Terrier takes on the non-English Web

Craig Macdonald
University of Glasgow
Computing Science
Glasgow G12 8QQ, U.K.
craigm@dcs.gla.ac.uk

Christina Lioma
University of Glasgow
Computing Science
Glasgow G12 8QQ, U.K.
xristina@dcs.gla.ac.uk

Iadh Ounis
University of Glasgow
Computing Science
Glasgow G12 8QQ, U.K.
ounis@dcs.gla.ac.uk

ABSTRACT

The aim of this work is to identify how standard Information Retrieval (IR) techniques can be adapted in Web retrieval for non-English queries. In particular, we address the challenge of stemming queries and documents in a multilingual setting. Experiments with a multilingual collection of over 20 languages, more than 800 queries, and various stemming strategies in these languages reveal that using no stemming results in satisfactory Web retrieval performance, that is overall stable. Moreover, we show that language-specific stemming requires an accurate identification of the language of each query.

Categories and Subject Descriptors

H.4 [**Information Storage and retrieval**]: Information Search and Retrieval

General Terms

Measurement, Performance, Experimentation

Keywords

Web retrieval, Known-item retrieval, Multilingual retrieval, Language-specific stemming

1. INTRODUCTION

The field of Information Retrieval (IR) addresses the general problem of how to retrieve information, which is relevant to a user need, from a large repository of information, such as a collection of documents. Information in the document collection is represented in the form of an index, which contains statistics of term frequencies in each document and in the whole collection. Typically, using these statistics, term weighting models compute weights for individual terms, which capture the importance of the terms to the content of each document. A matching function then estimates the likely relevance of a document to a query, on the basis of these term weights, and the most relevant documents are identified and retrieved [26].

In brief, IR systems typically contain an *indexing* component, which stores a collection of information, and a *matching* component, which retrieves relevant information in response to a user query. This very basic architecture is typically enriched with a variety of retrieval-enhancing techniques, aiming to facilitate the system's efficiency and effectiveness. Examples of such techniques are removing stopwords or reducing variants of the same word to a single form (*stemming*). These IR system techniques were originally engineered for English collections of documents and queries.

Nowadays, it is reported that the majority of Web users are non-native English speakers. This means that most people wishing to retrieve information relevant to their need from the Web are likely to do so in a language other than English [4]. It is estimated that non-English queries and unclassifiable queries are not only numerous, but also that they grow increasingly bigger in number. This fact creates a problem for most search engines, which are typically optimised to process mainly English queries. For example, most search engines do not take full account of diacritics or the use of capitals in a user query. Such limitations in processing non-English queries make multilingual retrieval less effective [9]. Consequently, it is usually acknowledged that international search engines (like Yahoo! and Google) are less effective with monolingual non-English queries. In fact, Google has only very recently announced the upcoming launch of a cross-lingual functionality.

In this paper, we investigate how the Terrier retrieval platform [19] can deal with non-English queries. Terrier is a robust and modular IR engine, with an established track record of solid high performance for English retrieval [14, 15]. By testing it on non-English queries, we aim to identify whether standard IR techniques implemented in it are appropriate for non-English retrieval. Specifically, the IR technique we investigate is the application of appropriate stemming in a multilingual Web IR environment.

Stemming consists of reducing morphological variants of a word to a single form (or stem). This technique has been popular with IR systems, because it allows for different word forms to be represented under a single entry. For example, by stemming singular and plural forms of a word to a common form, the occurrence of that word in a document is represented more accurately, and hence retrieval performance and system efficiency improves [10].

Nevertheless, in a multilingual Web IR setting, stemming is not a straightforward process. Firstly, before stemming is applied, the language of the query/document needs to be known, so that the correct stemmer is used. Secondly, morphological complexity varies greatly per language, from the relatively simple (e.g. English), to the relatively more elaborate (e.g. Hungarian). This practically means that, whereas stemming might work for some languages, it might not work for others. Finally, as with other types of lan-

SIGIR '07 iNEWS07 workshop, July 27, 2007, Amsterdam, The Netherlands.

guage resources (e.g. part-of-speech taggers, named entity extractors, and so on), the availability of stemmers for many languages is sparse. In such cases, what is the best strategy: applying no stemming, or using stemmers designed for other languages?

These are the main issues we address in this paper. By doing so, we seek to gain insights into what is the most appropriate way for an IR system to process words in many languages, so that they are accurately indexed and efficiently matched to user queries.

The remainder of this paper is organised as follows. Section 2 gives an overview of studies relating to this work. Section 3 describes how we adapt Terrier to multilingual retrieval. Section 4 presents our experiments and discusses the experimental results. Section 5 concludes this paper with lessons learnt and opted future research directions.

2. RELATED STUDIES

The Web is an heterogeneous environment, in which information may appear in a great variety of different languages. The workshops on the evaluation of multilingual Web IR (WebCLEF) [4, 24] constitute an organised effort into looking at how Web IR systems can scale up to retrieval in a multilingual setting. These workshops have produced literature on a variety of techniques that can extend standard English IR systems to perform multilingual retrieval. One such reported technique is the extension of Web-based features (for example document structure) for retrieval in a multilingual setting [1, 8, 16, 17, 18, 25]. Another technique is applying language-specific stemming when retrieving documents in different languages [16, 17, 25]. An alternative to stemming in a multilingual environment is the use of character n-grams to represent the terms in the index [12]. Further techniques used with retrieval in different languages include normalising diacritics and accents [13]. Encoding issues, one of the biggest problems with non-English retrieval, have been dealt with either by adapting the retrieval system to process specific encodings, such as UTF-8 for example [16], or by transliterating characters into encodings that the system can process [13].

Overall, the above work draws an encouraging yet incomplete picture of multilingual Web IR: encouraging, because the community addresses the problem with organised efforts for standard evaluation. Incomplete, because these efforts reveal that technical difficulties, such as character encoding, are not yet overcome, while there is not a clear consensus on whether standard IR techniques, such as stemming, are beneficial to multilingual IR.

It is this last point that motivates the work in this paper: we address the technical difficulties in doing Web IR across languages by extending the modular Terrier platform, and we investigate the usability of stemming by experimenting with different combinations of stemmers and languages.

3. ADAPTING TERRIER TO MULTILINGUAL RETRIEVAL

In this section, we present how we adapt Terrier's functionalities for non-English retrieval. There are two main components in the overall architecture of the Terrier platform, namely *indexing* (described in Section 3.1), and *matching* (described in Section 3.2). *Indexing* describes the process during which Terrier parses a document collection and represents the information in the collection in the form of an index that contains statistics on term frequency in each document and in the whole collection. Term weights are generated for each term based on these statistics. *Retrieval* describes the process during which Terrier weights each document term and estimates the likely relevance of a document to a query, on the basis of these term weights.

In order to adapt Terrier into a multilingual environment, we focus on the application of appropriate stemming strategies. This technique is part of the system's indexing process, which is presented next.

3.1 Indexing

Indexing consists in parsing a document collection and *appropriately* indexing the information contained in it. In a multilingual setting, indexing collections in an *appropriate* way means being able to support retrieval in different languages, so that the IR system can accurately and uniquely represent each term in the corpus. To meet this requirement, we use a Terrier version that supports multiple character set encodings[1], ensuring that we have a robust representation of the collection.

Terrier achieves modularity in indexing collections of documents by splitting the process into four stages, where, at each stage, plugins can be added to alter the indexing process. The four stages of indexing with Terrier are [19]:

- handling a collection of documents,
- handling and parsing each individual document,
- processing terms from documents, and
- writing the index data structures.

During indexing, Terrier assigns to each term extracted from a document three fundamental properties, namely

- the actual string textual form of the term,
- the position of the term in the document, and
- the document fields in which the term occurs (fields can be arbitrarily defined by the document plugin, but typically relate to HTML/XML tags).

During indexing, the terms pass through a configurable 'Term Pipeline', which transforms them in various ways, using plugins such as stemming, removing stopwords in various languages, expanding acronyms, and so on. The outcome of the Term Pipeline is passed to the Indexer, which writes the data structures of the final index.

We adapt Terrier's indexing component as follows: during the parsing of the collection, we use heuristics to identify the correct character set encoding of each document. In particular, we examine the Content-Type HTTP header of the request, and any equivalent META tag in the header of the HTML document. If neither of these are found, then a default encoding is assumed based on the language of the document (as described below). For example a Czech document is likely to be encoded in ISO8859-2. Once the correct encoding for each document is determined, the collection

[1]The latest open source release of Terrier (version 1.1.0) supports various encodings of documents, and the use of non-Latin character sets. More details can be found at: `http://ir.dcs.gla.ac.uk/terrier/`

is parsed, each term being read and converted into UTF-8 representation. Hence, we ensure that terms from different languages encoded using different character sets are accurately represented in the index.

Terrier's modular architecture allows for any number of different stemmers to be easily applied at this stage. In particular, to determine the language of each document, we use the language identification tool TextCat [5], combined with evidence from the URL and the HTML of each document. For instance, if the identifier fails to identify the language of a document, then we can assume that documents from the .fr domain are likely to be in French. Alternatively, the `HTML` tag of an HTML document can have a `lang` attribute describing the language of the document. In this work, in addition to English stemming, we use several language-specific stemmers, appropriately selected using the language identification data. The application of stemmers is detailed in Sections 4.1 and 4.2.

Because in this paper we investigate the effect of different combinations of stemming upon multilingual retrieval performance, we create different indices of the collection, so that each index applies a different type of stemming strategy. This point is further detailed in Sections 4.1 and 4.2. Overall, we apply several stemming combinations to index the collection. This means that we create different indices of the collection. In each index, we keep field information for each term, so that we can identify which terms occur in which fields of the documents. This is motivated by the fact that, for Web IR, knowing where in a document terms occur may help retrieval performance [6]. In this work, we use different document fields when matching relevant documents to queries, as explained next.

3.2 Matching

So far we have seen how Terrier indexes a collection, so that terms in different languages are represented accurately, and how information on the location of the terms in the documents is also kept. This positional information for terms takes into account document structure in order to enhance retrieval performance. By document structure we denote specific document sections, also referred to as fields in the literature. It has been shown that using document fields can enhance retrieval performance in a Web IR setting [6, 16, 22]. The specific document fields we use in this work are the *body* of the document, the *title* of the document, and the *anchor text* information for a document (i.e. the text associated with the incoming links of a Web document).

We consider these different sources of evidence when matching a document to a query, using a weighting model that is specifically designed to combine term frequencies from different document fields. Specifically, we use the PL2F weighting model from the Divergence From Randomness (DFR) framework [2]. PL2F is a derivative of the PL2 model, which is specifically adapted to combine evidence from different fields. Using the PL2F model, the relevance score of a document d for a query Q is given by:

$$score(d,Q) = \sum_{t \in Q} qtw \cdot \frac{1}{tfn+1}\big(tfn \cdot \log_2 \frac{tfn}{\lambda} + (\lambda - tfn)\cdot \log_2 e + 0.5 \cdot \log_2(2\pi \cdot tfn)\big) \quad (1)$$

where λ is the mean and variance of a Poisson distribution, given by $\lambda = F/N$; F is the frequency of the query term t in the collection, and N is the number of documents in the whole collection. The query term weight qtw is given by qtf/qtf_{max}; qtf is the query term frequency. qtf_{max} is the maximum query term frequency among the query terms.

tfn corresponds to the weighted sum of the normalised term frequencies tf_f for each used field f, known as *Normalisation 2F* [16]:

$$tfn = \sum_f \left(w_f \cdot tf_f \cdot \log_2(1 + c_f \cdot \frac{avg\_l_f}{l_f}) \right), (c_f > 0) \quad (2)$$

where tf_f is the frequency of term t in field f of document d; l_f is the length in tokens of field f in document d, and avg_l_f is the average length of the field across all documents. The contribution of the field is controlled by the weight w_f; c_f is a hyper-parameter for each field, which can be set automatically [11], and which controls the term frequency normalisation. The c_f and w_f values used in this work are given in Section 4.1, along with the rest of the experimental settings.

4. EVALUATION

The aim of our experiments is to investigate whether the standard IR techniques implemented in Terrier are appropriate for non-English retrieval, with special focus on the use of stemming in a multilingual setting. Section 4.1 describes the datasets and resources used, while Section 4.2 describes how we organise our experiments. Experimental results are presented in Section 4.3.

4.1 Experimental Settings

We adapt Terrier for multilingual Web IR (as presented in Section 3), and we evaluate it on the *mixed monolingual* task from WebCLEF 2005 and 2006. The mixed monolingual task simulates a user searching for a known-item page in a European language. This task uses known-item topics, namely homepage finding and named page finding queries. The homepage topics are names of a site that the user wants to reach, and named page topics concern non-homepages that the user wants to reach. The mixed monolingual retrieval task is based on a stream of known-item topics in a range of languages.

The mixed-monolingual retrieval task uses the EuroGOV test collection [23], and more than 800 monolingual known-item topics in various languages.

EuroGOV consists of Web documents crawled from European governmental sites. As such, it is a multilingual Web corpus, containing 3.5 million pages from 27 primary domains, and covering over 20 languages. Specifically, EuroGOV contains documents from the following (top-level) domains:

at(=austria)	cy(=cyprus)
de(=germany)	ee(=estonia)
eu(=european union)	fr(=france)
hu(=hungary)	it(=italy)
lu(=luxemburg)	mt(=malta)
pl(=poland)	ru(=russia)
si(=slovenia)	uk(=united kingdom)
be(=belgium)	cz(=czech republic)
dk(=denmark)	es(=spain)
fi(=finland)	gr(=greece)
ie(=ireland)	lt(=lithuania)
lv(=latvia)	nl(=the netherlands)

```
pt(=portugal)        se(=sweden)
sk(=slovakia)
```

There is no single language that dominates the corpus, and its linguistic diversity provides a natural setting for multilingual Web search. Files in EuroGOV have the following format:

```
 <EuroGOV:bin
domain="" <!-- The top level domain -->
id=""> <!-- The name of the file -->
<EuroGOV:doc
url="" <!-- URL of the page -->
id="" <!-- DocID of the format Exx-yyy-z -->
<!-- E is E and stands for EuroGOV -->
<!-- xx is the top level domain -->
<!-- yyy is the file name -->
<!-- z is the character offset of the document
-->
md5="" <!-- MD5 checksum of the content of
the page -->
fetchDate="" <!-- Fetch date of the page -->
contentType=""> <!-- contentType as given by
the web server -->
<EuroGOV:content>
<![CDATA[
...  content ...  <!-- This is the actual page
-->
]]>
</EuroGOV:content>
</EuroGOV:doc>
...
</EuroGOV:bin>
```

The structure of documents in EuroGOV is clearly marked by the annotation shown above.

An example of the topic format used at WebCLEF 2005 is:

```
 <topic>
<num>WC0006<\num>
<title>Minister van buitenlandse zaken<\title>
<metadata>
<topicprofile>
<language language="NL"/>
<translation language="EN">
dutch minister of foreign affairs </translation>
</topicprofile>
<targetprofile>
<language language="NL"/>
<domain domain="nl"/>
</targeprofile>
<userprofile>
<native language="IS"/>
<active language="EN"/>
<active language="DA"/>
<active language="NL"/>
<passive language="NO"/>
<passive language="SV"/>
<passive language="DE"/>
<passive_other>Faroese</passive_other>
<countryofbirth country="IS"/>
<countryofresidence country="NL"/>
</userprofile>
</metadata>
</topic>
```

The topics used in WebCLEF include a large amount of metadata, as can be seen above. Real-life user queries on the Web do not come with such a variety of metadata. In fact, they typically consist of very few keywords [20]. In order to simulate as much as we can real user queries, in our experiments we only use the title field of the topics.

There is a significant amount of queries available for the 2005 and 2006 mixed-monolingual task. Specifically, the 2005 topics contain 547 queries, consisting of 242 homepage finding queries, and 305 named page finding queries. These queries have been created manually by humans and target pages in 11 different languages: Spanish, English, Dutch, Portuguese, German, Hungarian, Danish, Russian, Greek, Icelandic, and French. The 2006 topics differ from the 2005 topics as follows: a great part of the 2006 topics has been created automatically, using Azzopardi and de Rijke's technique for automatically generating known-item topics [3]. The 2006 topic set also includes a number of manual (human-generated) topics. Specifically, there is a total of 1120 new topics for 2006, 817 of which are automatic, and 303 of which are manual. The 2006 manual queries cover only languages for which human expertise was available (Dutch, English, German, Hungarian, and Spanish) and are supplemented by including some of the queries from the 2005 topic set, while the 2006 automatic queries cover almost all languages. However, in this work, we consider only the manual queries, as the evaluation using the automatic queries did not correlate highly with the true performance of the IR systems as measured by the manual queries [4].

Section 3 presented how we extend Terrier's indexing component to take into account various stemmers, and how we match documents to queries using a field-based weighting model. Specifically, we apply the following stemmers:

- For English, we use Porter's English stemmer;
- For all other languages, we use their corresponding Snowball stemmer[2], with the exception of languages for which there was no stemmer available:
  - For Icelandic, we use the Danish Snowball stemmer; our reasonsing is that Danish is 'linguistically' relatively close to Icelandic.
  - For Hungarian, we use Hunstem[3] as the Snowball stemmer for Hungarian was not available at the time of our experiments.

We do not remove stopwords during indexing, because we do not have stopword lists for all languages, and we do not wish to give an unfair advantage to some languages over others. For retrieval, we use the language topic metadata to select the appropriate stemmer and stopword list for that language. Moreover, we use the body, title, and anchor text[4] fields of documents, which we weight using the PL2F model (Section 3.2). The setting of the parameters c_f and field

[2] http://snowball.tartarus.org/

[3] http://magyarispell.sourceforge.net

[4] During indexing, anchor text from a document with a different language to the target document is stemmed using the stemmer of the language of the source document.

weights w_f presented in Section 3.2 is taken from [16], and is the following:

- $c = 4.10$ & $w = 1$ for the body of the document;
- $c = 100$ & $w = 40$ for the title and anchor text of the document.

Finally, we mentioned earlier that the WebCLEF topics are known-item topics, where a unique URL is targeted. This means that an early precision measure is more suitable to evaluate retrieval in this case. We use the metric also used in WebCLEF, namely the *mean reciprocal rank* (MRR). The reciprocal rank is calculated as 1 divided by the rank at which the (first) relevant page is found. The mean reciprocal rank is obtained by averaging the reciprocal ranks of a set of topics.

4.2 Experimental Methodology

We hypothesise that being able to apply the correct stemmer to a document and a topic can increase retrieval performance. To test this hypothesis, we create three indices of the EuroGOV collection:

1. we index the collection without applying stemming;
2. we index the collection by applying Porter's English stemmer to all documents, regardless of their domain and language;
3. we index the collection by applying stemming to each document according to the language of the document. The language of each document is determined by the language identification data provided by the TextCat utility described in Section 3.1.

We organise our experiments as follows:

- **NoStem**: retrieval without stemming the documents or the queries. This is our baseline.
- **PorStem**: retrieval using Porter's English stemmer for all documents and queries, regardless of their language. This run is a simple baseline showing the effects of applying an English-oriented IR system. For languages not in the Latin character set, Porter's stemming should have no effect.
- **AllStem**: retrieval using language-specific stemming, where the language of the query is defined by the topic-metadata.
- **SelStem**: retrieval using language-specific stemming, where the language of the query is guessed using the TextCat language identifier. When the language identifier fails to identify a language, no stemming is applied to the query and the the unstemmed index is used.

While the run **AllStem** is not realistic in the sense that users would likely not state the language of their query at submission time, it allows us to determine the extent to which the language identification of the queries adds noise to the **SelStem** run. In addition to the runs described above, we compare the system's retrieval performance on a per-language basis, so that we may distinguish between 'harder' and 'easier' languages. The next section details the findings of our experiments.

4.3 Experimental Results

Table 1 displays the retrieval performance of Terrier on the 2005 topic set. We display the MRR scores according to the topic language, the named-page (NP) and home-page (HP) topics, and for all topics in total (All). In Table 1 we observe the following:

- Applying no stemming is generally the most effective approach. This is the general conclusion for all languages. However, on a per-language basis, stemming helps retrieval for German.
- Applying Porter's English stemmer for all languages results in the most stable retrieval performance (the deviation in MRR across all topics is the smallest of all, σ=0.426). However, applying Porter's stemming to all languages significantly harms retrieval performance, yet less than using language-specific stemming. This is the general conclusion for all languages. On a per-language basis, language-specific stemming is better for Danish, German, and Greek. The particularly low performance when applying the correct stemmer to the Hungarian topics (**AllStem**) implies that the Hungarian stemmer is not effective.
- There exists a considerable amount of variation across languages. This point is also displayed graphically in Figure 1(a). This observation is consistent with the general trend observed in WebCLEF 2005 [24], namely that some languages were hard for all systems. Specifically, in WebCLEF 2005, it was reported that most systems scored relatively high for Dutch, relatively low for Russian and Greek, and close to average for German. We observe that Terrier is not only consistent with this, but also generally robust across different languages, including Russian.
- Named page runs score higher than home page runs. This is consistent with the general trend reported in WebCLEF 2005 [24], and also the English monolingual experiments of the Text REtrieval Conference (TREC)[5] for the Web track of 2003 and 2004 [6, 7].
- As expected, the selective application of stemming using the language identifier (**SelStem**) normally decreases in performance compared to the **AllStem** run. This happens when the inaccuracy of the language identifier has caused the wrong stemmer to be selected. For some languages the performance of **SelStem** is better than when the correct stemmer is used (**AllStem**); we suggest that this is mostly the case when the language identifier fails to guess a language, and in these cases the system used the unstemmed query with the unstemmed index was used (which has a better performance).

Table 2 displays the retrieval performance of Terrier on the 2006 topic set. From the table, we observe the following:

- Similarly to before, applying no stemming is the most effective approach, overall, and for both NP and HP tasks, as well as for most languages.

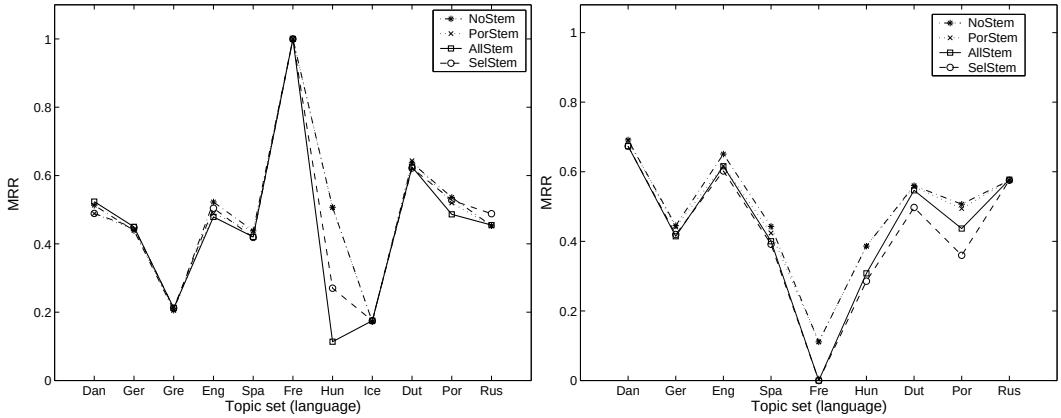
[5] `http://trec.nist.gov/`

Lang.	NoStem	PorStem	(Δ%)	AllStem	(Δ%)	SelStem	(Δ%)
Dan	0.5130	0.4886	(–4.8%)	0.5263	(+2.6%)	0.4891	(–4.7%)
Ger	0.4389	0.4421	(+0.7%)	0.4498	(+2.5%)	0.4476	(+2.0%)
Gre	0.2056	0.2056	(0.0%)	0.2119	(+3.1%)	0.2119	(+3.1%)
Eng	0.5226	0.4892	(–6.4%)	0.4789	(–8.4%)	0.5045	(–3.5%)
Spa	0.4381	0.4370	(–0.3%)	0.4203	(–4.1%)	0.4188	(–4.4%)
Fre	1.0000	1.0000	(0.0%)	1.0000	(0.0%)	1.0000	(0.0%)
Hun	0.5071	0.5062	(–0.2%)	0.1137	(–77.6%)	0.2702	(–46.7%)
Ice	0.1722	0.1722	(0.0%)	0.1750	(+1.6%)	0.1750	(+1.6%)
Dut	0.6371	0.6433	(+1.0%)	0.6251	(–1.9%)	0.6222	(–2.3%)
Por	0.5361	0.5197	(–3.1%)	0.4866	(–9.2%)	0.5277	(–0.2%)
Rus	0.4530	0.4530	(0.0%)	0.4549	(+0.4%)	0.4883	(+7.8%)
σ	0.429		0.426		0.428		0.430
All NP	0.5142	0.4928	(–4.2%)	0.4630	(–10.0%)	0.4909	(–4.5%)
All HP	0.4597	0.4643	(+1.0%)	0.4254	(–7.5%)	0.4320	(–6.0%)
All	0.4900	0.4802**	(–2.0%)	0.4464**	(–8.9%)	0.4648**	(–5.1%)

Table 1: Mean Reciprocal Rank (MRR) of WebCLEF 2005 mixed monolingual runs. Statistically significant differences on All from the NoStem baseline (Wilcoxon Signed Rank Test) are denoted * and ** for ($p < 0.05$) and ($p < 0.01$) respectively. Lang. = topic language. (Δ%) = % diff. from NoStem. σ=st. deviation. NP = named page. HP = homepage.

Lang.	NoStem	PorStem	(Δ%)	AllStem	(Δ%)	SelStem	(Δ%)
Dan	0.6914	0.6901	(–0.2%)	0.6735	(–2.6%)	0.6735	(–2.6%)
Ger	0.4451	0.4415	(–0.8%)	0.4145	(–6.9%)	0.4196	(–5.7%)
Eng	0.6509	0.6167	(–5.3%)	0.6158	(–5.4%)	0.6024	(–7.5%)
Spa	0.4428	0.4237	(–4.3%)	0.4002	(–9.6%)	0.3916	(–11.6%)
Fre	0.1111	0.1111	(0.0%)	0.0000	(n/a)	0.0000	(n/a)
Hun	0.3862	0.3862	(0.0%)	0.3080	(–20.2%)	0.2855	(–26.1%)
Dut	0.5601	0.5573	(–0.5%)	0.5467	(–2.4%)	0.4974	(–11.2%)
Por	0.5068	0.4942	(–2.5%)	0.4367	(–13.8%)	0.3600	(–29.0%)
Rus	0.5755	0.5755	(0.0%)	0.5772	(+0.3%)	0.5755	(0%)
σ	0.423	0.418		0.425		0.425	
All	0.5150	0.5031*	(–2.3%)	0.4733**	(–8.1%)	0.4530**	(–12.0%)

Table 2: Mean Reciprocal Rank (MRR) of the WebCLEF 2006 mixed monolingual runs (manual topics). Statistically significant differences on All from the NoStem baseline (Wilcoxon Signed Rank Test) are denoted * and ** for ($p < 0.05$) and ($p < 0.01$) respectively. Lang. = topic language. (Δ%) = % diff. from NoStem. σ = st. deviation. n/a = non applicable.



(a) WebCLEF 2005 Topics (b) WebCLEF 2006 (Manual) Topics

Figure 1: MRR per language with different stemming combinations for the WebCLEF 2005 and 2006 topics.

- Similarly to before, applying Porter's stemming gives the most stable retrieval performance throughout (smallest deviation among languages throughout), which is however not the best performance in terms of retrieval effectiveness.
- The considerable amount of variation across languages reported for the 2005 topics is observed here as well (see Figure 1(b)). This trend was also reported in WebCLEF 2006 [4], namely that some languages were hard for all systems.
- The **SelStem** run never outperforms the **AllStem** run for any language. This suggests that, unlike for the 2005 topics, the language identifier has failed to suggest a language for only a few queries, meaning that there has been insufficient fallback (cf **NoStem**) to increase the overall performance for those languages. This is confirmed by Table 3, which is described below.

Overall, we can summarise the observations drawn from Tables 1 and 2 as follows:

- In a multilingual Web IR environment, applying no stemming at all is generally the most effective approach. As predicted, applying Porter's English stemming to all languages results in a signficant decrease compared to applying no stemming. However, unexpectedly, applying Porter's English stemmer does achieve the most stable retrieval performance across both tasks. Applying language-specific stemming is neither the most stable, nor the most effective retrieval approach, and in particular, always results in a statistically significant degradation in overall MRR.
- In a realistic Web IR environment, the languages of each query are not available. However, using modern language identification tools to select an appropriate stemmer can affect the performance of a selective stemming system. In particular, Table 3 shows the accuracy and the number of unknowns generated by the language identification tool for the topic and documents respectively. While 94% accuracy is achievable for the language identification of the documents, due to the much shorter nature of the queries, only 50% accuracy is achieved in query language identification. This explains the difference in performance exhibited between the **AllStem** and **SelStem** runs in Tables 1 and 2.

This conclusion is not entirely generalisable, but subject to the quality of the stemming resources used. The different stemmers used for various languages are not necessarily of the same quality. For example, the performance of the Hungarian stemmer is not entirely satisfactory; the stemmer used for Icelandic is in fact designed to stem Danish. On the contrary, Porter's stemmer for English is a generally popular and well-established stemmer, the performance of which can be expected to be relatively reliable. More and better resources are needed in order to have a more accurate idea of whether language-specific stemming is indeed not beneficial for multilingual Web IR. Additionally, the accuracy of language-specific stemming is partly depicted by the extent to which the language of the queries can be identified, and hence we believe that it is in this area that future research should also be directed.

Language Identification				
	Accuracy		Unknown	
	2005	2006	2005	2006
Topics	55.9%	51.5%	43.3%	13.2%
Relevant Documents	94.4%	94.7%	2.5%	1.7%
All Documents	n/a	n/a	2.8%	

Table 3: Accuracy of the language identification for the language of the topics, and the language of the target documents of the topics. Unknown is the fraction that the classifier failed to suggest any languages. Note that there is only a language identification ground truth available for the relevant documents, not all documents in the collection.

WebCLEF Year	
2005	2006
0.5135	**0.5150**
0.4900	0.3145
0.4780	0.1396
0.2860	0.0923

Table 4: Terrier's best runs (bold) versus top 3 submitted runs for WebCLEF 2005 & 2006 (mixed monolingual task).

Finally, Table 4 displays the best MRR scores reported in our experiments next to the top three runs on the manual queries submitted to WebCLEF 2005 and 2006 from all participating groups. However, because these are the official submitted runs of participating groups, they all use more than baseline settings: for example, they make use of retrieval-enhancing techniques, such as some knowledge about the document URL, query expansion, Natural Language Processing (NLP) functionalities, and so on. In fact, the best scoring run for the manual runs of 2005 (MRR of 0.5135) uses the same retrieval system and weighting model on fields as our reported runs. Nevertheless, that run outperforms our equivalent run (MRR of 0.4900), because it uses URL evidence and acronym expansion, while we only use the baseline weighting model with document fields. Note that for the 2006 manual topics, our reported run obtains the best overall performance. Naturally, the retrieval performance reported here could be improved by using retrieval-enhancing techniques, such as the ones mentioned above, and by further optimising the system's settings.

5. CONCLUSIONS

We investigated whether the standard IR techniques implemented in Terrier are appropriate for non-English retrieval, with special focus on the use of stemming in a multilingual setting. The bare-system approach of applying no stemming at all is very effective, and in addition is a safe and stable option, where the results are significantly better than those produced by the best stemming approach for that language. It is not clear that stemming with respect to a language can assist retrieval performance, and in particular the performance of such is partly depicted by the accuracy of the language identifier tool used for the documents and the queries.

With regards to the retrieval platform used, we have shown how Terrier's modular configuration allows for some simple extensions that easily solve some well-noted technical problems in the field (e.g. character encoding). Experiments in a mixed monolingual environment show that the platform is thoroughly robust in dealing with queries in 11 European languages.

Future work includes using more realistic settings as well as more and better quality resources (e.g. non-English stemmers). Moreover, we will aim to adapt Terrier to non-European languages with different writing systems, such as Chinese or Japanese, where the tokenisation performed is much more important. In particular, the success of Terrier on retrieval in a Japanese content can be evaluated using collections from the NTCIR evaluation forum[6].

6. REFERENCES

[1] M. Adriani and R. Pandugita. Using the Web information structure for retrieving Web pages. In Peters et al. [21], pages 892–897.

[2] G. Amati. *Probabilistic Models for Information Retrieval based on Divergence from Randomness*. PhD thesis, University of Glasgow, Glasgow, 2003.

[3] L. Azzopardi and M. de Rijke. Automatic construction of known-item finding test beds. In *Proceedings of SIGIR 2006*, pages 603–604, 2006.

[4] K. Balog, L. Azzopardi, J. Kamps, and M. de Rijke. Overview of WebCLEF 2006. In A. Nardi, C. Peters, and J. Vicedo, editors, *Working Notes CLEF 2006*, 2006.

[5] W. B. Cavnar and J. M. Trenkle, N-Gram-Based Text Categorization. In Proceedings of SDAIR'94, pages 161–175, 1994.

[6] N. Craswell and D. Hawking. Overview of the TREC-2004 Web track. In *Proceedings of TREC-2004.*, 2005.

[7] N. Craswell, D. Hawking, R. Wilkinson, and M. Wu. Overview of the TREC 2003 Web track. In *Proceedings of TREC-2003*, 2004.

[8] C. G. Figuerola, J. L. A. Berrocal, Á. F. Z. Rodríguez, and E. R. V. de Aldana. Web page retrieval by combining evidence. In Peters et al. [21], pages 880–887.

[9] F. C. Gey, N. Kando, and C. Peters. Cross language information retrieval: a research roadmap. *SIGIR Forum*, 36(2):72–80, 2002.

[10] D. Harman. A failure analysis on the limitations of suffixing in an online environment. In *Proceedings of SIGIR 1987*, pages 102–108, 1987.

[11] B. He and I. Ounis. A study of the dirichlet priors for term frequency normalisation. In *Proceedings of SIGIR 2005*, pages 465–471, 2005.

[12] N. Jensen, R. Hackl, T. Mandl, and R. Strötgen. Web retrieval experiments with the EuroGOV corpus at the University of Hildesheim. In Peters et al. [21], pages 837–845.

[13] J. Kamps, M. de Rijke, and B. Sigurbjörnsson. Combination methods for crosslingual Web retrieval. In Peters et al. [21], pages 856–864.

[14] C. Lioma, C. Macdonald, B. He, V. Plachouras, and I. Ounis. University of Glasgow at TREC 2006: Experiments in Terabyte and Enterprise Tracks with Terrier. In *Proceedings of TREC-2006*, 2007.

[15] C. Macdonald, B. He, V. Plachouras, and I. Ounis. University of Glasgow at TREC 2005: Experiments in Terabyte and Enterprise Tracks with Terrier. In *Proceedings TREC-2005*, 2006.

[16] C. Macdonald, V. Plachouras, B. He, C. Lioma, and I. Ounis. University of Glasgow at WebCLEF 2005: Experiments in per-field normalisation and language specific stemming. In Peters et al. [21], pages 898–907.

[17] T. Martínez, E. Noguera, R. Muñoz, and F. Llopis. University of Alicante at the CLEF 2005 WebCLEF track. In Peters et al. [21], pages 865–868.

[18] Á. Martínez-González, J. L. Martínez-Fernández, C. de Pablo-Sánchez, and J. Villena-Román. MIRACLE at WebCLEF 2005: Combining Web specific and linguistic information. In Peters et al. [21], pages 869–872.

[19] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and C. Lioma. Terrier: A High Performance and Scalable Information Retrieval Platform. In *Proceedings of OSIR 2006*, 2006.

[20] S. Ozmutlu, A. Spink, and H. C. Ozmutlu. A day in the life of Web searching: an exploratory study. *Inf. Process. Manage.*, 40(2):319–345, 2004.

[21] C. Peters, F. C. Gey, J. Gonzalo, H. Müller, G. J. F. Jones, M. Kluck, B. Magnini, and M. de Rijke, editors. *Accessing Multilingual Information Repositories, 6th Workshop of the Cross-Language Evalution Forum, CLEF 2005, Vienna, Austria, 21-23 September, 2005, Revised Selected Papers*, volume 4022 of *Lecture Notes in Computer Science*. Springer, 2006.

[22] S. E. Robertson, H. Zaragoza, and M. J. Taylor. Simple BM25 extension to multiple weighted fields. In Proceedings of CIKM 2004, pages 42–49, 2004.

[23] B. Sigurbjörnsson, J. Kamps, and M. de Rijke. EuroGOV: Engineering a multilingual Web corpus. In Peters et al. [21], pages 825–836.

[24] B. Sigurbjörnsson, J. Kamps, and M. de Rijke. Overview of WebCLEF 2005. In Peters et al. [21], pages 810–824.

[25] S. Tomlinson. Danish and Greek Web search experiments with Hummingbird SearchServer^{TM} at CLEF 2005. In Peters et al. [21], pages 846–855.

[26] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, 1979.

[6] `http://research.nii.ac.jp/ntcir/`