IJDAR (2007) 10:199–209
DOI 10.1007/s10032-007-0060-2

ORIGINAL PAPER

Genre as noise: noise in genre

Andrea Stubbe · Christoph Ringlstetter ·
Klaus U. Schulz

Received: 20 March 2007 / Revised: 16 July 2007 / Accepted: 27 August 2007 / Published online: 30 November 2007

Abstract Given a specific information need, documents of the wrong genre can be considered as noise. From this perspective, genre classification helps to separate relevant documents from noise. Orthographic errors represent a second, finer notion of noise. Since specific genres often include documents with many errors, an interesting question is whether this "micro-noise" can help to classify genre. In this paper we consider both problems. After introducing a comprehensive hierarchy of genres, we present an intuitive method to build specialized and distinctive classifiers that also work for very small training corpora. Special emphasis is given to the selection of intelligent high-level features. We then investigate the correlation between genre and micro noise. Using special error dictionaries, we estimate the typical error rates for each genre. Finally, we test if the error rate of a document represents a useful feature for genre classification.

Keywords Genre hierarchies · Features · Genre classification · Error dictionaries · Noisy corpora

1 Introduction

The technical term "genre" refers to the partition of documents into distinct classes of texts with similar function and form. When analyzing documents, genre represents an independent dimension, ideally orthogonal to topic. Traditionally, most of the work in the area of text classification has concentrated on the problem of how to recognize thematic domains. However, since the genre of a document often gives strong hints on its value for a given user, genre classification also helps to distinguish between "noise" and "music,"—between wanted and unwanted documents.

In the context of documents and genres, the technical term "noise" has two possible meanings. In a narrower sense, it refers to data contaminated, for example by spelling/typing errors or by errors resulting from OCR recognition. In a wider sense, depending on the task at hand, each genre can represent a class of noisy documents. For example, cooking recipes and forums on fishing represent a kind of "macro-noise" if someone collects scientific articles on fish. Obviously, classifying genre helps to recognize "macro-noise". Observing web pages of certain genres, with an eye-catching number of orthographic errors, for example forums, it is a natural question whether this "micro-noise" can help to classify genre. Our main contributions are the following:

1. We introduce a fine-grained hierarchy of genres with maximal coverage, including web-specific genres.
2. We present a collection of hand-crafted high-level textual features for the hierarchy. On this basis, we designed classifiers for each genre that only use a selection of few relevant features. The resulting system of classifiers is compared with statistical methods from machine learning.

A. Stubbe · K. U. Schulz
CIS, University of Munich, Oettingenstr 67,
80538 Munich, Germany
e-mail: andrea@cis.uni-muenchen.de

K. U. Schulz
e-mail: schulz@cis.uni-muenchen.de

C. Ringlstetter (✉)
AICML, Department of Computing Science, University of Alberta,
Edmonton, Canada T6G 2E8
e-mail: kristof@cs.ualberta.ca

3. We present a detailed evaluation of the distribution of error rates for orthographic errors found in distinct genres.
4. We show that for a number of genres an automated analysis of the error rate of a given document can be used as an additional feature to improve the classification.

Our genre hierarchy extends previous work by [2,4]. We tried to reach maximal completeness with regard to general search applications, at the same time avoiding fuzzy and overlapping genre classes. With the use of two levels and 32 leaf categories in the genre hierarchy we want to guarantee sufficient granularity for practical applications, simultaneously offering the possibility to return to a coarser scheme where this is preferable. Our main application scenarios that motivated the construction of the hierarchy were genre-qualified search and genre-specific corpus collection.

Our work on features and classifiers is motivated by the practical experience that standard classifiers based on learning (e.g., support vector machines [7]) do not lead to satisfactory results if only a small amount of training data is available. In our test, a total of 1,280 files in the complete corpus is composed of 40 documents available for each genre. When using 20 documents for training of a genre, standard classifiers and uniform feature sets produced poor results. We were then interested to see if a heuristic classifier based on a small set of features motivated by class-specific knowledge would lead to better results. Considerable effort was put into the selection of powerful features. As another refinement, several methods for combining the classifiers for distinct genres have been tested. For the given scenario, our classifiers in fact outperform standard methods from machine learning.[1]

We illuminate the correlation between the genre of a document and the percentage of orthographic errors found in the texts focusing on *spelling* and *typing* errors. For detecting this "micro noise", we use huge special error dictionaries that capture the main part of errors introduced by the respective noisy channels. In fact, the results show a strong correlation between genre and the number of orthographic errors, with a significant trend towards higher error rates in documents that belong to the more private oriented genres. As one application, genres and documents with high error rates can be excluded from corpus construction.

Since some genres typically come with a particularly low or high error rate, it is natural to assume that the error rate of a given document can provide valuable hints on its genre. We use error dictionaries to derive additional classification features and integrate them into our classifiers. Our experiments show that in fact for some of the genres the precision of classifiers is improved when using the error rate as a new feature.

The paper is structured as follows. In Sect. 2, we describe our hierarchy of document genres and introduce the corpora used for our experiments. Section 3 addresses the extraction of genre-specific features and their contribution to the classifiers. In Sect. 4, we consider strategies for combining the individual classifiers into a decision network. Section 5 describes the construction and application of error dictionaries. In Sect. 6, we describe our experimental results. First we evaluate our genre classifiers over a test corpus of 640 annotated documents, comparing the new technique with traditional methods from machine learning. Two case studies exemplify how genre classification can be fruitfully used in different application scenarios. We then present an evaluation that characterizes the distribution of error rates for orthographic errors in distinct genres. In a final series of experiments we report on the effect of using error rates as an additional feature for genre classifiers. The conclusion summarizes the results and comments on the future work.

2 A hierarchy of genres

Starting from a preceding system [4], we developed a new, finer-grained hierarchy of genres, meeting the demands of genre focused corpus construction and in particular, the filtering of noise from a macro perspective. The 11 classes proposed by Dewe et al., were rearranged to eight container classes. We split up the class *other running text* into the literature genres (B), mail (F.1), and diverse genres for knowledge communication (C); *interactive web pages* together with *discussions* and *letters* were assigned to the container class communication (F); *private* and *public homepages* were merged into presentation (C.7); *error messages, empty pages,* and *frame sets* were put into class "nothing" (G.1). Concerning the second level of the hierarchy, several new genres below the container classes are meant to increase the coverage of the classification. Because of their functional similarity the journalistic genres were additionally scrutinized by an expert leading to minor rearrangements. The final hierarchy is presented in Table 1.

To better judge its quality and transparency, the hierarchy was evaluated by a non-expert not involved into the construction process. The test person had the task to classify two documents of each genre according to the classification schema. For 76.8% of the texts the human choose the originally tagged class. 2.9% of the classifications were also correct, since the documents were mixed documents for example a presentation that contains a lot of programming code.[2] For 13.1%

[1] This should not be interpreted as a general claim - typically classifiers from machine learning are trained with at least hundreds of documents.

[2] During the collection of the corpus we tried to include only unequivocal documents. Nevertheless, estimated 22% of the documents contain material that belongs to a second genre.

Table 1 A hierarchy of genres

A. Journalism	**C. Information**	D.3 protocol
A.1 commentary	C.1 science report	**E. Directory**
A.2 review	C.2 explanation	E.1 person
A.3 portrait	C.3 recipe	E.2 catalog
A.4 marginal note	C.4 faq	E.3 resource
A.5 interview	C.5 lexicon, word list	E.4 timeline
A.6 news	C.6 biling. dictionary	**F. Communication**
A.7 feature	C.7 presentation	F.1 mail,talk
A.8 reportage	C.8 statistics	F.2 forum, guestbook
B. Literature	C.9 code	F.3 blog
B.1 poem	**D. Documentation**	F.4 formular
B.2 prosa	D.1 law	**G. Nothing**
B.3 drama	D.2 official report	G.1 nothing

the class selected by the test person joined the same container class and was very similar as, for example, a reportage and a feature document. Only 7.2% of the texts were classified in a completely wrong manner. The following confusions were observed: F.4 → G, C.7 → G, A.7 → A.8, A.4 → F.1 and E.2 → G.[3]

The containers of the hierarchy define a first classification level usable for coarse corpus partition. With regard to classification errors, a hierarchical classification schema helps to evaluate the severity of a false classification: depending on the application, errors that happen within a container can be defined to cause lower loss of accuracy than those that cross the top-level classes.

For each of the 32 genres, 20 English HTML web documents for training and 20 documents for testing were collected, leading to a corpus with 1,280 files.[4]

We were careful to gather a broad distribution of topics, authors, and sources for each genre in order to avoid a bias towards specific values of these dimensions. The balanced data set for training and for testing allows us to compare the performance of the different classifiers and abstracts from the specifics of document spaces for different applications with their individual distribution of genres.

3 Genre specific classifiers

As we argued above, genre classification helps to recognize unwanted documents. A kernel issue behind document classification is the selection of features. While [4] and others use global feature sets, we decided to use specialized features for each genre. The goal was to allow only a small set of significant and natural features for each single classifier. Since training corpora were small, we used human knowledge on the given genre and tried to avoid effects caused by accidental similarities between documents of distinct genres that result in overfitting. In an iterative process, we investigated all training documents for the given genre, identifying important characteristics and sometimes defining clues.[5] We evaluated these features for all classes and tried to separate the training files of the chosen genre from the other files by determining thresholds that maximized the F1-value for those features and their combinations. These intuitive hypotheses (e.g., catalogs indeed contain a lot of prices) were tested on the complete training collection. For classification, features were arranged into a simple decision tree. If the use of a certain feature led to a performance improvement, it was added, otherwise it was discarded. During this process, when a previously acknowledged feature became degraded it was removed. For practical reasons the iteration was terminated when the classifier reached values for recall and precision of about 90% on the training corpus. For some genres which are exceedingly difficult to identify, a threshold for precision of 75% was set.[6] The final result of this procedure is a form of hand-crafted decision tree for each genre.

Many different kinds of features were considered including form, vocabulary and parts of speech, complex patterns, and combinations of all these. *Form* features can be further divided into statistical clues such as average line length or number of sentences, document structure, formatting of the text and HTML meta-information such as content-to-code-ratio. *Vocabulary* includes specialized word lists as well as dictionaries, for example positive adjectives or the 200,000 most common English words. Also multi word lexemes, bigrams, signs (emoticons) or phrases (such as “to whom it may concern” in letters) were applied. *Patterns* include more complex units such as repetitions of characters, dates or bibliographic references. *Combinations* of these features result in high-level structures. For example a casual style of writing can be recognized by the number of contractions (e.g., “won’t”) and the use of vague, informal and generalizing words. The occurrence of some kind of agents can be recognized through quotation marks (as only agents can speak), pronouns, names and living entities. Sometimes it was necessary to distinguish different styles of writing or structure within genres. Commentaries, for example, can either be polemic pamphlets or could show the pros and cons of a topic. In these cases, we had to construct rules of the form *feature-set-1* ∨

[3] These human errors had only marginal coherency with those induced by automatic genre recognition (s. b.).

[4] For research purposes the corpus is available at http://www.cis.uni-muenchen.de/~andrea/genre/corpus.

[5] Examples for clues are specific form features for FAQs, interviews and poems.

[6] This lower threshold concerned the genres commentary (A.1), portrait (A.3), marginal note (A.4), explanation (C.2), presentation (C.7) and mail (F.1).

feature-set-2. To avoid misclassification, special features that help to separate between similar genres were used.

The classifiers were then constructed as a conjunction of single rules. As an example the classifier of the genre *reportage* is defined by the following conjunction.[7]

textlength, HTML-form-elements
$number\_of\_chars > 2500 \wedge number\_of\_chars < 45000 \wedge$
$HTML\_form\_elements < 10$
is a continuous text
$number\_of\_verbs > 18 \wedge number\_of\_conjunctions > 2$
not too dispassionate, literary or casual language
$number\_of\_sentiment\_bearing\_adjectives > 17 \wedge$
$sentiment\_adjectives/adjectives > 0.5 \wedge$
$sentiment\_adjectives/adjectives < 4 \wedge$
$contractions < 2.5 \wedge casual\_language < 3$
filter commentaries, faq, interview
$arguing\_language < 1.3 \wedge generalizing\_language < 3.8 \wedge$
$questionmarks < 3$
filter scientific reports and portraits
$science\_bigrams < 0.01 \wedge (portraitWords < 1 \vee$
$names + 3^{rd}\_person\_pronouns < 7)$
not too many date-expressions or past-markers
$date\_expressions < 0.6 \wedge past\_markers < 1$
1^{st}_person**, not too many (but at least some) names**
$1^{st}\_person\_pronouns > 1.6 \wedge 3^{rd}\_person\_pronouns < 8 \wedge$
$names > 0.5 \wedge names < 6.5$
consequently past or present tense
$verbs\_in\_past\_tense > verbs\_in\_present\_tense \wedge$
$(verbs\_in\_past\_tense > 0.2 \vee verbs\_in\_present\_tense > 0.2)$
about people and creatures or past adventures and voyages
$(3^{rd}\_person\_pronouns > 3 \vee names > 4 \vee$
$living\_entities > 2) \vee$
$(geographical\_names > 0.5 \wedge past\_markers > 0.4)$

Difficulties. The limits of the described method are reached for text documents that neither possess specific structure nor specific vocabulary. Such texts often can only be recognized by POS-characteristics or by the kind of language used. Still, the stylistic differences between two authors can be more severe than those between two genres. Another problem is that certain genres have strong similarities. Examples are commentaries and marginal notes, which both express the opinion of an author in a somewhat casual manner.

4 Classifier combination

Endowed with specialized classifiers for each genre, we had to fix their interplay and their global behavior. An evaluation of 10% of our training corpus (two files per class) showed that 22% of the documents show aspects of more than one genre. Therefore, depending on the application it could be better to allow multiple classification. On the other hand, sometimes it might be desirable to have an unequivocal classification, and thus, a decision on the most probable class has to be made.

4.1 Multiple classification

The default case of multiple classification is an independent application of all classifiers to an input document. Since each classifier can make a positive decision, a document can end up in more than one class.

Filtering. A variant of multiple classification that exploits knowledge about the interdependencies of the classifiers is filtering. To remove erroneously classified texts of a certain genre from another class, filters can be used. These filters improve the precision of an individual classifier, restricting the set of hits. The filter rules operate as a disqualification criterion: if a text has been recognized as A, it cannot be simultaneously classified as B. This approach is highly efficient if A texts are often erroneously assigned to B, but conversely only a few B texts are recognized as A. In order to find appropriate rules, one may compute a confusion matrix on the training data. All classes that are only misclassified in an unidirectional way are suitable for filtering.

4.2 Mono classification

One solution is to compute the results for each single classifier and apply well-known techniques such as the behavior knowledge space(BKS) method, to determine the best class [6]. Instead of computing in advance all the classifications and then filtering the results afterwards, a more efficient alternative is to determine an evaluation sequence a text has to go through. As soon as a text is classified, the process stops. This procedure prevents multi-classifications, but a poor ordering of the applied classifiers can lead to deterioration of precision and recall. For example, if the classifier for class A leads to wrong classifications of documents that belong to class B, its use before the classifier of class B will lead to lower precision for the classifier of class A and to lower recall for the classifier of class B.

Ordering by F1 value. A possible solution to set up a reasonable unequivocal classification is to use the classifiers in order of their F1 values reached at the training set.[8] The underlying idea is that a higher F1 value indicates a higher probability that the classifier will make the correct decision.

Ordering by dependencies and recall. Ordering only by the F1 values misses the possible advantages that result from a reordering triggered by the local dependencies between the classifiers. To determine an improved ordering, a

[7] Explanations of all features and prototypical implementations of the classifiers for the different genres are available at http://www.cis.uni-muenchen.de/~andrea/genre/.

[8] $F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$.

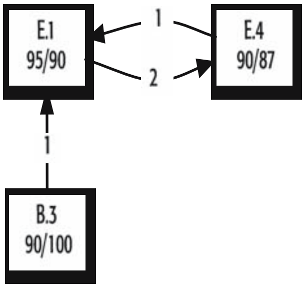

Fig. 1 Cut-out of the dependency-graph

dependency graph is generated. In a first round each document of the training corpus is classified by all classifiers. By that we get for each classifier N_i a value for recall and precision. Additionally we get a confusion matrix for the classifiers.

A first version of the classification sequence is established by declining recall values with precision as a secondary ordering criterion:

$$N_i \succ N_j \text{ iff}$$
$$(Recall(N_i) > Recall(N_j)) \vee$$
$$(Recall(N_i) = Recall(N_j) \wedge Precision(N_i) > Precision(N_j))$$

Then with the help of the confusion matrix a dependency graph is generated. When finding texts of class N_j misclassified as class N_i, we create a directed edge from N_i to N_j, labelled by the number of missclassified texts. The final version of the classification sequence is received by rearranging classifiers in the sequence according to their dependencies with their successors: if a classifier N_i is followed by N_j and has a dependency edge with N_j, N_j is put before N_i. The procedure is not applied recursively: that means if rearranged the dependencies of N_j are not taken into account.

For the case of a cycle in the dependency graph, N_i is only rearranged if the label of the outgoing edge to N_j is higher than that of the incoming edge from N_j. An illustrating cut-out of the dependency graph is shown in Fig. 1. From our training corpus, the following sequence arose.

G.1→E.2→F.4→F.2→F.3→C.9→C.6→C.5→B.3→B.1→D.1→D.3→D.2→E.4→E.1→E.3→C.8→C.1→A.5→A.7→A.8→F.1

Compared to the F1 model, the ordering by dependencies lead to improvements an precision and recall on the training and on the test collection.[9] In our experiments (cf. Sect. 6), we used mono-classification, ordered by dependencies and multiple classification with filtering.

5 Finding errors with error dictionaries

Our method to investigate the correlation between genre and orthographic errors is based on error dictionaries [1,13]. Assuming that errors in texts result from a structured and elucidable process, it is possible to generate and store errors in a systematic way, applying a generative algorithm to a language base. In [13] huge error dictionaries including typing errors, spelling errors and OCR-errors have been employed to estimate the number of orthographic errors in web documents. These dictionaries were found to capture most of the orthographic errors found in the web. In our present study, OCR-errors did not play any role. Hence, regarding the correlation between genre and noise we concentrated on the error channels *typing* and *wrong cognitive representation*.

Typing errors. Ignoring less important classes, typing errors can be divided into transpositions, deletions, insertions, and substitutions [10]. While transpositions and deletions may affect arbitrary symbols, insertions occur when pressing two keys together instead of one. Therefore, any inserted letter is neighbored on the keyboard to one of its adjoint letters in the text. Similarly substitutions only affect two symbols neighbored on the keyboard. Taking these restrictions into account, we created a rule set for producing typing errors from correct words. These rules were applied to a conventional English dictionary with 100,000 entries (high-frequency words of a larger dictionary). We did not simultaneously apply two rules to a correct word; thus, the tokens that are produced contain exactly one error. We never modified the first letter of a given word. On average, 135 mutations were produced per English input token.

Obviously, when applying an error pattern, a correct word may be produced and thus an additional filtering step is required (s.b.). After deletion of duplicates and correct words (filtering), the dictionary of typing errors $D_{\text{err}}(\textit{English,typing})$ contains $9,427,051$ entries.

Cognitive errors. We define cognitive errors as orthographic errors that result from an incorrect cognitive representation, caused, for example, by a disagreement between phonetic and orthographic form of a word. To find characteristic patterns for such errors, a bootstrapping method was used. Starting from a small set of prominent errors, we collected error prone documents from the web. From these documents, new high-frequent errors were extracted. The bootstrapping was terminated when no new errors with a reasonable frequency were found. From the list of errors, we derived patterns for cognitive errors and built up a production program for cognitive error dictionaries. Applying the error rules to a standard English dictionary $D(\textit{English})$ with $315,300$ entries, we obtained a list containing $1,223,128$ garbled tokens. After the standard filtering step (s.b.), the dictionary for cognitive spelling errors, $D_{\text{err}}(\textit{English,spell})$, is composed of $1,202,997$ entries.

[9] In general an improvement by this method can be expected if only the direct successors are switched according to their dependencies.

Table 2 Size of filter dictionaries

Dictionary	Number of entries
$D(English)$	315,300
$D(German)$	2,235,136
$D(French)$	85,895
$D(Spanish)$	69,634
$D(Geos)$	195,700
$D(Names)$	372,628
$D(Abbreviations)$	2,375

Filtering step. The filtering procedure needs as input an unfiltered error dictionary and in addition a filtering dictionary $\mathcal{D}^{\text{Filter}}$. For our experiments, $\mathcal{D}^{\text{Filter}}$ represents the union of diverse conventional dictionaries presented in Table 2. Each garbled token found in $\mathcal{D}^{\text{Filter}}$ is excluded from the respective error dictionary. Note that the classification of a token as an error is always related to the applied filter lexicon. This can have profound effects on the values of precision and recall. For example, when analyzing the orthography of multilingual documents, the overgeneration of an error dictionary can be reduced drastically by adding a missing lexicon of one of the involved languages to the filter procedure.

Detecting and counting errors with error dictionaries. In a study (cf. [13]) on 1,000 real errors and on 4,000 tokens recognized by the error dictionaries, we found a recall of 62.4% and a precision of 85% for error recognition using our error dictionaries. For the worst documents (>10 errors per 1,000 tokens), recall (66.93%) and precision (95.00%) turned out to be higher. The results show that the number of hits of the error dictionary could be seen as a lower approximation of the real number of errors. For English texts, the ratio between both numbers is ~ 1.4 (cf. [13]). The approximation is more reliable for "bad" documents with a large number of hits. In what follows, the *error rate* of a text is defined as the average number of hits of the error dictionary in 1,000 tokens of the text.

6 Experimental results

In this section, we present the experimental results. In the first part we evaluate the behaviour of our specialized classifiers for genre classification, comparing them with conventional statistical classifiers. In the second subsection, we study the correlation between genre and the percentage of orthographic errors found in the texts. Finally, we describe experiments of using error rates as an additional feature for genre classification.

6.1 Genre classification by a combination of specialized classifiers

In our experiments for genre classification initially we applied the unequivocal classification model (cf. Sect. 4.2). Each document of an evaluation corpus is treated by specialized binary classifiers until a classifier makes a positive decision. The application sequence of the classifiers is controlled by a dependency graph.

Precision and recall are used as evaluation measures for genre classification. An evaluation corpus $D = \{D_1, ..., D_T\}$ is a multi set of documents of T different genres, with $D_i = \{d_{i1}, ..., d_{iN}\}$ as the document set of genre i; recall and precision of a classifier for genre i, according to the set of recognized documents C_i, are defined as follows:

Precision is the number of documents of genre i in the result set C_i divided by the total number of documents in the result set. $Precision = \frac{|C_i \cap D_i|}{|C_i|}$.

Recall is the fraction of documents of genre i in the result set C_i divided by the total number of documents of genre i in the evaluation corpus. $Recall = \frac{|C_i \cap D_i|}{|D_i|}$.

In Table 3, we show a survey of the classification results using the genre-specialized classifiers combined by the dependency graph. The precision of the classification into original classes is 72.2% with an overall recall of 54.0%.[10] The quality of classification differs considerably between certain classes, ranging from an F1 value of 14.7% for marginal notes (A.4) to 100% for "nothing" (G.1). Genres with a definite structural appearance such as directories, poems, FAQ and forums involve certain form features and because of that are better recognized than average. If we consider documents as correctly classified that do not end up in their original class, but in a class that is also *well-justified* in the sense of a multi-classification (cf. Sect. 4.1) the precision rises to 80.5%. We regard a document as *well-justified* to suit for a class if it either is a mixture of genre (like a presentation in form of a timeline) or contains a certain amount of material that belongs to a different genre: for example, a scientific report with a great part of statistical information that has been classified to statistics or a presentation with a great amount of programming code. Reducing the hierarchy to the more coarse-grained first level, we obtain a precision of 77.8%, showing clearly the effect of improvements in classification when using fewer genres.

An analysis of the confusion matrix shows a high quantity of minor classification errors where true class and classification result are close neighbors. For example, marginal notes are confused with features (4) or commentary (6)—all of them fall into the journalism container and express somehow the view of the author. An excerpt of the confusion matrix presented in Table 4 shows frequently confused genres that lead to more serious classification errors. The given examples of confusion errors show the direction to possible improvements of the classifiers by either maintaining separative ranges of feature values or introducing additional separative features.

[10] With original class we denote the class that the document was sorted into during corpus construction.

Table 3 Precision (P) and recall (R) of genre classification using specialized classifiers

Genre	P	R
A. Journal.	57.0	38.1
A.1 comm.	50.0	30.0
A.2 review	72.7	40.0
A.3 portr.	76.9	50.0
A.4 m.not.	14.3	5.0
A.5 interv.	81.3	65.0
A.6 news	40.0	30.0
A.7 feat.	53.8	35.0
A.8 repo.	50.0	50.0
B. Lit.	78.0	53.3
B.1 poem	85.7	60.0
B.2 prosa	66.7	60.0
B.3 drama	88.9	40.0
C. Info.	74.0	55.3
C.1 sci.rp.	88.9	40.0
C.2 explan.	50.0	35.0
C.3 recipe	81.3	65.0
C.4 faq	86.7	65.0
C.5 lexicon	70.0	70.0
C.6 bil.dic.	88.9	40.0
C.7 presen.	30.0	35.3
C.8 stat.	80.0	40.0
C.9 code	100	85.0
D. Docu.	77.5	51.7
D.1 law	83.3	50.0
D.2 off.rp.	61.5	40.0
D.3 prot.	86.7	65.0
E. Dir.	76.1	63.8
E.1 pers.	90.9	50.0
E.2 catal.	94.4	85.0
E.3 res.	82.4	70.0
E.4 timel.	47.6	50.0
F. Comm.	73.9	63.8
F.1 mail,talk	40.0	20.0
F.2 for.,gueb.	64.0	80.0
F.3 blog	92.9	65.0
F.4 formular	90.0	90.0
G. Noth.	100	100
G.1 noth.	100	100

Ordering of classifier application by a dependency graph. Results for classification of texts from the test corpus into original class

Trends on bigger samples. During an application experiment (cf. bellow) and a classifier adaption experiment [16] trends for precision and recall on bigger samples were investigated. These data are to be seen as a complement to the given results since the used corpora are not carefully balanced either with regard to unbiased sources (recall) or to the genre distribution of the corpus (precision). On a 160-document set for each of 4 selected genres we got the following recall values: blog(57.50), catalog(40.00), faq(52.50), interview (55.00).[11] With regard to precision on a corpus of 30,000 webpages measuring on random test samples of 50 documents we got the following values: blog(64.00%), forum (72.00%), interview(56.00%).

Table 4 Excerpt of the confusion matrix showing more serious classification errors and their explanation

Genre	Class	Freq	Remark
A.5	B.3	2	similar structure
A.4	F.1	4	personal style, freq. use of I, you
A.5	A.4	4	no simple explanation
A.5	F.1	5	welcome and goodbye
B.1	F.1	5	no simple explanation
B.3	A.5	1	similar structure
C.1	A.5	4	scientific texts with marginal notes
C.9	C.6	4	code words recogn. as foreign words
D.1	C.2	4	no simple explanation
F.2	E.4	5	series of dates
F.3	B.2	4	some blogs have narrative style
F.3	E.4	4	series of dates
F.3	F.1	8	personal style, freq. use of I, you

Comparison with Machine Learning Methods. For the sake of comparison, several machine-learning (ML) methods have been applied to the data, using as a global feature set the union of all feature sets introduced for the specialized classifiers.[12] The first ML method is the *Naive Bayes Classifier* using the maximum likelihood expectation criterion to make a decision. The term "naive" refers to the assumption of statistical independence of features, which leads to a simple multiplication of probabilities obtained for the single features. The second method is the *decision tree J.48*, a variant of C4.5, that turns the feature combination into a series of if-then-tests [12]. With the *k-nearest-neighbor* algorithm (KNN), an object is assigned to the nearest cluster in the feature space.[13] Finally, we applied *Support Vector Machines* (SVMs) [7], which divide the data into classes by a separating hyper plane.[14] The SVM was trained by the

[11] For the genres faq, blog, and catalog we used the corpus provided by [15]. The recall values could be improved to blog(72.50), catalog(52.50), faq(77.50, and interview(67.50) by a range adaption algorithm [16].

[12] All ML applications were realized with the help of the the WEKA implementations [19].

[13] We obtained the best results for $k = 1$.

[14] SVMs have been tested in a variant that employs the sequential minimal optimization algorithm that compares classes in pairs leading to a complexity of On^2. We used a linear kernel. Joachims has shown that for thematic text classification, SVMs outperform the other three methods [7]. This has been confirmed for genre classification in [3].

Table 5 Precision (P) and recall using specialized classifiers, support vector machines, Naive Bayes, J48-decision-tree and k-Nearest-neighbor algorithm (mono classification)

Method	Precision (%)	Recall (%)
Specialized classifiers	72.2	54.0
Support vector machines	51.9	47.8
Naive Bayes	48.3	44.8
J48 decision tree	40.4	37.5
k-Nearest neighbor	35.7	31.7

Table 6 Precision (P) and recall (R) of queries sent to a search engine to retrieve scientific documents on fish

Method	P_{raw} (%)	R_{raw} (%)	P_{gen} (%)	R_{gen} (%)	P_{perf} (%)
rank 5	26.0	14.33	34.0	19.5	66.0
rank 10	22.0	22.6	25.0	26.4	48.0
rank 15	22.7	40.0	24.7	44.1	38.7
rank 20	25.5	61.5	23.0	62.2	29.5
rank 30	19.7	100	19.7	100	19.7

Values for the original ranking (P, R_{raw}), the rearranged ranking by genre recognition (P, R_{genre}) and the perfect ranking (P_{perf})

WEKA implementation of John Platt's sequential minimal optimization algorithm [11]. Multi class problems are converted to a set of 1-vs-1 classifications (pairwise classification) and combined using pairwise coupling [5]. In comparison to statistical methods (cf. Table 5), our method is superior by 39% in precision and 13% in recall. This result, of course, depends on the small training corpus and we claim superiority only under this condition.[15] Still, for many classification tasks it is not realistic to annotate thousands of training documents. Here we consider the proposed method as a strong alternative.

Comparison with previous work on genre classification. Comparing our results to previously published work, the small size of our training corpora and the high number of possible classes should be emphasized. In [3], using a training corpus with 10,000 documents and only 7 genres, an F1 value of 89.1% is reached that sharply decreases with the reduction of training documents. In [18], a Bayes classifier is used to classify documents into nine classes of the Brown corpus. Recall of 57.8% and precision of 62.2% are reported. In [9] the influence of the number of genres on classification quality is documented with a decline from 73% precision using four different genres to 52% using 15 Brown categories.

In two application studies, we further tested the strength of our method to filter noise by classifying and excluding undesired genres.

Application scenario 1: Collecting scientific articles on fish. The first study deals with the improvement of the ranking of a search engine by genre classification. As an application scenario we assume a user who is interested in scientific articles on fish, which he hopes to extract from the Internet by sending queries like e.g., $cod \wedge habitat$ to a search engine. The evaluation runs over the 30 highest-ranked documents of each query. We used ten different queries. In Table 6, we present the macrovalues for recall and precision on the ranked document sets at cut points 5, 10, 15, 20 and the complete set of 30 documents.[16] We compare the findings of the search engine to the sets reranked by genre recognition. To mark the upper bound, we give values for precision as achieved with a perfect ranking. It turns out that both precision and recall are improved by the genre classification. On the other hand, as the perfect ranking shows, the improvements by far do not reach the upper bound. This gap is caused by the weak recall (40%) of our classifier for science documents.

Application scenario 2: supporting the construction of language models for speech recognition. In a second application experiment, we collected a corpus for the improvement of language models for speech recognition. A serious problem in this domain is that training corpora of spoken language are notoriously sparse. A widely used technique is to extend the spoken material by documents of written text, thus boosting the language models [14]. A shortfall of this method is that arbitrary written documents are collected, ignoring matters of language style. In our experiments, we collected documents of written text from genres where the use of language is similar to that found in spoken corpora. We approved forum/guestbooks, interviews and blogs using language similar to that in speech, and tried to exclude all other documents as noise. Sending 200 combined utterances (3 g) of the Verbmobil spoken language corpus [17] to a search engine, we collected ca. 30,000 web pages. From these, 1,631 were classified as forum/guestbook, 1,327 as interview, and 1,355 as blog. For each genre, a random sample of 50 answer documents was annotated by hand to estimate precision.

For forum/guestbook, we obtained a precision of 72%. With 6 blog documents in the sample, this increases to a value of 84% desired documents. By the term *secondary precision*, we denote the ratio of all desired documents in a sample divided by the sample size. For the interview class, we achieved 56% primary precision and, with 6 forum documents and 7 blog documents, a secondary precision of 82%. The blog genre comes with 64% primary precision containing

[15] For example Joachims [7] used 9,603 training documents, nearly 1,000 for each training class. Additionally, we did not tune the WEKA ML methods that are not especially designed for problems with many classes. Transductive SVMs that performed well with small training sets for topic classification [8] require many sparse but relevant features, a premise not given in our setting.

[16] $P_{\text{Macro}} = \frac{\sum_{i=1}^{N} P\text{Query}_i}{N}$, $R_{\text{Macro}} = \frac{\sum_{i=1}^{N} R\text{Query}_i}{N}$.

13 forum documents and 1 interview document leading to a secondary precision of 92%. Compared to the above results for our test collection, the genre classifiers on average show slightly lower precision, but taking desired genres into account (*interview, blog, forum*), the classifiers work remarkably well. If we approximate the recall for the three desired classes by the recall values obtained for the test collection of our genre corpus, we obtain a reduction of noise in absolute values of 24,000 excluded files or a residue of only 2.5%.

6.2 Correlation between genre and orthographic errors

Table 7 shows the mean rate of orthographic errors (*err*) for each of our 32 genres. As we argued earlier, the error rate represents a lower approximation for the real number of errors. In addition, values for the eight container classes are given. We find extraordinary high differences between the genres, and also that significant deviations within the container classes exist. Error rates reach from 0.23 for law to 6.89 for forum/guestbook. In the journalism class, the subclasses review and interview come with values $err > 2.0$. In the container literature, poems are exceptionally erroneous with $err > 5.0$. In the information class, the two lexica genres have higher error rates. For the documentation container class, the subclass law—with a mean error rate of only 0.23—is a candidate for classifier tuning by error rate. For the communication container class, the guestbook/forum subclass has an outstanding error rate. Somewhat surprizingly, the value for blog is nearly as high as the former. Evidently, for some of the blogs, no spellcheckers have been used (s. b.). These two classes also hold the highest rates over the whole classification. Naturally, the guestbook/forum genre is a candidate for improvement of genre classification by using the error rate of a document as an additional feature.

Since the error rate for blogs was very high, we collected another corpus of 200 blog documents using Google. We found that due to the page-ranking mechanism, we had now much more professional blogs in our selection. Here the mean error rate was 3.03 (standard deviation 2.26). This indicates that for some genres details of the corpus collection have significant influence on the kind of documents that are attracted.

In the right columns of Table 7 we show the mean error rate for 80% of the documents with the lowest error rate. This cut will help to eliminate the outliers with a high deviation of the error rate compared to the rest of the class. The relative order between the genres is not changed too drastically. In the "information" container, the FAQ genre moves to a more prominent position, which makes sense since FAQs are usually dynamic, technically oriented web pages, possibly not well maintained from an orthographic point of view.

Table 7 Mean error rates (err) and standard deviation (σ) for different genres in the training part of the genre corpus

Genre	All		Best 80%	
	err	σ	err	σ
A. Journalism	1.49	2.70	0.57	1.96
A.1 comment.	0.96	1.51	0.30	0.64
A.2 review	2.74	4.60	0.72	1.21
A.3 portrait	1.48	1.62	0.85	0.63
A.4 marg. nt.	1.04	1.29	0.55	0.77
A.5 interview	2.08	2.32	1.14	1.29
A.6 news	1.22	4.18	0.19	0.74
A.7 feature	0.99	1.26	0.47	0.69
A.8 reportage	1.18	2.33	0.29	0.48
B. Literat.	3.33	6.10	1.37	1.64
B.1 poem	5.17	8.90	1.73	2.32
B.2 prosa	2.51	3.78	1.24	1.07
B.3 drama	2.30	2.93	1.14	1.04
C. Informat.	2.29	4.11	0.74	1.09
C.1 science.rep.	0.79	0.88	0.49	0.49
C.2 explanation	1.77	1.58	0.83	0.91
C.3 recipe	2.10	2.09	1.24	1.22
C.4 faq	2.42	2.39	1.39	1.14
C.5 lexicon	3.26	4.54	1.21	1.62
C.6 biling. dict.	4.04	7.27	0.42	0.68
C.7 presentation	1.83	3.55	0.57	0.93
C.8 statistics	1.69	4.68	0.22	0.48
C.9 code	2.78	6.20	0.26	1.00
D. Document.	0.85	1.14	0.43	0.76
D.1 law	0.23	0.46	0.04	0.09
D.2 off. report	0.91	0.96	0.56	0.66
D.3 protocol	1.41	1.45	0.87	0.98
E. Directory	1.72	3.70	0.39	0.69
E.1 person	0.31	0.44	0.30	0.21
E.2 catalog	1.72	2.11	0.82	1.09
E.3 resource	1.94	5.47	0.18	0.32
E.4 timeline	1.34	3.23	0.21	0.41
F. Communic.	5.20	8.49	2.33	2.55
F.1 mail,talk	2.84	5.92	0.79	1.21
F.2 for., guestb.	6.89	7.90	3.68	3.55
F.3 blog	6.65	7.74	3.65	1.45
F.4 formular	4.44	10.94	1.20	1.60
G. Nothing	0.00	0.00	0.00	0.00
G.1 nothing	0.00	0.00	0.00	0.00

Columns 2 and 3 (4 and 5) refer to all (the 80% best) documents

Figure 2 shows the deviation of error rates between training and test corpora with remarkable stability for all corpora except "code" (C.9).[17]

[17] As we already knew from previous experiments the code genre is problematic in regards to the precision of the error dictionaries. If a

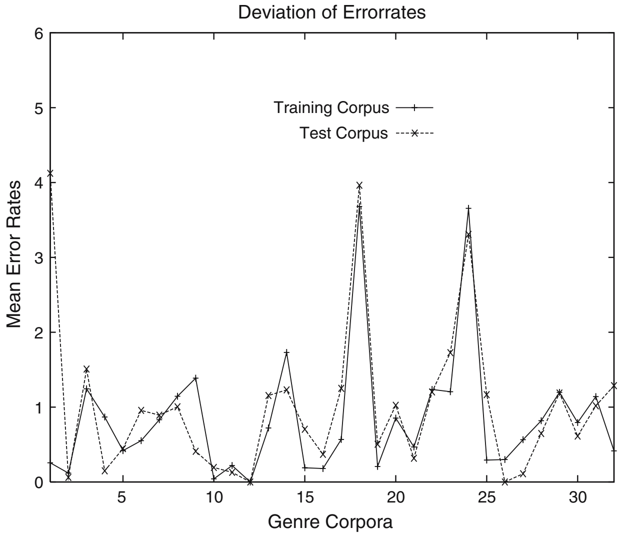

Fig. 2 Deviation of error rates for genres (best 80% documents) between training and test corpora

Topicality and Genre. Thinking of corpus collection for computer-aided language learning (CALL), it is important to know whether the mean error rate for documents of a given genre depends on the topics that are covered by the document. If, for example, the error rates of a highly error-prone genre are acceptable for more professional topics, its exclusion from a pedagogical corpus as noise only by genre is not justified. We conducted a prestudy for the genre *forum/guestbooks* on corpora that cover four distinct topics ranging from hobbies to science: fish, neurology, mushrooms, and holocaust. With a range of 4.11–6.89, the forum genre seems to have high mean error rates for all topics. The corresponding picture for all other genres remains to be studied.

6.3 Using error rate for classification

Observing a significant correlation between genre and mean error rate, we tried to exploit this for the improvement of classification. We used the error rate as an additional feature for our genre classifiers. Not surprisingly, an improvement was obtained only for some of the genres. For the genres features(A.7), persons(E.1), timeline(E.4), the precision of classifiers could be improved without any loss of recall, see Table 8. For three other genres that were tested (portrait(A.3), reportage(A.7), presentation(C.7)), classification results even became worse. A partial explanation is the high variance of error rates. For the statistical classifiers we obtained a similar picture. For example, SVM classification improved for class prosa (B.2) from 65.2 to 71.4% precision. But again for other classes a negative effect was obtained.

Table 8 Results for precision and recall when using the error rate as an additional feature

Genre	P^{Orig} (%)	R^{Orig} (%)	P^{Err} (%)	R^{Err} (%)
A.7 features	37	35	41	35
E.1 persons	80	60	86	60
E.4 timeline	36	13	46	13

Original classifiers versus new classifiers

7 Conclusion

In this paper we showed that genre classification can be successfully applied to compute meaningful partitionings of document repositories. As we indicated in two case studies, a division of documents into genre classes can help to better satisfy the needs of a user or support special corpus construction tasks. We introduced a new fine-grained hierarchy of genres which offers an adequate granularity for a wide range of applications. With the focus on hand-crafted high level features, a system of classifiers for the hierarchy was designed. We think that the manual and careful design of special features deserves much more attention in the literature on text classification and machine learning. Our specialized genre classifiers are extremely easy to implement and they work even for very small training corpora.

We also showed that a significant correlation exists between the genre of a document and its percentage of orthographic errors. Using this knowledge we could further improve the behavior of the classifiers for some genres by using the mean error rate as an additional feature.

In our future work we intend to deepen this picture. For genres where the error rate has a high variance it might be interesting to see if further subdivision into "professional-public" versus "non-professional-private" subgenres makes sense. We also intend to look at further application scenarios, from ranking of search results to focused corpus construction.

Acknowledgements We want to thank the reviewers of IJDAR. This work was supported by AICML, iCORE, and DFG.

References

1. Arning, A.: Fehlersuche in großen Datenmengen unter Verwendung der in den Daten vorhandenen Redundanz. Ph.D. thesis, University of Osnabrück (1995)
2. Crowston, K., Williams, M.: Reproduced and emergent genres of communication on the world-wide web. In: 30th Hawaii International Conference on System Sciences (HICSS) (6), pp. 30–39 (1997)
3. Dewdney, N., VanEss-Dykema, C., MacMillan, R.: The form is the substance: classification of genres in text. In: Proceedings of the workshop on Human Language Technology and Knowledge Management, pp. 1–8. Association for Computational Linguistics, Morristown (2001)

programming language includes a keyword that is part of the error dictionary, the mean error rate will be very high.

4. Dewe, J., Karlgren, J., Bretan, I.: Assembling a balanced corpus from the internet. In: Proceedings of 11th Nordic Conference of Computational Linguistics. Copenhagen (1998)
5. Hastie, T., Tibshirani, R.: Classification by pairwise coupling. In: M.I. Jordan, M.J. Kearns, S.A. Solla (eds.) Advances in Neural Information Processing Systems, vol. 10. MIT, Cambridge (1998)
6. Huang, Y., Suen, C.: The behavior-knowledge space method for combination of multiple classifiers. In: Proceedings of Computer Vision and Pattern Recognition CVPR '93, pp. 347–352 (1993)
7. Joachims, T.: A statistical learning learning model of text classification for support vector machines. In: SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 128–136. ACM Press, New York (2001)
8. Joachims, T.: Transductive learning via spectral graph partitioning. In: Proceedings of the International Conference on Machine Learning, pp. 290–297 (2003)
9. Karlgren, J., Cutting, D.: Recognizing text genres with simple metrics using discriminant analysis. In: Proceedings of the 15th. International Conference on Computational Linguistics (COLING 94), vol. II, pp. 1071–1075. Kyoto (1994)
10. Kukich, K.: Techniques for automatically correcting words in texts. ACM Comput. Surv. pp. 377–439 (1992)
11. Platt, J.: Machines using sequential minimal optimization. In: Schoelkopf, B., Burges, C., Smola, A. (eds.) Advances in Kernel Methods—Support Vector Learning. MIT, Cambridge (1998)
12. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993)
13. Ringlstetter, C., Schulz, K.U., Mihov, S.: Orthographic errors in web pages: Towards cleaner web corpora. Comput. Lingusit. **32**(3), 295–340 (2006)
14. Rosenfeld, R.: Two decades of statistical language modeling: where do we go from here? Proc. IEEE **88**(8), 1270–1278 (2000)
15. Santini, M.: Common criteria for genre classification: Annotation and granularity. In: Workshop on Text-based Information Retrieval (TIR-06). Riva del Garda, Italy (2006)
16. Stubbe, A., Ringlstetter, C., Goebel, R.: Elements of a learning interface for genre qualified search. In: Proceedings of the Workshop Towards Genre-Enabled Search Engines:The Impact of NLP (RANLP-2007). Borovets, Bulgaria (2007)
17. Wahlster, W., (ed.): Verbmobil: Foundations of Speech-to-Speech Translation. Springer, Heidelberg (2000)
18. Wastholm, P., Kusma, A.: Using linguistic data for genre classification. In: Proceedings of the Swedish Artificial Intelligence and Learning Systems Event SAIS-SSLS. Mälardalen University, Schweden (2005)
19. Witten, I.H., Eibe, F.: Data mining: practical machine learning tools and techniques. 2nd edn. Morgan Kaufmann, San Francisco. http://www.cs.waikato.ac.nz/ml/weka (2005)