LCI-INSA Linguistic Experiment for CLEF-IP Classification Track

Jean Beney

LCI, Département Informatique, INSA de Lyon F69621 Villeurbanne,
Université de Lyon, jean.beney@insa-lyon.fr

Abstract. We present the experiment the LCI group has performed to prepare our submission to CLEF-IP Classification Track. In this preliminary experiment we used a part of the available *target* documents as test set and the rest as train set. We describe the systems AGFL used for extracting these triples and the LCS used for classification by the Winnow algorithm. We show that the use of linguistic triples in place of bags of words improves the accuracy, as well as using the names and addresses of the applicants. we found that using the complete descriptions as bags of words does not really perform better than using only abstracts and titles. Some simple mathematics show that the official measures are redundant and that R@N should be used to evaluate a ranking, P@1 to evaluate routing and that the usual precision, recall and F1 should be used on the results of a real classification, that is a selection of the classes performed internally by the classifier.

Keywords: Supervised Classification, Document Cathegorization, Winnow, Natural Language Processing, Affix grammars, Linguistic triples.

1 Introduction

The CLEF-IP competition gave us the opportunity to repeat on a widely available document set some experiments that we have done on other less widely accessible data. The goal of these experiments is to investigate the effect of compound linguistic terms on document classification quality.

Most automatic classifications are based on the *Bag of Words* representation of the texts: the words[1] are just counted but there order is not taken into account. It seems obvious that this order, so important for human understanding of the sentences, can have an effect in the classification, at least on the precision by disambiguating homonyms.

The linguistic system AGFL allows us to parse texts efficiently and to build dependency parse trees. We unnested them to dependency triples (two words and a relation) that we use as classification features. This *Bag of Triples* is supposed to better represent the *aboutness*[1] of the text.

[1] wordforms or lemmatized words.

2 The classification system LCS and the Winnow algorithm

The system used is the Linguistic Classification System [5] which was developed by the University of Nijmegen (RU) in the course of the DORO and PEKING Esprit Projects[2]. LCS is a generical system for classification of documents after training on a corpus of known documents (supervised classification). The terms (features) managed by LCS may be either simple words or linguistic triples

The classifiers were trained with balanced Winnow [8], a heuristical learning algorithm with nice mathematical convergence properties. It can be seen as a Perceptron with multiplicative updates. The balanced version [2] can cope with-large numbers of features and can tolerate large variations in document length, because it uses positive and negative weights.

According to previous experiments, the Winnow parameters received the following values: promotion 1.02, demotion 0.92, 10 iterations over the documents, thick threshold [0.6,2]. The general parameters are: term strength LTC, term selection based on χ^2.

3 The linguistic system AGFL and the associated grammars

The linguistic system AGFL[3] (see [6]) reads the grammatical description of a natural language and builds a parser for this language.

The input is an affix grammar [3] that is a context-free grammar, endowed with enumerated parameters, which has been found well-suited for linguistic needs. There are no restrictions on left or right recursion, the huge lexicon is efficiently managed as a trie. The output of the generated transducer is described as a compositional transduction inside the grammar.

AGFL and the associated grammars are under GNU GPL license.

The English grammar NPX[4] used in the present experiment has been written by C.H.A. Koster in Nijmegen, the French grammar FR4IRhas been written by J. Beney in Lyon. Both output dependency trees, where the subtrees (including the leaves, the words) are linked by a grammatical relation.

The trees are unnested to lists of head-modifier pairs that can easily be interpreted as linguistic triples because, if the head is a simple word, the modifier is composed of a relation (or a preposition) and a word. Examples of triples:

[N:rouleau, ATTR A:cylindrique]
[N:machine, P:à N:café]
[N:apparatus, SUBJ V:eliminating]
[N:flocks, through N:duct]

[2] http://www.cs.ru.nl/peking
[3] http://www.agfl.cs.ru.nl/
[4] This grammar has been improved and is now available under the name AEGIR.

4 Experimental setup

4.1 Document selection

We have selected the documents that have at least one abstract (usually an A file, some have 2 abstracts) and an IPC code[5] (to be found in a corresponding B file). To avoid a bias, we have kept only 1 file per patent (sometimes, there are 2 versions of the abstract(s)). We experimented on the 3 languages separately, but also on the concatenation of the abstracts in different languages. Table 1 gives the number of documents we have used, the average document length, the average number of classes and subclasses per document and the number of classes and subclasses that have at least 1 document. The last line gives the number of unique words for the Bag of Words representation of the abstracts and titles.

	All languages		English	French	German
documents	544,126		366,804	55,876	148,631
number of classes	121		118	118	120
number of subclasses	631		617	617	631
words/document			132	121	99
classes/document	1.33		1.32	1.32	1.32
subclasses/document	1.45		1.45	1.42	1.44
	Abstracts	Descriptions			
unique words	1,337,592	4,635,879	288,471	91,734	781,308

Table 1. statistics on the documents.

The number of words in the complete set of documents is larger than the sum of the 3 language sets separately because most of the documents have only 1 abstract but 3 titles. Therefore we have more text (English and French title added to a German abstract, and so on).

4.2 Experimental process

To prepare the CLEF-IP sumission, we have experimented on the training documents we have selected, using 20% of them as a test set. The documents were represented as Bag of Words or as bags of linguistic triplets obtained by parsing the abstracts and titles by the corresponding AGFL parser, concatenating the different unnested tree in case several abstracts are present. As we have no German analyser yet, the German abstracts were kept as bags of words.

For some experiments, we had time to repeat the classification test on 10 different shuffles 80%/20% of the document set (a form of cross-validation). In

[5] we have used the <main-classification> and <further-classification> subelements of <classification-ipc>

these cases, we give the mean and (between parentheses) the standard deviation over the 10 runs.

The full results are given for class level, then we will have a quick look at subclass level.

4.3 Accuracy measures

LCS has been designed to perform a classification, that is, do decide whether a given unknown document belongs to a given class or not. In fact, Winnow computes a score[6] and the documents whose score are greater than 1 are retrieved[7].

During training, the measure F1 (harmonic mean between precision and recall, see [9]) is optimized. The Winnow score allows us to *ra* the documents, but we can only hope that the best F1 leads to the best ranking accuracy measures.

The value given in the following tables are micro-averaged F1 values on all documents. The macro-averaged F1 on all classes is generally lower because the accuracy is very bad for the small classes.

5 Experimental results

We have compared the words and triples representation, expecting that the triples will improve the accuracy. We then studied the effect of the document number on this improvement.

We also compared the use of different XML elements (abstracts, names, addresses and description) to see to what extent the full description performs better than the abstracts. And we briefly compared the accuracy at class and subclass level.

5.1 Triples vs words

The main purpose of this experiment is to find whether linguistic triples can help the classifier or not. In table 2, we compare the classification accuracy (F1) at class level, with words only, with triples only and with both, for French abstracts, English abstracts and all abstracts including German.

We see that the triples-only representation give an accuracy lower than the words-only but that words and triples together give a better accuracy. The same result was found by us on other document collections [4].

The English abstracts show a larger gain than the French abstracts, but this difference can be explained by the number of available training documents (see 5.3). The fact that the German abstracts give a poor result (lower than

[6] This scores allow us to output rankings of the documents in each class and vice versa. These ranking is the result sent to CLEF-IP.

[7] In addition to this Scut, we also use a Rcut, that is each document is retrieved in at least 1 class (which is very useful) and at most 7 classes (4 classes gives very similar results).

	words only	triples only	words+triples	gain
English	69.94%(0.15%)	66.11%(0.11%)	73.05%(0.12%)	3.11%(0.11%)
French	68.41%(0.19%)	50.02%(0.26%)	70.07%(0.20%)	1.66%(0.20%)
German	62.10%	N.A.	N.A.	N.A.
All	75.38%	missing	76.02%	.64%

Table 2. F1 for the different document representations.

for French, with more documents) could be explained by this hypothesis: in German, composed words can be built as in English, but the components are glued together[8]. In the table 1, we see that the german word forms are much more numerous, while the average document length is smaller. Each word form is then more rare and statistically less discriminative.

The result obtained with all the documents is hard to interpret: the accuracy is better because we have more documents and more titles, but why is the gain so small?

5.2 Abstracts, names, descriptions

The patent documents in the CLEF-IP collection are composed of mayny other XML elements besides the abstracts and /or the description. We investigated the use of (words and triples from) the abstracts, titles, descriptions and applicant names and addresses.

The table 3 show the results for different parts of the XML documents[9].

	words	words+triples
abstracts+titles	75.38%	76.02%
abstracts+titles+names+addresses	76.50%	77.15%
descriptions	75.52%	missing

Table 3. F1 for different parts of the documents, all languages together, class level.

The names and addresses of the applicants bear information because most companies work in a restricted domain. It is not surprising that using these parts of the patent helps a bit.

We can note that the improvement is almost the same for the two representations, which means that the information brought by the names and addresses is independent from the information brought by the triples.

[8] For example *software engineering* is translated by *Softwaretechnik*.

[9] For the name and adresses, we used the following XML elements: name, last-name, street, city.

The result obtained with the descriptions is disappointing especially when we consider the time needed to train a classifier with the huge vocabulary of this representation: 9 days when the training for the best result was obtained in 9 hours only.

We did not find time to parse the descriptions and to train with the triples obtained.

5.3 The number of documents

On the English abstracts, which are the more numerous, we also looked at the influence of the number of training documents. Keeping the test set fixed (20% of the available English abstracts), we computed F1 for words, triples, triples+words and for different number of training document randomly choosen in the training set. The results is shown in the training graph for subclass level: figure 1.

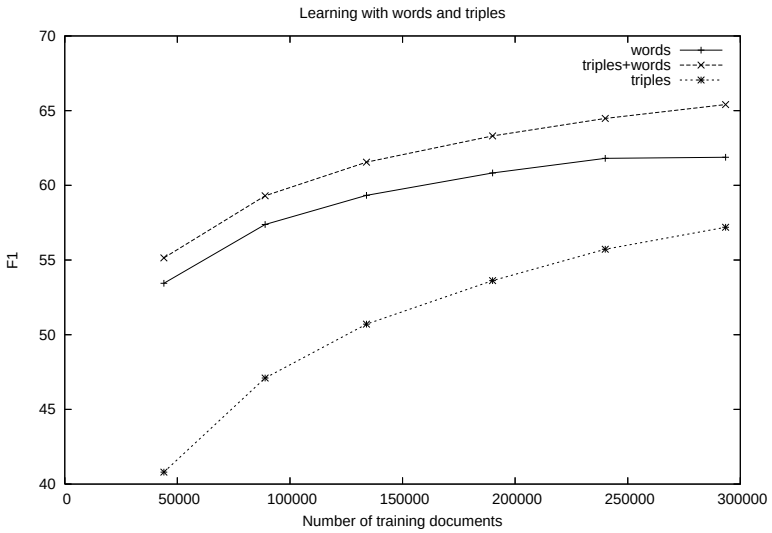

Fig. 1. Training curve, English abstracts,

We clearly see that the classification power of triples is lower than that of words, but it grows faster. We do not know whether, with a much larger number of document, the triples alone could give a better classifier than the words alone.

The effect of the triples on the accuracy of the combination words+triples (gain) also grows with the number of training documents.

These results can be explained as follows: because the triples are necessarily more rare than the words they are composed of, they need more documents in

order to appear a number of times large enough for them to have a statistical effect.

The class size In figure 2, the vertical axe shows the *gain* obtained when using the words and the triples together compared to the use of the words alone. We have excluded the 202 classes that have less than hundred training documents for which the following effect is even more pronounced: the classification into small classes can gain or loose very much from the use of triples. The few train documents available are not enough to get a sample that is representative of the class. Therefore the result is rather uncertain: the gain can be negative or very large in the cases where the words gave a very low accuracy.

However, for classes that have more than 1500 training documents, the gain is always positive, even if it is very small for some very large classes that were already well classified with the words only.

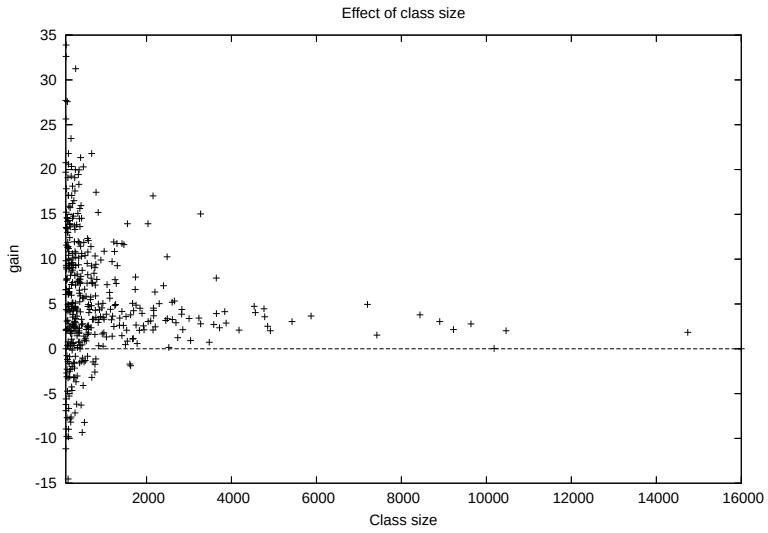

Fig. 2. Effect of the class size.

5.4 The subclass level

The following table can be compared with table 2, which concerns the class level.

As expected, the accuracy with more than 600 subclasses (level for CLEF-IP competition) is always lower than with 120 classes.

The fact that the words and triples together do not perform better than the words is probably due to the use of the subclasses, which are in average 5 times smaller than the classes.

Subclass	words only	triples only	words+triples
All	68.26%	missing	67.80%
English	61.88%	56.86%	65.41%
French	59.93%	41.46%	61.31%
German	60.38%	N.A.	N.A.

Table 4. Subclass level

6 Ranking for CLEF-IP

In the final runs, we have submitted rankings obtained from the Winnow scores with the abstracts in the three languages together. We expected that the differences to appear on the ranking measures would be in agreement with those found on F1 in our preliminary experiment.

But most of the measures give a lower value for triples and words together than for the words alone; for the other measures, the difference is very low (less than 0.2%).

It turned out that the official results were computed using IPC-R codes, so that several subclasses (2.4% of the occurences) never appeared in our training set, making them impossible to be found by us.

Furthermore, the trec_eval program introduces a bias so that teams that do not submit full ranking get better scores. Therefore, it is hard to compare different run results.

6.1 Note on the trec_eval measures

We can easily find a link between precision and recall at the same Rcut (Rank cut-off, see [10]).

Let us define M as the number of documents, T_i as the number of relevant classes for document d_i, $S_i@N$ as the number of relevant classes in the first N classes selected for document d_i, $T = \sum_i T_i$ and $S@N = \sum_i S_i@N$.

Then the micro-averaged precision at N is: $P@N = \frac{S@N}{N \times M}$

the micro averaged recall at N is: $R@N = \frac{S@N}{T}$

and F1 at N is: $F1@N = \frac{2P@N \times R@N}{P@N + R@N} = \frac{2S@N}{T + N \times M}$.

It follows that:

$$N \times M \times P@N = T \times R@N = \frac{T + N \times M}{2} F1@N$$

The 3 values are linearly dependent and monotically increasing one with the other. This is a situation completely different from the usual P/R graphs, where the differents pairs of values are obtained with the same classification method by varying the number of selected classes. Here, the number of selected classes is fixed; the different points are obtained by different methods (or the same method with different parameters).

Figure 3 has been built using the values of P@25 and R@25 for all the submited runs. It clearly shows the linearity for complete rankings and the bias for incomplete rankings. The linearity is not perfect (the straight line is the theoretical line) and we can see in the data some cases where there is obviously an error, or at least a rounding error.

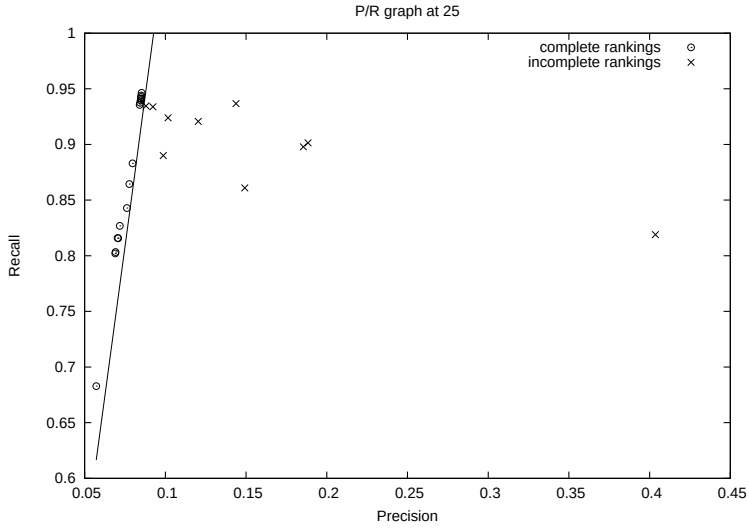

Fig. 3. P/R graph at 25

We also clearly see on this graph the bias introduced by trec_eval for results that do not contain a complete ranking.

The linear dependency means that we do not need the 3 values to compare the results of 2 teams, 2 methods or 2 sets of parameter values: one is enough. The next section will help us to choose.

6.2 Ideal values

In some of cases, it is obvious to know what would be the value of a given measure if the classifier was perfect.

Let $P_M@N$ and $R_M@N$ be the largest possible value of precision and recall when selecting N classes per document. You cannot select more than N classes, you cannot have in your selection for document d_i more than T_i relevant classes, therefore the largest possible value of relevant and selected classes is:

$$S_M@N = \sum_i min(T_i, N)$$

$$Then\ P_M@N = \frac{\sum_i min(T_i, N)}{N \times M}\ \ and\ R_M@N = \frac{\sum_i min(T_i, N)}{T}$$

For small values of N, these values are computed by looking at the number of classes of each document in the qrel file, but:

$$when\ N \geq max(T_i) : S_M@N = \sum_i T_i = T, then : R_M@N = 1$$

Which allows us to see if a method retrieves all the classes (R=1) or almost all the classes with a decent number of selections.

$$when\ N \leq min(T_i) : S_M@N = \sum_i N = N \times M, then : P_M@N = 1$$

This correspond to another practical situation: routing to a single examiner who will, if necessary, forward the patent application to others examiners. For this task, we also have a human reference: at EPO, human routers reached a *precision of the first choice* of 81.2% (see [7]).

6.3 ranking versus classification

The *Rcut* method is often used is Information Retrieval: "give me the first 10 results and, if I do not find what I want, I will modify my request".

A classifier should be able to decide if a given document belongs to a given class or not. The number of classes for a given document can vary very much. Then, a *Scut* (score cut-off) is generally used: a document is selected for a class when its score (computed by the classifier) is larger than a given threshold (1 for us). This can be combined by an *interval Rcut*: assign to each document at least x classes and at most y classes (we used 1-4). This can also be combined by a *Pcut* (proportionnal cut-off): assign to each class a number of document proportionnal to the size of the class in the training set.

To evaluate the quality of these strategies, we need to know which document-class pairs were selected, and the complete ranking does not help. Then, precision, recall and F1 could be computed on that selection.

7 Conclusion and further works

The results we have obtained in these experiments confirm those that we have found on other patent sets (from EPO and WIPO): when using as terms besides the words the linguistic triples, there is a statistically significant improvement. The gain is not large but very stable in cross-validation. Furthermore this gain grows with the number of training documents, when we work on a single language.

When we used all the available abstracts and titles together (3 languages) the accuracy was much larger than with the languages separated, probably because

of the much larger number of available train documents and maybe, for a small amount, of the additional titles.

In this situation, the applicant names and addresses bring extra information that seems independent of the information brought by the triples. The huge document representation by the description words does not perform better after a very long training. We therefore recommend using the triples (from the abstracts and titles) plus the names and addresses.

The French and English grammars are still under improvement and we hope to get even larger improvements with the new versions.

Using the MAREC collection, we plan to repeat the experiment with more documents to see whether the triples alone can perform better than the words with many more occurences. The new LCS3, that is much faster than the previous version, should allow to train on more than 1 million documents in a reasonable time.

References

1. Bruza, P.D., Huibers, T.W.C.: A study of aboutness in information retrieval. Artificial Intelligence Review 10, 1–27 (1996)
2. Dagan, I., Karov, Y., Roth, D.: Mistake-driven learning in text categorization. In: Proceedings of the Second Conference on Empirical Methods in NLP. pp. 55–63 (1997)
3. Koster, C.H.A.: Affix grammars for natural languages. In: In Attribute Grammars, Applications and Systems, International Summer School SAGA. pp. 469–484. Springer-Verlag (1991)
4. Koster, C.H.A., Beney, J.: Phrase-based document categorization revisited. In: Proceedings CIKM 2009, PAIR'09 workshop. pp. 49–55 (2009)
5. Koster, C.H.A., Seutter, M., Beney, J.: Multi-classification of patent applications with Winnow. In: Proceedings of PSI 2003. pp. 545–554. LNCS 2890, Springer-Verlag (2003)
6. Koster, C.H.A., Verbruggen, E.: The AGFL grammar work lab. In: Proceedings FREENIX/Usenix 2002. pp. 13–18 (2002)
7. Krier, M., Zaccà, F.: Automatic categorisation applications at the european patent office. World Patent Information 24, 187–196 (2002)
8. Littlestone, N.: Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. Machine Learning 2, 285–318 (1988)
9. van Rijsbergen, C.J.: Information retrieval. Butterworths, Londres (1979)
10. Yang, Y.: An evaluation of statistical approaches to text categorization. Information Retrieval 1(1), 69–90 (1999)