Benefits of deep NLP-based lemmatization for information retrieval

Péter Halácsy
Budapest University of Technology and Economics
Centre for Media Research
hp@mokk.bme.hu

Abstract

This paper reports on our system used in the CLEF 2006 ad hoc mono-lingual Hungarian retrieval task. Our experiments focus on the benefits that deeper NLP-based lemmatization (as opposed to simpler stemmers) can contribute to mean average precision. Our results show that these benefits counterweight the disadvantage of using an off-the-shelf retrieval toolkit.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.3.7 Digital Libraries; H.2.3 [**Database Managment**]: Languages—*Query Languages*

General Terms

Measurement, Performance, Experimentation

Keywords

Information retrieval, Stemming, Morphological Analysis, Hungarian Language

1 Background for the Hungarian experiments

The first Hungarian test collection for information retrieval appeared last year at the CLEF 2005 conference. Previously there have been no empirical experiments measuring the effect of Hungarian language features on retrieval performance.

The Hungarian language[1] is highly inflectional, rich in compound words, and has an extensive inflectional and derivational morphology. Only nominals and verbs can be suffixed with *inflectional* suffixes. Nominals and verbs can also be suffixed with productive *derivational* suffixes but there are few derivational suffixes which can attach to adverbs, preverbs and postpositions.

The set of inflectional suffixes after nominals are more or less the same, therefore nouns, adjectives and numerals cannot be defined solely on the base of inflectional morphology. The stem can be followed by 7 types of Possessive (`POSS`), 3 Plural, 3 Anaphoric Possessive (`ANP`) and 17 Case suffixes that results in as many as 1134 possible forms. In grammar textbooks we often find examples such as: *botjaitokéinak* = 'for the sg. of your (pl) sticks', which is analyzed as:
`bot/NOUN<PLUR><POSS<2><PLUR>><ANP<PLUR>><CAS<DAT>>`

[1] A more detailed descriptive grammar of Hungarian is available at `http://mokk.bme.hu/resources/ir`

The plural and case suffixes have several alternants. The alternation is governed by several morpho-phonological processes such as vowel harmony and depends on the stem and on the types of the case suffixes. For example the nominal plural is expressed with the suffix *-k*, but after consonant-final stems it is preceded by a non-high vowel (*a e o ö*), the so-called linking vowel. The quality of the linking vowel depends on the stem-vowel(s) and other lexical properties of the stem. Table 1 shows the variants of the plural suffix for different stems.

Singular	**Plural**	GLOSS
kar	*karok*	'arm(s)'
vár	*várak*	'castle(s)'
bér	*bérek*	'wage(s)'
bör	*börök*	'skin(s)'
kapu	*kapuk*	'gate(s)'
lufi	*lufik*	'balloon(s)'
hajó	*hajók*	'boat(s)'
fa	*fák*	'tree(s)'
kefe	*kefék*	'brush(es)'

Table 1: Examples for the different forms of the plural suffix

Suffixing of foreign names presents a specific challenge for information retrieval because of the effort needed to maintain proper name lexicons. If a foreign name is suffixed, the selection of suffix allomorphs can be sensitive to the normal Hungarian pronunciation of the stem as Table 1 shows some examples for vowel harmony ($a \sim e$), consonant assimilation, and for final vowel lengthening ($a \sim á$, $e \sim é$, $o \sim ó$). Note that in the last type the stem-final *a e o* changes to the long (accented) versions.

suffixed form	GLOSS
Clintonnal	'with Clinton'
Reagannel	'with Reagan'
Austerrel	'with Auster'
O'Connorral	'with O'Connor'
Balzackal	'with Balzac'
Lucasszal	'with Lucas'
Bachhal	'with Bach'
Hessével	'with Hesse'
Tzarával	'with Tzara'
Hugóval	'with Hugo'

Table 2: Some examples for suffix alternations of foreign names.

Verbs have fewer forms than nouns: 104 forms (in our classification) that express various person, number, tense, and transitivity distinctions. The phonological alternations are very similar to those in the nominal system. For example: *vár+ok, kér+ek, üt+ök* = 'I wait, I request, I hit'.

Similarly to German and Finnish, compounding is very productive in Hungarian. Almost any two (or three) nominals next to each other can form a single compound written without an intervening whitespace. Examples are *vizumkötelezettség* = 'obligation (to carry) visa', *ópiumelőállítás* = 'opium manufacture', *üvegházhatás* = 'glass house effect (greenhouse effect)'.

The use of hyphens can also cause problems, as it is governed by complex orthographic rules. Some examples are given in Table 1.

Hungarian	English
közép-kelet-európai országok	Central and Eastern European countries
mobiltelefon-felhasználók	cell phone users
Harry Potter-jelenség	Harry Potter phenomenon
USA-ban	'in the USA'
szeptember 11-i terroristatámadások	September 11 terrorist attacks
James Bond-filmek	James Bond films
Starr-ra[2]	on Starr
New York-i[3]	from N.Y. (adjective)

Table 3: Using hyphenation sign.

2 Stemming algorithms

It should be evident from the foregoing that extensive stemming is especially beneficial for Hungarian information retrieval. All of last year's top five systems (Table 2) had some method for handling the rich morphology of Hungarian: either words were tokenized to n-grams or an algorithmic stemmer was used.

part	run	map	stemming method
jhu/apl	aplmohud	41.12%	4gram
unine	UniNEhu3	38.89%	Savoy's stemmer + decompounding
miracle	xNP01ST1	35.20%	Savoy's stemmer
humminngbird	humHU05tde	33.09%	Savoy's stemmer + 4gram
hildesheim	UHIHU2	32.64%	5gram

Table 4: The top five runs for Hungarian ad hoc monolingual task of CLEF 2005.

The best result was achieved by JHU/APL in the run called `aplmohud` [8]. They used a character 4-gram based tokenization in a language modeling information retrieval system. The n-gram technique solves the problem of rich agglutinative morphology and compounding. For example the word *atomenergia* = 'atomic energy' in the query is tokenized to *atom, tome, omen, mene, ener, nerg, ergi, rgia* strings. When the text only contains the form *atomenergiával* = 'with atomic energy', the system still finds the relevant document.

Although the Snowball stemmer was also used together with the n-gram tokenization for the English and French tasks, the Hungarian results were nearly as good: English 43.46%, French 41.22% and Hungarian 41.12%. From these results it seems that the difference between the isolating and agglutinating languages can be eliminated by character n-gram methods. But note that JHU/APL used a state-of-the-art retrieval model, achieving much better results than other contestants using n-gram methods.

Unine [11], Miracle [2] and Hummingbird [12] employ the same algorithmic Hungarian stemmer that removes the nominal suffixes corresponding to the different cases, the possessive and the number (plural). Above this, UniNEhu3 [11] also utilizes a language independent decompounding algorithm that tries to segment compounds according to corpus statistics calculated from the document collection.[10] This is only triggered by long words, composed by more than 8 characters.

[12] isolates the effect of the stemmer which increases the mean average precision from 18.24% to 27.4%. However, they also mention that in some cases the aggressive overstemming causes a drop in performance. The *bank*='bank' word is stemmed to *ban*='in' because the stemmer assumes this this k is the plural suffix. The other example is *német*='German' which is stemmed after accent removal to *nem* which is a homonym word meaning 'no' or 'gender'. In their best runs they also use 4-grams instead of stemming and decompounding.

This suggests that a lexical stemmer – that incorporates a lexicon and the morphological rules

– can be more beneficial for retrieval. The n-gram experiments show that the decompounding is more than salutary. This hypothesis is confirmed by [7]. They measured that for Finnish, which is very similar to Hungarian, lexicon based lemmatization increase mean average precision from 32.8% to 56.1%. A pure algorithmic stemmer can only achieve as high as 42.4% score. They also emphasize the positive effect of decompounding.

[6] got a poor performance at the Multilingual Web Track when using a stemmer somewhat similar to ours. This stemmer was based on the Hungarian ispell wordlist, created for the Hunspell spellchecker. It reduced the accuracy of the retrieval. They do not mention specific faults in their report, therefore we cannot reconstruct what might have caused the decline. However we suggest that the 'take-the-first-stem' heuristic did not work well with the spellchecker's default settings.

3 Our system

Our research group has been working on a Hungarian morphological analyzer for three years. First we extended the codebase of `MySpell`, a reimplementation of the well-known `Ispell` spellchecker, yielding a generic word analysis library[9, 15]. At this point the development of the library has forked. Now the extended `MySpell`, called `HunSpell`, is part of the OpenOffice.org multilingual office suite. `Hunmorph` is the program tuned to morphological analysis.

Our technology, like the `Ispell` family of spellcheckers it descends from, enforces a strict separation between the language-specific resources (known as `dic`tionary and `aff`ix files), and the runtime environment, which is independent of the target natural language.

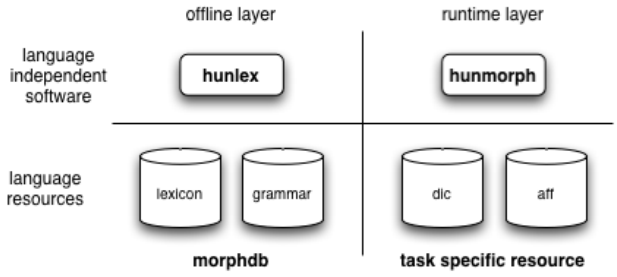

Figure 1: Architecture of the hunmorph word analysis framework.

Compiling accurate wide coverage machine-readable dictionaries and coding the morphology of a language can be an extremely labor-intensive task, so the benefit expected from reusing the language-specific input database across tasks can hardly be overestimated. To facilitate this resource sharing and to enable systematic task-dependent optimizations from a central lexical knowledge base, we designed and implemented a powerful offline layer we call `hunlex`. `Hunlex` offers an easy to use general framework for describing the lexicon and morphology of any language. Using this description it can generate the language-specific `aff/dic` resources, optimized for the task at hand.

`morphdb.hu`[14] is the Hungarian lexical database and morphological grammar for the `hunmorph` framework. It is the outcome of a several-year collaborative effort and represents a resource with the widest coverage and broadest range of applicability presently available for Hungarian. The grammar resource is the formalization of well-founded theoretical decisions handling inflection and productive derivation.

The coverage of `morphdb.hu` was measured running the analyzer on two Hungarian Corpora. One of them is the Szeged Corpus [1] which contains 1 million words, and on which the recall of our analyzer is 90%. The missing 10% are mostly proper names and acronyms not analyzed partly due to the difficulty of multi-word named entity tokenization. The other corpus is the 700 million word Hungarian Webcorpus [3, 5] on which the proportion of out-of-vocabulary items is 7%.

For the CLEF 2006 experiments we have developed two types of lemmatizers. The first is context independent: we choose the analysis with the shortest lemma, ie. we try to strip the longest possible suffix. The second algorithm is more sophisticated: the choice between alternative morphological analyses is resolved using the output of a POS tagger[4]. When there are several analyses that match the output of the tagger, we choose one with the biggest number of identified morphemes.

The morphological analyzer is able to guess possible analyses even if the input word is out of vocabulary. This feature works similarly to resourceless algorithmic stemmers, but hunmorph knows much more about suffix rules (also including derivations and compounding), and it only gives results in accordance with Hungarian morpho(-phono)logical rules. For example it is known that the word *bank* cannot be plural as suggested by Savoy's stemmer (see Section 2), because following final-consonant a linking-vowel is needed before the 'k'.

Decompounding is similar to guessing, as hunmorph has to analyze words that are not included in their lexicon. An unknown word is split if its components are known nominals. It is a grammatical rule that non-nominals can't be combined.

Another feature of hunmorph is *blocking*. If this option is set, the algorithm gives back less analyses. A lexical (non-affixed) partial analysis always blocks one that involves affixation. Out of two partial analyses, only the ones that are not equivalent are kept. Blocking effectively implements the idea that productive generation of an item by affixation or compounding is a fallback option in case the item is not found lexicalized. The 'blocking' and 'compounds' options can be used alongside in which case blocking also suppresses a compound analysis if the compound is entered as a lexical item.

morphdb.hu contains lexicalized words even with derivational suffixes and compounds. For example the word *üvegház*='glass house (greenhouse)' is represented as one entry in the lexicon, therefore the blocking option suppresses the decompounded analysis. Although blocking is only an option of hunmorph, during our experiments we always used it, as it can prevent the stemmer from overstemming.

For lemmatization we implemented a post-hoc filter that first chooses only one analysis which the lemma is extracted from in the second step. If there is only one possible analysis (as in 50% of the cases), use this analysis. Otherwise the ambiguity can be reduced by the following rules:

- If there is at least one analysis that is neither compound nor guessing then only this result is considered.
- If there is no such 'simple' output, the compounds of known words dominate over the guessed lemmas (analyses of unknown words).

The used analysis is the one with the shortest lemma. The next option is to decide whether to split the compound words or not. In decompounding mode, the components are added to the index next to each other as different tokens. Finally, and independently from decompounding, there is an option whether to strip derivations or not. Because of the blocking option mentioned above, derivation stripping has no effect on lexicalized words.

The more sophisticated lemmatization procedure involves a POS tagger[4] in the first step. This is a time consuming task that attaches an inflectional tag to all tokens in the corpus. To decrease ambiguity, only the matching analyses (with the same POS tags) are chosen from the output of the morphological analyzer. After this, the same post-hoc filter is applied.

We used Lucene (off-the-shelf) for indexing and retrieval with its standard vector space model. Only the stemmed tokens were added to the index. All fields of documents were concatenated. We used our own rule based tokenizer based on Lucene's `StandardTokenizer`. We prohibited the recognition of web hosts and acronyms, as these can be confused with periods between two sentences (wrongly) written without an intervening whitespace. And we allowed words to contain hyphens, since foreign names often are suffixed with linking hyphens. But after lemmatization the words were split at remaining hyphens. Before indexing the text was converted to lowercase (the

morphological analyzer can handle uppercase words) but the accented characters were unmodified. In addition, we used the same stopword list as [13][4]. Both the index the queries were stopped.

The disambiguation needs sentence boundary detection which is unusual in information retrieval. For this we trained a maximum entropy model on the Hungarian Webcorpus [3].

4 Evalutation

All measurements were performed with the topics of CLEF 2005 and 2006 to be able to compare our results to others'. Table 5 shows the performance of our baseline system which doesn't use any stemming.

stem	deriv	comp	year	MAP	MRR	ret/rel
no	no	no	2005	21.31	48.17	648/939
			2006	18.30	44.95	759/1308

Table 5: Baseline: without stemming

In ad hoc retrieval experiments, the most used evaluation measure is "average precision". For a topic, it is the average of the precision figures obtained after each new relevant document is observed (using zeros as the precision for relevant documents which are not retrieved). By convention, it is based on the first 1000 retrieved documents for the topic. The score ranges from 0.0 (no relevant found) to 1.0 (all relevant ones were at the top of the list). The *Mean average precision (MAP)* is the mean of average precision scores over all of the test topics. This measure is higher for systems which retrieve relevant documents early in the ranking list.

Reciprocal rank (RR) and *Precision at 10 documents (P10)* focus only on the head of the ranked list. For a topic, RR equals to $\frac{1}{r}$, where r is the rank of the first retrieved relevant documents. If $RR = 1$ the first retrived document is relevant. If $RR = 0.5$ then the second. If no relevant document was found, $RR = 0$. *Mean reciprocal rank* is the mean of the reciprocal ranks over all topics. *P10* is the precision after the first 10 document retrieved. This is important for web applications, as it's well known that the average user doesn't typically look at the second page of search results. The last measure is *overall recall*, *ret/rel*, i.e. the number of retrieved relevant documents divided by the number of all relevant documents.

Table 6 shows the performance gain caused by stemming. The second and third rows isolate the impact of stripping derivations (deriv) and splitting compounds (compound). As the results show decompounding has much more (positive) effect on accuracy. But an unexpected result was that stripping derivations also increase precision. This might be caused by the fact that hunmorph blocks stripping of non-productive derivations, as lexicalized words are contained in the high coverage lexical database. So during stemming only productive and transparent derivations are stripped, like the frequent verbal derivations, e.g. *klón+oz* = 'to clone'.

It is encouraging that in the 2005 (Hungarian monolingual ad hoc) task we achieved better MAP than all other CLEF participants except JHU/APL. Since we used the off-the-shelf Lucene toolkit for retrieval, we can deduce that the good results are due to lemmatization.

Table 7 shows the retrieval performance when the POS tagger is utilized. We can see the unexpected result that morphological disambiguation has no significant positive effect on performance. What's more, in some cases a decline is experienced.

Figure 2 shows the overall recall vs. precision graph. At some recall levels the simple rule-based lemmatizer performs better, at others the disambiguator-based does.

Retrospectively, it is not too surprising that the morphological disambiguator did not bring improvements. First, for most of the tokens, there is nothing to disambiguate, as the lemma is unique. Second, when the disambiguator solves some nontrivial problem, it is often irrelevant from

[4]The stopword list is downloadable at `http://ilps.science.uva.nl/Resources/HungarianStemmer/`

deriv	comp	year	MAP	MRR	P10	ret/rel
no	no	2005	0.3227	0.6491	0.3400	795/939
		2006	0.2797	0.6465	0.3860	987/1308
yes	no	2005	0.3361	0.6983	0.3500	805/939
		2006	0.2933	0.6307	0.4020	1042/1308
no	yes	2005	0.3746	0.7074	0.3660	870/939
		2006	0.3317	0.6827	0.4180	1099/1308
yes	yes	2005	0.3926	0.7698	0.3800	882/939
		2006	0.3482	0.6967	0.4300	1152/1308

Table 6: Results of different stemming methods *without* disambiguation

deriv	comp	year	MAP	MRR	P10	ret/rel
no	no	2005	0.3027	0.6896	0.2900	798/939
		2006	0.2650	0.6548	0.3660	945/1308
yes	no	2005	0.3289	0.7221	0.3300	814/939
		2006	0.2855	0.6542	0.3880	1009/1308
no	yes	2005	0.3648	0.7352	0.3520	884/939
		2006	0.3229	0.7135	0.4040	1070/1308
yes	yes	2005	0.3861	0.7711	0.3740	893/939
		2006	0.3416	0.7214	0.4360	1120/1308

Table 7: Result of different stemming methods *after* disambiguation

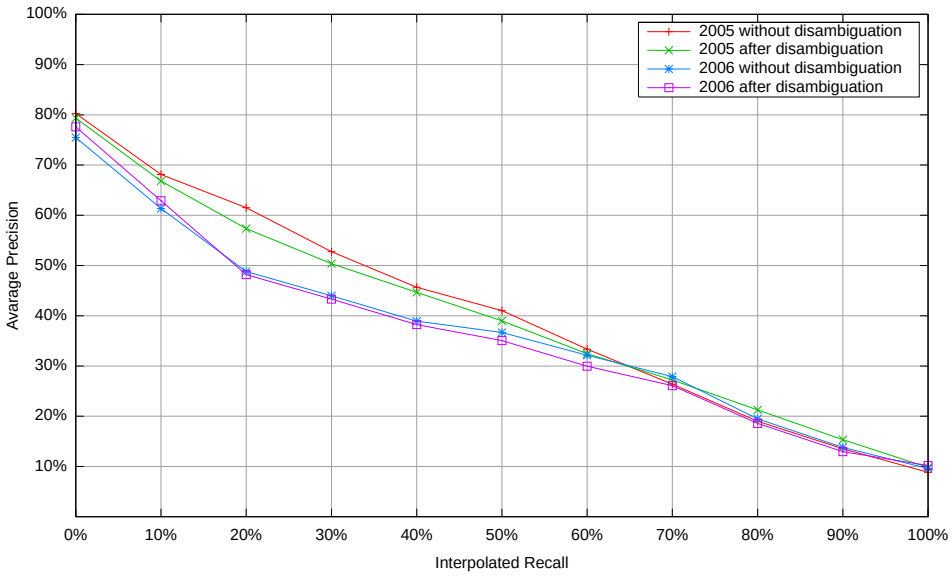

Figure 2: Interpolated Recall vs. Avarage Precision for runs with context insensitive lemmatization and lemmatization after disambiguation

a retrieval point of view. E.g. disambiguation of the frequent *egy* homonym ('one/NUM' or 'a/DET') is a hard task, but either of the lemmata is discarded by stopword filtering.

Another example is the classification of definite or indefinite verbal conjugation.[5] The tagger has to decide on the definiteness of *csináltam*='I did' based on the context, although the same lemma will be used in the two cases. So the effort of the tagger is irrelevant in retrieval.

Actual morphological homonym resolution also does not seem to be very useful for Hungarian IR. For example, the word *falunk* can be segmented as *falu+unk* = 'our village' or as *fal+unk* ='our wall'. But such ambiguities are more likely to appear in theoretical textbooks than in real-world language usage, very seldom affecting search results.

Using the POS tagger can even lead to new kinds of errors. For example, in Topic 367, an error of the morphological analyzer amplified an error of the POS tagger: The analyzer gave two analyses for the word *drogok*. The drog/PLUR inflected and the (arguably overgenerated) drog+ok (drug-cause) compounded one. The post-hoc filter presented in Section 3 would throw away the compounded version, but the POS tagger chooses it, preferring the singular to the plural form. This problem could be resolved by integrating the post-hoc filter into the POS tagger.

Table 8 shows our official results submitted to CLEF 2006. Only after the CLEF submission deadline did we realize that the simple rule-based lemmatizer can achieve the precision of the disambiguator-based, more advanced lemmatizer. So all submitted results are for the disambiguator-based lemmatizer. The values reported here are slightly different from the fourth row of Table 7, since we've reimplemented the lemmatizer meanwhile, and a bug has been eliminated.

	2005	2006
map	0.3831	0.3495
MRR	0.7187	0.7257
P10	0.3820	0.4360
ret/rel	891/939	1150/1308

Table 8: Official run (2006) using disambiguator with derivation stripping and decompunding. The results of the same system with the topics of CLEF 2005 are also shown.

5 Conclusion

The experiments on which we report in this paper confirm that a deep NLP-based lemmatization in Hungarian greatly improves retrieval accuracy. Our system outperformed all CLEF 2005 systems that use algorithmic stemmers even though our retrieval toolkit is using a not-state-of-the-art off-the-shelf basic vector space model. The good results are due to the high coverage lexical resources and the morphological analyzer which allow us the aggressive stemming and decompounding without overstemming.

We compared two different morphological analyzer-based lemmatization methods. We have found that a more advanced method based on high-precision context-sensitive morphological disambiguation does not bring improvements compared to a simpler context-insensitive greedy lemmatization algorithm.

Our Hungarian lemmatizer (together with its morphological analyzer and a Hungarian descriptive grammar) is released under a permissive LGPL-style license, and can be freely downloaded from `http://mokk.bme.hu/resources/ir`. We hope that members of the CLEF community will incorporate these into their IR systems, closing the gap in effectivity between IR systems for Hungarian and for major European languages.

[5]The verb forms have to be agreed with the definiteness of the *object* in the sentence. If the verb is intransitive or the object is an indefinite noun phrase, the indefinite value has to be used. The definite value on a verb refers a definite object.

References

[1] Dóra Csendes, Csaba Hatvani, Zoltán Alexin, János Csirik, Tibor Gyimóthy, Gábor Prószéky, and Tamás Váradi. Kézzel annotált magyar nyelvi korpusz: a Szeged Korpusz. In *II. Magyar Számítógépes Nyelvészeti Konferencia*, pages 238–245. Szegedi Tudományegyetem, 2003.

[2] José Miguel Goñi Menoyo, José C. González, and Julio Vilena-Román. Miracle's 2005 approach to monolingual information retrieval'. Working Notes for the CLEF 2005 Workshop, 21-23 September, Vienna, Austria, 2005.

[3] Péter Halácsy, András Kornai, László Németh, András Rung, István Szakadát, and Viktor Trón. Creating open language resources for Hungarian. In *Proceedings of Language Resources and Evaluation Conference (LREC04)*. European Language Resources Association, 2004.

[4] Péter Halácsy, András Kornai, Csaba Oravecz, Viktor Trón, and Dániel Varga. Using a morphological analyzer in high precision POS tagging of Hungarian. In *Proceedings of LREC 2006*, pages 2245–2248, 2006.

[5] András Kornai, Péter Halácsy, Viktor Nagy, Csaba Oravecz, Viktor Trón, and Dániel Varga. Web-based frequency dictionaries for medium density languages. In *Proceedings of the EACL 2006 Workshop on Web as a Corpus*, 2006.

[6] Craig Macdonald, Vassilis Plachouras, He Ben, Lioma Christina, and Ounis Iadh. University of Glasgow at WebCLEF 2005: Experiments in per-field normalisation and language specific stemming. Working Notes for the CLEF 2005 Workshop, 21-23 September, Vienna, Austria, 2005.

[7] Craig Macdonald, Vassilis Plachouras, Ben He, Christina Lioma, and Iadh Ounis. Finnish, Portuguese and Russian Retrieval with Hummingbird SearchServerTM at CLEF 2004, 2004.

[8] Paul McNamee. Exploring New Languages with HAIRCUT at CLEF 2005. Working Notes for the CLEF 2005 Workshop, 21-23 September, Vienna, Austria, 2005.

[9] László Németh, Viktor Trón, Péter Halácsy, András Kornai, András Rung, and István Szakadát. Leveraging the open-source ispell codebase for minority language analysis. In *Proceedings of SALTMIL 2004*. European Language Resources Association, 2004.

[10] Jacques Savoy. Report on CLEF-2003 monolingual tracks: Fusion of probabilistic models for effective monolingual retrieval. Results of the CLEF-2003, cross-language evaluation forum, 2003.

[11] Jacques Savoy and Pierre-Yves Berger. Report on CLEF-2005 Evaluation Campaign: Monolingual, Bilingual, and GIRT Information Retrieval. Working Notes for the CLEF 2005 Workshop, 21-23 September, Vienna, Austria, 2005.

[12] Stephen Tomlinson. Europan Ad hoc retireval experiments with HummingbirdTM at CLEF 2005'. Working Notes for the CLEF 2005 Workshop, 21-23 September, Vienna, Austria, 2005.

[13] Anna Tordai and Maarten de Rijke. Hungarian Monolingual Retrieval at CLEF 2005. Working Notes for the CLEF 2005 Workshop, 21-23 September, Vienna, Austria, 2005.

[14] Viktor Trón, Péter Halácsy, Péter Rebrus, András Rung, Péter Vajda, and Eszter Simon. Morphdb.hu: Hungarian lexical database and morphological grammar. In *Proceedings of LREC 2006*, pages 1670–1673, 2006.

[15] Viktor Trón, György Gyepesi, Péter Halácsy, András Kornai, László Németh, and Dániel Varga. Hunmorph: open source word analysis. In *Proceeding of the ACL 2005 Workshop on Software*, 2005.