Computational Linguistics in the Netherlands Journal 2 (2012) 52–70 Submitted 09/2012; Published 12/2012

Using skipgrams and PoS-based feature selection for patent classification

Eva D'hondt, Suzan Verberne, Niklas Weber, Kees Koster, Lou Boves
Radboud University Nijmegen

Abstract

Until recently, phrases were deemed suboptimal features for text classification because of their sparseness (Lewis 1992). In recent work (Koster et al. 2011, D'hondt et al. Forthcoming), however, it was found that for classifying English patent documents, combining phrasal and unigram representations leads to significantly better classification results, because phrases are better suited to catch the Multi-Word Terms (MWT) abundant in the terminology-rich technical patent texts.

In this article, we consider the task of patent classification of English abstracts at the class level (about 120 classes) of the International Patent Classification (IPC). We compare (a) the impact of two types of phrases to capture meaningful information (bigrams and skipgrams); and (b) the impact of performing additional filtering of the classification features, based on their Part of Speech (PoS). For this purpose we performed a series of classification experiments using different phrasal text representations and feature selection to determine which representation is most beneficial to English patent classification. We further investigated which type of information (as captured by the PoS-filtered skipgrams) has most impact during classification.

The results show that combining unigrams and PoS-filtered skipgrams leads to a significant improvement in classification scores over the unigram baseline. Additional experiments show that the most important phrasal features are bigrams and additional useful phrases can be captured by allowing at most 2 skips in the skipgram approach. Deeper analysis revealed that the noun-noun combinations and – to a lesser extent – the adjectival-noun combinations are the most informative phrasal features for patent classification.

1. Introduction

Patent classification is a large-scale, unbalanced, multi-class, multi-label text classification problem. Most studies seeking to improve patent classification have focussed on exploiting the hierarchical structure of the IPC[1], the clustering possibilities offered by patent metadata or the imbalance of data in the classes. Relatively little attention, however, has gone to another salient aspect of patent text classification: the language use in patents.

Patents are written in *patentese*: a version of English wrought with genre-specific formulations, terminological Multi-Word Terms (MWT[2]), simplex terms (Kando 2000) and generic terms. The latter are especially interesting: in trying to keep the patent's coverage as broad as possible, while being specific enough to claim novelty, a patent attorney will describe the invention in generic terms: this results in (complex) noun phrases that consist of a generic noun with a function indicator, for example, 'fastening device' to indicate any kind of screw, nail, rope, etc, or 'means establishing fluid communication' to mean 'valve' (Lawson 1997).

[1] The International Patent Classification (IPC) is a complex hierarchical classification system comprising sections, classes, subclasses and groups. For example, the 'A42B 1/12' class label which groups designs for bathing caps, falls under section A *"Human necessities"*, class 42 *"Headwear"* , subclass B *"Head coverings"*, group 1 *"Hats; caps; hoods"*. The latest edition of the IPC contains eight sections, about 120 classes, about 630 subclasses, and approximately 69,000 groups. The IPC covers inventions in all technological fields in which inventions can be patented.

[2] A Multi-Word Term (MWT) is a term that is composed of more than one word. The exact semantics of a Multi-Word Term differ per knowledge area and cannot be inferred directly from its parts (SanJuan et al. 2005, Frantzi et al. 1998).

Because of their peculiar language use, patents pose an interesting problem for text classification: the abundance and variety of technical jargon – patents cover every possible technological field, from flower cutting devices to rocket launchers – results in a large term vocabulary with many very specific, low-frequency terms. The components that make up the generic terms are used in many different combinations across the different categories and are themselves too general to be clear indicators of specific categories. The combinations, however, can be salient features for the categories. Because of the existence of generic terms and the large number of Multi-Word Terms in the terminologies, it seems that phrasal[3] features might be of aid to patent classification.

The use of phrasal features for text classification has been hotly debated: Lewis (1992) did extensive research on using phrasal features for text classification, but found no improvements due to their sparseness. This was later confirmed by Apté et al. (1994) and for a long time, the prevailing idea in the text classification community was that phrasal features have no impact on text classification. However, with the advent of larger data sets and faster algorithms this has been re-examined (Bekkerman and Allan 2003).

Recently, Özgür and Güngör (2010) and Özgür and Güngör (2012) found that for certain text genres adding dependency triples[4] can lead to significant improvements in classification accuracy. The impact differed between genres and, interestingly, could be attributed to different dependency triple types (grammatical relations) for the different genres. In the case of scientific abstracts (the genre in their studies which is most closely related to patent texts) they found a large and significant improvement by adding noun-phrase internal dependency triples to unigrams.

Koster et al. (2011) found a similar result for patent classification: classification accuracy improved significantly when dependency triples were added to the unigrams. Here too, the most important triples were those that contained a noun-noun compound or a noun with an adjectival modifier. D'hondt et al. (Forthcoming) used the same data set and classification algorithm as Koster et al. (2011) to compare the impact of bigrams and two types of dependency triples. They found that adding phrases always leads to significant improvements in classification accuracy, but bigrams are by far the most powerful phrasal features. Deeper analysis showed that although linguistic parsers output some informative features, they struggle with the syntactic structures in the long, complexly embedded noun phrases and, consequently, make consistent errors that result in many noisy, low-frequency triples. Analysis of the high-ranking bigrams brought more flaws to light: It was found that some salient phrases are missed because of function words. A phrase like 'divide and conquer' is cut up into somewhat less meaningful features *divide_and and_conquer* in the bigram approach. Furthermore, even in the best-scoring classifier they found an abundance of phrasal features that are made up of nouns and function words (e.g. *the_device*), which contribute little or limited semantic content to the unigram features.

In this paper, we build on the results found in D'hondt et al. (Forthcoming). Our goal is twofold: First we want to examine a new text representation which overcomes the limits of the bigram representation. *Skipgrams* (cf. Section 2.1) are less bound by the specifics of the surface text and might more effectively capture meaningful phrases from the long and complexly embedded noun phrases in patent texts.

However, the skipgram representation has drawbacks as well: The combinatory possibilities of the skipgrams will lead to a large increase in the number of features, many of which will be combinations of nouns and frequent function words like determiners and prepositions. So even though more meaningful phrases are captured by the skipgram representation, it is not unlikely that these would drown in a sea of noisy, semantically uninformative terms. Consequently, a fair comparison between the different text representations must be coupled with a stricter feature selection, to ensure that we only select those n-gram features that capture the MWT and generic phrases which are so important

[3] By a phrase we mean an index unit consisting of two or more words, generated through either syntactic or statistical methods.

[4] A 'dependency triple' is a triple [word,relation,word] obtained by unnesting a dependency tree. For example, the sentence 'John smokes' can be described as *[John,SUBJ,smoke]*.

in patent texts. This brings us to our second goal: experiment with PoS filtering of the features, that is, we only allow those phrases whose components are nouns, adjectives or verbs. In this way we will attempt to find optimal features for phrase-based automatic classification of English patent texts.

In this paper, we will perform classification experiments using different text representations, namely (a) unigrams; (b) bigrams and (c) skipgrams. The phrasal features will be used in isolation as well as in combination with the unigrams. We will also experiment with PoS filtering on the phrases and words to select the features that have the most *aboutness*[5] and combine these in new classification experiments.

In the analyses we will further investigate (1) the differences between bigrams and skipgrams in the class profiles; (2) the impact of allowing wider skips in phrases; and (3) which subtypes of features (based on PoS combinations) contain the most important information for patent classification.

2. Background

For an extensive overview of the previous literature on the use of statistical and linguistic phrases as features for text classification, see D'hondt et al. (Forthcoming) and the references therein.

2.1 From bigrams to skipgrams

The skipgram representation originates from the field of speech processing, but was introduced into (text) language modelling by Guthrie et al. (2006). It is a combinatorial representation in which n-grams are formed (bigrams, trigrams, etc.) but in addition to allowing adjacent sequences of words, the representation also allows tokens to be 'skipped'. Skipgrams for a given skip distance k allow a maximum of k words skipped to construct the n-gram. Therefore, '3-skip-n-gram' results include 3 skips, 2 skips, 1 skip, and 0 skips (the latter are typical n-grams formed from adjacent words).

Guthrie et al. (2006) compared the coverage of 4-skip-2-grams and regular bigrams, as well as 4-skip-3-grams and regular trigrams. They found that – in case of bigrams – allowing up to 4 skips increased the number of relevant phrases (i.e. bigrams that occur in an unseen test document) with 5 percentage points (raising coverage to 85%). In other words, the skip-2-gram method does uncover relevant phrases that could not be found through the bigram method. In the case of trigrams they found a similar but less pronounced effect. As can be expected, the combinatorial explosion of skip-2-grams results in many noisy, low-frequency phrases, but in additional experiments it was shown that the skipgram phrases are not too variable and can still be used to model context.

Ptaszynski et al. (2011) looked at the usability of skipgrams with more skips and compared these to a regular n-gram approach in language modelling. Their pattern extraction system allows for k-skip-n-grams where k equals sentence length. Their aim is to extract frequent patterns, which they define as occurring at least two times in the corpus. They find that skipgrams are good phrasal features for modelling language in sparse data sets: While the number of frequent n-grams decreases rapidly with the increase in number of elements (larger n-grams), the number of frequent patterns increased for skip-2-grams (compared to unigrams) and then gradually decreased for larger skip-n-grams, providing approximately 5 to 20 times more frequent patterns for the different test sets.

Siefkes et al. (2004) examined a variant of skipgrams called Orthogonal Sparse Bigrams (OSB) which is primarily aimed at reaching the same coverage as regular skipgrams but with less redundancy in the feature space. OSBs are created by moving a window of size k over a string of words, creating bigrams by taking two out of the k words, under the condition that the right-most one is always present. For example: for a sequence of words T with t1 to tN words, the first set of bigrams created with a window size of w = 5, would consist of (t1, t5), (t2, t5), (t3, t5), and (t4, t5). The features

[5] The notion of *aboutness* originates from the library science domain and refers to the conceptual content of a unit of text, stripped of all pragmatic and syntactic detail. For a detailed explanation of the aboutness concept and how it relates to text categorization, see Koster et al. (2011).

resulting from the process described above are orthogonal to each other in the sense that they all span different axes in feature space. Unlike 'regular' skipgrams, OSBs contain information on the number of skips as part of the skipgram. Using OSBs resulted in a much smaller feature space (2.4 times smaller than that of the best scoring n-gram baseline) and achieved a 30% decrease in error rates over a regular skipgram baseline in a spam filtering task using the Winnow classifier.

In addition, skipgrams have been used in a number of NLP application such as irony detection (Reyes et al. 2012), machine translation (Lin and Och 2004) and plagiarism detection (Hartrumpf et al. 2010).

2.2 Feature selection of phrasal features for text classification

The combinatorial explosion (Ptaszynski et al. 2011) of features raises the problem of selecting only those features that are truly representative for a class in text classification. While standard feature selection methods like TF-IDF, Information Gain, etc., are applicable to phrasal features, these features generally have low frequencies and as such might be discarded too easily by the standard feature selection methods, especially when combined with unigram features. In this section we give an overview of the different approaches reported in the literature that are specifically aimed at selecting phrasal features: (a) based on the unigram models of the phrase components; (b) through human selection; and (c) based on linguistic criteria.

2.2.1 Phrase selection based on unigram model scores of the components

Braga et al. (2009) used a Multinomial Naive Bayes classifier to investigate classification performance with uni- and bigrams by comparing multi-view classification, (the results of two independent classifiers trained with unigram and bigram features are merged) with mono-view classification (unigrams and bigrams are combined in a single feature set).[6] They found that there is little difference between the output of the mono- and multi-view classifiers. In the multi-view classifiers, the unigram and bigram classifiers make similar decisions in assigning labels, although the latter generally yielded lower confidence values. Consequently, in the merge the unigram and bigram classifiers affirm each other's decisions, which does not result in an overall improvement in classification accuracy. The authors suggest to combine unigrams only with those bigrams for which it holds that the whole provides more information than the sum of the parts.

Tan et al. (2002) proposed to select highly representative and meaningful bigrams based on the Mutual Information scores of the words in a bigram compared to the unigram class model. They selected only the top 2% of the bigrams as index terms, and found a significant improvement over their unigram baseline, which was low compared to state-of-the-art results. Bekkerman and Allan (2003) failed to improve over their unigram baseline when using similar selection criteria based on the distributional clustering of unigram models.

2.2.2 Phrase selection through human selection

König and Brill (2006) developed an interactive system in which top ranking features from a skipgram-based classifier were presented to human annotators who selected the most discriminating patterns. These patterns were then added to a unigram representation of the texts and the classifier was re-run. The system achieves significantly better results than runs with features selected through statistical feature selection, but the human interaction, although automatized as much as possible, still requires considerable effort.

A related study extracted phrases by capturing frequently occurring keyword combinations within short segments using a rule-based algorithm (Ghanem et al. 2002). These were then filtered through a terminology list supplied by human domain experts for one run, and through a list of keywords

[6] The difference between multi-view and mono-view classification corresponds to what is called *late* and *early fusion* in the pattern recognition literature.

extracted from evidence files (written by domain experts) that were supplied with the training data. The algorithm yielded improved results, but the experiments were done only on a specific and not widely used data set.

2.2.3 Linguistic selection

Pinna and Brett (2012) use Part-of-Speech-grams (PoS-grams) to extract meaningful phrases from corpora for corpus linguistic purposes. A PoS-tagged text corpus can be seen as a sequence of pairs (token, PoS-tag). A PoS-gram is a sequence of PoS-tags drawn from such a PoS-tagged corpus. Hence, in each slot of the PoS-gram, any word can occur as long as it belongs to the PoS category of that particular slot (Pinna and Brett 2012).

Luo et al. (2011) used PoS-based term selection of unigrams and bigrams to examine the impact of different PoS categories and PoS combinations for Chinese text classification. They found that for unigrams nouns are by far the most effective terms. In the case of bigrams, noun-verb combinations proved to be the most effective phrases.

3. Experimental Set-up

3.1 Data Selection

Our experiments were conducted on a subset of the CLEF-IP 2010[7] corpus, which is a subset of the MAREC patent collection. It contains 2.6 million patent documents, which pertain to a total of about 1.3 million patents (each patent can consist of multiple patent documents). The patents included in the corpus have been published between 1985 and 2001.

The documents are encoded in a customized XML format and may contain text in English, French and/or German. They consist of the following text sections: title, abstract, claims and description. They also include meta-information, such as inventor, date of application, assignee, etc. Because our focus lies on text classification, we disregard the meta-data. We only use the abstract section for our experiments. Although previous research (Verberne et al. 2010) has shown that adding text from the description section to abstracts leads to a small but significant improvement over classifying abstracts only, we are more interested in comparing the relative gains between the different text representations. Therefore, the restriction to abstracts will not change our findings but reduce the amount of data to a more manageable level.

The classification is carried out on the class level in the IPC-8 hierarchy. Consequently, only documents having at least one IPC class in the <classification-ipcr> field have been used. The selection is further narrowed down by only choosing documents containing an English abstract. Filtering based on these criteria leaves us with 532,264 abstracts, divided into 121 classes. The majority of these documents have one to three class labels, with an average of 2.12 labels per document.

For classification, these documents have been split in a train set of 425,811 (80% of the corpus) and a test set of 106,453 (20%) documents, respectively.[8]

3.2 Data Preprocessing

3.2.1 General preprocessing

General preprocessing of the texts in the training and test files included cleaning up character conversion errors and removing references to claims, image references and in-text list designators from the original texts. This was done automatically using regular expressions. We then ran a Perl

[7] Available through the IRF at `http://www.ir-facility.org/collection`

[8] No cross-validation has been carried out, based on the results of Verberne et al. (2010), who demonstrated that for this corpus there is little variance between different train/test splits (with a standard deviation of less than 0.3%).

script to divide the running text into sentences, by splitting on end-of-sentence punctuation such as question marks and full stops. In order to minimize incorrect splitting of the terminology-rich technical texts, the Perl script was supplied with a list of common English abbreviations and a list containing abbreviations and acronyms that occur frequently in technical texts, derived from the Specialist lexicon.[9]

3.2.2 Part-of-Speech tagging

The preprocessed sentences were then tagged using an in-house Part-of-Speech (PoS) tagger (van Halteren 2000).[10] The tagger was trained on the annotated subset of the British National Corpus and uses the CLAWS-6 tag set.[11] We chose this particular tagger because it is highly customizable to new lexicons and word frequencies: Language usage in the patent domain can differ greatly from that in other genres. For example, the past participle *said* is often used to modify nouns as in *'for said claim'*. While this usage is very rare and archaic in general English where *said* is most often used as a perfect or past tense verb, it is a very typical modifier in patent language. Consequently, for tagging text from the patent genre, a PoS tagger must be updated to account for these differences in language use, so as to output more accurate and better informed tags and tag sequences. For this experiment, we have adapted the tagger to use word frequency information and associated PoS tags from the patent domain, taken from the AEGIR lexicon.[12] However, we have not retrained the tagger on any annotated patent texts. Such annotations are very expensive to make and were not possible within the scope of this article. Consequently, the tagger is still only trained on the labelling sequence distribution of the original training texts, i.e. the British National Corpus.

To ease later filtering, the detailed CLAW-6 tag set, containing 148 tags, was mapped to a more basic set of only 10 Part-of-Speech (PoS) tags, which resembles the set used in the AEGIR lexicon. More specifically, this means that all noun-related tags ($N*$) were mapped to N, all verb-related tags ($V*$) to V and adjectives (JJ) to A. The conversion table can be found in Appendix 6. Please note that this approach does not distinguish between main and auxiliary verbs. The high frequencies of the latter ensure that they are removed during local term selection (see section 3.4).

3.2.3 Lemmatisation

We used the AEGIR lexicon to lemmatize all words in the tagger output based on their PoS tag. In a final step we performed de-capitalization and removed all remaining punctuation except for "-". The special punctuation rule for "-" is present because the hyphen frequently connects two words which, together, form one unit of sense (e.g. *data-driven* in the example sentence). Therefore, we deemed it useful to treat the resulting sequence as one word. A sentence like 'Performance of data-driven processing increased greatly.' results in the following output:

3.2.4 Feature generation

Unigrams were extracted from the lemmatised tagger output. For the filtered unigram variant, we only selected lemmas with a noun (N), adjective (A) or verb (V) tags.[13] For our sample sentence the respective output is given below:

(1) performance, of, data-driven, processing, increase, greatly

(2) performance, data-driven, processing, increase

[9] Both the splitter and abbreviation file can be downloaded from `http://lands.let.kun.nl/~dhondt/`.

[10] Tokenization was performed by the tagger.

[11] `http://ucrel.lancs.ac.uk/claws6tags.html`

[12] The AEGIR lexicon is part of the AEGIR parser, a hybrid dependency parser that is designed to parse technical text. For more information, see (Oostdijk et al. 2010).

[13] We opted not to include adverbs in the feature selection, based on the results of Koster et al. (2011) which showed adverbs are not informative features for patent classification.

Bigrams, i.e. pairs of adjacent words, were created by a Python script that extracted bigrams with zero skips from the tagger output. Like Guthrie et al. (2006), we only created intra-sentential bigrams. For the filtered bigram variant, we selected bigrams that contain combinations of nouns, adjectives and verb tags. The respective unfiltered and filtered output for the example sentence is given below:

(3) performance_of, of_data-driven, data-driven_processing, processing_increase, increase_greatly

(4) data-driven_processing, processing_increase,

Skipgrams were created by a similar Python script. In these experiments we opted to use 2-skip-2-grams since this range covers the most informative phrases without increasing the feature space too much (see section 5.2). As for bigrams, we only allow intra-sentential skipgrams. Furthermore, no information about what words have been skipped or how many of them have been skipped is encoded in the resulting Skipgrams. For the filtered skipgram variant, we only selected skipgrams consisting of nouns, adjectives and verb combinations. The respective skipgrams generated for our example sentence are given below:

(5) performance_of, performance_data-driven, performance_processing, of_data-driven, of_processing, of_increase, data-driven_processing, data-driven_increase, data-driven_greatly, processing_increase, processing_greatly, increase_greatly

(6) performance_data-driven, performance_processing, data-driven_processing, data-driven_increase, processing_increase

3.3 Feature Statistics

A summary of the statistics for the different representations after feature creation is given in Table 1 below.

As can be expected, the more variable and sparse phrases have a much lower token/type ratio than the unigrams. The impact of PoS filtering is much smaller for the unigrams than for bigrams and skipgrams. Filtering out the high-frequency function words does not reduce the number of unigram features (types) much, but does – predictably – lower the token/type ratio. In case of the phrasal features, PoS filtering has a slightly bigger effect on bigram features than on skipgrams, reducing the number of features by 42% and 37% respectively. The lowered token/type ratios of the filtered phrases are caused by filtering out phrases containing function words. As function words appear frequently, types containing them tend to be instantiated by many tokens.

3.4 Classification Experiments

Classification was done using the Linguistic Classification System (LCS, cf. (Koster et al. 2003)). Within this framework one may select a classifier from the following set: Naive Bayes, SVM Light and Balanced Winnow. Earlier work (Verberne et al. 2010) has shown that for patent classification, SVM Light and Balanced Winnow perform similarly well, both outperforming Naive Bayes. Of those two, Balanced Winnow offers the higher speed and, more importantly, human-readable class profiles[14]. Again following (D'hondt et al. Forthcoming) we therefore choose to use Balanced Winnow.

We also use the same LCS configuration as D'hondt et al. (Forthcoming) and Koster et al. (2011) which was based on experiments on two different development sets:

- Global Term Selection: minimal document frequency = 2, minimal term frequency = 3.
- Local Term Selection: Simple Chi Square (Galavotti et al. 2000), selecting the 10,000 most representative term per class.

[14] For each class, the Winnow algorithm outputs a set of the discriminating terms and their associated winnow weights for that class.

- After local term selection all of the remaining terms are combined into one common vocabulary which is then used as a starting point for training the individual classes, i.e. aggregation of term vocabularies.

- Term Strength Calculation: LTC algorithm, a variant of the TF*IDF algorithm (Salton and Buckley 1988).

- Training Method: Ensemble learning based one-versus-rest binary classifiers. This means that there is not one classifier assigning all the class labels, but every class has its own binary classifier. Each of these classifiers independently assigns a score to every given document, representing the confidence that this document belongs to that class. To each document is assigned at least one and at most four of these class labels (if the classifier confidence score is greater than the threshold of 1.0).

- Winnow Configuration: $\alpha = 1.02$, $\beta = 0.98$, $\theta+ = 2.0$, $\theta- = 0.5$, with a maximum of 10 training iterations. We refer to Koster and Beney (2007) for more details on these parameters.

4. Results

In this section we present the classification results, both from the isolation (one text representation only) and combination (unigrams + phrasal representation) runs. The combination runs were done using filtered unigrams. We also performed similar runs for all unigram-phrase combinations using unfiltered unigrams, but the results were nearly identical to the runs reported here.

Like D'hondt et al. (Forthcoming) and Koster et al. (2011) we found consistent improvements in classification accuracy when phrasal features are added to unigrams in the combination runs. Interestingly, all classifiers trained on phrases only (except the filtered bigrams) also outperform the *F1* score of the unigram baseline. To our knowledge, this is unprecedented. This shows the pervasiveness of (linguistic) phrases in the patent texts, whether they be generic terms or Multi-Word Terms. Experiments on the same data sets in Koster et al. (2011) that only used dependency triples as features achieved much lower scores than the unigram baseline (which is comparable to ours). We hypothesize that the syntactic parser's treatment of complex noun phrases (as discussed in section 1) had an adverse effect on the effectiveness of phrases in that experiment.

The impact of using bigrams as opposed to skipgrams is less clear: When we consider the scores of the unfiltered phrases in the isolation runs, we can see that the precision does not change significantly, which implies that the features capture similar information. The major difference lies in the recall which, unsurprisingly, correlates with the data spread, recorded in Table 1. Since the skipgram representation has most features, it achieves higher recall scores. The much sparser filtered bigrams and filtered skipgrams have the lowest recall scores.

When we consider the impact of performing PoS filtering on the different text representations, we can see some interesting results: First, filtering unigrams has absolutely no effect on classification accuracy. Manual inspection of the resulting class profiles also showed that in both filtered and unfiltered profiles the same terms were selected. Filtering the phrases in the isolation runs has little impact on precision but limiting the data causes a drop in recall scores. The filtered bigram run has the lowest recall score of all the isolation runs. We suspect that our approach of PoS filtering is too strict for the bigram representation: Any meaningful phrase that is split up by at least one function word, like for example *'divide and conquer'* is completely discarded in this approach.

In the combination runs, however, there is a marked difference in the impact of PoS filtering for bigrams and skipgrams. The overall bigram performance does not improve, while filtering the skipgrams leads to improvements both in precision and recall, signifying that more discriminative features were found. This results in the unigram + filtered skipgram run to significantly outperform all other runs.

5. In-depth Analysis

5.1 The impact of skipgrams versus bigrams

In this section we investigate why filtered skipgrams outperform filtered bigrams in the combination runs. More specifically, we will investigate whether the skipgram representation creates new, more informative terms that replace regular bigrams in the class profiles, or whether the improvement in classification accuracy is caused by a long tail of skipgram features that give additional information to the same set of features as can be found in the unigram+filtered bigram class profile.

To get a better understanding of the differences, we examined the class profiles of the different classes in the unigrams+filtered bigrams and unigrams+filtered skipgrams runs. A class profile is the model created for an individual class during the training phase. It consists of the set of features that best distinguish that particular class from the rest of the corpus. These features are ranked according to the weight assigned by the Winnow algorithm during training.

We first examined to what extent the global term sets, i.e. the full set of terms that occur in the 121 class profiles, of the two filtered combination runs overlap. The results are given in table 4 which shows feature counts for different subsets of the global term sets. It is organised as follows: The rows distinguish between the different feature representations, i.e. unigrams and phrases. The columns show whether the features:

1. occur only in the term set of the unigrams+filtered bigrams run (column 'UniBi-only'); or
2. occur both in the term set of the unigrams+filtered bigrams and in the term set of the unigrams+filtered skipgrams run (column 'UniBi∩UniSkip'); or
3. occur only in the term set of the unigrams+filtered skipgrams run (column 'UniSkip-only').

In short, the union of columns UniBi-only and UniBi∩UniSkip describes the global term set of the unigrams+filtered bigrams run. The union of columns UniBi∩UniSkip and UniSkip-only describes the term set of the unigrams+filtered skipgrams runs.

The table shows some interesting data: Firstly, in the global term set of the combined filtered skipgram run more phrases (skipgrams) are selected, both in absolute numbers and relative to the number of unigrams. This means that during term selection and training, skipgram phrases prove more informative features than unigrams. Secondly, less than 50% of the phrasal features in the combined skipgram run occur in the bigram runs. In other words, less than half of the selected terms are regular bigrams. The nature of the other selected terms is less clear. As was shown in Table 1, there are around three times more skipgram features than bigram features in the corpus. It is likely that the new features are phrases with one or two skips. We will hereafter refer to this particular subset of skipgram features as 'novel (non-bigram) features'.

Table 4 gives evidence that these novel (non-bigram) features actually replace some of the bigram features: The left-most cell in the 'phrase' row shows that 136,223 of the 335,692 (136,223+199,469) regular bigrams which were deemed informative phrasal features in the combined bigram run, are not selected in the combined skipgram run. Since these terms were present in the corpus in the combined skipgram run, the fact that they were not selected suggests that other, novel (non-bigram) phrases were better at distinguishing between categories during the training phase.

This raises the question where these novel (non-bigram) skipgram features are situated in the (ranked) class profiles. If they replace bigram features with a large Winnow weight, that is, high ranks in the class profiles, we can conclude that allowing skips in the skipgram representation creates more informative phrases that are better at distinguishing between categories than bigram features. If, on the other hand, the novel (non-bigram) features are located at lower rankings in the class profiles, it would seem that the most effective features can be captured by the bigram representation, and that the non-bigram skipgram phrases merely provide additional information.

To answer this question we first looked at the distribution of novel (non-bigram) phrases in the top k phrasal features extracted from the class profiles of the combined filtered skipgram run (figure

1). This figure shows where these novel (non-bigram) terms are situated in the class profiles and whether there is a trend in the distributions that holds for all 121 classes.

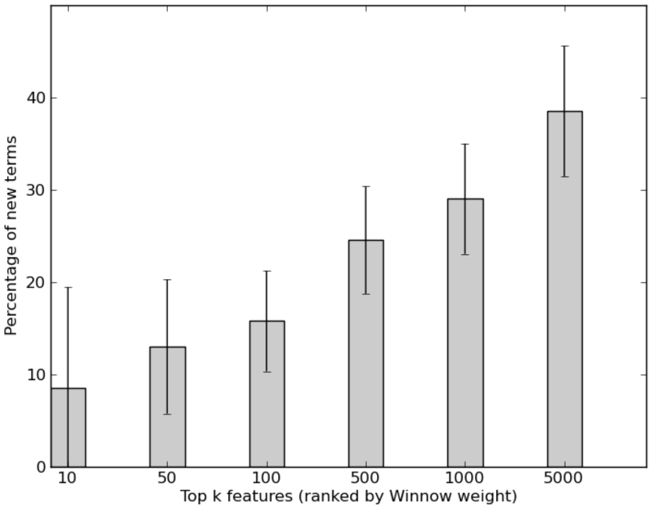

Figure 1: Percentage of non-bigram terms in top k phrasal terms for the unigram+filtered skipgram combination run, averaged over 121 classes.

Figure 2 shows (a) the cosine similarities between the top k unigram terms extracted from the class profiles from the unigrams+filtered bigram run and unigrams+filtered skipgrams run; and (b) the cosine similarities between the top k phrasal terms (bigrams and skipgrams) from the same runs. This gives us an overview of how the class profiles differ at different rankings and whether this is caused by differences in the selection of unigram or phrasal features. Both figures show the averages and standard deviations over all 121 classes.

Figure 1 shows that, on average, only 1 of the top 10 phrasal features in the unigrams+filtered skipgram class profiles is a feature that did not occur in the bigram class profiles. In other words, the highest ranking phrasal terms in the skipgram class profiles are mostly bigrams. The high standard deviation shows that this does not hold for all classes. We did a further analysis of those classes that select more novel non-bigram features at higher ranking, but found no correlation with class size or classification performance. In general, we can conclude that novel non-bigram features are more frequent at lower rankings and continue to replace bigrams at lower levels in the class profile.

In Figure 2 we see a similar pattern: Looking at the cosine similarities of the bigrams and skipgrams in the combined run class profiles, we see a high similarity between the selections of the higher ranking terms. At lower levels in the ranking, the average cosine similarity drops steadily. The drop in cosine similarity scores between the top 100 and top 500 mirrors the increase of non-bigram features at the same ranks in Figure 1. This shows that the differences between the class profiles are not due to reordering of the available bigrams, but caused by a difference in selected features.

Figure 2 also shows that the increase in classification performance in the unigram+filtered skipgram run is a direct consequence of the selection of different phrasal features and not caused by an interaction of these features with the selection of the unigrams. The cosine similarity of the unigrams in the class profiles of the unigram+filtered bigrams and unigram+filtered skipgrams is relatively high and remains stable for lower ranked terms. This indicates that in both runs, nearly the same

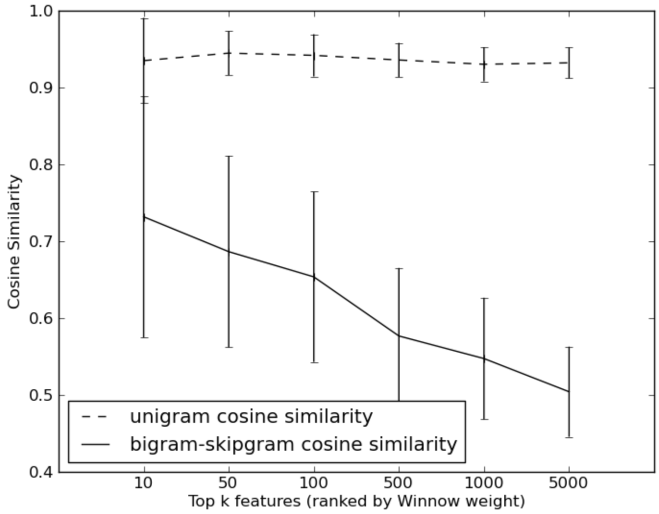

Figure 2: Cosine similarities of top k unigram and phrasal terms in class profiles from combination runs, averaged over 121 classes

unigrams were selected in the class profiles of the 121 categories. In other words, the selection of different features in the two runs does not have an impact on the selection of the unigrams. The small standard deviation shows consistent behaviour across the 121 different categories.

We can conclude that the improvement in classification accuracy of the unigrams+filtered skipgrams run is a direct consequence of the selection of new terms, which were not available to or not selected by the unigrams+filtered bigrams classifier. These terms replace the bigram terms to a certain extent, but are mostly found at lower rankings in the class profiles, indicating that although the most meaningful phrases are captured through the bigram approach, by allowing more skips additional qualitative phrasal features can be found.

5.2 The impact of allowing wider skips

The results reported in section 4, showed that filtered 2-skip-2-grams significantly outperform a classifier trained on bigram features in the combination runs. In the previous section, we found that the additional phrases created through the skipgram approach are features that are sufficiently meaningful and informative for a classifier to select them instead of more general unigrams. We furthermore found evidence that phrases created through the skipgram method replace bigram features, at least at lower rankings in the class profiles.

In this section, we investigate where these informative phrases are situated in the surface text, and if allowing wider skipgrams, that is, skipgrams with more skips, might have a positive impact on classification accuracy. To do so, we ran additional experiments in the isolation runs with a variable number of skips. Note that 1-skip-2-grams incorporate 0-skip-2-grams (i.e. bigrams), 2-skip-2-grams incorporate 1-skip-2-grams as well as 0-skip-2-grams and so on. The results can be seen in figure 3.

The increase in *F1* scores is clearly caused by the improvements in recall of the different k-skip-2-grams. We find the biggest improvement between zero and two skips. This implies that the most effective phrases –after bigrams– consist of words separated by at most two function words or modifiers. For more skips the increase in accuracy tapers off. Clearly, ‘wider’ phrases have less impact during the classification process. Since we find parallel effect in the filtered and unfiltered

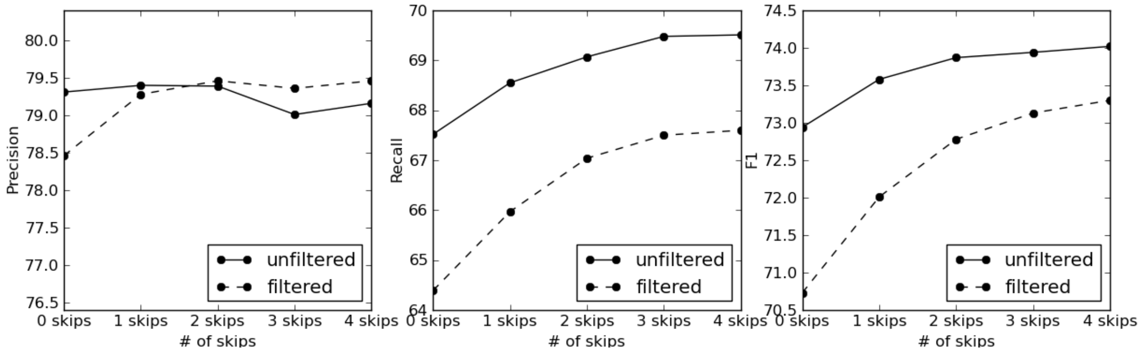

Figure 3: Classification accuracy (precision, recall and F1) for filtered k-skip-2-grams

skipgram runs, it appears that the lack of increase is not caused by the number of features available – for unfiltered skipgrams wider skips lead to a rapid increase in the number of features – but by a property of the newly generated wider skipgram phrases.

5.3 Optimal features

In section 4 we found that –for skipgrams at least– PoS filtering based on the 'aboutness' of terms (as captured by the PoS tags) leads to the best classification results. In this section we describe additional experiments to determine which subtype of information contained in the filtered skipgrams has most impact on classification and whether further, more stringent feature selection can lead to bigger improvements in classification accuracy.

In the filtered skipgram experiments, we allowed all combinations of nouns (N), verbs (V) and adjectives (A) as phrasal features. In table 5 we give an overview of the frequency with which the six possible combinations that occur in this feature set. We do not take the ordering of the phrase elements into account when dividing the features into different combination categories. For example, we consider both 'john_smoke' and 'smoke_cigarette' to be instances of NV combinations.

Table 5 shows a clear division in the data between frequent phrases which contain at least one noun (N) and the much less frequent adjectives and verbs combinations. On the basis of these frequencies, we performed a series of classification experiments for the four largest combination categories. In these experiments we used all filtered skipgram features minus the features from that PoS combination category. By comparing the relative drops in classification accuracy (compared to the filtered skipgram baseline) we expect to see which features contribute most to the overall classification accuracy. Results are given in table 6 which shows the differences with the filtered skipgram baseline, the classification scores and corresponding confidence intervals for the different runs.

The classification results of the PoS combination experiments clearly show that NN and AN combinations make up the most important features in the classification experiments. This is in line with the findings by Koster et al. (2011) and D'hondt et al. (Forthcoming) that noun-noun compounds and adjectival modifier-noun combinations have the most impact during classification. We also find some similarity to the results discussed in Özgür and Güngör (2012) where the features that contributed most when classifying scientific abstracts were noun-noun compounds and nouns with adjectival modifiers.

Given the impact of these features, we wanted to examine if selecting only these phrases would yield comparable results to the filtered skipgrams experiments. We therefore performed a second experiment with only these two subtypes, i.e. NN and NA, both in an isolation and a combination run. The results are shown in table 7.

In the isolation runs, limiting the data to NN- and NA-features only leads to a significant decrease in classification accuracy, compared to allowing all filtered skipgram combinations. In the combination runs we find a similar but insignificant deterioration in classification results. Selecting only noun-noun and adjective-noun combinations discards too many other lower impact terms. However, with only around 15% of the number of initial[15] (unfiltered) skipgram terms, we were able to achieve a similar accuracy to the best-scoring classifier, i.e. unigram+filtered skipgrams.

6. Conclusion

In this paper we investigated different approaches to generate and select phrasal features to improve the classification of abstracts from English patent texts on the class level of the International Patent Classification (IPC). We performed classification experiments using unigrams, bigrams and 2-skip-2-grams features and found that phrases make for informative features for patent classification. In the isolation runs, we found that (unfiltered) phrases outperform the unigram baseline, which – to our knowledge – is unprecedented. In the combination runs, where we added phrasal features to unigram features, we saw significant improvements in classification accuracy over the unigram baseline. These improvements stemmed mostly from increased recall.

We further investigated the impact of Part-of-Speech (PoS) filtering on the different text representations. We ran additional experiments with a filtered set of features that consisted of only nouns (N), adjectives (A), verbs (V), or combinations thereof in case of the phrasal features. PoS filtering of unigrams and bigrams has no positive effect on classification accuracy. In case of the latter, we suspect that too many features are discarded by the strict filtering. For the skipgrams, PoS filtering proved effective: In the combination run, adding filtered skipgrams led to an improvement both in precision and recall, which indicates that more discriminative terms were selected.

An extensive analysis of the class profiles of the combined filtered skipgram run shows that the most important two-word phrases for classification can be captured by bigrams, and that the additional phrases generated through the skipgram approach can be found at lower positions in the ranked class profiles. The skipgram features replace some of the unigram and bigram features, indicating that more informative phrases are generated through the skipgram approach.

We performed additional experiments to determine if more informative phrases can be extracted from patent texts by allowing wider skips in the skipgrams. We found that most effective phrases can be captured by skipgrams with zero (bigrams) upto two skips.

An additional analysis of the relative impact of different PoS combinations in the filtered skipgrams showed that noun-noun and adjective-noun combinations make up the most important features for patent classification. This confirms previous findings by Koster et al. (2011) and D'hondt et al. (Forthcoming) that noun-noun compound and adjectival modifier-noun combinations have the most impact during patent classfication.

We can conclude that adding phrases to unigrams results in significant improvements in classification accuracy for English patent classification. We found that the most effective two-word phrases for patent classification consist of words that lie at most two words apart in the surface texts and capture noun-noun compounds or adjectival modifier-noun combinations.

References

Apté, Chidanand, Fred Damerau, and Sholom Weiss (1994), Automated learning of decision rules for text categorization, *ACM Transactions on Information Systems* **12** (3), pp. 233–251.

Bekkerman, Ron and John Allan (2003), Using bigrams in text categorization, *Technical Report IR-408*, Center of Intelligent Information Retrieval, UMass Amherst.

[15] By initial we mean the number of types in the corpus, i.e. the number of features available before the Global and Local Term Selection are carried out by the LCS.

Braga, Igor, Maria Monard, and Edson Matsubara (2009), Combining unigrams and bigrams in semi-supervised text classification, *Proceedings of Progress in Artificial Intelligence, 14th Portuguese Conference on Artificial Intelligence (EPIA 2009)*, Aveiro, pp. 489–500.

D'hondt, Eva, Suzan Verberne, Cornelis Koster, and Lou Boves (Forthcoming), Text Representations for Patent Classification, *Computational Linguistics.*

Frantzi, Katerina T., Sophia Ananiadou, and Jun-ichi Tsujii (1998), The c-value/nc-value method of automatic recognition for multi-word terms, *Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries*, ECDL '98, Springer-Verlag, London, UK, UK, pp. 585–604.

Galavotti, Luigi, Fabrizio Sebastiani, and Maria Simi (2000), Experiments on the use of feature selection and negative evidence in automated text categorization, *Proceedings 4th European Conference on Research and Advanced Technology for Digital Libraries*, pp. 59–68.

Ghanem, Moustafa M., Yike Guo, Huma Lodhi, and Yong Zhang (2002), Automatic scientific text classification using local patterns: Kdd cup 2002 (task 1), *SIGKDD Explor. Newsl.* **4** (2), pp. 95–96, ACM, New York, NY, USA.

Guthrie, David, Ben Allison, Wei Liu, Louise Guthrie, and Yorick Wilks (2006), A Closer Look at Skip-gram Modelling, *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, European Language Resources Association (ELRA), Paris, pp. 1222–1225.

Hartrumpf, Sven, Tim vor der Brück, and Christian Eichhorn (2010), Semantic duplicate identification with parsing and machine learning, *in* Sojka, Petr, Aleš Horák, Ivan Kopecek, and Karel Pala, editors, *Text, Speech and Dialogue*, Vol. 6231 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 84–92.

Kando, Noriko (2000), What shall we evaluate?–Preliminary discussion for the NTCIR patent IR challenge (PIC) based on the brainstorming with the specialized intermediaries in patent searching and patent attorneys., *Proceedings of the ACM SIGIR 2000 Workshop on Patent Retrieval.*

König, Arnd Christian and Eric Brill (2006), Reducing the human overhead in text categorization, *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, ACM, New York, NY, USA, pp. 598–603.

Koster, Cornelis and Jean Beney (2007), On the importance of parameter tuning in text categorization, *Perspectives of Systems Informatics* pp. 270–283, Springer.

Koster, Cornelis, Jean Beney, Suzan Verberne, and Merijn Vogel (2011), Phrase-based document categorization., *in* Lupu, Mihai, Katja Mayer, John Tait, and Anthony J. Trippe, editors, *Current Challenges in Patent Information Retrieval*, Vol. 29, Springer, pp. 263–286.

Koster, Cornelis, Marc Seutter, and Jean Beney (2003), Multi-classification of patent applications with winnow, *in* Broy, Manfred and Alexandre V. Zamulin, editors, *Ershov Memorial Conference*, Vol. 2890 of *Lecture Notes in Computer Science*, Springer, pp. 546–555.

Lawson, Veronica (1997), The terms and arts of patentese: wolves in sheep's clothing, *in* Wright, S.E. and G. Budin, editors, *Handbook of Terminology Management*, Vol. 1: Basic Aspects of Terminology Management, John Benjamins, Amsterdam, chapter 2.1.3, pp. 171–183.

Lewis, David D. (1992), An evaluation of phrasal and clustered representations on a text categorization task, *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '92)*, pp. 37–50.

Lin, Chin-Yew and Franz Josef Och (2004), Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics, *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Association for Computational Linguistics, Stroudsburg, PA, USA.

Luo, Xi, Wataru Ohyama, Tetsushi Wakabayashi, and Fumitaka Kimura (2011), A study on automatic chinese text classification, *Proceedings of the 2011 International Conference on Document Analysis and Recognition*, ICDAR '11, IEEE Computer Society, Washington, DC, USA, pp. 920–924.

Oostdijk, Nelleke, Suzan Verberne, and Cornelis Koster (2010), Constructing a broadcoverage lexicon for text mining in the patent domain, *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta.*

Özgür, Levent and Tunga Güngör (2010), Text classification with the support of pruned dependency patterns, *Pattern Recognition Letters* **31** (12), pp. 1598–1607.

Özgür, Levent and Tunga Güngör (2012), Optimization of dependency and pruning usage in text classification, *Pattern Analysis and Applications* **15** (1), pp. 45–58.

Pinna, Antonio and David Brett (2012), Fixedness and Variability : Using PoS-grams to Study Phraseology in Newspaper Articles, *Collected Abstracts from the10th Teaching and Language Corpora Conference*, Warsaw.

Ptaszynski, Michal, Rafal Rzepka, Kenji Araki, and Yoshio Momouchi (2011), Language combinatorics: A sentence pattern extraction architecture based on combinatorial explosion, *International Journal of Computational Linguistics (IJCL)* **2**, pp. 24–36.

Reyes, Antonio, Paolo Rosso, and Tony Veale (2012), A multidimensional approach for detecting irony in twitter, *Language Resources and Evaluation* pp. 1–30, Springer Netherlands.

Salton, Gerard and Christopher Buckley (1988), Term-weighting approaches in automatic text retrieval, *Information Processing Management* **24** (5), pp. 513–523.

SanJuan, Eric, James Dowdall, Fidelia Ibekwe-SanJuan, and Fabio Rinaldi (2005), A symbolic approach to automatic multiword term structuring, *Computer Speech and Language* **19** (4), pp. 524–542, Academic Press Ltd., London, UK.

Siefkes, Christian, Fidelis Assis, Shalendra Chhabra, and William S Yerazunis (2004), Combining Winnow and Orthogonal Sparse Bigrams for Incremental Spam Filtering, *Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pp. 410—421.

Tan, Chade-Meng, Yuan-Fang Wang, and Chan-Do Lee (2002), The use of bigrams to enhance text categorization, *Information Processing and Management* **38** (4), pp. 529–546.

van Halteren, Hans (2000), The detection of inconsistency in manually tagged text, *Proceedings of LINC-00.*

Verberne, Suzan, Merijn Vogel, and Eva D'hondt (2010), Patent Classification Experiments with the Linguistic Classification System LCS, *Proceedings of the Conference on Multilingual and Multimodal Information Access Evaluation (CLEF 2010)*, Padua.

Appendix A. Tag conversion table

performance	N
of	PREP
data-driven	A
processing	N
increase	V
greatly	X

Representation		#Tokens	#Types	#Tokens/#Types	% of Hapaxes
unigrams	unfiltered	60,583,174	355,589	170.37	47.81
	filtered	34,441,600	311,976	110.40	47.74
bigrams	unfiltered	58,539,569	4,066,190	14.40	48.68
	filtered	18,016,900	2,391,233	7.53	49.12
skipgrams	unfiltered	169,695,978	11,789,369	14.39	49.44
	filtered	50,565,821	7,392,686	6.84	49.90

Table 1: Corpus statistics for the text representations.

Terms	% Precision	% Recall	% F1
Unfiltered unigrams	76.62 ± 0.25	66.68 ± 0.28	71.31 ± 0.27
Filtered unigrams	76.74 ± 0.25	66.58 ± 0.28	71.30 ± 0.27
Unfiltered bigrams	79.31 ± 0.24	67.52 ± 0.28	72.94 ± 0.27
Filtered bigrams	78.46 ± 0.25	64.39 ± 0.29	70.73 ± 0.27
Unfiltered skipgrams	79.39 ± 0.24	**69.07 ± 0.28**	**73.87 ± 0.26**
Filtered skipgrams	**79.60 ± 0.24**	67.04 ± 0.28	72.78 ± 0.27

Table 2: Classification scores of isolation runs, micro-averaged (95 % conf. value)

Terms	% Precision	% Recall	% F1
Filtered unigrams+unfiltered bigrams	79.36 ± 0.24	71.00 ± 0.27	74.95 ± 0.26
Filtered unigrams+filtered bigrams	79.74 ± 0.24	70.60 ± 0.27	74.89 ± 0.26
Filtered unigrams+unfiltered skipgrams	79.42 ± 0.24	71.13 ± 0.27	75.04 ± 0.26
Filtered unigrams+filtered skipgrams	**80.16 ± 0.24**	**71.54 ± 0.27**	**75.60 ± 0.26**

Table 3: Classification scores of combination runs, micro-averaged (95 % conf. value)

	UniBi-only	UniBi∩UniSkip	UniSkip-only
# of unigrams	19,223	29,212	0
# of phrases	136,223	199,469	230,956

Table 4: Feature counts in the global term sets of the unigrams+filtered bigrams run (col. ‘UniBi-only’ & ‘UniBi∩UniSkip’) and the unigrams+filtered skipgrams run (col. ‘UniBi∩UniSkip’ & ‘UniSkip-only’).

Tag	#Tokens	#Types	#Tokens/Types	% of filtered skipgrams terms
All filtered skipgrams	*50,565,821*	*7,392,686*	*6.84*	*100*
NN	14,025,274	2,153,128	6.51	29.1
AN	12,976,938	2,254,880	5.76	30.5
NV	13,998,047	1,574,830	8.89	21.3
AV	4,801,841	798,498	6.01	10.8
AA	1,584,446	430,288	3.68	5.8
VV	3,179,273	179,676	17.69	2.4

Table 5: Distribution of subtypes of filtered skipgrams

Terms	% Precision	% Recall	% F1
Filtered skipgrams baseline	*79.60 ± 0.24*	*67.04 ± 0.26*	*72.78 ± 0.26*
Filtered skipgrams noNN	**-1.84 (77.76) ± 0.25**	**-4.13 (62.91) ± 0.29**	**-3.23 (69.55) ± 0.28**
Filtered skipgrams noNA	-1.03 (78.57) ± 0.25	-2.76 (64.28) ± 0.29	-2.07 (70.71) ± 0.27
Filtered skipgrams noNV	-0.66 (78.94) ± 0.25	-2.15 (64.89) ± 0.29	-1.56 (71.23) ± 0.27
Filtered skipgrams noVA	+0.02 (79.62) ± 0.24	-0.24 (66.80) ± 0.28	-0.13 (72.65) ± 0.26

Table 6: Classification scores for the four PoS combination experiments

Terms	% Precision	% Recall	% F1
Filtered skipgrams	79.69 ± 0.24	67.03 ± 0.28	72.81 ± 0.27
OnlyNNandNA	78.86 ± 0.25	64.82 ± 0.29	71.16 ± 0.27
Unigrams + filtered skipgrams	**80.17 ± 0.24**	**71.33 ± 0.27**	**75.49 ± 0.26**
Unigrams + onlyNNandNA	79.88 ± 0.24	71.06 ± 0.27	75.21 ± 0.26

Table 7: Classification scores for the combination and isolation runs with NN and NA phrases

CLAW-6	AEGIR	CLAW-6	AEGIR	CLAW-6	AEGIR	CLAW-6	AEGIR	CLAW-6	AEGIR
APPGE	D	II	PREP	NP1	N	RGR	P	VHG	V
AT	D	IO	PREP	NP2	N	RGT	P	VHI	V
AT1	D	IW	PREP	NPD1	N	RL	X	VHN	V
BCL	X	JJ	A	NPD2	N	RP	X	VHZ	V
CC	X	JJR	A	NPDM1	N	RPK	X	VM	V
CCB	X	JJT	A	NPDM2	N	RR	X	VMK	V
CS	X	JK	A	PN	P	RRQ	X	VV0	V
CSA	X	MC	Q	PN1	P	RRQV	X	VVD	V
CSN	X	MC1	Q	PNQO	P	RRR	X	VVG	V
CST	X	MC2	Q	PNQS	P	RRT	X	VVGK	V
CSW	X	MCGE	Q	PNQV	P	RT	X	VVI	V
DA	D	MCMC	Q	PNX1	P	TO	X	VVN	V
DA1	D	MD	A	PPGE	P	UH	X	VVNK	V
DA2	D	MF	Q	PPH1	P	VB0	V	VVZ	V
DAR	D	ND1	N	PPHO1	P	VBDR	V	XX	UNK
DAT	D	NN	N	PPHO2	P	VBDZ	V	YBL	UNK
DB	D	NN1	N	PPHS1	P	VBG	V	YBR	UNK
DB2	D	NN2	N	PPHS2	P	VBI	V	YCOL	UNK
DD	D	NNA	N	PPIO1	P	VBM	V	YCOM	UNK
DD1	D	NNB	N	PPIO2	P	VBN	V	YDSH	UNK
DD2	D	NNL1	N	PPIS1	P	VBR	V	YEX	UNK
DDQ	D	NNL2	N	PPIS2	P	VBZ	V	YLIP	UNK
DDQGE	D	NNO	N	PPX1	P	VD0	V	YQUE	UNK
DDQV	D	NNO2	N	PPX2	P	VDD	V	YQUO	UNK
EX	X	NNT1	N	PPY	P	VDG	V	YSCOL	UNK
FO	X	NNT2	N	RA	X	VDI	V	YSTP	UNK
FU	X	NNU	N	REX	X	VDN	V	ZZ1	UNK
FW	X	NNU1	N	RG	X	VDZ	V	ZZ2	UNK
GE	X	NNU2	N	RGQ	X	VH0	V		
IF	PREP	NP	N	RGQV	X	VHD	V		

Table 8: Table for mapping Claws-6 tags to AEGIR tags