Extraction of definitions using grammar-enhanced machine learning

Eline Westerhout
Utrecht University
Trans 10, 3512 JK, Utrecht, The Netherlands
E.N.Westerhout@uu.nl

Abstract

In this paper we compare different approaches to extract definitions of four types using a combination of a rule-based grammar and machine learning. We collected a Dutch text corpus containing 549 definitions and applied a grammar on it. Machine learning was then applied to improve the results obtained with the grammar. Two machine learning experiments were carried out. In the first experiment, a standard classifier and a classifier designed specifically to deal with imbalanced datasets are compared. The algorithm designed specifically to deal with imbalanced datasets for most types outperforms the standard classifier. In the second experiment we show that classification results improve when information on definition structure is included.

1 Introduction

Definition extraction can be relevant in different areas. It is most times used in the domain of question answering to answer 'What-is'-questions. The context in which we apply definition extraction is the automatic creation of glossaries within elearning. This is a new area and provides its own requirements to the task. Glossaries can play an important role within this domain since they support the learner in decoding the learning object he is confronted with and in understanding the central concepts which are being conveyed in the learning material.

Different approaches for the detection of definitions can be distinguished. We use a sequential combination of a rule-based approach and machine learning to extract definitions. As a first step a grammar is used and thereafter, machine learning techniques are applied to filter the incorrectly extracted data.

Our approach has different innovative aspects compared to other research in the area of definition extraction. The first aspect is that we address less common definition patterns also. Second, we compared a common classification algorithm with an algorithm designed specifically to deal with imbalanced datasets (experiment 1), which seems to be more appropriate for us because we have some data sets in which the proportion of "yes"-cases is extremely low. A third innovative aspect is that we examined the influence of the type of grammar used in the first step (sophisticated or basic) on the final machine learning results (experiment 1). The sophisticated grammar aims at getting the best balance between precision and recall whereas the basic grammar only focuses at getting a high recall. We investigated to which extent machine learning can improve the low precision obtained with the basic grammar while keeping the recall as high as possible and then compare the results to the performance of the sophisticated grammar in combination with machine learning. As a last point, we investigated the influence of definition structure on the classification results (experiment 2). We expect this information to be especially useful when a basic grammar is used in the first step, because the patterns matched with such a grammar can have very diverse structures.

The paper is organized as follows. Section 2 introduces some relevant work in definition extraction. Section 3 explains the data used in the experiments and the definition categories we distinguish. Section 4 discusses the way in which grammars have been applied to extract definitions and the results obtained with them. Section 5 then talks about the machine learning approach, covering issues such as the classifiers, the features and the experiments. Section 6 and section 7 report and discuss the results obtained in the experiments. Section 8 provides the conclusions and presents some future work.

Proceedings of the EACL 2009 Student Research Workshop, pages 88–96,
Athens, Greece, 2 April 2009.

2 Related research

Research on the detection of definitions has been pursued in the context of automatic building of dictionaries from text, question-answering and recently also within ontology learning.

In the area of automatic glossary creation, the DEFINDER system combines shallow natural language processing with deep grammatical analysis to identify and extract definitions and the terms they define from on-line consumer health literature (Muresan and Klavans, 2002). Their approach relies entirely on manually crafted patterns. An important difference with our approach is that they start with the concept and then search for a definition of it, whereas in our approach we search for complete definitions.

A lot of research on definition extraction has been pursued in the area of question-answering, where the answers to 'What is'-questions usually are definitions of concepts. In this area, they most times start with a known concept (extracted from the question) and then search the corpus for snippets or sentences explaining the meaning of this concept. The texts used are often well structured, which is not the case in our approach where any text can be used. Research in this area initially relied almost totally on pattern identification and extraction (cf. (Tjong Kim Sang et al., 2005)) and only later, machine learning techniques have been employed (cf. (Blair-Goldensohn et al., 2004; Fahmi and Bouma, 2006; Miliaraki and Androutsopoulos, 2004)).

Fahmi and Bouma (2006) combine pattern matching and machine learning. First, candidate definitions which consist of a subject, a copular verb and a predicative phrase are extracted from a fully parsed text using syntactic properties. Thereafter, machine learning methods are applied on the set of candidate definitions to distinguish definitions from non-definitions; to this end a combination of attributes has been exploited which refer to text properties, document properties, and syntactic properties of the sentences. They show that the application of standard machine learning methods for classification tasks (Naive Bayes, SVM and RBF) considerably improves the accuracy of definition extraction based only on syntactic patterns. However, they only applied their approach on the most common definition type, that are the definitions with a copular verb. In our approach we also distinguish other, less common definition types. Because the patterns of the other types are more often also observed in non-definitions, the precision with a rule-based approach will be lower. As a consequence, the dataset for machine learning will be less balanced. In our approach we applied – besides a standard classification algorithm (Naive Bayes) – also a classification algorithm designed specifically to deal with imbalanced datasets.

In the domain of automatic glossary creation, Kobylinski and Przepiórkowski (2008) describe an approach in which a machine learning algorithm specifically developed to deal with imbalanced datasets is used to extract definitions from Polish texts. They compared the results obtained with this approach to results obtained on the same data in which hand crafted grammars were used (Przepiórkowski et al., 2007) and to results with standard classifiers (Degórski et al., 2008). The best results were obtained with their new approach. The differences with our approach are that (1) they use either only machine learning or only a grammar and not a combination of the two and (2) they do not distinguish different definition types. The advantage of using a combination of a grammar and machine learning, is that the dataset on which machine learning needs to be applied is much smaller and less imbalanced. A second advantage of applying a grammar first, is that the grammar can be used to add information to the candidate definitions which can be used in the machine learning features. Besides, applying the grammar first, gives us the opportunity to separate the four definition types.

3 Definitions

Definitions are expected to contain at least three parts. The definiendum is the element that is defined (Latin: that which is to be defined). The definiens provides the meaning of the definiendum (Latin: that which is doing the defining). Definiendum and definiens are connected by a verb or punctuation mark, the connector, which indicates the relation between definiendum and definiens (Walter and Pinkal, 2006).

To be able to write grammar rules we first extracted 549 definitions manually from 45 Dutch text documents. Those documents consisted of manuals and texts on computing (e.g. Word, Latex) and descriptive documents on academic skills and elearning. All of them could be relevant learn-

Type	Example sentence
to be	Gnuplot is een programma om grafieken te maken *'Gnuplot is a program for drawing graphs'*
verb	E-learning omvat hulpmiddelen en toepassingen die via het internet beschikbaar zijn en creatieve mogelijkheden bieden om de leerervaring te verbeteren . *'eLearning comprises resources and application that are available via the Internet and provide creative possibilities to improve the learning experience'*
punctuation	Passen: plastic kaarten voorzien van een magnetische strip, die door een gleuf gehaald worden, waardoor de gebruiker zich kan identificeren en toegang krijgt tot bepaalde faciliteiten. *'Passes: plastic cards equipped with a magnetic strip, that can be swiped through a card reader, by means of which the identity of the user can be verified and the user gets access to certain facilities. '*
pronoun	Dedicated readers. Dit zijn speciale apparaten, ontwikkeld met het exclusieve doel e-boeken te kunnen lezen. *'Dedicated readers. These are special devices, developed with the exclusive goal to make it possible to read e-books.'*

Table 1: Examples for each of the definition types.

ing objects in an elearning enivronment and are thus representative for the glossary creation context in which we will use definition extraction.

Based on the connectors used in the found patterns, four common definition types were distinguished. The first type are the definitions in which a form of the verb *to be* is used as connector. The second group consists of definitions in which a verb (or verbal phrase) other than *to be* is used as connector (e.g. *to mean, to comprise*). It also happens that a punctuation character is used as connector (mainly *:*), such patterns are contained in the third type. The fourth category contains the definitory contexts in which relative or demonstrative pronouns are used to point back to a defined term that is mentioned in a preceding sentence. The definition of the term then follows after the pronoun. Table 1 shows an example for each of the four types. To be able to test the grammar on unseen data, the definition corpus was split in a development and a test part. Table 2 shows some general statistics of the corpus.

	Development	Test	Total
# documents	33	12	45
# words	286091	95722	381813
# definitions	409	140	549

Table 2: General statistics of the definition corpus.

4 Using a grammar

To extract definition patterns two grammars have been written on the basis of 409 manually selected definitions from the development corpus. The XML transducer *lxtransduce* developed by Tobin (2005) is used to match the grammars against files in XML format. Lxtransduce is an XML transducer that supplies a format for the development of grammars which are matched against either pure text or XML documents. The grammars are XML documents which conform to a DTD (lxtransduce.dtd, which is part of the software).

The grammars consist of four parts. In the first part, part-of-speech information is used to make rules for matching separate words. The second part consists of rules to match chunks (e.g. noun phrases, prepositional phrases). We did not use a chunker, because we want to be able to put restrictions on the chunks. For example, to match the definiendum, we only want to select relatively simple NPs (mainly of the pattern (Article) - (Adjective) - Noun(s)). The third part contains rules for matching and marking definiendums and connectors. In the last part the pieces are put together and the complete definition patterns are matched. The rules were made as general as possible to prevent overfitting to the corpus.

Two types of grammars have been used: a basic grammar and a sophisticated grammar. With the basic grammar, the goal is to obtain a high recall without bothering too much about precision. The number of rules for detecting the patterns is 26 of which 6 fall in the first category (matching words), 15 fall in the third part (matching parts of definitions) and 5 fall in the fourth category (matching complete definitions). There are no rules of the second category in this grammar (matching chunks), because the focus is on the connector patterns only and not on the pattern of the definiendum and definiens. In the sophisticated grammar the aim is to design rules in such a way that a high recall is obtained while at the same time the precision does not become very low. This grammar contains 40 rules, which is 14 more than contained in the basic grammar. There are 12 rules in part 1,

5 in part 2, 11 rules in the third part and 12 rules in the last part.

The first difference between the basic and the sophisticated grammar is thus the number of rules. However, the main difference is that the basic grammar puts fewer restrictions on the patterns. Restrictions on phrases present in the sophisticated grammar such as 'the definiendum should be an NP of a certain structure' are not present in the basic grammar. For example, to detect *is* patterns, the basic grammar simply marks all words before a form of *to be* as definiendum and the complete sentence containing a form of *to be* as definition. (Westerhout and Monachesi, 2007) describes the design of the sophisticated grammar and the results obtained with it in more detail.

Table 3 shows that the recall is always higher with the basic grammar is considerably, which is what you would expect because fewer restrictions are used. The consequence of using a less strict grammar is that the precision decreases. The gain of recall is much smaller than the loss in precision, and therefore the f-score is also lower when the basic grammar is used.

type	corpus	precision	recall	f-measure
is	SG	0.25	0.82	0.38
	BG	0.03	0.98	0.06
verb	SG	0.29	0.71	0.41
	BG	0.08	0.81	0.15
punct	SG	0.04	0.67	0.08
	BG	0.01	0.97	0.02
pron	SG	0.05	0.47	0.10
	BG	0.03	0.66	0.06
all	SG	0.13	0.70	0.22
	BG	0.03	0.86	0.06

Table 3: Results with sophisticated grammar (SG) and basic grammar (BG) on the complete corpus.

5 Machine learning

The second step is aimed at improving the precision obtained with the grammars, while trying to keep the recall as high as possible. The sentences extracted with the grammars are input for this step (table 3). We thus have two datasets: the first dataset contains sentences extracted with the basic grammar and the second dataset contains sentences extracted with the sophisticated grammar. Because the datasets are relatively small, both development and test results have been included to get as much training data as possible. As a consequence of using the output of the grammars as dataset, the definitions not detected by the grammar are lost already and cannot be retrieved anymore. So, for example, the overall recall for the *is* type where the sophisticated grammar is used as a first step can not become more than 0.82.

The first classifier used is the Naive Bayes classifier, a common algorithm for text classification tasks. However, because some of our datasets are quite imbalanced and have an extremely low percentage of correct definitions, the Naive Bayes classifier did not always perform very well. Therefore, a balanced classifier has been used also for classifying the data. After describing the classifiers, the experiments and the features used within the experiments are discussed.

5.1 Classifiers

5.1.1 Naive Bayes classifier

The Naive Bayes classifier has often been used in text classification tasks (Lewis, 1998; Mitchell, 1997; Fahmi and Bouma, 2006). Because of the relatively small size of our dataset and sparseness of the feature vector, the calculated numbers of occurrences were very small and we expected them to provide no additional information to the classifier. For this reason, we used supervised discretization (instead of normal distribution), in which numeric attributes are converted to nominal ones, and in this way removed the information on the number of times *n*-grams occurred in a particular sentence.

5.1.2 Balanced Random Forest classifier

The Naive Bayes (NB) classifier is aimed at getting the best possible overall accuracy and is therefore not the best method when dealing with imbalanced data sets. In our experiments, all datasets are more or less imbalanced and consist of a minority part with definitions and a majority part with non-definitions. The extent to which the dataset is imbalanced differs depending on the type and the grammar that has been applied. Table 4 shows for each type the proportion that constitutes the minority class with definitions. As can be seen from this table, the sets for *is* and *verb* definitions obtained with the sophisticated grammar are the most balanced sets, whereas the others are heavily imbalanced.

The problem of heavily imbalanced data can be addressed in different ways. The approach we adopted consists in a modification of the Random

	SG (%)	BG (%)
is	24.6	3.0
verb	28.9	8.1
punct	4.8	1.0
pron	5.4	2.9

Table 4: Percentage of correct definitions in sentences extracted with sophisticated (SG) and basic (BG) grammar.

Forest classifier (RF; (Breiman, 2001)). In Balanced Random Forest (BRF; (Chen et al., 2004)), for each decision tree two bootstrapped sets of the same size, equal to the size of the minority class, are constructed: one for the minority class, the other for the majority class. Jointly, these two sets constitute the training set. In our experiments we made 100 trees in which at each node from 20 randomly selected features out of the total set of features the best feature was selected. The final classifier is the ensemble of the 100 trees and decisions are reached by simple voting. We expect the BRF classifier to outperform the NB classifier, especially on the less balanced types.

5.2 Experiments

Two experiments have been conducted. Because the datasets are relatively small 10-fold cross validation has been used in all experiments for better reliability of the classifier results.

5.2.1 Comparing classifier types

In the first experiment, the Naive Bayes and the Balanced Random Forest classifiers are compared, both on the data obtained with the sophisticated and basic grammar. As features *n*-grams of the part-of-speech tags were used with *n* being 1, 2 and 3. The main purpose of this experiment is to compare the performance of the two classifiers to see which method performs best on our data. We expect the advantage of using the BRF method to be bigger when the datasets are more imbalanced, since the BRF classifier has been designed specifically to deal with imbalanced datasets. The second purpose of the experiment is to investigate whether combining a basic grammar with machine learning can give better results than a sophisticated grammar combined with machine learning. Because the datasets will be more imbalanced for each type when the basic grammar is used, we expect the BRF method to perform better than the NB classifier on the definition class. However, the counter effect of using the balanced method will be that the scores on the non-definition class will be worse.

5.2.2 Influence of definition structure

In the second experiment, we investigated whether the structure of a definition provides information that helps when classifying instances for the datasets created with the basic grammar. As features the part-of-speech tag *n*-grams of the definiendum, the first part-of-speech tag *n*-gram of the definiens and the part-of-speech tag *n*-grams of the complete sentence. Because we have seen when developing the sophisticated grammar that the structure of the definiendum is very important for distinguishing definitions from non-definitions, we decided to add information on the structure of this part in the features of the data obtained with the basic grammar. Also the first part of the definiens often seemed to have a comparable structure, therefore we included this part as well in our features. We expect that including this information will result in a better classification result.

6 Results

6.1 Comparing classifier types

Table 5 shows the results of the different classifiers. When we look at the results for the sophisticated grammar, we see that for the less balanced datasets (i.e. the *punct* and *pron* types) the BRF classifier outperforms the NB classifier. For these two types there were no definitions classified correctly and as a consequence both the precision and the recall are 0. For the other two types the results of the different classifiers are comparable. When the classifiers are used after the basic grammar has been applied, the recall is substantially better for all four types when the BRF method is used. However, the precision is quite low with this approach, mainly due to the low scores for the *punct* and *pron* types. The accuracy of the results, that is, the over all proportion of correctly classified instances, is in all cases higher when the Naive Bayes classifier is used. This is due to the fact that the number of misclassified non-definition sentences is higher when the BRF classifier is used.

Table 6 shows a comparison of the final results obtained with the sophisticated grammar and the basic grammar in combination with the two machine learning algorithms. The performance varies largely per type and the overall score is highly influenced by the *is* and *verb* type, which together

	Naive Bayes							
	Sophisticated grammar				Basic grammar			
	precision	recall	f-measure	accuracy	precision	recall	f-measure	accuracy
is	0.82	0.76	0.79	0.90	0.26	0.66	0.38	0.93
verb	0.77	0.75	0.76	0.86	0.67	0.17	0.27	0.93
punct	0	0	0	0.95	0	0	0	0.98
pron	0.36	0.30	0.33	0.93	0	0	0	0.97
all	0.72	0.61	0.66	0.92	0.29	0.32	0.31	0.95
	Balanced Random Forest							
	Sophisticated grammar				Basic grammar			
	precision	recall	f-measure	accuracy	precision	recall	f-measure	accuracy
is	0.77	0.79	0.78	0.89	0.18	0.82	0.30	0.88
verb	0.76	0.78	0.77	0.87	0.29	0.65	0.40	0.84
punct	0.13	0.61	0.22	0.79	0.06	0.61	0.10	0.79
pron	0.18	0.62	0.28	0.83	0.08	0.41	0.13	0.83
all	0.43	0.74	0.55	0.84	0.15	0.68	0.24	0.85

Table 5: Performance of Naive Bayes classifier and Balanced Random Forest classifier on the results obtained with the grammars.

contain 69.8 % of the definitions. For the other two types, the BRF classifier performs considerably better, independent of which grammar has been used in the first step. The overall f-measure is best when the sophisticated grammar is used, where the recall is higher with the BRF classifier and the precision is better with the NB classifier.

Naive Bayes				
	grammar	precision	recall	f-measure
is	SG	0.82	0.62	0.70
	BG	0.26	0.65	0.37
verb	SG	0.77	0.53	0.63
	BG	0.67	0.14	0.23
punct	SG	0	0	0
	BG	0	0	0
pron	SG	0.36	0.14	0.20
	BG	0	0	0
all	SG	0.72	0.43	0.54
	BG	0.29	0.27	0.28
Balanced Random Forest				
	grammar	precision	recall	f-measure
is	SG	0.77	0.65	0.70
	BG	0.18	0.80	0.30
verb	SG	0.76	0.55	0.64
	BG	0.29	0.53	0.37
punct	SG	0.13	0.42	0.20
	BG	0.06	0.52	0.10
pron	SG	0.18	0.29	0.22
	BG	0.08	0.27	0.12
all	SG	0.43	0.52	0.47
	BG	0.15	0.57	0.24

Table 6: Final results of sophisticated grammar (SG) and basic grammar (BG) in combination with Naive Bayes classifier and Balanced Random Forest classifier.

6.2 Influence of definition structure

Table 7 shows the results obtained with the BRF classifier on the sentences extracted with the basic grammar when sentence structure is taken into account. When we compare these results to table 5, we see that the overall recall is higher when structural information is provided to the classifier. However, to which extent the structural information contributes to a correct classification of the definitions is different per type and also depends on the amount of structural information provided. When only information on the definiendum and first part of the definiens are included, the precision scores are lower than the results obtained with *n*-grams of the complete sentence. Providing all information, that is, information on definiendum, first part of the definiens and the complete sentence, gives the best results.

All information				
	precision	recall	f-measure	accuracy
is	0.24	0.82	0.38	0.92
verb	0.29	0.81	0.43	0.82
punct	0.04	0.84	0.08	0.58
pron	0.09	0.54	0.16	0.83
all	0.14	0.78	0.24	0.82
Definiendum and first *n*-gram of definiens				
	precision	recall	f-measure	accuracy
is	0.19	0.82	0.31	0.89
verb	0.25	0.78	0.38	0.80
punct	0.03	0.96	0.05	0.23
pron	0.05	0.57	0.09	0.65
all	0.09	0.78	0.16	0.71

Table 7: Performance of Balanced Random Forest classifier with information on sentence structure in features applied on the results obtained with the basic grammar.

For the *is* type, the recall remains the same when structural information is added and the precision increases, especially when all structural in-

formation is used. Information on the structure of the definiens and the first *n*-gram of the definiens thus improves the classification results for this type.

The recall of *verb* definitions is higher when structural information is used whereas the precision does not change. The fact that the precision is hardly influenced by adding structural information might be explained by the fact that connectors and connector phrases are quite diverse for this type. As a consequence, different types of first *n*-grams of the definiens might be used and the predicting quality of structural information is smaller.

The classification of the *punct* patterns is quite different depending on the amount of structural information used. The recall increases when structural information is added, whereas the precision decreases. Adding structural information thus results in a low accuracy, especially when only the *n*-grams of the definiendum and the first *n*-gram of the definiens are used. For this type of patterns the structure of the complete definition is thus important for obtaining a reasonable precision.

For the *pronoun* patterns the recall is higher when structural information is included. The precision is slightly higher when all structural information is included, but remarkably lower when only the *n*-grams of the definiendum and the first *n*-gram of the definiens are used. From this we can conclude that for this pattern type information on the structure of the complete definition is crucial to get a reasonable precision.

7 Evaluation and discussion

Which classifier performs best depends on the balance of the corpus. For the more balanced datasets the results of the NB and the BRF method are almost the same. The more imbalanced the corpus, the bigger the difference between the two methods, where BRF outperforms the NB classifier. The accuracy is in all cases higher when the NB classifier is used, due to the fact that this classifier scores better on the majority part with non-definitions. The inevitable counter effect of using the BRF method is that the scores on this part are lower, because the two classes now get the same weight.

The answer to the question which grammar should be used in the first step can be viewed from different perspectives, by looking either at the goal or the definition type.

When aiming at getting the highest possible recall, the BRF method in combination with the basic grammar gives the best overall results. However, when using these settings, the precision is quite low. When the goal is to obtain the best balance between recall and precision, this might therefore not be the best choice. In this case, the best option would be to use a combination of the sophisticated grammar and the BRF method, in which the recall is slightly lower than when the basic grammar is used, but the precision is much higher.

We can also view the question which grammar should be used from a different perspective, namely by looking at the definition type. To get the best result for each of the separate types, we would need to use different approaches for the different types. When the BRF method is used, for two types the recall is considerably higher when the basic grammar is used, whereas for the other two types the recall scores are comparable for the two grammars. However, again this goes with a lower precision score, and therefore this may not be the favourable solution in a practical application. So, also when looking at a per type basis, using the sophisticated grammar seems to be the best option when the aim is to get the best balance.

We are now able to answer the questions addressed in the first experiment and summarize our conclusions on which classifier and grammar should be used in table 8. The conclusions are based on the final results obtained after both the grammar and machine learning have been applied (table 6). Although the recall is very important, because of the context in which we want to apply definition extraction the precision also cannot be too low. In a practical application a user would not like it to get 5 or 6 incorrect sentences for each correct definition.

	Best recall	Best balance
is	BG + BRF	SG + NB / BRF
verb	SG + NB / BRF	SG + NB / BRF
punct	BG + BRF	SG + BRF
pron	SG / BG + BRF	SG + BRF

Table 8: Best combination of grammar and classifier when aiming at best recall or best balance.

Information on structure in all cases results in a higher number of correctly classified definitions. The recall for the definition class is for all types remarkably higher when only the *n*-grams of the

definiendum and the first *n*-gram of the definiens are considered. However, this goes with a much lower precision and f-score and might therefore not be the best option. When using all information, the best results are obtained: the recall goes up while the precision and f-score do not change considerably. However, although the results are improved, they are still lower then the results obtained with the sophisticated grammar.

A question that might rise when looking at the results for the different types, is whether the punctuation and pronoun patterns should be included when building an application for extracting definitions. Although these types are present in texts – they make up 30 % of the total number of definitions – and can be extracted with our methods, the results are poor compared to the results obtained for the other two types. Especially the bad precision for these types gives reasons to have a closer look at these patterns to discover the reason for these low scores. The bad results might be caused by the amount of training data, which might be too low. Another reason might be that the patterns are more diverse than the patterns of the other types, and therefore more difficult to detect.

It is difficult to compare our results to other work on definition extraction, because we are the only who distinguish different types. However, we try to compare research conducted by Fahmi and Bouma (2006) on the first pattern and Kobyliński and Przepiórkowski (2008) on definitions in general. Fahmi and Bouma (2006) combined a rule-based approach and machine learning for the detection of *is* definitions in Wikipedia articles. Although they used more structured texts, the accuracy they obtained is the same as the accuracy we obtained in our experiments. However, they did not report precision, recall, and f-score for the definition class separately, which makes it difficult to compare their result to ours. Kobyliński and Przepiórkowski (2008) applied machine learning on unstructured texts using a balanced classifier and obtained a precision of 0.21, a recall of 0.69 and an f-score of 0.33 with an overall accuracy of 0.85. These scores are comparable to the scores we obtained with the basic grammar in combination with the BRF classifier. Using the sophisticated grammar in combination with BRF outperforms the results they obtained. From this we can conclude that using a sophisticated grammar has advantages over using machine learning only.

8 Conclusions and future work

On the basis of the results we can draw some conclusions. First, the type of grammar used in the first step influences the final results. With the features and classifiers used in our approach, the sophisticated grammar gives the best results for all types. The added value of a sophisticated grammar is also confirmed by the fact that the results Kobyliński and Przepiórkowski (2008) obtained without using a grammar are lower then our results with the sophisticated grammar. A second lesson learned is that it is useful to distinguish different definition types. As the results vary depending on which type has to be extracted, adapting the approach to the type to be extracted will result in a better overall performance. Third, the degree to which the dataset is imbalanced influences the choice for a classifier, where the BRF performs better on less balanced datasets. As there are many other NLP problems in which there is an interesting minority class, the BRF method might be applied to those problems also. From the second experiment, we can conclude that taking definition structure into account helps to get better classification results. This information has not been implemented in other approaches yet and other work on definition extraction can thus profit from this new insight.

The results obtained so far clearly indicate that a combination of a rule-based approach and machine learning is a good way to extract definitions from texts. However, there is still room for improvement, and we will work on this in the next months. In near future, we will investigate whether our results improve when more linguistic information is added in the features. Especially for the basic grammar, we expect it to be possible to get a better recall when more information is added. We can make use of the grammar rules implemented in the sophisticated grammar to see there which information might be relevant. To improve the precision scores obtained with the sophisticated grammar, we will also look at linguistic information that might be relevant. However, improving this score using linguistic information will be more difficult, because the grammar already filtered out a lot of incorrect patterns. To improve results obtained with this grammar, we will therefore look at different features, such as features based on document structure, keywordiness of definiendum and similarity measures.

References

S. Blair-Goldensohn, K. R. McKeown, and A. Hazen Schlaikjer, 2004. *New Directions In Question Answering*, chapter Answering Definitional Questions: A Hybrid Approach. AAAI Press.

L. Breiman. 2001. Random Forests. *Machine Learning*, 46:5–42.

C. Chen, A. Liaw, and L. Breiman. 2004. Using random forest to learn imbalanced data. Technical Report 666, University of California, Berkeley.

Ł. Degórski, M. Marcińczuk, and A. Przepiórkowski. 2008. Definition extraction using a sequential combination of baseline grammars and machine learning classifiers. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation, LREC 2008*.

I. Fahmi and G. Bouma. 2006. Learning to identify definitions using syntactic features. In R. Basili and A. Moschitti, editors, *Proceedings of the EACL workshop on Learning Structured Information in Natural Language Applications*.

Ł. Kobyliński and A. Przepiórkowski. 2008. Definition extraction with balanced random forests. In B. Nordström and A. Ranta, editors, *Advances in Natural Language Processing: Proceedings of the 6th International Conference on Natural Language Processing, GoTAL 2008*, pages 237–247. Springer Verlag, LNAI series 5221.

D. D. Lewis. 1998. Naive (Bayes) at forty: The independence assumption in information retrieval. In Claire Nédellec and Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, number 1398, pages 4–15, Chemnitz, DE. Springer Verlag, Heidelberg, DE.

S. Miliaraki and I. Androutsopoulos. 2004. Learning to identify single-snippet answers to definition questions. In *Proceedings of COLING 2004*, pages 1360–1366.

T. M. Mitchell. 1997. *Machine learning*. McGraw-Hill.

S. Muresan and J. Klavans. 2002. A method for automatically building and evaluating dictionary resources. In *Proceedings of the Language Resources and Evaluation Conference (LREC 2002)*.

A. Przepiórkowski, Ł. Degórski, M. Spousta, K. Simov, P. Osenova, L. Lemnitzer, V. Kubon, and B. Wójtowicz. 2007. Towards the automatic extraction of denitions in Slavic. In *Proceedings of BSNLP workshop at ACL*.

E. Tjong Kim Sang, G. Bouma, and M. de Rijke. 2005. Developing offline strategies for answering medical questions. In D. Mollá and J. L. Vicedo, editors, *Proceedings AAAI 2005 Workshop on Question Answering in Restricted Domains*.

R. Tobin. 2005. Lxtransduce, a replacement for fsgmatch. `http://www.ltg.ed.ac.uk/~richard/ltxml2/lxtransduce-manual.html`.

S. Walter and M. Pinkal. 2006. Automatic extraction of definitions from German court decisions. In *Proceedings of the workshop on information extraction beyond the document*, pages 20–28.

E. N. Westerhout and P. Monachesi. 2007. Extraction of Dutch definitory contexts for elearning purposes. In *Proceedings of CLIN 2006*.