Genre as Noise - Noise in Genre

Andrea Stubbe[1], **Christoph Ringlstetter**[2], **Klaus U. Schulz**[1]

[1] CIS, University of Munich, Oettingenstr 67, D-80538 München, Germany

E-mail: stubbe,schulz@cis.uni-muenchen.de

[2] AICML, University of Alberta, Edmonton, Canada T6G 2E8

E-mail: kristof@cs.ualberta.ca

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. We then investigate the correlation between genre and micro noise. Using special error dictionaries, we estimate the typical error rates for each genre. We finally 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 it represents an independent dimension, ideally orthogonal to topic. Traditionally, most work in the area of text classification has been 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, also genre classification helps to distinguish between "noise" and "music", i.e., between wanted and unwanted documents.

In the context of documents and genres, the technical terms "noise" has two possible readings. In a narrower sense, it refers to data that is contaminated e.g. by spelling/typing errors, or by errors resulting from OCR recognition. In a coarser sense, depending on the task at hand, each genre can represents a class of noisy documents. For example, when collecting scientific articles in fish, cooking recipes and forums on fishing represent a kind of "macro-noise". Obviously, classifying genre helps to recognize "macro-noise". Observing web pages of certain genres, like for example forums, with an eye-catching number of orthographic errors, a natural question is whether this "micro-noise" can help to classify genre. These two problems represent the kernel of this paper. 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 textual features for the hierarchy. On this basis we designed classifiers for each genre. In our approach, the features used for the classification depend on the genre. In a detailed evaluation we compare the resulting system of classifiers with statistical methods from machine learning.
3. We present a detailed evaluation of the distribution of error rates for orthographic errors found in distinct genres.
4. We show first results in how far an automated analysis of the error rate of a document can be used as an additional feature to improve genre classification.

As to 1, our genre hierarchy extends previous work by [Crowston and Williams, 1997; Dewe *et al.*, 1998]. We tried to reach maximal completeness, 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 work on features and classifiers is motivated by the practical experience that standard classifiers based on learning (e.g., support vector machines [Joachims, 2001]) do not lead to satisfactory results if only a small amount of training documents 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 intelligent hand-crafted fea-

tures would lead to better results. Considerable effort was put in 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]

As to micro noise and point 3, we illuminate the correlation between the genre of a document and its degree of errorness, focusing on the main sources of errors at pattern level occurring in web documents, *spelling* and *typing* errors. For detecting these errors we utilized huge special error dictionaries that capture the main part of errors introduced by the respective noisy channel. In fact the results show a strong correlation between genre and errorness, with a clear trend for documents to be more erroneous that belong to the more private oriented genres. As one application, genres and documents with high error rates can be excluded from corpus construction.

It is natural to assume that the error rate of a given document can give valuable hints on its genre, in particular if this genre typically comes with a striking (low or high) error rate (compared to all other genres, or to the genres neighbored in the hierarchy). We use error dictionaries to derive additional classification features and integrate them into our classifiers. In a first series of experiments we could show positive effects on precision for at least some of the genres.

The paper is structured as follows. In Section 2, we describe our hierarchy of document genres and introduce the corpora used for our experiments. Section 3 addresses to the extraction of genre-specific features and their contribution to the classifiers. In Section 4, we consider strategies for combining the individual classifiers into a decision network. Section 5 refers to the construction and application of error dictionaries. In Section 6, we give the experimental results of classifying a test corpus of 640 annotated documents and present two application studies on detecting noise from a macro perspective. Subsequently, we investigate the distribution of noise from a micro perspective within the genre classification and, finally in a series of first experiments, we report on the effects of applying error rates as a feature to genre classification. The Conclusion summarizes the results and comments on future work.

2 A hierarchy of genres

Genre should exclusively represent the dimensions of the form and function of a text. The classification ought to be task oriented and hierarchical. It has to be logically consistent and complete. A certain text can be assigned to different classes; on the other hand, this should not be the norm.[2] Starting from a predecessor system [Dewe *et al.*, 1998] 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 Dewde et al. were rearranged to 8 container classes. We split up their 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), and private and public homepages were merged into *presentation*(C.7). Error messages, empty pages, and frame sets were put into the *nothing* class(G.1). Several additional new genres below the container classes are meant to increase the coverage of the hierarchy. The journalistic genres were scrutinized by an expert. The final hierarchy is presented in Table 1.

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 receipt	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

Table 1: A hierarchy of genres

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.[3] We tried to gather a broad distribution of topics for each genre in order to avoid specific content.

3 Genre specific classifiers

As we argued above, genre classification helps to recognize unwanted documents. A kernel question behind document classification is the selection of features. While [Dewe *et al.*, 1998] and others use global feature sets, we decided to use specialized features for each genre. The goal was to have 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. In an iterative process, we investigated all training documents for the given genre and identified characteristic clues. Those intuitive and sometimes trivial hypotheses (e.g., catalogs indeed contain a lot of prices) were tested on the complete training collection. For classification, features were arranged to 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 previously acknowledged features can become degraded, and therefore, are removed. For practical reasons the iteration was terminated

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

[2] An exception is constituted by combined documents that are however beyond the scope of this paper.

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

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% has been set.[4] 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 could 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* included specialized word lists as well as rather huge 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 considered. *Patterns* included more complex units, such as repetitions of characters, dates or bibliographic references. Combinations of these features resulted 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 show the pros and cons of a topic. In these cases, we had to construct rules of the form *feature-set-1* $\vee$ *feature-set-2*. To avoid misclassifications, excluding features for otherwise easily confused 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.[5]

textlength, forms
$length > 2500 \wedge length < 45000 \wedge forms < 10$
is a text
$verb > 18 \wedge conjunction > 2$
not too dispassionate, literary or casual language
$adj > 17 \wedge adjPosNeg > 0.5 \wedge adjPosNeg < 4 \wedge contractions < 2.5 \wedge casual < 3$
filter commentaries, faq, interview
$arguing < 1.3 \wedge generalizing < 3.8 \wedge questionmarks < 3$
filter scientific reports and portraits
$science - bigrams < 0.01 \wedge (portraitWords < 1 \vee name + he < 7)$
not too many date-expressions or past-markers
$time < 0.6 \wedge thePast < 1$
first person, not too many (but at least some) names
$we + I > 1.6 \wedge he < 8 \wedge name > 0.5 \wedge name < 6.5$
consequently past or present
$past > present \wedge (past > 0.2 \vee present > 0.2)$
about people and creatures or past adventures and voyages
$(he > 3 \vee name > 4 \vee living > 2) \vee (land > 0.5 \wedge past > 0.4)$

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

Indued with specialized classifiers for each genre, the interplay and global behavior has to be fixed. A trivial option is the use of a multiple classification scheme where all classifiers are applied independently. In other cases, an unequivocal classification is needed, and thus, a decision for the most probable class has to be made. In what follows we describe three possible approaches used in our experiments.

4.1 Filtering

A variant of multiple classification 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. The filter rules operate as a disqualification criterion: if a text has been recognized as A, it can not 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 we computed a confusion matrix on the training data. All classes that are only unidirectional misclassified are suitable for filtering.

4.2 Ordering by dependencies and recall

Instead of computing in advance all the classifications and then filtering the results afterwards, an alternative is to determine an optimal evaluation sequence a text has to go through. As soon as a text is classified, the process stops. This procedure prevents multi-classifications. First of all, the classifiers with the highest recall and precision values are applied. Dependencies between the individual classifiers then have to be considered. To determine the ordering, a dependency graph is used. Each class is represented by a node annotated with recall and precision values. When finding texts of class A recognized as class B we create a directed edge from A to B labeled by the number of texts. The prevention of cycles is managed by first traversing edges with smaller values guaranteeing fewer texts being classified into the wrong class. For even values in both directions, the

[4] 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).

[5] All features for the different genres are available at http://www.cis.uni-muenchen.de/~andrea/genre/features.

node with the higher recall is preferred. 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

4.3 Ordering by F1 value

An alternative solution to set up an unequivocal classification is to use the classifiers in order of their F1 values.[6] The underlying idea is that a higher F1 value indicates a higher probability that the classifier will make the correct decision.

5 Finding errors with error dictionaries

Our method to investigate the correlation between genre and orthographic errors is based on channel specific error dictionaries [Arning, 1995; Ringlstetter *et al.*, 2006]. Assuming errors result from a structured and elucidable process, it is possible to generate them applying a generative algorithm to a language base. In [Ringlstetter *et al.*, 2006] huge error dictionaries have been employed on the basis of character transitions, which were found to comprise the main part of erroneous tokens emerging usually from words of a given language inventory transmitted through a certain noisy channel. Since we did not find many OCR-errors in arbitrary web documents regarding the correlation between genre and noise we concentrated on the error channels *typing* and *wrong cognitive representation*.

5.1 Construction principle

For the composition of a specialized error dictionary, we had to determine the most important error transitions of the particular channel. Depending on the characteristics of the channel, these transitions can be discovered in an analytical way or they have to be derived from observations made in a training corpus containing an expressive number of errors. From the obtained character transitions we deduced transition rules. For the productivity of the rules, it is important how far contextual properties are to be taken into account. A general transition rule has the following shape:

$$R_i := l\alpha r \rightarrow l\beta r \text{ with } l, r, \alpha, \beta \in \Sigma^*$$

Here the pattern α given the left context l and the right context r transforms to the pattern β. As usual Σ represents the character set. The set of all basic rules of the respective error channel is denoted by $\mathcal{R}$. We talk about a level n of an error dictionary, if an input token passes at most through n rules. The application of the rules on an input dictionary D results in an unfiltered rewrite lexicon $\hat{D}_{err}^{Rew}$ that is defined as the set of the transition relations (w^{orig}, R_i, w^{err}). Here w^{orig} denotes a token of the input dictionary D that has been transformed into an error token w^{err} by a rule $R_i \in \mathcal{R}^n$. For our application, measuring the correlation between genre and errorness, we separated the erroneous part of the rewrite dictionary $\hat{D}_{err}$, represented by the set of error tokens $\bigcup w_i^{err}$. For $n = 1$ the use of the base rules does not bear any problems: each rule is applied once to each token of the input dictionary D. For $n > 1$ the question on the composition of the base rules arises. As an example, for $n = 2$ we get a $\mathcal{R} \times \mathcal{R}$ relation and initially $|\mathcal{R}|^2$ production rules.

The number of generated error tokens depends on the specific structure of the rules and the tokens they are applied to. For higher levels n this number can grow very fast.[7] In this investigation we only used level-1 dictionaries.

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

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

Table 2: Size of filter dictionaries.

5.2 Filtering step

Using a final filtering step, from the raw error dictionary $\hat{D}_{err}$, two lexica D_{err}, representing the non-word errors and D_{err-ff} representing the word errors, were generated. For our experiments on genre we only used the dictionary for non-word errors. The filtering procedure needs as input the unfiltered error dictionary and in addition a filter dictionary $\mathcal{D}^{Filter}$. For our experiments, $\mathcal{D}^{Filter}$ represents the union of divers conventional dictionaries presented in Table 2. A lookup procedure performs the ruling into which of the two error dictionaries a token is stored. Note that, as a result, the classification of a token as error is always related to the applied filter lexicon. This can yield considerable effects on the values of precision and recall. For example, the overgeneration (precision) of an error dictionary within the context of multilingual documents can be reduced drastically by adding a missing lexicon of one of the involved languages to the filter procedure.

5.3 Error dictionaries built

In our experiments we used error dictionaries for two different channels that are briefly described below. As base vocabulary, we used a conventional English dictionary with 100,000 entries.

Typing errors. The first type of error dictionary we used is one for typing errors. Typing errors can be divided into transpositions, deletions, insertions and substitutions. Note that the cause of an error can be ambiguous. It is very difficult to decide whether a deletion is a typing error or a cognitive

[7] For example experiments with a level-2 production on OCR error patterns led to an error dictionary with 780 million entries.

error. Transpositions, deletions, substitutions and insertions are covering by far the greater part of typing errors [Kukich, 1992]. After finding no other error patterns in our investigations of text corpora, we only allowed transformations between letters. The first letter being nearly never hit by a typing error, was excluded from the generation process [Kukich, 1992]. Applying all rules together at level-1 we get on average a productivity of 135 mutations per English input token. After clearing out duplicates, resulting because of rule ambiguities and removals by the filtering step against the union of the base lexica D_{Filter}, the English dictionary of typing errors $D_{err}(\textit{English,typing})$ is constituted of $9,427,051$ entries.

Cognitive errors. To find characteristic patterns for cognitive 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 terminates if no new errors with a reasonable frequency can be found. From the list of errors, we derived cognitive error transitions and built up a production program for cognitive error dictionaries. Applying every rule by the *first match principle* to $D(\textit{English})$, we obtained a list containing $1,223,128$ potential error tokens. After the standard filtering step, the error dictionary for English cognitive spelling errors $D_{err}(\textit{English,spell})$, is composed of $1,202,997$ entries. In a study on 1,000 error tokens and on 4,000 tokens recognized by the error dictionaries we found a recall of 62.4% and a precision of 85%. With a clear tendency of higher recall (66.93%) and higher precision (95.00%) for the worst documents (>10 errors per 1,000 tokens).

6 Experimental Results

In this section, we present evaluation results. In the first part we report on the classification of genre by specialized classifiers on our test collection. In addition, we give results for statistical classifiers. In the second subsection we investigate the correlation between genre and orthographic errors. Finally, we describe some preliminary experiments on using errors as an additional feature for genre classification.

6.1 Macro perspective: genre classification by specialized classifiers

As evaluation measure for genre classification we use precision and recall. Difficulties for measurement arise with multiple classifications respectively if a document can not be assigned to any class. 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%.[8] As it turns out 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 shape such as directories, poems, FAQ and forums are better recognized than average. If we allow multiple classification and consider documents as correctly classified that end up not in their original class, but in a class that is also well-justified (such as a scientific report with a great part of statistical information that has been classified to statistics) the precision rises to 80.5%. 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.

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

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

Table 3: Precision (P) and recall (R) of genre classification using specialized classifiers. Ordering of classifier application by a dependency graph. Results for classification into original class.

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 recognized 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

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

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 explanations lead to obvious possibilities to improve the classifiers.

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. The first ML method is the *Naive Bayes Classifier* using the maximum likelihood expectation criterion to make a decision. The naive component is the assumption of sta-

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 5: Precision (P) and recall using Specialized Classifiers, Support Vector Machines, Naive Bayes, J48-Decision-Tree and k-Nearest-Neighbor algorithm.

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%

Table 6: Precision (P) and Recall (R) of queries sent to a search engine to retrieve scientific documents on fish. Values for the original ranking (P, R_{raw}), the rearranged ranking by genre recognition (P, R_{genre}) and the perfect ranking (P_{perf})

tistical independence that allows simple multiplication of 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. With the *k-nearest-neighbor* algorithm (KNN) an object is assigned to the nearest cluster in the feature space. Finally, we applied *Support Vector Machines* (SVMs) [Joachims, 2001], which divide the data into classes by a separating hyper plane.[9] All ML applications were realized with the help of the the WEKA implementations [Witten and Eibe, 2005].[10]

In comparison to statistical methods (cf. Table 5), our method is superior by 39% in precision and 13% in recall. Note that this result completely depends on the small training corpus and we state superiority only under this condition. [11] On the other hand, if one has a classification task it is often unrealistic to annotate thousands of training documents. For these cases we consider the proposed method as a strong alternative. Additionally we did not tune the ML methods which are remarkably strong for some of the genres. Here classifier combination would be a first option as for some of the genres Bayes and for others SVMs lead to better classification.

Comparison with previous work on genre classification. Comparing our results to former published work, the small size of our training corpora and the high number of possible classes should be emphasized. In [Dewdney *et al.*, 2001] using a training corpus with 10,000 documents and only 7 genres a F1 value of 89.1% is reached that sharply decreases with the reduction of training documents. In [Wastholm and Kusma, 2005] for a classification based on the 9 classes of Brown corpus using a Bayes classifier, 57.8% recall and 62.2% precision is reported. In [Karlgren and Cutting, 1994] the influence of the number of genres on classification quality is documented with a decline from 73% precision using 4 different genres to 52% using the 15 Brown categories.

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

Ap.1: 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 an 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 conducted 10 different queries. In Table 6 we present the macro values for recall and precision on the ranked document sets at cut points 5,10,15,20 and the complete set of 30 documents. We compare the result 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, room for further improvements is left. This is caused by the weak recall (40%) of our classifier for science documents.

Ap.2: 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 printed documents and thus boosting the language models [Rosenfeld, 2000]. A shortfall of this method is that all documents are collected, ignoring matters of language style. In our experiments we collected documents from genres where the use of language is similar to that of the spoken corpora. We approved forum/guestbooks, interviews and blogs, and tried to exclude all other documents as noise. Via a web crawl sending combined utterances of a spoken language corpus 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 interview documents a secondary precision of 82%. The blog genre comes with 64% primary precision containing 13 forum documents and 1 interview document leading to a secondary precision of 92%. Compared to the above

[9] 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.

[10] Joachims has shown that for thematic text classification, SVMs outperform the other three methods [Joachims, 2001]. SVMs are more tolerant against the sparseness of the feature space. This has been confirmed for genre classification in [Dewdney *et al.*, 2001].

[11] For example Joachims used for his experiments [Joachims, 2001] 9,603 training documents, nearly 1,000 for each training class.

results for our test collection, the genre classifiers on average show slightly lower precision, but taking desired genres into account it works quite well. If we approximate the recall for the 3 classes by the recall values obtained for the test collection, we obtain a reduction of noise in absolute values of 24,000 files or a residue of only 2.5%.[12]

6.2 Correlation between genre and noise

Table 7 shows the mean rates of errors per 1,000 token (*err*) for each of our 32 genres, as measured with the help of the introduced error dictionaries. In addition, values for the 8 container classes are given. We find extraordinary high differences between the genres. Also 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 up 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 as expected the "guestbook/forum" subclass has an outstanding error rate. Somewhat surprisingly, the value for "blog" is nearly as high as the former. Evidently, spellcheckers are not used too often in this genre. 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 errorness as an additional feature.

In Table 8 we refer to the mean error rates of the 80% documents of a class with the lowest error rate. This cut shall help to eliminate the outliers with a high deviation of the error rate compared to the rest of the class. The picture is not strongly changed. 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, that are possibly not well maintained from a orthographic point of view.

Figure 1 shows the deviation of error rates between training and test corpora with a remarkable stability for all of the corpora except code (C.9).[13]

6.3 Using noise for classification

Observing a significant correlation between genre and mean error rate, we tried to exploit this for the improvement of classification. In a series of first experiments, several of our clas-

[12]The improvements regarding the perplexity of the language model and of the recognition accuracy using this methods will be reported in a forthcoming paper.

[13]As we already knew from previous experiments the code genre is problematic concerning the precision of the error dictionaries. If a programming language includes a keyword that is part of the error dictionary, the mean error rate will be very high. language

A.Journalism	1.49	**C. Information**	2.29	D.3 protocol	1.41
A.1 comment.	1.18	C.1 science.rep.	0.79	**E Directory**	1.72
A.2 review	2.74	C.2 explanation	1.77	E.1 person	0.31
A.3 portrait	1.48	C.3 receipt	2.10	E.2 catalog	1.72
A.4 marginal note	1.04	C.4 faq	2.42	E.3 resource	1.94
A.5 interview	2.08	C.5 lexicon	3.26	E.4 timeline	1.34
A.6 news	1.22	C.6 biling. dict.	4.04	**F. Communication**	5.20
A.7 feature	0.99	C.7 presentation	1.83	F.1 mail,talk	2.84
A.8 reportage	1.18	C.8 statistics	1.69	F.2 forum, guestbook	6.89
B. Literature	3.33	C.9 code	2.78	F.3 blog	6.65
B.1 poem	5.17	**D. Documentation**	0.85	F.4 formular	4.44
B.2 prosa	2.51	D.1 law	0.23	**G. Nothing**	0.00
B.3 drama	2.30	D.2 off. report	0.91	G.1 nothing	0.00

Table 7: Mean error rates (errors per 1,000 token) for different genres in the training part of the genre corpus.

A.Journalism	0.57	**C. Information**	0.74	D.3 protocol	0.87
A.1 comment.	0.30	C.1 science.rep.	0.49	**E Directory**	0.39
A.2 review	0.72	C.2 explanation	0.83	E.1 person	0.30
A.3 portrait	0.85	C.3 receipt	1.24	E.2 catalog	0.82
A.4 marginal note	0.55	C.4 faq	1.39	E.3 resource	0.18
A.5 interview	1.14	C.5 lexicon	1.21	E.4 timeline	0.21
A.6 news	0.19	C.6 biling. dict.	0.42	**F. Communication**	2.33
A.7 feature	0.47	C.7 presentation	0.57	F.1 mail,talk	0.79
A.8 reportage	0.29	C.8 statistics	0.22	F.2 forum, guestbook	3.68
B. Literature	1.37	C.9 code	0.26	F.3 blog	3.65
B.1 poem	1.73	**D. Documentation**	0.43	F.4 formular	1.20
B.2 prosa	1.24	D.1 law	0.04	**G. Nothing**	0.00
B.3 drama	1.14	D.2 off. report	0.56	G.1 nothing	0.00

Table 8: Mean error rates in the training part of the genre corpus using the best 80% of the documents.

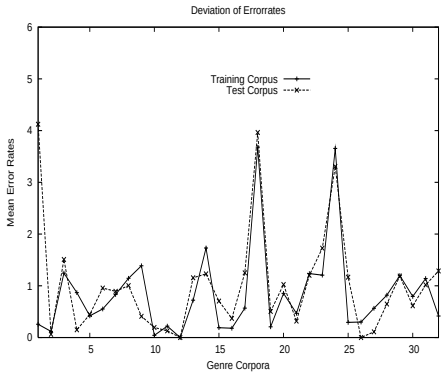

Figure 1: Deviation of error rates for genres (best 80% documents) between training and test coropra.

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%
A.8 reportage	37%	50%	38%	45%
C.7 presentation	21%	45%	30%	35%
A.3 portrait	52%	50%	50%	45%

Table 9: Results for precision and recall by integrating noise as a feature into the genre classifiers compared to the original classifiers.

sifiers have been extended by filters measuring the errorness of a document. The intention was to improve precision by rejecting documents with exceptional high error rates for genres with a low mean error rate. Table 9 shows that these initial results do not show a unique tendency. Some of the classifiers could be improved. An example ist "timeline", where from initially 9 wrongly recognized forum/blog documents only 2 remained after a filtering demanding a mean error rate ≤ 3. On the other hand some of the classifiers lost performance. The failures are explained by the high variance of errorness for some of the genres that leads to wrong rejections. 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.

7 Conclusion

In this paper, we showed that techniques for genre classification can be successfully used to partition document repositories. Dependent on the type of application, some of the genres within a collection constitute noise and have to be excluded. With a new hierarchy of genres, the essential level of granularity for informed corpus construction was realized. Our specialized genre classifiers are extremely easy to implement and they work even for very small training corpora. The results show a competitive performance that has been confirmed in two application studies. Concerning noise in a micro perspective, it turned out that certain genres are correlated to a level of errorness significantly above or below average. This knowledge can be used to reduce noise during corpus construction at the micro level. In a series of first experiments, we were able to show improvements of classification performance by using the level of errorness as an additional feature for some of the classifiers. However since some of the classifiers lost performance additional research has to be done.

Our future research is directed to applications of genre classification such as improving the ranking of search results or focused corpus construction. On the other hand, we want to strengthen the knowledge about the correlation between genre and errorness and the possibilities of its application for genre recognition. Finally we expect to facilitate further insights on the comparison between global feature sets and specialized classifiers and the use of heuristics respectively ML methods by an expansion of our document corpora.

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

References

[Arning, 1995] Andreas Arning. *Fehlersuche in großen Datenmengen unter Verwendung der in den Daten vorhandenen Redundanz*. PhD thesis, University of Osnabrück, 1995.

[Crowston and Williams, 1997] Kevin Crowston and Marie Williams. Reproduced and emergent genres of communication on the world-wide web. In *30th Hawaii International Conference on System Sciences (HICSS) (6)*, pages 30–39, 1997.

[Dewdney *et al.*, 2001] Nigel Dewdney, Carol VanEss-Dykema, and Richard MacMillan. The form is the substance: classification of genres in text. In *Proceedings of the workshop on Human Language Technology and Knowledge Management*, pages 1–8, Morristown, NJ, USA, 2001. Association for Computational Linguistics.

[Dewe *et al.*, 1998] Johan Dewe, Jussi Karlgren, and Ivan Bretan. Assembling a balanced corpus from the internet. In *Proceedings of 11th Nordic Conference of Computational Linguistics*, Copenhagen, Denmark, 1998.

[Joachims, 2001] Thorsten Joachims. 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*, pages 128–136, New York, NY, USA, 2001. ACM Press.

[Karlgren and Cutting, 1994] Jussi Karlgren and Douglass Cutting. Recognizing text genres with simple metrics using discriminant analysis. In *Proceedings of the 15th. International Conference on Computational Linguistics (*COLING 94*)*, volume II, pages 1071 – 1075, Kyoto, Japan, 1994.

[Kukich, 1992] Karen Kukich. Techniques for automatically correcting words in texts. *ACM Computing Surveys*, pages 377–439, 1992.

[Ringlstetter *et al.*, 2006] Christoph Ringlstetter, Klaus U. Schulz, and Stoyan Mihov. Orthographic errors in web pages: Towards cleaner web corpora. *Computational Lingusitics*, 32(3):295–340, September 2006.

[Rosenfeld, 2000] R. Rosenfeld. Two decades of statistical language modeling: where do we go from here? *Proceedings of the IEEE*, 88(8):1270–1278, 2000.

[Wastholm and Kusma, 2005] Peter Wastholm and Annette Kusma. Using linguistic data for genre classification. In *Proceedings of the Swedish Artificial Intelligence and Learning Systems Event SAIS-SSLS*, Mälardalen University, Schweden, 2005.

[Witten and Eibe, 2005] Ian H. Witten and Frank Eibe. Data mining: practical machine learning tools and techniques. 2nd edition. Morgan Kaufmann, San Francisco. http://www.cs.waikato.ac.nz/ ml/weka, 2005.