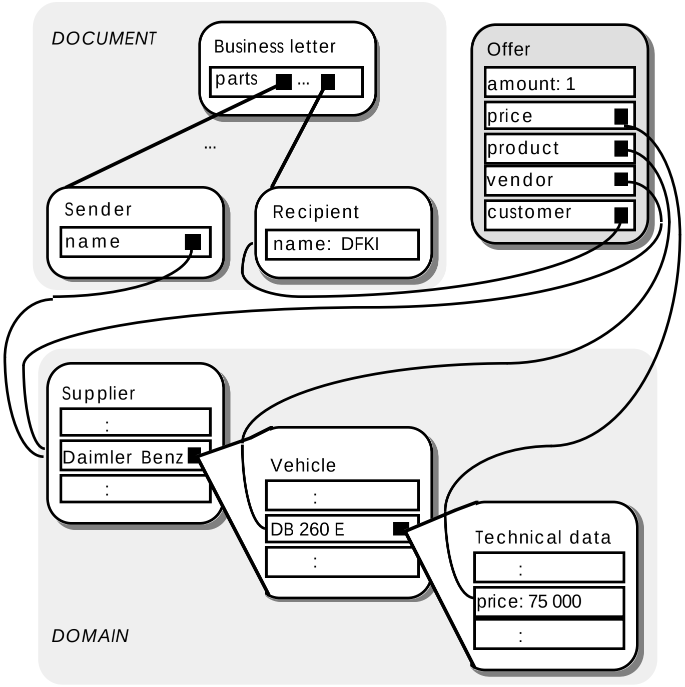

Figure 2: Example of message identification.

3 System Overview

INFOCLAS is a knowledge-based system enhancing the capabilities of our document analysis system Π_{ODA} being presented in [17]. The name of INFOCLAS stands for **in**dexing, **fo**cussing and **clas**sifying business letters reflecting the three principal tasks of the system. Consequently, the system consists of three main modules: the central indexer, the focuser as well as the classifier which are illustrated in Fig. 3. While the indexer is a prerequisite for further analysis by extracting weighted index terms of a document, the two other modules can be used independently from each other or can mutually support themselves sharing temporary results.

While simple (i.e., well-structured) parts of the letter such as *recipient, sender* and *date* are checked syntactically for verification of recognition results, the INFOCLAS system concentrates on those parts containing free text, especially on the *subject* and the letter‘s *body*. For these two logical objects, an examination is most promising because statistical methods usually rely on natural language texts in contrast to the rather syntactically limited and well-structured constituents of a letter (sender, recipient, date).

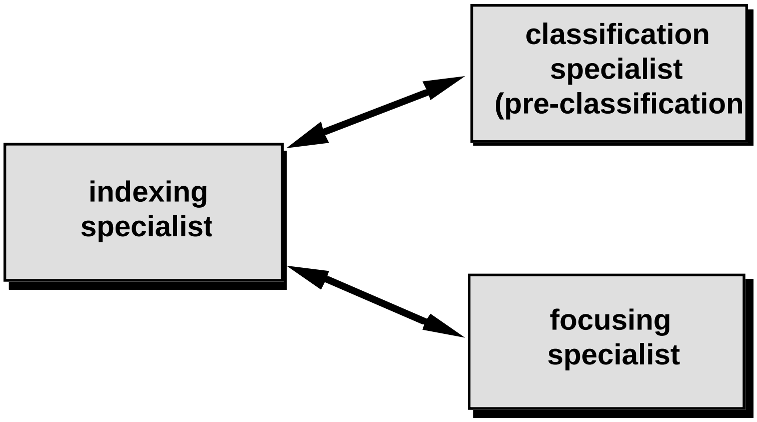

Figure 3: INFOCLAS system architecture.

As input, INFOCLAS expects a list of word hypotheses (e.g., of body) ascertained from character recognition. Fig. 4 gives an impression of how a typical German business letter is represented by a Lisp datastructure in our database where the last sublist reveals the logical object identification.

As output INFOCLAS produces a list of weighted hypotheses about possible types of message recognized, or a focus of text indicating the letter‘s most important phrase or sentence, respectively. To our mind the focus of a letter may be indicated by a cluster of high-weighted index terms within a small range of text, e.g., at the beginning of a letter‘s body. This focal information can be used from other specialists, for example, from a flexible and robust syntactic parser taking respective index terms as the starting-point (seed) for island parsing. [2]

[2] We admit that this definition of focus should be interpreted with caution. Even human readers have difficulties with localizing the focus of a letter, as our tests reveal.

In the next sections, we will explain processing steps of all components in detail.

```
((("Deutsche" #\newline "Forschungsanstalt" #\newline "für" "Luft-" #\newline
"und" "Raumfahrt" "e." "V.")("DLR" "Hauptabteilung" "Beschaffung" #\newline
"Postfach" "90" "60" "58," "5000" "Köln" "90")
("DeutschesForschungszentrum" #\newline "für" "Künstliche" "Intelligenz" "GmbH"
#\newline "Erwin-Schrödinger-Str." "(Bau" "57)" #\newline #\newline "6750"
"Kaiserslautern")
("Köln-Porz," "30.01.90")
("Betr.:" "Vertrag-Nr." "5-112-4308" #\newline "ü" "b" "e" "r" "\"Anfertigung"
"einer" "Studie" ""Wissensbank'\"")
("Sehr" "geehrte" "Damen" "und" "Herren,")
("als" "Anlage" "übersenden" "wir" "Ihnen" "den" "Entwurf" "des" "obigen"
"Vertrages" #\newline "in" "dreifacher" "Ausfertigung" "mit" "der" "Bitte" "um"
"Unterzeichnung" "und" "Rück-" #\newline "sendung" "von" "zwei" "Exemplaren"
"an" "die" "DLR," "Hauptabteilung" "Beschaffung," #\newline "Linder" "Höhe,"
"5000" "Köln" "90." #\newline #\newline "Nach" "Gegenzeichnung" "durch" "uns"
"erhalten" "Sie" "ein" "Original" "des" "Vertrages" #\newline "für" "Ihre" "Akten.")
("Mit" "freundlichen" "Grüßen")
("i.A." "R." "Derkum" "i.A." "H." "D'moch")
("Anlagen:" "3" "Vertragsexemplare"))
((company) sender recipient date subject salutation body regards subscript enclosure))
```

Figure 4: Internal representation of the letter 48 of database (in German).

4 System Components

4.1 Indexer

Applying IR techniques we extract keywords or phrases from a business letter. In the IR literature, these keywords are designated as *index terms* or *descriptors* and the process of ascertaining terms is known as *automatic indexing*. Additionally, weights are computed assigning indications of importance to terms [2, 3].

In this way, a letter can be represented by an n-dimensional vector of pairs

$L_j = ((d_{j1}, w_{j1}), (d_{j2}, w_{j2}), ... , (d_{jn}, w_{jn}))$

where L_j = letter j, d_{ji} = descriptor i in letter j, and w_{ji} = weight of descriptor d_{ji}.

The external interface and processing model of the indexing specialist are depicted in Fig. 5. Initial character recognition yields the needed word candidates for the indexer. In addition, contextual information is provided during logical labeling [17].

Typically, indexing with INFOCLAS is performed on the logical objects *subject* and *body* of a letter. Furthermore, there is no fixed vocabulary for indexing, so all text words are used for content identification (free text indexing).

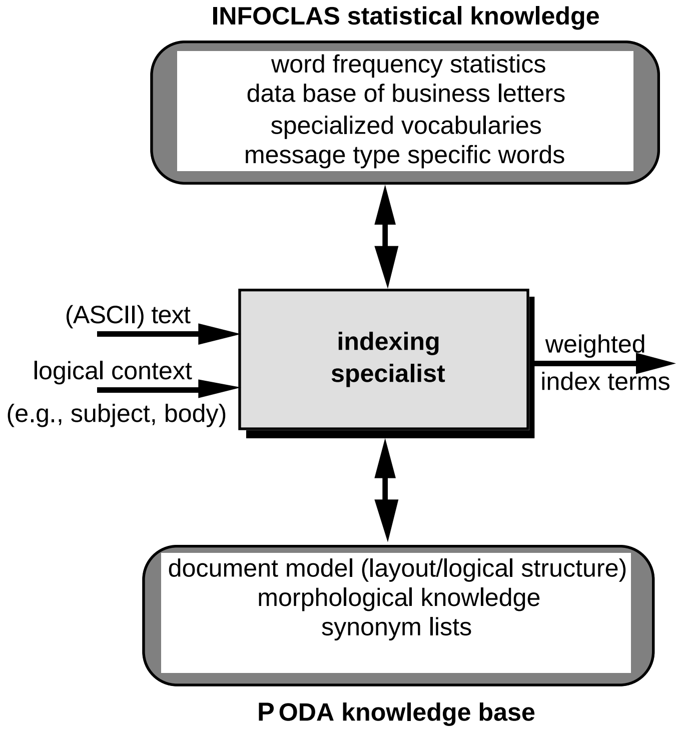

Figure 5: External interface of indexing module.

INFOCLAS engages two kinds of knowledge sources: statistical knowledge as well as the Π_{ODA} knowledge base [19] (cf. Fig. 5). Statistical knowledge comprises common word frequencies of German, some specialized vocabularies (common abbreviations, cities, countries, employee names, etc.), message type specific words (for offer, order, confirmation, etc.) and a database of already analyzed business letters. The Π_{ODA} knowledge base makes use of a document model for business letters, a morphological analysis tool and simple synonym lists. These knowledge sources are used for improving classification results.

Computation with the indexer now proceeds in four steps (see Fig. 6): morphological analysis, stop word reduction, frequency analysis, and index term weighting.

Morphological analysis. First, a morphological tool for German reduces all input words to their respective stems. We use the freely available morphological analysis component MORPHIX 3.0 [21]. MORPHIX handles all inflectional phenomena of the German language by considering morphologic regularities as the basis for defining fine-grained word-class specific subclassification. Besides morphosyntactic features, there are also phonological aspects which are considered in refining the classification. In spite of the complexity of German inflections, the tool is very fast. The average time for analyzing one word lies between 0.01 and 0.02 cpu-seconds, though the system is implemented in Common Lisp.

Stop word reduction. As output MORPHIX yields morphological and syntactical information for each input word form. For instance, word category (part-of-speech), case, gender, number, tense, etc. are conveyed. In contrast to traditional information retrieval systems which initially eliminate stop words and then apply suffix-stripping to reduce the remaining words to their word stems, our approach is reversed. After morphological analysis we delete irrelevant stop words using part-of-speech information. Only words of category noun, verb, adjective, and adverbs, or unknown words to MORPHIX, respectively, are further considered. These *valuable* words directly correspond to the so-called *open category* of words in a language which is rather dynamic in opposition to the *closed category* of articles, conjunctions, prepositions, pronouns, determiners, etc. [22]. Our opinion is that only elements of the open word class are significant for content identification of a letter.

Because the internal lexicon of MORPHIX is small, we enlarged it, in particular entering domain-specific words (but also other general German word forms). Additionally, we improved its I/O-interface to deal with German umlauts (ä, ö, ü) and s-zet (ß). However, one crucial problem still remains. MORPHIX does not handle word composites (compound nouns) which frequently occur in the German language. However, common string matching algorithms can solve this problem.

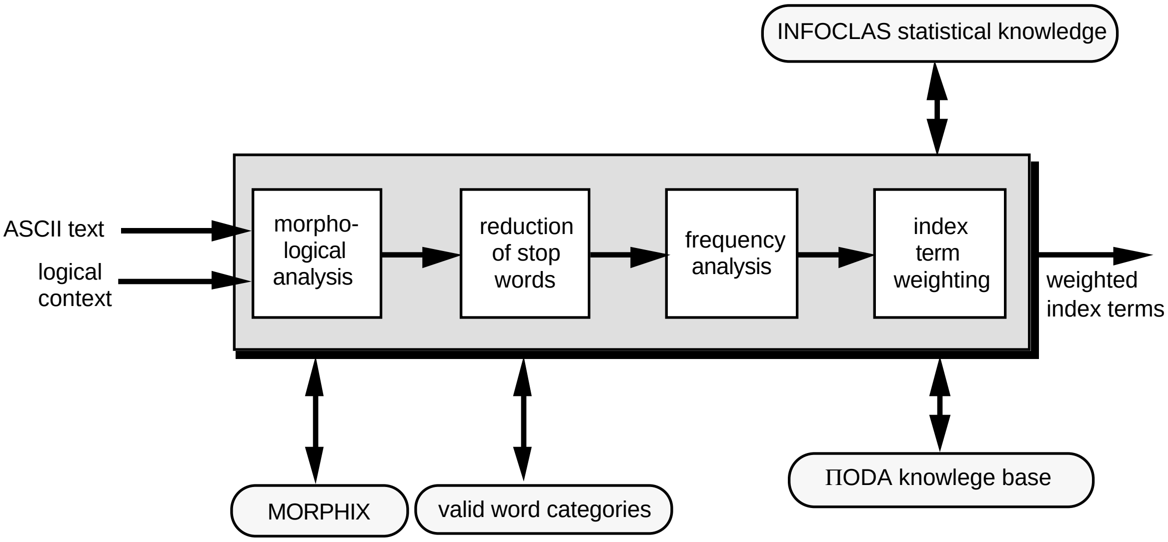

Figure 6: Components of indexing module.

Frequency analysis. In a third step, the indexer performs a frequency analysis of all remaining word stems, i.e., words of category noun, verb, adjective, and adverb. We distinguish between *relative* and *absolute frequency* measures for the identification of content indicators [2, 3]. Frequencies relative to the actual letter are evaluated, i.e., how often each word stem occurs within one letter locally, as well as absolute measures corresponding with the entire database of already analyzed letters (i.e., the collection). Moreover, absolute frequencies are stored in an inverted index file of terms for efficient retrieval.

Note that these frequency measures also represent primitive weighting functions. In practice, however, they are too crude for content identification. For example, an initial phrase often used in German business letters is the salutation “Sehr geehrte Damen und Herren” (Dear Sirs/Madams), but has no deep significance. Therefore, we apply other conventional IR weighting functions derived from these basic frequencies.

Index term weighting. Finally, the central component of index term weighting is invoked. The user can now select between three different weighting functions, including either an *inverse document frequency* function (W1), the *information value* of a term suggested by information theory (W2), or, optionally, the *term discrimination value* (W3) [2]. In the following, we will explain these functions briefly.

The idea behind inverse document frequency is assigning high weights of importance to terms occurring in only a few documents. Hence, the weight of an index term is proportional to its relative frequency in a letter and inversely proportional to the number of letters containing this term. Formula (W1) mirrors this fact:

$\text{weight}_{ik} = \text{freq}_{ik} * \log_2(n) - (\log_2(\text{docfreq}_k) + 1)$ (W1)

where i = document i, k = index term k, docfreq = number of documents in collection containing index term k.

Computation of the information value of a term is more complicated. In information theory, the information content of a message/term can be measured as an inverse function of the probability of occurrence of the words in the text. Moreover, a so-called *noise* of information may be defined which varies inversely with the concentration of the term within the collection. While for perfectly even distributions the noise is maximized, the noise becomes zero when a term only appears in one document. An inverse function of this noise, i.e., the signal value of a term, might then be used as term value. Thus, the corresponding weighting function may be defined as (for details see [2]):

$\text{signal}_k = \log_2(\text{totalfreq}_k) - \text{noise}_k$ (W2)

where i = document i, k = index term k, totalfreq = count of occurrences of term k within collection.

One of the most sophisticated approaches to automatic indexing is the computation of discrimination values. Here, each value indicates how well a term helps to distinguish documents from each other. If the index terms of all documents are equal, the *density* of the document space is maximal (= 1) which is not desired for retrieval. Typically, distance or similarity measures are used to represent the similarity of documents (e.g., cosine coefficient, Dice, Jaccard, etc.). The discrimination value of a term k is then computed by subtracting the density of the document space from the density of the space where term k is eliminated by means of similarities. So, respective weighting of term k is defined as:

$\text{weight}_{ik} = \text{freq}_{ik} * \text{discvalue}_k$ (W3)

As our run time measurements reveal, computation of the term discrimination value is very expensive. The discrimination value measures the degree to which the use of each index term of a document will help to distinguish the documents of each other. In particular, the dynamics of our letter database as well as the usage of a free index term vocabulary leads to computational load, though we have used a centroid for efficient similarity computation [23]. In Section 5, we compare results of the three functions W1, W2 and W3 as well as term frequency.

4.2 Focuser

The focuser has the task to hypothesize phrases or sentences which are supposed to contain the relevant information of a message. In this sense such a focus represents a center of attention for subsequent analysis steps. For instance, we use the focus of a letter, i.e., a small range of text, as the starting point for robust island parsing also dealing with word alternatives. Note that a complete text understanding and full semantic analysis, however, is not intended.

In fact, the process of focusing is rather simple. Initially, the smallest unit of focusing is determined either based on text sentences (e.g., one sentence focus) or a range of words (e.g., five word focus). Incidentally, the first option presumes a correct segmentation of punctuation marks when recognizing characters. Using a sliding window technique the weights of the corresponding index terms are added and normalized by the length of a unit. Parameters of the focuser include the logical object(s) of interest, the weighting function of indexing, the length of focus, and the divisor for normalization. Fig. 8 illustrates one result of focusing.

4.3 Classifier

The classifier has the task to generate weighted hypotheses about the message type of a business letter. In fact, we are able to analyze five message types including order, offer, inquiry, confirmation, and advertisement; other messages are being modelled (e.g., invoice). Our terminology of message types has its origin in the EDIFACT standard [24]. EDIFACT is an application-driven standard for a common representation to interchange data of transport, commerce and administration. We take just these message types from EDIFACT which were adequate for our initial database consisting of about 90 incoming DFKI letter (the learning set). Since the system is open, new message types can easily be integrated.

At the moment, each message type is represented by lists of *primary, secondary and tertiary words* (more precisely, word stems), so-called *message type specific words*. While primary words are most significant and characteristic for one certain message type, secondary words as well as tertiary words may be shared from several messages. For example, Fig. 7 gives all primary and secondary words of message type “offer”. The message type specific words were evaluated carefully, first by means of frequency analyses and then improving the quality of the resulting lists manually. Usage of synonym lists or a thesaurus could further improve the quality of these lists.

During classification of a letter all weighted index terms, computed by the indexer, are matched against specific words of each message type. There are different parameters (called multipliers) controlling this matching process. Primary words have a higher multiplier when they match index terms in comparison with secondary and tertiary words (lowest value), thus indicating their greater importance and allowing a fine-tuning of classification. In this manner all index terms of a letter are matched against all specific words and multiplied with their respective parameters; the resulting values are added and finally normalized by the number of index terms. The process is then repeated for each single message type. As result the classifier generates an ordered list of weighted hypotheses about the type of message recognized.

```
   primary:  (("angebot" "bestell" "bitt" "bezugnehm" "lieferung" "liefer" "netto" "skonto"
              "zahlungsbedingung" "bestellung" "rechnung" "schick" "send")
   secondary:      ("auflage" "beschreibung" "verfuegung" "anfertigung" "anlage"
              "eventuell" "garantie" "rueckfrage" "summe" "zahlbar" "adresse"
              "beabsichtig" "bedingung" "beitrag" "bemuehung" "direkt" "einzelpreis"
              "obig" "rabatt" "telefon" "telefongespraech" "anhang" "auffuehr" "auftrag"
              "bedarf" "beifueg" "beiliegend" "beinhalt" "einzeln" "erbitt" "erhalt"
              "gesamtsumme" "herstell" "lieferbedingung" "moeglich" "moeglichst" "neu"
              "neuauflage" "probeexemplar" "schnell" "schreib" "telefonat" "uebersend"
              "umfang" "verfuegbar" "version" "vorlieg" "zahlung" "zwischensumme"
              "zahl" "nachfolg")
```

Figure 7: Primary and secondary German words for message type “order”.

Fig. 8 presents classification and focusing results of letter 48 of our database. In parentheses absolute weights of computation are shown. The first table reveals that this letter has a high probability of being a confirmation while the other message type hypotheses have equal probabilities and thus are less probable. The second table shows that the central message of the letter might be found by analyzing sentences one and two.

```
***************************************
Classification results of letter 48 of database:
-----------------------------------------------------------
BEBE (confirmation):   41.81 %   (71.92)
ANGE (offer):          19.52 %   (33.59)
ANFR (inquiry):        15.14 %   (26.05)
BEST (order):          14.86 %   (25.57)
WERB (advertisement):   8.66 %   (14.90)
***************************************
Focusing results of letter 48 (decreasing order):
-----------------------------------------------------------
1st:                 sentence: 2     (2.44)
2nd:                 sentence: 1     (2.28)
3rd:                 sentence: 3     (0.36)
***************************************
```

Figure 8: Results of classifying and focusing the letter 48 of database.

5 Experimental Results

INFOCLAS is completely implemented in Common Lisp/CLOS and currently runs on Sun SPARCstation. There are three interfaces to INFOCLAS, a functional programmer‘s interface, a powerful menu-driven Lisp user interface as well as a comfortable graphical user interface implemented with the Window Tool Kit of Common Lisp. The menu interface allows a change of parameter settings interactively and browsing the actual letter database.

There are two inherent problems when indexing letters from our database. On the one hand, typical business letters on their own are small comprising one or two pages at most. On the other hand, the experimental letter database is currently not very large (about 100 letters) for reasons of lacking resources, thus implying a small set of reasonable message types being modelled. Typically, between 10 and 25 letters were analyzed to extract specific words of each message type. However, the essential question is when is a database of letters large enough for expressiveness of empirical results? In a paper by Croft [25] this issue is thoroughly discussed.

The learning set for INFOCLAS consisted of 61 incoming DFKI letters which were selected at random and scanned with a commercial character recognition system (from about 100 incoming letters we discarded the English ones). All letters are typical in some sense: they comprise one or two pages, do not contain tables or figures in the body (company logos are ignored), have some well-structured parts such as addresses, date, enclosures, and company specific information as well as more complex objects (subject, body) of correspondence. We transformed then the output of the OCR system into the internal list representation of INFOCLAS (cf. Fig. 4).

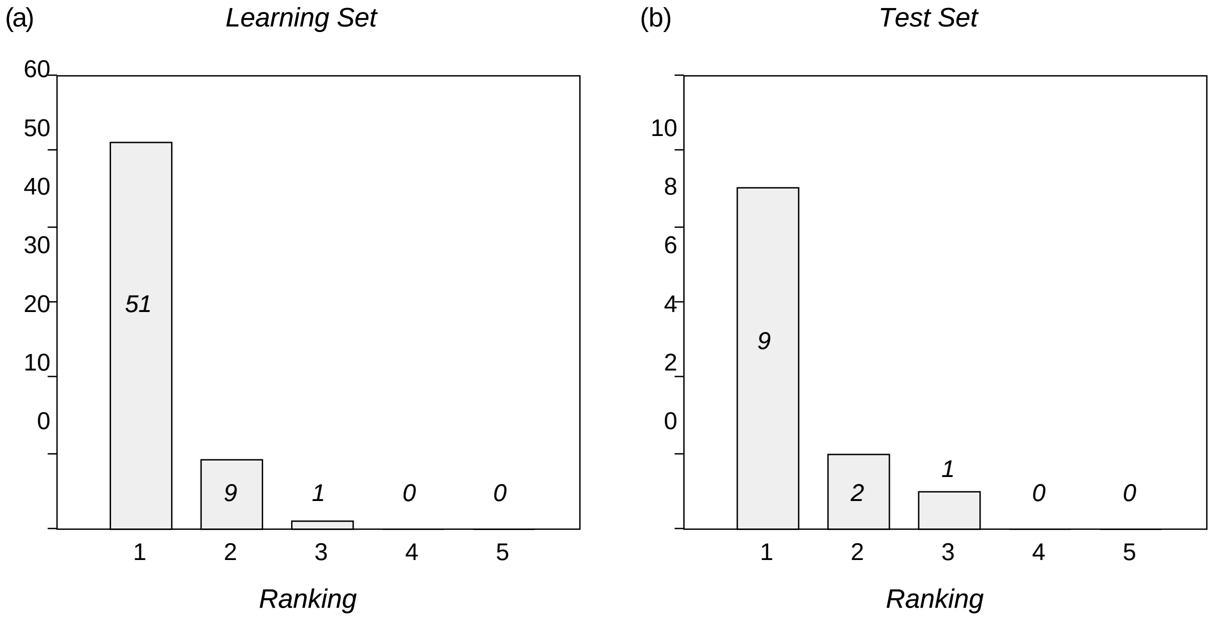

Figure 9: Classification results of learning set versus test set.

The left bar chart of Fig. 9 shows the results of classifying the 61 letters of the learning set using inverse document frequency as the weighting function. 51 letters were classified correctly, i.e., INFOCLAS yields the right message at the first position of the message type hypotheses list (cf. Fig. 8). Concerning 9 letters, classification was nearly correct, just when letters include more than one single message; classification fails for one letter more or less.

Real test data consisted of 12 DFKI random letters, analyzed by a prototype of our document analysis system and also transferred into the INFOCLAS internal representation. Surprisingly for us, classification results hardly degrade, 9 letters were well categorized and 2 nearly correct. However, we agree that the test set should be enhanced in the near future.

We have also compared run time measurements of the distinct weighting functions and how they influence classification results. However, differences in the quality of classification were negligible in contrast to run time. Fig. 10 and 11 show that computation of the term discrimination value is rather expensive in contrast to the other weighting functions and correlates with the size of the letter database directly.

6 Conclusion and Future Work

To summarize, we presented a system called INFOCLAS for the indexing, focusing and the classification of German business letters into different message types. The system primarily applies statistical methods of information retrieval, but also employs additional knowledge sources, such as word frequency statistics for German, message type specific words, morphological knowledge, and knowledge about the document structure (logical structure). As input, INFOCLAS takes (ASCII-) words either correctly or incorrectly recognized by our document analysis system.

Our future work concentrates on several topics:

Word alternatives. Currently, INFOCLAS deals with word alternatives of recognition (i.e., word ambiguities) in a best-first manner. An advanced approach could also take into account word alternatives, perhaps using recognition scores for index term weighting.

Elimination of word alternatives. The classification results of INFOCLAS can be used to prefer word candidates which have a lower credibility with respect to recognition. For instance, in a business letter of a bus travel agency the word “seats” in the context of ordering a bus is much more common than the words “beats” or “seals”.

Local and global similarities. The context of a word can be considered for the recognition of content similarities between individual text units, e.g., sections, paragraphs, sentences, and phrases. Classification is therefore not restricted to the global similarity between texts as reflected by corresponding index terms, but also uses the local environments in which the individual words are placed [4, 26]. In addition, the automatic extraction of links between letters is possible.

Parsing techniques. The indexing and focusing results will be used as starting-point for a flexible and robust (feature-based) island parsing which can deal with incomplete recognition results and lexical word dependencies (e.g., verb and noun valency using a dependency grammar) in order to solve ambiguities and eliminating irrelevant word hypotheses.

Skimming techniques. We also concentrate on skimming techniques such as those implemented in the FRUMP system [7], CODER [8], SCISOR [27], TCS [9], FERRET [10], etc. These systems accurately extract certain conceptual information from texts in preselected topic areas (e.g., news stories). Even the FRUMP system proved that an expectation-driven strategy was useful for interpreting texts in constrained domains. We belief that our domain of business letters and a corresponding message type model will allow similar skimming techniques for natural language processing (NLP). In particular, our message types are comparable with the sketchy script idea presented in FRUMP. Note that these NLP tools are yet not implemented in our system.

Acknowledgements

We would like to thank Stefan Dittrich who has implemented the main parts of the INFOCLAS system.

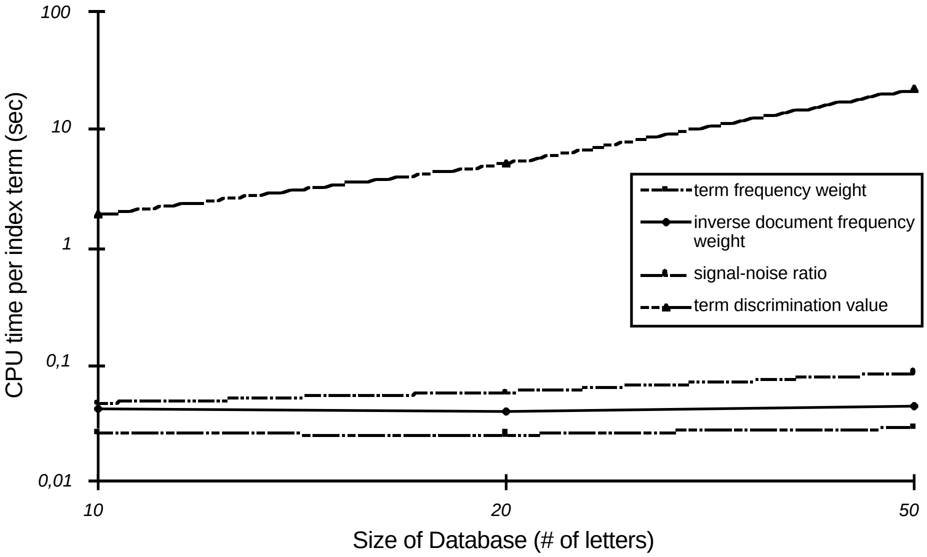

Figure 10: Run time measurements for index term weighting.

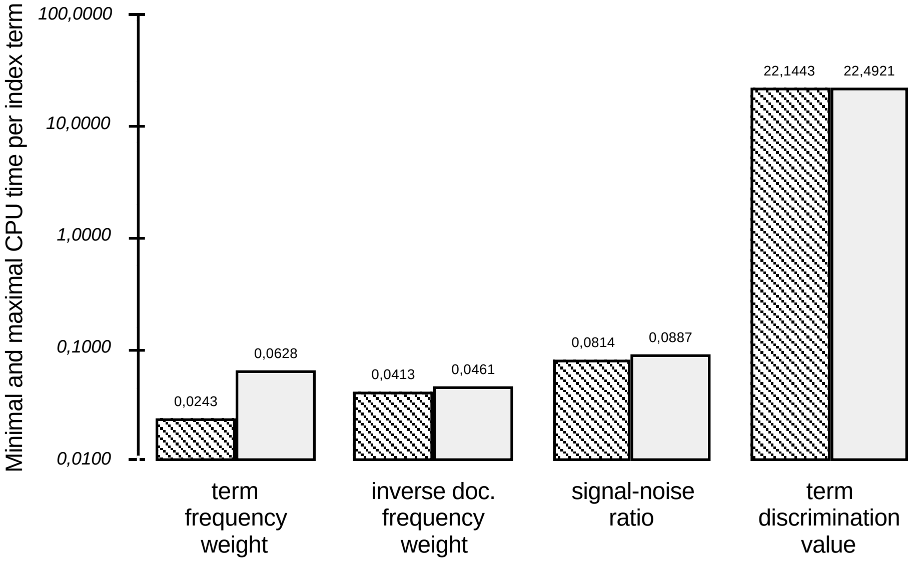

Figure 11: Minimal and maximal CPU time per index term (database = 50).

References

[1] C. Faloutsos, “Access Methods for Text,” Computing Surveys, vol. 17, no. 1 (1985), pp. 49-74.

[2] G. Salton and M. J. McGill, “*Introduction to Modern Information Retrieval,”* McGraw-Hill, Inc. (1983).

[3] G. Salton, “*Automatic Text Processing,”* Addison Wesley, Reading, MA (1989).

[4] G. Salton, J. Allan, and C. Buckley, “Automatic Determination of Content Relationships in Natural-Language Texts,” *Proc. Electronic Publishing ‘92*, Lausanne, Cambridge University Press (1992), pp. 183-198.

[5] H. Eirund and K. Kreplin, “Knowledge Based Document Classification Supporting Integrated Document Handling,” *Proc. COIS*, Palo Alto, CA (1989), pp. 189-196.

[6] N. M. Mattos and A. Dengel, “The Role of Abstraction Concepts and Their Built-In Reasoning in Document Representation and Structuring,” *Proc. ISAI-91*, Cancún, Mexico (1991), pp. 136-142.

[7] G. DeJong, “An Overview of the FRUMP System,” in: W. G. Lehnert, M. H. Ringle (eds.), *Strategies for Natural Language Processing*, Lawrence Erlbaum Assoc., Hillsdale (1982), pp. 149-175.

[8] E. A. Fox, “Development of the CODER system: A testbed for artificial intelligence methods in information retrieval”, *Information Processing & Management*, vol. 23, no. 4 (1987), pp. 341-366.

[9] P. J. Hayes, P. M. Andersen, I. B. Nirenburg, and L. M. Schmandt, “TCS: A Shell for Content-Based Text Categorization,” *Proc. 6th Conference on AI Applications*, Santa Barbara, CA (1990), pp. 320-326.

[10] M. L. Mauldin, “Retrieval Performance in FERRET—A Conceptual Information Retrieval System,” SIGIR Forum, Special issue of *14th Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval* (1991), pp. 347-355.

[11] T. Bayer, J. Franke, U. Kressel, E. Mandler, M. Oberländer, and J. Schürmann, “Towards the Understanding of Printed Documents,” in: H. Baird, H. Bunke, K. Yamamoto (eds.), *Structured Document Image Analysis*, Springer-Verlag (1992), pp. 3-35.

[12] G. P. Michalski, “The World of Documents,” *BYTE*, vol. 16, no. 4 (1991), pp. 159-170.

[13] R. G. Casey and D. F. Ferguson, “Intelligent Forms Processing,” *IBM Systems*

Journal, vol. 29, no. 3 (1990), pp. 421-435.

[14] A. Dengel and R. Hoch, “Intelligent Interfaces between Paper and Computer,” *Proc. Int‘l Symposium on Intelligent Workstations for Professionals,* Munich, Germany, March 1992 (in press).

[15] G. Nagy, S. Seth and M. Viswanathan, “A Prototype Document Analysis System for Technical Journals,” *IEEE Computer*, vol. 25, no. 7 (1992), pp. 10-24.

[16] G. A. Story, L. O‘Gorman, D. Fox, L. L. Schaper and H. V. Jagadish, “The RightPages: An Electronic Library for Alerting and Browsing,” *IEEE Computer,* vol. 25, no. 9 (1992).

[17] A. Dengel, R. Bleisinger, R. Hoch, F. Fein, F. Hönes, “From Paper to Office Document Standard Representation,” *IEEE Computer,* vol. 25, no. 7 (1992), pp. 63-67.

[18] A. Dengel, A. Pleyer and R. Hoch, “Fragmentary String Matching by Selective Access to Hybrid Tries,” *Proc. Int‘l Conference on Pattern Recognition ICPR92,* The Hague, The Netherlands (1992), pp. 149-153.

[19] A. Dengel, R. Bleisinger, R. Hoch, F. Fein, F. Hönes, and M. Malburg, “PODA: The Paper Interface to ODA,” *DFKI Research Report RR-92-02,* Kaiserslautern (1992), 53 pages.

[20] ISO 8613 Information Processing, Text and Office Systems, *Office Document Architecture and Interchange Format (ODA/ODIF)*, parts 1-8 (1988).

[21] W. Finkler and G. Neumann, “MORPHIX—A Fast Realization of a Classification-Based Approach to Morphology,” *Proc. 4. Österreichische Artificial-Intelligence-Tagung*; Springer-Verlag, Berlin, (1988), pp. 11-19.

[22] M. D. Harris, “*Introduction to Natural Language Processing,”* Reston Publishing Company Inc., Reston, Virginia (1985).

[23] P. Willett, “An Algorithm for the Calculation of Exact Term Discrimination Values,” *Information Processing & Management,* vol. 21, no. 3 (1985), pp. 225-232.

[24] ISO 9735 *Electronic date interchange for administration, commerce and transport (EDIFACT),* application level syntax rules (1988).

[25] W. B. Croft, “Retrieval from large text databases,” *Proc. Symposium on Document Analysis and Information Retrieval, Las Vegas,* Nevada, USA (1992), pp. 96-101.

[26] G. Salton and C. Buckley, “Global Text Matching for Information Retrieval,” *Science,* vol. 253 (1991), pp. 1012-1015.

[27] L. F. Rau and P. S. Jacobs, “Integrating top-down and bottom-up strategies in a text processing system,” *Proc. Second Conference on Applied NLP,* Austin, Texas (1988), pp. 129-135.